Pengembangan Sistem Pembelajaran Interaktif Berbasis *Web* untuk Cipta dan Evaluasi Soal Otomatis Menggunakan OpenAI

TUGAS AKHIR

Diajukan untuk Memenuhi Persyaratan Akademik dalam

Menyelesaikan Pendidikan pada Program Studi

S1 Teknik Informatika Universitas Kristen Maranatha

Oleh

**Laurentius Gusti Ontoseno Panata Yudha**

**2172028**



**PROGRAM STUDI S1 TEKNIK INFORMATIKA**

**FAKULTAS TEKNOLOGI DAN REKAYASA CERDAS**

**UNIVERSITAS KRISTEN MARANATHA**

**BANDUNG**

**2024**

# LEMBAR PENGESAHAN

**Pengembangan Sistem Pembelajaran Interaktif Berbasis Web untuk Cipta dan Evaluasi Soal Otomatis Menggunakan OpenAI**

**Dengan ini, saya menyatakan bahwa  
isi CD ROM Laporan Penelitian sama dengan hasil revisi akhir**

**Bandung, Tanggal Bulan Tahun**

**(Laurentius Gusti Ontoseno Panata Yudha)**

**(2172028)**

**Menyetujui,**

|  |  |
| --- | --- |
| **Pembimbing I** | **Pembimbing II (Jika Ada)** |
|  |  |
| **Dr. Hapnes Toba, M.Sc.** | **Hendra Bunyamin, S.Si** |
| **NIK: 710004** | **NIK: 720001** |
|  |  |
| **Penguji I** | **Penguji II (Jika Ada)** |
|  |  |
| **Prof.Dr.Ir.Mewati Ayub, M.T** | **Oscar Karnalim, S.T., M.T., Ph.D.** |
| **NIK: 720140** | **NIK: 720309** |

**Mengetahui,**

**Ketua Program Studi Teknik Informatika**

**Robby Tan, S.T., M. Kom**

**NIK: 720307**

# PERNYATAAN ORISINALITAS LAPORAN PENELITIAN

Dengan ini, saya yang bertanda tangan di bawah ini:

|  |  |  |
| --- | --- | --- |
| Nama | : | Laurentius Gusti Ontoseno Panata Yudha |
| NRP | : | 2172028 |
| Fakultas/ Program Studi | : | Teknologi dan Rekayasa Cerdas / Teknik Informatika |

Menyatakan bahwa laporan penelitian ini adalah benar merupakan hasil karya saya sendiri dan bukan duplikasi dari orang lain.

Apabila pada masa mendatang diketahui bahwa pernyataan ini tidak benar adanya, saya bersedia menerima sanksi yang diberikan dengan segala konsekuensinya.

Demikian pernyataan ini saya buat.

Bandung, 16 Oktober 2024

A black background with a black square

Description automatically generated with medium confidence

Laurentius Gusti Ontoseno Panata Yudha

NRP: 2172028

# PERNYATAAN PUBLIKASI LAPORAN PENELITIAN

Saya yang bertanda tangan di bawah ini:

|  |  |  |
| --- | --- | --- |
| Nama | : | Laurentius Gusti Ontoseno Panata Yudha |
| NRP | : | 2172028 |
| Fakultas/ Program Studi | : | Teknologi dan Rekayasa Cerdas / Teknik Inomatika |

Dengan ini, saya menyatakan bahwa:

1. Demi perkembangan ilmu pengetahuan, saya menyetujui untuk memberikan kepada Universitas Kristen Maranatha Hak Bebas Royalti non eksklusif (*Non* *Exclusive* *Royalty* *Free* *Right*) atas laporan penelitian saya yang berjudul Pengembangan Sistem Pembelajaran Interaktif Berbasis *Web* untuk Cipta dan Evaluasi Soal Otomatis Menggunakan OpenAI.
2. Universitas Kristen Maranatha Bandung berhak menyimpan, mengalihmediakan/ mengalihformatkan, mengelola dalam bentuk pangkalan data (*database*), mendistribusikannya, serta menampilkannya dalam bentuk *softcopy* untuk kepentingan akademis tanpa perlu meminta izin dari saya selama tetap mencantumkan nama saya sebagai penulis/ pencipta.
3. Saya bersedia dan menjamin untuk menanggung secara pribadi tanpa melibatkan pihak Universitas Kristen Maranatha Bandung, segala bentuk tuntutan hukum yang timbul atas pelanggaran hak cipta dalam karya ilmiah saya ini.

