

# BAB 1

## PENDAHULUAN

### 1.1 Latar Belakang

Penelitian ilmiah merupakan komponen kritis dalam pengembangan akademik, khususnya di lingkungan perguruan tinggi. Salah satu tahap paling fundamental dalam penyusunan penelitian adalah pemilihan judul yang relevan, orisinal, dan sesuai dengan perkembangan keilmuan. Di Fakultas Ilmu Komputer Universitas Pasir Pengaraian mahasiswa sering kali menghadapi kesulitan dalam menemukan judul penelitian yang unik dan berbasis inovasi terkini karena terbatasnya akses terhadap referensi yang terintegrasi serta minimnya alat pendukung berbasis teknologi. Selain itu, proses pencarian judul yang masih manual atau mengandalkan *database* statis berpotensi menyebabkan duplikasi topik, ketidakefisienan waktu, dan kurangnya inspirasi bagi mahasiswa.

Sistem pencarian judul penelitian yang ada saat ini umumnya hanya menyediakan fungsi pencarian berbasis kata kunci sederhana tanpa kemampuan analisis kontekstual. Hal ini mengakibatkan hasil pencarian kurang akurat dan tidak mampu mengakomodasi kebutuhan kompleks mahasiswa, seperti rekomendasi topik berdasarkan tren penelitian terbaru atau kesesuaian dengan bidang minat. Oleh karena itu, diperlukan solusi yang memanfaatkan kecerdasan buatan (*AI*) untuk meningkatkan kualitas dan efisiensi sistem pencarian.

Berdasarkan hasil wawancara dengan Ketua Program Studi Teknik Informatika & Sistem Informasi diketahui bahwa proses penentuan judul magang dan skripsi di lingkungan program studi masih dilakukan secara manual. Mahasiswa umumnya mencari referensi dari skripsi kakak tingkat, internet, atau

bertanya langsung kepada dosen pembimbing. Namun, metode ini dinilai kurang efektif karena memakan waktu, tidak adanya sistem digital khusus, dan sering kali mahasiswa merasa ragu untuk berkonsultasi di tahap awal. Kendala yang sering dihadapi meliputi kurangnya ide kreatif, kebingungan menentukan topik sesuai minat, dan ketidaktahuan apakah judul yang diajukan sudah pernah digunakan.

Hasil kuesioner yang disebarakan kepada mahasiswa skripsi dan magang juga menunjukkan pola yang sejalan. Dari total *responden*, sebagian besar mengaku mengalami kesulitan dalam menemukan ide penelitian atau topik magang yang sesuai minat, dengan kendala utama berupa “sulit menemukan ide” dan “kurang referensi”. Mayoritas responden menyatakan minat pada *platform* berbasis *website*, topik yang relevan dengan bidang mereka, serta metode penelitian berbasis *NLP*. Lebih dari 80% *responden* menyatakan “sangat tertarik” terhadap gagasan sistem berbasis kecerdasan buatan yang dapat merekomendasikan ide judul skripsi atau magang. Fitur yang dianggap penting meliputi *input* minat/topik, rekomendasi judul *otomatis*, dan penyimpanan hasil rekomendasi.

*Natural Language Processing (NLP)* adalah disiplin ilmu komputer yang bertujuan untuk memahami konsep dan maksud dari bahasa manusia. Penggunaan *Natural Language Processing (NLP)* ini dapat diintegrasikan dengan model *Convolutional Neural Network (CNN)* untuk meningkatkan analisis data berbasis gambar, seperti identifikasi hama, penyakit atau kondisi tanaman. Kombinasi ini mendukung pengambilan keputusan yang lebih tepat oleh petani tanpa memerlukan pengetahuan teknis mendalam. Beberapa penelitian telah menunjukkan potensi teknologi ini. Khoirunisa (2020) mengembangkan *chatbot* berbasis *Natural*

*Language Processing (NLP)* untuk membantu petani mendapatkan informasi seperti harga bibit dan prediksi cuaca. *Chatbot* ini menggunakan teknik *rule-based* dengan tokenization dan stemming untuk memahami input pengguna. Namun, pendekatan ini hanya fokus pada analisis *teks* tanpa integrasi dengan *data visual* untuk diagnostik berbasis gambar. Di sisi lain, Rahman (2023) menerapkan algoritma *Convolutional Neural Network (CNN)* untuk mengidentifikasi kualitas tanaman selada berdasarkan citra daun. Penelitian ini berhasil mencapai akurasi 90% pada pelatihan model dengan optimasi *dropout* dan *learning rate*. Namun, penelitian ini hanya mencakup *analisis visual* dan belum mengintegrasikan teknologi berbasis *teks* seperti konsultasi berbasis bahasa alami [1].

*Natural Language Processing (NLP)* memungkinkan mesin ke mesin atau interaksi manusia ke mesin menggunakan bahasa alami terkait manusia. Disiplin ini difokuskan pada mempelajari kemampuan komputer untuk memahami dan memproses bahasa manusia untuk selanjutnya menghasilkan tanggapan darinya. Beberapa metode adalah digunakan untuk memahami kata-kata dan niat dari pengguna yang ada berkomunikasi dalam suatu konteks: dari pencarian *teks* yang sederhana pola dalam pesan pengguna ke *artificial* yang lebih canggih teknik kecerdasan diterapkan pada bahasa manusia [2].

Dalam menghadapi era pesatnya perkembangan teknologi, perangkat *lunak* telah menjadi kebutuhan utama dalam berbagai aspek kehidupan. Kemajuan *LLM* sebagai bagian dari kecerdasan buatan telah melaju pesat, dengan implementasinya yang meluas di berbagai aplikasi dan bidang. Saat ini, *Google* telah meluncurkan *LLM* terbaru mereka yang diberi nama *Gemini*, *Google* membuka akses ke kunci

*API*-nya untuk para pengembang aplikasi yang ingin memanfaatkannya dalam pembuatan aplikasi[3].

Berdasarkan penelitian yang berjudul “Penerapan *Gemini Ai* Dalam Pembuatan Deskripsi Produk *E-Commerce*” Integrasi *Gemini AI* dalam deskripsi produk *e-commerce* telah merevolusi interaksi bisnis dengan pengguna. Penelitian ini memaparkan implementasi *API Gemini AI* dalam menghasilkan deskripsi produk pada sebuah *website e-commerce*. Penelitian ini mengikuti model *waterfall*, dimulai dengan analisis untuk memahami kebutuhan integrasi *API Gemini AI* [4].

Integrasi *Generative AI* melalui *Gemini API* menawarkan potensi besar dalam mengatasi tantangan tersebut. *Gemini API*, dengan kemampuan *Natural Language Processing (NLP)* dan *machine learning*, dapat menganalisis data penelitian sebelumnya, memahami konteks permintaan pengguna, serta memberikan rekomendasi judul yang inovatif dan relevan. Pemanfaatan teknologi ini diharapkan dapat mentransformasi sistem pencarian judul menjadi lebih dinamis, interaktif, dan responsif terhadap kebutuhan akademik.