Demikian pernyataan ini saya buat dengan sebenarnya dan untuk dapat dipergunakan sebagaimana mestinya.

Bandung, 16 Oktober 2024

A black background with a black square

Description automatically generated with medium confidence

Laurentius Gusti Ontoseno Panata Yudha

NRP:2172028

# PRAKATA

Puji syukur saya panjatkan ke hadirat Tuhan Yang Maha Esa atas limpahan rahmat dan karunia-Nya sehingga saya dapat menyelesaikan tugas akhir ini dengan judul "Pengembangan Sistem Pembelajaran Interaktif Berbasis *Web* untuk Cipta dan Evaluasi Soal Otomatis Menggunakan OpenAI".

Tugas akhir ini merupakan salah satu syarat untuk memperoleh gelar Sarjana pada Program Studi Teknik Informatika di Universitas Kristen Maranatha. Penulisan karya ilmiah ini menjadi bagian dari proses pembelajaran yang harus saya lalui untuk menyelesaikan studi di tingkat sarjana.

Karya ilmiah ini bertujuan untuk mengembangkan sistem pembelajaran berbasis web yang dapat memfasilitasi pembuatan dan evaluasi soal secara otomatis, dengan memanfaatkan teknologi kecerdasan buatan dari OpenAI. Pengembangan sistem ini diharapkan dapat membantu para pendidik dalam menciptakan soal dengan lebih efisien.

Dalam penyusunan tugas akhir ini, saya sangat berterima kasih kepada dosen pembimbing, Dr. Hapnes Toba, M.Sc., dan Hendra Bunyamin, S.Si., M.T., yang telah memberikan banyak arahan, bimbingan, serta bantuan yang sangat berarti dalam proses penyusunan tugas akhir ini. Tanpa dukungan dan nasihat dari beliau berdua, saya tidak akan dapat menyelesaikan karya ini dengan baik.

Saya menyadari bahwa tugas akhir ini masih jauh dari sempurna. Oleh karena itu, dengan rendah hati, saya terbuka terhadap segala kritik dan saran yang konstruktif demi perbaikan dan penyempurnaan karya ini di masa mendatang.

Saya berharap, hasil dari tugas akhir ini dapat memberikan manfaat, baik bagi pengembangan ilmu pengetahuan maupun bagi siapa saja yang tertarik di bidang teknologi pendidikan.

Bandung, 16 Oktober 2024

A black background with a black square

Description automatically generated with medium confidence

Laurentius Gusti Ontoseno Panata Yudha

# ABSTRAK

(Style: abstrak) Jumlah kata pada abstrak adalah 150-200 kata. Gunakanlah style abstrak. Pada prinsipnya, abstrak akan memberikan rangkuman dari laporan Anda secara keseluruhan. Jika seseorang membaca abstrak, maka pembaca akan tahu apa keseluruhan isi laporan dari latar belakang sampai simpulan dan saran. Abstrak disarankan untuk memuat hal-hal berikut ini yaitu pokok masalah yang dibahas (Mengapa melakukan KP/ TA dengan topik ini?); tujuan pembahasan (Apa yang ingin dicapai?); teori yang digunakan, sumber data; metode dan teknik penelitian (metode adalah cara menganalisis/ memecahkan sedangkan teknik penelitian adalah cara pengumpulan data); temuan ilmiah/ jawaban pembahasan/ hasil dan simpulan yang dicapai.

Kata kunci: frase 1, frase 2, dst. (maksimum 6, urutkan abjad)

# ABSTRACT

*(Style: Abstract) Jumlah kata pada abstract adalah 150-200 kata. Abstract Bahasa Inggris harus dibuat sesuai dengan abstrak Bahasa Indonesia-nya. Harap menggunakan English grammar yang baik dan benar. Tidak disarankan menggunakan Google Translate karena belum tentu sesuai dengan kalimat yang dimaksud. Namun Google Translate bisa membantu Anda untuk mencari kata-kata yang Anda tidak tahu Bahasa Inggris-nya.*

*Keywords: phrase 1, phrase 2, etc. (max 6 phrases, sorted alphabetically)*

# DAFTAR ISI

[1. LEMBAR PENGESAHAN i](#_Toc179925319)

[2. PERNYATAAN ORISINALITAS LAPORAN PENELITIAN ii](#_Toc179925320)

[3. PERNYATAAN PUBLIKASI LAPORAN PENELITIAN iii](#_Toc179925321)

[4. PRAKATA iv](#_Toc179925322)

[5. ABSTRAK v](#_Toc179925323)

[6. ABSTRACT vi](#_Toc179925324)

[7. DAFTAR ISI vii](#_Toc179925325)

[8. DAFTAR GAMBAR viii](#_Toc179925326)

[9. DAFTAR TABEL ix](#_Toc179925327)

[10. DAFTAR SINGKATAN x](#_Toc179925328)

[11. DAFTAR ISTILAH xi](#_Toc179925329)

[BAB 1 PENDAHULUAN 1](#_Toc179925330)

[1.1 Latar Belakang 1](#_Toc179925331)

[1.2 Rumusan Masalah 2](#_Toc179925332)

[1.3 Tujuan Pembahasan 2](#_Toc179925333)

[1.4 Ruang Lingkup 3](#_Toc179925334)

[1.5 Sumber Data 4](#_Toc179925335)

[1.6 Sistematika Penyajian 5](#_Toc179925336)

[BAB 2 KAJIAN TEORI 7](#_Toc179925337)

[BAB 3 ANALISIS DAN RANCANGAN SISTEM 12](#_Toc179925338)

[3.1 Gambaran Umum 12](#_Toc179925341)

[3.2 *Stakeholder* 13](#_Toc179925342)

[3.3 Otomatisasi Proses Pembuatan dan Evaluasi Soal 13](#_Toc179925343)

[3.4 Rancangan Antarmuka 14](#_Toc179925344)

[3.5 Rancangan Penyimpanan Data 17](#_Toc179925345)

[3.6 Alur Sistem 18](#_Toc179925346)

[3.7 Rancangan Sistem 19](#_Toc179925347)

[12. DAFTAR PUSTAKA 21](#_Toc179925348)

# DAFTAR GAMBAR

[Gambar 3.2.1 *Use Case Diagram* 13](#_Toc179921724)

[Gambar 3.5.1 *Entity Relationship Diagram* 17](#_Toc179921725)

[Gambar 3.6.1 *Activity Diagram* 18](#_Toc179921726)

# DAFTAR TABEL

# DAFTAR SINGKATAN

|  |  |
| --- | --- |
| ERD | *Entity Relationship Diagram* |
| JSON | JavaScript *Object Notation* |
| LLM | *Large Language Model* |
| MVC | *Model View Controller* |
| PDF | *Portable Document Format* |
| REST API | *Representational State Transfer Application Programming Interface* |

# DAFTAR ISTILAH

|  |  |
| --- | --- |
| *treshold* | Nilai atau titik tertentu di mana suatu perubahan atau tindakan terjadi. |
| *Framework* | Struktur atau panduan yang mempermudah pengembangan sistem, aplikasi, atau proses tertentu dengan menyediakan komponen, aturan, dan alur kerja yang sudah terdefinisi. |

# PENDAHULUAN

## Latar Belakang

Dalam era digitalisasi yang semakin berkembang pesat, teknologi telah menjadi elemen penting dalam transformasi sistem pembelajaran. Institusi pendidikan kini dituntut untuk menghadirkan metode pembelajaran yang lebih interaktif dan efisien, terutama dalam hal evaluasi dan penyampaian materi kepada mahasiswa ​[1], [2].