Berdasarkan latar belakang tersebut, peneliti mengusulkan pengembangan “Integrasi *Natural Language Processing (NLP)* dan *Gemini Application Programming Interface (API)* untuk Rekomendasi Judul Magang dan Skripsi”.

## **1.2 Rumusan Masalah**

Rumusan masalah yang diangkat dalam penelitian ini adalah bagaimana menerapkan integrasi *Natural Language Processing (NLP)* Dan *Gemini API* untuk merekomendasikan judul magang dan skripsi?

### 1.3 Tujuan Penelitian

Tujuan dari penelitian ini adalah untuk menerapkan integrasi *Natural Language Processing (NLP)* Dan *Gemini API* dalam rekomendasi judul magang dan skripsi.

### 1.4 Batasan Masalah

Adapun batasan masalah pada penelitian ini sebagai berikut:

1. Integrasi *GEMINI API* difokuskan pada fungsi pencarian dan rekomendasi judul magang serta skripsi.
2. Teknologi utama yang digunakan meliputi *framework Laravel* untuk *backend*, *React.js* untuk *frontend*, dan *MySQL* sebagai basis data.
3. Pengujian sistem terbatas pada kelompok mahasiswa dan dosen Fakultas Ilmu Komputer sebagai pengguna utama.

### 1.5 Manfaat Penelitian

Manfaat Penelitian:

1. Bagi Mahasiswa: Mempercepat proses pencarian judul magang dan skripsi, mengurangi duplikasi, serta meningkatkan orisinalitas penelitian.
2. Bagi Fakultas: Mengoptimalkan manajemen data penelitian dan mendukung pengembangan kurikulum berbasis inovasi.
3. Bagi Peneliti: Memberikan kontribusi dalam penerapan *AI* generatif untuk solusi akademik yang lebih modern dan efisien.

## 1.6 Metode Pengumpulan Data

Metode yang digunakan meliputi:

1. Studi Literatur: Pengumpulan referensi dari jurnal, buku, artikel, dan sumber online terkait penerapan *AI*, *Natural Language Processing (NLP)*, dan *Gemini Api* dalam sistem pencarian judul.
2. Survei dan Kuesioner: Distribusi kuesioner kepada mahasiswa dan dosen untuk mengidentifikasi kebutuhan, tantangan, dan tren dalam pemilihan judul magang serta skripsi.
3. Wawancara: Diskusi terstruktur dengan *stakeholder* seperti dosen pembimbing dan praktisi industri untuk mendapatkan insight tentang tren penelitian dan relevansi judul.

## 1.7 Sistematika Penulisan

Sistematika penulisan dari skripsi ini terdiri dari lima bagian utama sebagai berikut:

### **BAB 1        PENDAHULUAN**

Bab ini akan menjelaskan mengenai latar belakang pemilihan judul tugas akhir “Integrasi *Natural Language Processing (NLP)* Dan *Gemini Application Programming Interface (API)* Untuk Rekomendasi Judul Magang Dan Skripsi”, rumusan masalah, Batasan masalah, tujuan penelitian dan manfaat penelitian, metode pengumpulan data, dan metodologi penelitian.

**BAB 2        LANDASAN TEORI**

Bab ini akan membahas teori-teori yang berkaitan tentang Integrasi *Natural Language Processing (NLP)* Dan *Gemini Application Programming Interface (API)* Untuk Rekomendasi Judul Magang Dan Skripsi.

**BAB 3        METODOLOGI PENELITIAN**

Bab ini menguraikan tentang pendahuluan dan kerangka kerja penelitian.

**BAB 4        ANALISA DAN PERANCANGAN SISTEM**

Bab ini berisi bagaimana menganalisa cara kerja sistem yang akan dibangun, dan menjelaskan tahap perancangan sistem berdasarkan hasil analisis agar dimengerti oleh pengguna.

**BAB 5        IMPLEMENTASI DAN PENGUJIAN**

Bab ini berisi tentang implementasi dan pengembangan perangkat lunak serta pengujian akhir terhadap sistem yang telah dibuat.

**BAB 6        PENUTUP**

Bab ini berisi mengenai kesimpulan terhadap sistem yang dibuat dan saran untuk pengembangan terhadap sistem yang telah dibuat.

## BAB 2

### LANDASAN TEORI

#### 2.1 Konsep Dasar Sistem

Sistem adalah rangkaian dari dua atau lebih komponen yang saling terhubung, yang berinteraksi untuk mencapai tujuan. Sebagian besar sistem terdiri dari sub sistem yang lebih kecil yang mendukung sistem yang lebih besar. Sistem adalah suatu jaringan *prosedure* yang dibuat menurut pola yang terpadu untuk melaksanakan kegiatan pokok perusahaan. Dapat penulis simpulkan bahwa sistem adalah kumpulan komponen yang saling terkait satu dengan yang lainnya kegiatan pokok untuk mencapai suatu tujuan tertentu [5].

Sistem adalah suatu jaringan kerja berdasarkan prosedur-prosedurnya yang saling berhubungan, berkumpul bersama-sama buat melakukan suatu aktivitas atau buat menuntaskan suatu target tertentu”. Sistem adalah suatu alat buat menghubungkan suatu jaringan kerja dari sekumpulan elemen-elemen yang saling berinteraksi antara satu dengan yang lainnya, sehingga mencapai tujuan yang telah ditetapkan atau direncanakan. Kesimpulan berdasarkan 2 artikel tersebut bahwa sistem merupakan *prosedur* & tata kerja yang teratur berdasarkan beberapa kumpulan elemen-elemen yang saling berinteraksi satu dengan yang lain buat mencapai tujuan tertentu [6].

Sistem adalah suatu jaringan kerja dari kumpulan prosedur yang berkumpul bersama-sama dan saling terkait untuk menyelesaikan suatu target tertentu atau untuk melakukan suatu kegiatan. Sistem adalah sekelompok komponen yang saling terkait, saling bekerja sama untuk mencapai tujuan bersama dengan menerima masukan serta menghasilkan keluaran dalam proses informasi yang teratur [7].



## **2.2 Konsep Dasar Informasi**

Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya. Kualitas dari suatu informasi tergantung tiga hal yaitu, informasi harus akurat, tepat pada waktunya, dan relevan [8].

Informasi adalah data yang telah diorganisasi dan telah memiliki kegunaan dan manfaat. Informasi adalah data yang telah dikelola dan diproses untuk memberikan arti dan memperbaiki proses pengambilan keputusan, sebagaimana perannya, pengguna membuat keputusan yang lebih baik sebagai kuantitas dan kualitas dari peningkatan informasi. Dapat penulis simpulkan, bahwa informasi adalah data mentah yang telah diolah sedemikian rupa sehingga menghasilkan sesuatu yang bermakna bagi penggunanya dalam mengambil sebuah keputusan [5].

Informasi adalah data yang diolah menjadi bentuk tertentu yang lebih bermanfaat dan lebih memiliki arti bagi yang menerimanya [7].