Salah satu tantangan utama dalam pendidikan adalah bagaimana mengevaluasi pemahaman mahasiswa secara efektif, terutama di kelas berskala besar yang dapat menguras waktu dan tenaga jika dilakukan secara manual [3].

Pengembangan sistem otomatis yang mengintegrasikan teknologi *Automated Question Generation* (AQG) dan penilaian otomatis menjadi solusi yang sangat dibutuhkan. Penelitian terbaru telah menunjukkan bahwa sistem AQG mampu menghasilkan pertanyaan yang relevan secara otomatis berdasarkan teks dari sumber pendidikan, seperti PDF, dan membantu dosen dalam mempercepat proses pembuatan soal ​[1], [4].

Sistem ini, seperti yang dikembangkan dalam proyek QGen, mampu mengekstraksi konten dari teks dan menghasilkan pertanyaan pilihan ganda lengkap dengan jawaban yang benar dan pengalih perhatian (*distractor*) yang valid [4].

Selain itu, penggunaan model kecerdasan buatan seperti GPT-4 dalam penilaian jawaban mahasiswa semakin memperkaya pengalaman pembelajaran. Model ini mampu menilai jawaban mahasiswa berdasarkan kesamaan konteks dengan jawaban yang diharapkan, menggunakan metrik seperti *Exact Match* dan F1-*score* untuk menentukan tingkat kesesuaian jawaban ​[2], [3]. Penelitian dalam konteks pendidikan medis menunjukkan bahwa penilaian berbasis LLM ini dapat memberikan umpan balik yang cepat dan akurat, sehingga memudahkan dosen untuk memantau perkembangan mahasiswa secara lebih efisien [3].

Dengan memanfaatkan framework seperti Laravel dan Flask, platform pembelajaran interaktif berbasis *web* dapat dikembangkan untuk mengintegrasikan kemampuan generasi pertanyaan dan penilaian otomatis ini. Sistem ini tidak hanya memudahkan dosen dalam pengelolaan soal dan evaluasi, tetapi juga memberikan pengalaman belajar yang lebih personal dan adaptif bagi mahasiswa, sehingga mahasiswa dapat mendapatkan umpan balik secara *real-time* dan memperbaiki pemahamannya terhadap materi ​[2], [4].

## Rumusan Masalah

Untuk mencapai tujuan yang diinginkan, fokus harus diberikan pada rumusan masalah yang perlu diselesaikan. Berikut adalah rumusan masalah yang telah disusun berdasarkan studi kasus pada topik ini:

1. Bagaimana mengembangkan sistem pembelajaran interaktif berbasis *web* yang dapat secara otomatis menghasilkan pertanyaan dari materi pembelajaran PDF menggunakan teknologi OpenAI untuk membantu dosen?
2. Bagaimana merancang sistem penilaian otomatis yang dapat mengevaluasi kesamaan konteks antara jawaban mahasiswa dengan jawaban yang benar, serta menggunakan *treshold* untuk menentukan tingkat pemahaman mahasiswa?
3. Bagaimana sistem dapat menghasilkan pertanyaan serupa secara otomatis ketika mahasiswa belum mencapai pemahaman yang memadai, dan memberikan panduan halaman untuk membantu mahasiswa menemukan jawaban yang benar?