## **2.3 Sistem Informasi**

Sistem Informasi adalah gabungan yang teratur antara manusia, perangkat lunak, perangkat keras, jaringan komunikasi dan sumber data, dalam upaya untuk mengumpulkan, mengubah, dan menyebarkan informasi dalam organisasi. Dalam arti yang sangat luas, istilah sistem informasi yang sering dipakai mengacu kepada interaksi antara teknologi, data, proses algoritmik, dan para penggunanya [7].

Sistem informasi adalah sebuah rangkaian prosedur formal dimana data dikelompokkan, diproses menjadi informasi, dan didistribusikan kepada pemakai. Sistem informasi adalah cara-cara yang diorganisasi untuk mengumpulkan, memasukkan dan mengolah serta menyimpan data dan cara-cara yang diorganisasi

untuk menyimpan, mengola, mengendalikan, dan melaporkan informasi sedemikian rupa sehingga sebuah organisasi dapat mencapai tujuan yang telah ditetapkan. Penulis menyimpulkan bahwa, sistem informasi adalah kumpulan dari beberapa sistem di dalam suatu organisasi yang mengumpulkan, memproses, menyimpan dan mendistribusikan informasi sebagai pendukung pengambilan keputusan dan pengendalian dalam organisasi [5].

Sistem informasi merupakan sebuah sistem terstruktur yang memiliki prosedur-prosedur untuk melakukan pengolahan data guna mendapatkan informasi dan mempermudah pekerjaan dari pemakai sistem informasi [9].

#### **2.4 Integrasi**

Integrasi dalam penelitian ini mengacu pada penggabungan modul–modul *AI*, *NLP*, dan *data mining* menjadi sistem yang mampu memahami pola pengguna dan memberikan rekomendasi konteks khusus (Azzahra et al., 2024).

#### **2.5 Artificial Intelligence (AI)**

*Artificial Intelligence (AI)* merupakan teknologi yang meniru kecerdasan manusia dengan kemampuan seperti pembelajaran dan penalaran melalui *machine learning* dan *data analytics*, terbukti memacu efisiensi dan kualitas dalam sistem informasi Manajemen (Azzahra et al., 2024).

#### **2.6 Natural Language Processing (NLP)**

*Natural Language Processing (NLP)* adalah kombinasi dari ilmu komputer dan bidang kecerdasan buatan yang terkait dengan linguistik. *Natural Language Processing (NLP)* berkaitan dengan bagaimana mesin didesain untuk berinteraksi dengan manusia menggunakan bahasa alami. Secara prinsip, bahasa alami merupakan representasi pesan yang digunakan untuk berkomunikasi antar manusia.

Bentuk utama dari representasi ini biasanya berupa suara atau ucapan, namun seringkali juga diwujudkan dalam bentuk tulisan. Beberapa bidang utama dalam *Natural Language Processing (NLP)* meliputi *question answering systems (QAS)*, *summarization*, *machine translation*, *speech recognition*, dan *document classification*. Dengan teknologi *Natural Language Processing (NLP)*, komputer dapat mempelajari dan memahami bahasa manusia, sehingga mampu berkomunikasi secara efektif dengan manusia.[1]

## 2.7 GEMINI AI

*Gemini Ai* dapat dipahami sebagai platform kecerdasan buatan (*AI*) yang dirancang untuk menghasilkan teks, menerjemahkan bahasa, menulis berbagai jenis konten kreatif, dan menjawab pertanyaan dengan cara yang informatif. Platform ini dibangun dari awal untuk *multimodality*, memungkinkannya bekerja dengan mulus di berbagai format data seperti teks, kode, gambar, dan audio.[11]

*Gemini Ai* merupakan sistem percakapan *multimodal* yang *revolusioner* yang telah dirancang untuk melampaui model Bahasa besar berbasis teks tradisional seperti *GPT-3* dan bahkan pendahulunya yang *multimodal*, *ChatGPT-4*. *Gemini* memiliki arsitektur yang dirancang untuk memproses berbagai jenis data seperti teks, gambar, audio, dan video, yang memungkinkan untuk mencapai pemahaman *multimodal* yang canggih.[4]

*Gemini Ai* adalah salah satu platform kecerdasan buatan (*AI*) yang dikembangkan untuk meningkatkan kemampuan analitik dan prediktif dalam berbagai bidang. *Gemini Ai*, yang merupakan bagian dari pengembangan teknologi Large Language Models (*LLM*), memanfaatkan pendekatan berbasis *Application Programming Interface (API)* untuk mengintegrasikan fitur-fitur canggihnya ke

dalam aplikasi web dan sistem lainnya. Platform ini dirancang untuk memberikan kemampuan prediksi yang lebih akurat serta rekomendasi yang relevan, sehingga dapat membantu dalam pengambilan keputusan yang lebih efisien dan responsif terhadap berbagai situasi yang kompleks. Dalam konteks keamanan siber, *Gemini Ai* memainkan peran penting dengan meningkatkan deteksi ancaman dan memberikan respons yang lebih cepat terhadap potensi risiko. Melalui teknologi ini, data yang bersifat dinamis dan kompleks dapat dianalisis secara lebih efektif, memungkinkan pengurangan waktu reaksi terhadap ancaman siber yang muncul.[12]

### **2.8 Application Programming Interface (API)**

*API* merupakan sebuah mekanisme yang memungkinkan berbagai aplikasi untuk berkomunikasi dan berinteraksi satu sama lain. Dengan *API*, suatu aplikasi dapat mengakses atau menggunakan fungsionalitas, layanan, atau data dari aplikasi maupun sistem lain tanpa perlu mengetahui detail implementasinya secara internal. Dalam era perkembangan teknologi informasi, terjadi penggabungan antara keterampilan manusia dan teknologi baru, yang dikenal dengan istilah kecerdasan buatan atau *Artificial Intelligence (AI)*. [13]

### **2.9 Judul Penelitian**

Judul penelitian adalah pencerminan dari keseluruhan isi karya ilmiah yang bersifat menjelaskan diri, menarik sehingga orang langsung menduga materi dan masalah apa yang diteliti, penjabaran dari topik yang lebih spesifik dan sering menyiratkan variabel yang akan dibahas. Penulis judul penelitian biasanya sudah dirumuskan setelah perumusan masalah penelitian atas batasan dan penentuan masalah yang akan diteliti.[14]

## **2.10 Magang**

Pemagangan adalah bagian dari sistem pelatihan kerja yang diselenggarakan secara terpadu antara pelatihan di lembaga pelatihan dengan bekerja secara langsung di bawah bimbingan dan pengawasan instruktur atau pekerja secara langsung di bawah bimbingan dan pengawasan instruktur atau pekerja/buruh yang lebih berpengalaman, dalam proses produksi barang dan/atau jasa di perusahaan, dalam rangka menguasai keterampilan atau keahlian tertentu.[15]

## **2.11 Skripsi**

Skripsi adalah istilah yang digunakan di Indonesia untuk mendeskripsikan tesis ilmiah berupa penjelasan hasil penelitian sarjana, yang menggunakan kaidah-kaidah yang berlaku untuk membahas masalah/fenomena dalam bidang keilmuan tertentu [16].

Skripsi merupakan karangan ilmiah yang wajib ditulis oleh mahasiswa sebagai bagian dari persyaratan akhir pendidikan akademisnya [9]. Pengertian skripsi sendiri adalah suatu karya ilmiah yang diwajibkan sebagai bagian dari persyaratan pendidikan akademis di Perguruan Tinggi [17].

Skripsi merupakan salah satu mata kuliah yang ditempuh mahasiswa pada semester delapan. Yang didalamnya merupakan hasil paparan atau tulisan hasil penelitian mahasiswa di jenjang strata satu [18].

Skripsi merupakan suatu karya ilmiah berdasarkan suatu kegiatan penelitian mandiri mahasiswa, disusun dalam jangka waktu satu semester atau lebih dibawah bimbingan seorang dosen pembimbing dan dapat dibantu seorang dosen pembantu pembimbing [8].

## 2.12 Flowchart

*Flowchart* adalah penggambaran secara grafik dari langkah-langkah dan urutan prosedur dari suatu program. *Flowchart* menolong analyst dan programmer untuk memecahkan masalah kedalam segmen-segmen yang lebih kecil dan menolong dalam menganalisis alternatif-alternatif lain dalam pengoperasian. *Flowchart* biasanya mempermudah penyelesaian suatu masalah khususnya masalah yang perlu dipelajari dan dievaluasi lebih lanjut [40]. Terkadang seorang mengalami kesulitan dalam membuat flowchart khususnya mahasiswa yang baru melek pemrograman. Untuk itu merancang perangkat lunak yang dapat mengimplentasikan *flowchart* menjadi output program [41].

*Flowchart* membantu para analis dan programmer untuk membuat suatu gambaran aplikasi yang dibuat dan memecahkannya ke dalam segmen yang lebih kecil agar lebih mudah dianalisis dengan ini flowchart mempermudah penyelesaian suatu masalah [42].

## 2.13 Alat Bantu Perancangan Aplikasi

### 2.13.1 Unified Modeling Language (UML)

*Unified Modeling Language (UML)* merupakan satu *set* standar teknik diagram yang memberikan representasi *grafis* yang cukup kaya untuk model setiap pengembangan sistem proyek dari analisis melalui implementasi. *UML* singkatan dari *Unified Modeling Language* yang berarti bahasa pemodelan standar. Saat ini, sebagian besar sistem berorientasi objek analisis dan desain pendekatan menggunakan *UML* untuk menggambarkan perkembangan sistem. *UML*

menggunakan berbagai diagram untuk menunjukkan berbagai sudut pandang dari sistem yang sedang berkembang [19].

*UML* juga dapat didefinisikan sebagai suatu bahasa standar *visualisasi*, perancangan, dan pendokumentasian sistem, atau dikenal juga sebagai bahasa standar penulisan *blueprint* sebuah *software* [20]. *UML* dibangun dari penggabungan beberapa bahasa pemodelan *grafis* berorientasi *objek* yang berkembang pesat pada akhir 1980-an dan awal 1990-an. Fungsi *UML* adalah untuk membantu pendeskripsian dan desain sistem perangkat lunak, khususnya sistem yang dibangun menggunakan pemrograman berorientasi objek [21].

#### **2.13.1.1 Use Case Diagram**

*Use case diagram* adalah sebuah diagram yang menunjukkan hubungan antara *actors* dan *use cases*. Digunakan untuk analisis dan desain sebuah sistem [19]. Diagram *use case* menyajikan interaksi antara *use case* dan aktor, dimana aktor dapat berupa orang, peralatan, atau sistem lain yang berinteraksi dengan sistem yang sedang dibangun [22].

#### **2.13.1.2 Class Diagram**

*Class Diagram* berfungsi untuk menjelaskan tipe dari objek sistem dan hubungannya dengan objek yang lain [23]. *Class diagram* adalah sebuah diagram yang menunjukkan hubungan antar *class* yang didalamnya terdapat atribut dan fungsi dari suatu objek [20]

#### **2.13.1.3 Activity Diagram**

*Activity Diagram* atau diagram aktivitas menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak, yang perlu diperhatikan disini adalah bahwa diagram aktivitas

menggambarkan aktivitas sistem bukan yang dilakukan aktor, jadi aktivitas dilakukan oleh sistem [24].

#### **2.13.1.4 *Sequence Diagram***

*Sequence Diagram* adalah sebuah diagram yang menggambarkan kolaborasi dari objek-objek yang saling berinteraksi antar elemen dari suatu *class* [20]. *Sequence Diagram* menggambarkan objek kelakuan pada *usecase* dengan mendeskripsikan objek waktu hidup dan pesan yang dikirimkan dan diterima antar objek [26].

### **2.14 Bahasa Pemograman**

#### **2.14.1 *Hyper Text Markup Language (HTML)***

*Hypertext Markup Language (HTML)* adalah Bahasa yang digunakan untuk menulis halaman web. *HTML* merupakan pengembangan dari standar pemformatan dokumen *teks*, yaitu *Standard Generalized Markup Language ( SGML )* [27].

#### **2.14.2 *Cascading Style Sheets (CSS)***

*CSS* atau *Cascading Style Sheet*, *CSS* pertama kali diusulkan oleh Hakon Wium Lie tahun 1994 dan selanjutnya distandarisasi oleh W3C. *CSS* memberikan cara yang mudah dan efisien bagi pemrogram untuk menentukan tata letak halaman web dan mempercantik halaman dengan elemen desain seperti warna, sudut bulat, *gradien*, dan animasi. seperti halnya *HTML*, versi *CSS* juga berkembang. Versi pertama yang ditetapkan oleh W3C adalah *CSS 1 (CSS Level 1)* pada bulan Desember 1996. Selanjutnya berkembang menjadi *CSS 2* (bulan Mei 1998). *CSS 2.1* (tahun 2011) dan *CSS 3* yang diusulkan pada bulan 2012. Versi yang sedang populer saat ini adalah *CSS 3* yang memiliki banyak kelebihan dibanding versi sebelumnya seperti penambahan fitur *media-queries* dan penanganan font web [28].



### 2.14.3 *Java Script*

*Javascript* ialah suatu bahasa scripting yang digunakan sebagai fungsionalitas dalam membuat suatu web.[29]. *Javascript* adalah bahasa script yang biasa diletakkan bersama kode *HTML* untuk menentukan suatu aksi”. *Javascript* dikembangkan oleh *Netscape*, sebagai bahasa pemrograman ”sederhana” karena dapat digunakan untuk membuat aplikasi ataupun *Applet*. Namun dengan *Javascript* kita dapat membuat halaman web yang interaktif dan mudah [30].

### 2.14.4 *Hypertext Preprocessor (PHP)*

*PHP* pertama kali ditemukan pada tahun 1995 oleh seorang *Software Developer* bernama Rasmus Lerdorf. Ide awal *PHP* adalah ketika itu Rasmus ingin mengetahui jumlah pengunjung yang membaca resume online-nya. *Script* yang dikembangkan baru dapat melakukan dua pekerjaan, yakni merekam informasi *visitor*, dan menampilkan jumlah pengunjung dari suatu website. Dan sampai sekarang kedua tugas tersebut masih tetap populer digunakan oleh dunia web saat ini. Kemudian, dari situ banyak orang di milis mendiskusikan *script* buatan Rasmus Lerdorf, hingga akhirnya Rasmus mulai membuat sebuah *tool/script*, bernama *Personal Home Page (PHP)* [31].

*PHP Hypertext Preprocessor* adalah suatu bahasa *scripting* khususnya digunakan untuk *web server*. *PHP* memiliki sifat *server side scripting* sehingga untuk menjalankan *PHP* harus menggunakan *web server* [27].

### 2.14.5 *Structured Query Language (SQL)*

*Structured Query Language (SQL)* adalah bahasa yang digunakan untuk mengelola data pada *DBMS*. *SQL* biasanya berupa perintah sederhana yang berisi instruksi-instruksi untuk manipulasi data. Perintah *SQL* ini sering juga disingkat

dengan sebutan ‘*query*’. Berdasarkan diatas, dapat disimpulkan bahwa *Structured Query Language (SQL)* merupakan sebuah Bahasa yang berisi perintah-perintah yang dikhususkan untuk mengolah dan mengakses basis data. Berikut Perintah-Perintah yang ada dalam *SQL* : 1. *DDL (Data Definition Language)* *DDL* : *Create, Rename, Alter, Drop* 2. *DML (Data Manipulation Language)* *DML* : *Select, Insert, Update, Delete* 3. *DCL (Data Control Language)* *Dcl* : *Grant, Revoke* [32].

## 2.15 Alat Bantu Pemograman

### 2.15.1 *MySQL*

*MySQL* adalah salah satu aplikasi *DBMS* yang sudah sangat banyak digunakan oleh para pemrograman aplikasi web.[27]. *MySQL* adalah sebuah implementasi dari sistem manajemen basis data relasional (*RDBMS*) yang didistribusikan secara gratis. Setiap pengguna dapat secara bebas menggunakan *MySQL*, namun dengan batasan perangkat lunak tersebut tidak boleh dijadikan produk turunan yang bersifat komersial. *MySQL* sebenarnya merupakan turunan salah satu konsep utama dalam basis data yang telah ada sebelumnya *Structured Query Language (SQL)*. *SQL* adalah sebuah konsep pengoperasian basis data, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis [31].

*MySQL* merupakan *software RDBMS (server database)* yang dapat mengelola database dengan sangat cepat, dapat menampung data dalam jumlah sangat besar, dapat diakses oleh banyak pengguna (*multi-user*), dan dapat melakukan suatu proses secara sinkron atau berbarengan (*multithreaded*). Bahasa ini secara *defacto* merupakan bahasa standar yang digunakan dalam manajemen basis data *relasional*. *MySQL* diciptakan dinegara swedia oleh perusahaan *MySQL*

AB. Adapun masing-masing nama yang berjasa dalam menciptakan *MySQL* adalah David Axmark, Allan Larson, dan Micheal Monty Widenius. Perangkat lunak ini tersebar secara gratis karna memiliki license *GNU (General Public Lincense)*. Sampai sekarang, tercatat ada beberapa bahasa pemrograman yang cukup populer yang biasa bersinkronisasi dengan *MySQL*, seperti *C*, *C++*, *C#*, bahasa pemrograman *Eiffel*, bahasa pemrograman *Smaltalk*, *Java*, *Lisp*, *Perl*, *PHP*, *Phyton*, *Ruby*, *REALbasic*, dan *Tcl* [33].

*MySQL* adalah salah satu aplikasi *Database Management System (DBSM)* yang sudah sangat banyak digunakan oleh para pemrogram aplikasi web [29].

### 2.15.2 XAMPP

*XAMPP* merupakan paket *PHP* berbasis open source. Informasinya dapat diperoleh di website resminya: <http://www.apachefriends.com>. *XAMPP* membantu memudahkan dalam mengembangkan aplikasi berbasis *PHP*. *XAMPP* mengkombinasikan beberapa paket *software* berbeda kedalam satu paket. Adapun *lisensi* masing masing paket *software* tersebut dapat ditemukan didirektori *\xampp\licence*. *XAMPP* menyediakan antar muka *control panel* tersendiri yang dapat digunakan untuk menjalankan semua *service* (paket *software* pendukung) yang telah terinstal. Pada sistem operasi *windows*, *control panel* dapat diakses melalui menu *[Start]-[Program]-[Apachefriends]-[xampp]-[control xampp server panel]*. Pada *web server* (lokal komputer, tidak di *server internet* sesungguhnya) pada *XAMPP*, akan menyediakan satu *folder* kerja yang bernama *htdocs*. Pada paket ini, folder kerja tersebut dapat ditemukan pada *subfolder* *C:\.\XAMPP* (sesuai lokasi dimana menyimpan hasil instalasinya) [31].

### 2.15.3 Website

*Website* dapat diartikan sekumpulan halaman yang terdiri dari beberapa halaman yang berisi informasi dalam bentuk data digital baik berupa *text*, gambar, video, audio, dan animasi lainnya yang disediakan melalui jalur *koneksi internet*. *Website* atau sering disebut situs merupakan kumpulan halaman web yang dijalankan dari suatu alamat web *domain*. *Website* merupakan kumpulan halaman-halaman yang digunakan untuk menampilkan informasi *text*, gambar diam atau gerak, animasi, suara, dan atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, yang masing masing masing dihubungkan dengan jaringan-jaringan halaman [33].

### 2.15.4 Visual Studio Code

*Visual Studio Code* adalah kode *editor* sumber yang dikembangkan oleh *Microsoft* untuk *Windows*, *Linux* dan *macOS*. Ini termasuk dukungan untuk *debugging*, kontrol *git* yang tertanam dan *GitHub*, penyorotan *sintaksis*, penyelesaian kode cerdas, *snippet*, dan *refactoring* kode. Ini sangat dapat disesuaikan, memungkinkan pengguna untuk mengubah tema, pintasan *keyboard*, preferensi, dan menginstal ekstensi yang menambah fungsionalitas tambahan [34].

### 2.15.5 Model View Controller (MVC)

*Model View Controller (MVC)* merupakan sebuah konsep yang diperkenalkan oleh penemu *Smalltalk* (*Tryangve Reenkaug*) untuk meng-*enkapsulasi* data bersama pemrosesan (*model*), mengisolasi dari proses manipulasi (*controller*) dan tampilan (*view*) untuk direpresentasikan pada tampilan muka. *MVC* dibagi menjadi tiga lapisan, yaitu : a) Model Model, digunakan untuk mengelola informasi dan memberitahu pengamat ketika ada perubahan informasi. Model

mengandung data dan fungsi yang berhubungan dengan pemrosesan data. b) *View* *View*, bertanggungjawab untuk pemetaan grafis ke sebuah perangkat. c) *Controller* *Controller*, menerima input dari pengguna dan mendistribusikan model dan *view* untuk melakukan aksi berdasarkan masukan tersebut. Sehingga *controller* bertanggungjawab untuk pemetaan aksi pengguna akhir terhadap respon aplikasi [35].

#### **2.15.6 Framework Laravel**

*Laravel* merupakan salah satu dari beberapa macam *framework* bahasa pemrograman *PHP* yang tersedia di *internet*. *Framework* merupakan sekumpulan library atau fungsi – fungsi yang telah disediakan untuk membantu mempermudah dalam pengembangan suatu sistem [9].

*Laravel* adalah “*framework open source PHP* berbasis web *gras* yang dibuat oleh *Taylor Otwell* dan ditujukan untuk mengembangkan aplikasi web mengikuti *model-view-controller (MVC)* atau pola arsitektur. Beberapa fitur dari *Laravel* adalah pengembangan sistem modul modul yang dapat di manajemen, mengenalkan cara yang berbeda untuk mengakses *database relasional*, *utilitas* yang membantu dalam penyebaran aplikasi dan pemeliharaan yang mudah” [30].

### **2.16 Pengujian Aplikasi Dan Metode**

#### **2.16.1 Black Box Testing**

*Black Box Testing* adalah metode pengujian perangkat lunak yang tes fungsionalitas dari aplikasi yang bertentangan dengan struktur internal atau kerja. Pengetahuan khusus dari kode aplikasi / struktur internal dan pengetahuan pemrograman pada umumnya tidak diperlukan. Tes ini dapat menjadi fungsional atau non-fungsional, meskipun biasanya fungsional. Perancang uji memilih input

yang *valid* dan tidak *valid* dan menentukan *output* yang benar. Metode uji dapat diterapkan pada semua tingkat pengujian perangkat lunak: unit, integrasi, fungsional, sistem dan penerimaan [36].

*BlackBox Testing* sendiri merupakan pengujian yang dilakukan hanya mengamati hasil eksekusi melalui data uji dan memeriksa fungsional dari perangkat lunak. Pengujian *blackbox* ini menitik beratkan pada fungsi sistem. Metode *Blackbox Testing* adalah sebuah metode yang dipakai untuk menguji sebuah *software* tanpa harus memperhatikan detail *software*. Pengujian ini hanya memeriksa nilai keluaran berdasarkan nilai masukan masing-masing. Tidak ada upaya untuk mengetahui kode program apa yang output pakai. Pengujian ini diperlukan untuk mengetahui program tersebut berjalan sesuai dengan yang dibutuhkan oleh perusahaan [37].

Metode *BlackBox Testing* adalah pengujian yang melihat hasil eksekusi melalui data uji dan memastikan fungsi dari *software*. Metode *Black Box Testing* mempunyai beberapa teknik pengujian, yaitu *Sample Testing*, *Boundary Value Analysis*, *Equivalence Partitions* dan lain-lain. Pengujian *blackbox* pengujian perangkat lunak yang tanpa perlu memperlihatkan hasil detail perangkat lunak [38].

#### **2.16.2 Metode BERT (*Bidirectional Encoder Representations from Transformers*)**

*BERT* merupakan model bahasa berbasis arsitektur *Transformer* yang dilatih secara *bidirectional* untuk memahami konteks kata dalam sebuah kalimat. Berbeda dengan model tradisional yang hanya membaca teks secara satu arah (kiri ke kanan atau kanan ke kiri), *BERT* memanfaatkan *self-attention* untuk menangkap hubungan antar kata dalam seluruh kalimat secara lebih mendalam [43].

*BERTScore* diperkenalkan oleh Zhang et al. (2019) sebagai metrik evaluasi teks generatif yang mengukur kemiripan semantik antar dua teks, yaitu *candidate* (hasil sistem) dan *reference* (teks acuan) [44].

Tidak seperti metrik tradisional (*BLEU*, *ROUGE*) yang hanya menghitung kesamaan n-gram, *BERTScore* menggunakan *embedding* kontekstual dari *BERT* untuk menghitung kesamaan antar token. Dengan demikian, meskipun kata berbeda tetapi memiliki makna serupa, *BERTScore* tetap memberikan nilai kesamaan yang tinggi. Secara matematis, *BERTScore* mendefinisikan *Precision*, *Recall*, dan *F1* sebagai berikut [44]:

$$Precision = \frac{1}{|C|} \sum_{x \in C} \max_{y \in R} \phi(x, y)$$

$$Recall = \frac{1}{|R|} \sum_{y \in R} \max_{x \in C} \phi(x, y)$$

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

Dengan keterangan:

- $CCC$  = himpunan token dari kalimat *candidate* (judul dari Gemini).
- $RRR$  = himpunan token dari kalimat *reference* (judul dari input form mahasiswa).
- $\phi(x,y)$  = skor kesamaan kosinus (*cosine similarity*) antara *embedding token xxx dan yyy*.

Interpretasi:

- *Precision* → seberapa relevan kata hasil sistem (*candidate*) terhadap referensi.
- *Recall* → seberapa banyak kata referensi berhasil ditemukan dalam hasil sistem.
- *F1* → keseimbangan antara *precision* dan *recall*, memberikan gambaran kualitas keseluruhan.

Lebih sensitif terhadap makna dibanding *BLEU/ROUGE*.

Bisa mendeteksi kesamaan semantik meskipun kata berbeda (contoh: “*sistem informasi*” dan “*SDLC*”). Telah terbukti memiliki korelasi lebih tinggi dengan penilaian manusia dalam tugas evaluasi *NLP* [44].

Namun, penelitian lain menunjukkan bahwa *BERTScore* juga memiliki keterbatasan. Misalnya, Hanna (2021) menemukan bahwa skor bisa bias pada panjang kalimat dan terkadang terlalu permisif dalam menilai teks yang berbeda secara sintaksis tetapi masih semantik serupa [45].

## 2.17 Penelitian Terdahulu

**Table 2. 1 Penelitian Terdahulu**

Judul	Peneliti	Objek	Objek
Penerapan Teknologi Ai Dari <i>GEMINI</i> Untuk Meningkatkan Layanan	Marthalena1, Nela Puspita Florensia2 , Yulia Patimah3 , Viktor Handrianus	Aplikasi Cozybook	Hasil penelitian ini menunjukkan bahwa pendekatan prototyping dalam pengembangan aplikasi perpustakaan digital mobile dapat



Peminjaman Buku Online Pada Aplikasi Cozybook	Pranatawijaya <sup>4</sup> , Nova Noor Kamala Sari		menghasilkan sistem yang lebih responsif dan sesuai dengan kebutuhan pengguna, serta memanfaatkan teknologi chatbot AI untuk meningkatkan kualitas layanan
Memperkaya Pemrograman Web Sistem Kasir Dengan Teknologi Ai: Implementasi <i>API GEMINI</i>	Novia Fitriani, Nadya Revelin Putri, Ardhy Dwiwicaksono Viktor Handrianus Pranatawijaya, Nova Noor Kamala Sari	Web Sistem Kasir	Dari hasil pembahasan, dapat disimpulkan bahwa penerapan <i>API GEMINI</i> pada sistem kasir coffee shop membawa dampak positif yang signifikan. Integrasi <i>API GEMINI</i> memungkinkan otomatisasi pembuatan deskripsi produk, mengurangi beban kerja manual bagi kasir dan meningkatkan efisiensi operasional. Selain itu, dengan menggunakan algoritma kecerdasan buatan, <i>API GEMINI</i> menghasilkan deskripsi produk yang konsisten, relevan, menarik, dan rata-rata waktu yang singkat yaitu 2,47s akan membantu meningkatkan pengalaman pelanggan dengan menyediakan informasi yang jelas dan mudah dipahami.

Penerapan <i>GEMINI</i> Ai Dalam Pembuatan Deskripsi Produk E-Commerce	Atong Nazarius, Ferry Saputra, Nova Noor Kamala Sari, Viktor Handrianus Pranatawijaya	E-Commerce	Berdasarkan penelitian yang dilakukan dapat disimpulkan bahwa penerapan <i>GEMINI</i> AI memberikan sebuah solusi yang inovatif untuk melakukan proses generate sebuah deskripsi produk pada e-commerce. Dimana dalam pembuatan deskripsi produk pada suatu e-commerce memberikan dampak positif dalam meningkatkan interaksi antara bisnis dan pengguna. Melalui integrasi <i>API GEMINI</i> AI, proses pembuatan deskripsi produk menjadi lebih efisien, menghemat waktu serta tenaga.
Implementasi Fitur Deskripsi Produk Berbasis <i>API GEMINI</i> Dalam Pengembangan E-Commerce Berbasis Mobile Menggunakan Framework Flutter	Cindi Lusia Kiareni, Cinda Sorisa, Nova Noor Kamala Sari, Viktor Handrianus Pranatawijaya	E-Commerce	Berdasarkan hasil dan pembahasan diatas dapat disimpulkan aplikasi E-commerce berbasis mobile yang dikembangkan menggunakan Flutter mengimplementasikan berbagai fitur yang dapat mempermudah pengguna dalam berbelanja online. Integrasi <i>API GEMINI</i> dari <i>Google</i> Generative AI untuk deskripsi produk otomatis terbukti efektif

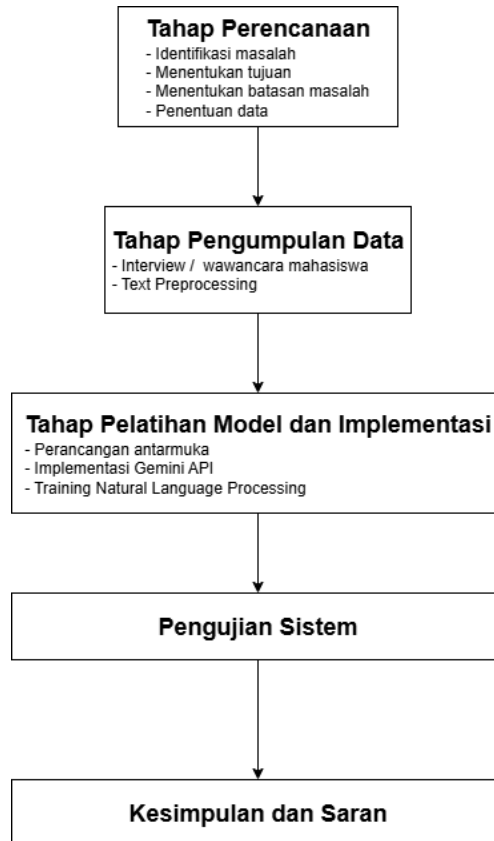
Visualisasi Data Cyber Security Attack Dengan Fitur Prediksi Serangan Dan Mitigasi Risiko: Perspektif Generative <i>GEMINI</i> AI	Purwanti	Prediksi Serangan Dan Mitigasi Risiko Cyber Security Attack	Hasil penelitian menunjukkan bahwa serangan siber dominan pada tahun tersebut terdiri dari jenis Malware, Intrusion, dan DDoS. Aplikasi dashboard yang dikembangkan mampu menyajikan visualisasi data yang informatif melalui grafik interaktif, peta, dan tabel. Integrasi teknologi <i>GEMINI</i> AI memungkinkan prediksi serangan yang mendukung tindakan pencegahan berdasarkan rekomendasi yang dihasilkan oleh AI. Aplikasi ini juga dirancang untuk memberikan respons yang sesuai dengan kebutuhan pengguna dan membantu dalam memahami data secara efektif. Pengujian aplikasi menunjukkan performa yang baik, terutama dari segi fungsionalitas dan konsistensi.
---	----------	---	---

## BAB 3

### METODOLOGI PENELITIAN

#### 3.1 Tahapan Penelitian

Tahapan yang ditempuh pada penelitian ini dapat dilihat pada Gambar 3.1:



**Gambar 3. 1 Metodologi Penelitian**

Penjelasan dari metodologi penelitian pada gambar 3.1 dapat dilihat pada penjelasan berikut:

#### 3.2 Tahap Perencanaan

Tahap ini merupakan Langkah yang harus dilakukan agar tujuan penelitian menjadi lebih jelas dan terarah.

### 3.2.1 Identifikasi Masalah

Identifikasi masalah di dapat melalui pengamatan secara langsung terhadap objek ini dan dilakukan dengan maksud agar dapat mengetahui secara jelas permasalahan yang terjadi di lapangan atau lokasi. Setelah diidentifikasi masalah bahwa mahasiswa seringkali menghadapi kesulitan dalam menemukan judul penelitian yang unik dan berbasis inovasi terkini karena terbatasnya akses terhadap referensi yang terintegrasi serta minimnya alat pendukung berbasis teknologi. Hingga saat ini, proses pencarian judul yang masih manual atau mengandalkan *database* statis berpotensi menyebabkan duplikasi topik, ketidakefisienan waktu, dan kurangnya inspirasi bagi mahasiswa.

### 3.2.2 Menentukan Tujuan

Menentukan target yang ingin dicapai pada penelitian. Tujuan dari penelitian ini adalah untuk menerapkan integrasi *Natural Language Processing (NLP)* Dan *Gemini Api* dalam rekomendasi judul magang dan skripsi guna meningkatkan kualitas, relevansi, serta efektivitas dalam mendukung proses pemilihan judul berdasarkan analisis data dan tren terkini.

### 3.2.3 Menentukan Batasan Masalah

Bertujuan agar penelitian berkonsentrasi pada cakupan objek penelitian dan tidak keluar dari cakupan yang telah ditentukan.

### 3.2.3 Penentuan Data

Untuk mendukung tujuan penelitian yang berfokus pada penerapan integrasi *Natural Language Processing (NLP)* dan *Gemini Api* dalam rekomendasi judul magang dan skripsi, data yang dikumpulkan harus relevan dengan permasalahan

yang diidentifikasi. Oleh karena itu, penentuan data diarahkan kepada pengumpulan sumber-sumber yang mencerminkan tren, inovasi, dan keberagaman topik penelitian terkini.

Mengumpulkan data dari jurnal ilmiah, konferensi, dan publikasi riset yang membahas perkembangan teknologi dan tren penelitian terbaru. Hal ini untuk memastikan bahwa rekomendasi judul tidak hanya unik, tetapi juga relevan dengan kebutuhan dan perkembangan zaman.

### **3.3 Tahap Pengumpulan Data**

Pada tahap ini dilakukan pengumpulan data yang relevan untuk mendukung integrasi *Natural Language Processing (NLP)* dan *Gemini Application Programming Interface (API)* dalam sistem rekomendasi judul magang dan skripsi. Data dikumpulkan melalui wawancara dengan ketua program studi dan mahasiswa, penyebaran kuesioner menggunakan *Google Form*, serta tahap *text pre-processing* untuk menyiapkan data agar siap digunakan dalam sistem.

#### **3.3.1 Wawancara**

Wawancara dilakukan dengan ketua program studi dan mahasiswa untuk memperoleh informasi mendalam mengenai kendala yang sering dihadapi dalam menentukan judul penelitian. Dari hasil wawancara, diketahui bahwa keterbatasan referensi terintegrasi dan proses pencarian manual menjadi faktor utama yang menyebabkan mahasiswa kesulitan dalam menemukan judul penelitian yang relevan dan orisinal.

### 3.3.2 Kuesioner (*Google Form*)

Selain wawancara, dilakukan penyebaran kuesioner kepada mahasiswa melalui *Google Form*. Tujuannya adalah untuk menjangkau data lebih luas terkait pengalaman, kesulitan, serta kebutuhan mahasiswa dalam menentukan judul penelitian. Data dari kuesioner digunakan sebagai dasar dalam perancangan sistem rekomendasi agar sesuai dengan kebutuhan nyata pengguna.

### 3.3.3 Text Pre-processing

Tahap *text pre-processing* dilakukan untuk menyiapkan data teks sebelum diproses oleh sistem. Proses ini meliputi pembersihan data dari entitas yang tidak relevan, normalisasi teks, serta penyusunan ulang informasi agar sesuai dengan kebutuhan analisis *Natural Language Processing (NLP)*. Dengan langkah ini, data yang digunakan menjadi lebih terfokus dan berkualitas untuk mendukung pelatihan model rekomendasi judul.

## 3.4 Tahap Pelatihan Model dan Implementasi

Pada tahap ini dilakukan proses pelatihan model kecerdasan buatan dan implementasi sistem untuk membangun aplikasi rekomendasi judul magang dan skripsi berbasis *NLP* dengan integrasi *Gemini API*. Tahapan ini meliputi perancangan antarmuka, implementasi *Gemini API*, serta pelatihan *Natural Language Processing (NLP)*.

### 3.4.1 Perancangan Antarmuka

Antarmuka sistem dirancang agar intuitif, sederhana, dan mudah digunakan mahasiswa dalam proses pencarian judul penelitian. Komponen utama seperti formulir input, tombol, dan navigasi menu disusun secara responsif sehingga dapat diakses melalui berbagai perangkat. Umpan balik dari pengguna

dimanfaatkan untuk memperbaiki desain selama tahap *prototyping*, sementara skema warna dan tipografi dipilih untuk meningkatkan kenyamanan *visual* dan keterbacaan. Hasil desain yang optimal kemudian diintegrasikan ke dalam sistem sebagai sarana utama interaksi antara pengguna dan aplikasi.

### **3.4.2 Implementasi *Gemini API***

*Gemini API* digunakan sebagai pusat penelusuran untuk memperoleh rekomendasi judul penelitian yang relevan dengan tren terbaru. Integrasi *API* dilakukan agar bekerja selaras dengan modul *NLP* sehingga sistem mampu memahami konteks input mahasiswa sebelum menghasilkan rekomendasi. Fokus utama implementasi meliputi kecepatan respon, akurasi hasil, serta keamanan pertukaran data dengan menerapkan autentikasi dan enkripsi. Dokumentasi *API* disiapkan untuk mendukung pemeliharaan serta pengembangan di masa depan.

### **3.4.3 Pelatihan *Natural Language Processing (NLP)***

Pelatihan *NLP* dilakukan dengan memanfaatkan data yang diperoleh dari wawancara kaprodi, mahasiswa, serta kuesioner *Google Form*, yang sebelumnya telah melalui tahap *text pre-processing*. Proses ini bertujuan agar sistem mampu mengenali pola kata, istilah penelitian, dan keterkaitan semantik antar topik. Model *NLP* kemudian dipadukan dengan *Gemini API* sehingga sistem dapat menghasilkan rekomendasi judul yang tidak hanya relevan secara teknis, tetapi juga sesuai dengan kebutuhan dan minat mahasiswa..

## **3.5 Pengujian Aplikasi**

Pada tahap ini dilakukan pengujian aplikasi untuk memastikan fungsionalitas dan kualitas sistem yang dikembangkan. Metode yang digunakan adalah *blackbox testing* dan evaluasi kualitas rekomendasi menggunakan



*BERTScore*. *Blackbox testing* berfokus pada pengujian fungsionalitas eksternal aplikasi tanpa memperhatikan detail internal kode atau struktur sistem. Pengujian dilakukan dengan memberikan input berupa data minat penelitian, bidang keilmuan, serta *metode* yang dimasukkan mahasiswa melalui form aplikasi. *Output* yang dihasilkan berupa rekomendasi judul dari sistem. Hasil pengujian digunakan untuk memastikan setiap fitur berjalan sesuai kebutuhan pengguna, termasuk validasi *input*, kecepatan respon, serta konsistensi hasil rekomendasi. Selain menguji fungsionalitas, dilakukan pula evaluasi kualitas rekomendasi menggunakan *metode BERTScore*. Metode ini digunakan untuk mengukur kesesuaian makna semantik antara judul yang dimasukkan mahasiswa dengan judul yang direkomendasikan sistem melalui *Gemini API*. Jika judul berbeda secara redaksi tetapi memiliki makna yang sama, maka nilai kesesuaian tetap tinggi. Evaluasi ini bertujuan untuk memastikan bahwa rekomendasi yang diberikan sistem tidak hanya berfungsi secara teknis, tetapi juga relevan, kontekstual, dan sesuai dengan kebutuhan akademik mahasiswa.

### **3.6 Kesimpulan Dan Saran**

Pada bagian ini, disajikan kesimpulan dari hasil penelitian yang telah dilakukan terkait Integrasi *Natural Language Processing (NLP)* Dan *Gemini Application Programming Interface (API)* Untuk Rekomendasi Judul Magang Dan Skripsi. Selain itu, diberikan pula saran-saran untuk pengembangan selanjutnya berdasarkan temuan yang ditemukan selama penelitian. Kesimpulan ini bertujuan untuk merangkum hasil penelitian secara komprehensif, sedangkan saran-saran diharapkan dapat memberikan arah bagi penelitian mendatang untuk meningkatkan kualitas dan efektivitas aplikasi rekomendasi judul magang dan skripsi.