## Tujuan Pembahasan

Tujuan dari pengembangan sistem *web* ini adalah untuk membantu meningkatkan efisiensi dalam proses pembelajaran, khususnya dalam pembuatan soal dan penilaian, serta untuk menguji tingkat pemahaman mahasiswa terhadap materi. Berdasarkan hal tersebut, berikut adalah tujuan yang ingin dicapai:

1. Mengembangkan sistem pembelajaran interaktif berbasis web yang dapat secara otomatis menghasilkan pertanyaan dari materi pembelajaran PDF menggunakan teknologi OpenAI untuk memudahkan dosen dalam pembuatan soal.
2. Merancang sistem penilaian otomatis yang mampu mengevaluasi kesamaan konteks antara jawaban mahasiswa dengan jawaban yang benar, serta menggunakan *treshold* untuk menentukan tingkat pemahaman mahasiswa terhadap materi.
3. Mengembangkan fitur sistem yang dapat menghasilkan pertanyaan serupa secara otomatis ketika mahasiswa belum mencapai pemahaman yang memadai, serta memberikan panduan halaman untuk membantu mahasiswa menemukan jawaban yang benar.

## Ruang Lingkup

Dalam tugas akhir ini, sistem pembelajaran interaktif berbasis *web* dikembangkan untuk membantu dosen dalam menghasilkan dan mengevaluasi pertanyaan secara otomatis menggunakan teknologi OpenAI. Ruang lingkup penelitian ini dibatasi pada beberapa aspek sebagai berikut:

1. **Aplikasi Berbasis *Web***

Sistem yang dikembangkan merupakan aplikasi berbasis web yang dapat diakses melalui *browser*, tanpa perlu menginstal aplikasi tambahan di perangkat pengguna.

1. ***Framework* Laravel 10**

Bagian *frontend* dan *backend* dari sistem ini dikembangkan menggunakan *framework* Laravel versi 10, yang memberikan struktur MVC untuk memudahkan pengembangan dan pengelolaan aplikasi.

1. **Flask (Python)**

Bagian *backend* untuk penilaian otomatis dan integrasi dengan OpenAI menggunakan Flask sebagai *framework* berbasis Python. Flask dipilih karena kesederhanaannya dan kemampuannya untuk diintegrasikan dengan model berbasis Python.

1. **API Integrasi dengan OpenAI**

Sistem ini terintegrasi dengan API OpenAI untuk menghasilkan pertanyaan otomatis berdasarkan materi pembelajaran yang diunggah dalam bentuk PDF. Integrasi ini memungkinkan otomatisasi proses pembuatan soal.

1. ***Prompting* OpenAI**

Sistem menggunakan pendekatan *prompting* dalam komunikasi dengan OpenAI, memastikan bahwa pertanyaan yang dihasilkan relevan dengan konten materi pembelajaran yang diberikan oleh dosen.

1. **Pengukuran Kesamaan Konteks Menggunakan *Cosine* *Similarity***

Evaluasi jawaban mahasiswa dilakukan dengan membandingkan kesamaan konteks antara jawaban mahasiswa dan jawaban yang benar menggunakan algoritma *cosine* *similarity*, yang diimplementasikan dalam sistem penilaian otomatis.

1. ***Embedding* OpenAI (text-embedding-ada-002)**

Sistem menggunakan embedding OpenAI model Ada 002 untuk mengukur representasi teks dan konteks dari jawaban mahasiswa. Embedding ini digunakan untuk memetakan teks ke dalam ruang vektor untuk menghitung kesamaan konteks dengan jawaban yang benar [5].

## Sumber Data

Dalam pelaksanaan tugas akhir ini, data yang digunakan untuk mendukung pengembangan sistem pembelajaran interaktif berbasis web terbagi menjadi dua jenis, yaitu data primer dan data sekunder.

* 1. Data Primer

1. Dokumen pembelajaran yang diunggah dosen

Data primer berasal dari dokumen-dokumen pembelajaran yang diunggah oleh dosen dalam bentuk berkas PDF. Dokumen ini berisi materi pembelajaran yang digunakan sebagai basis untuk menghasilkan pertanyaan otomatis menggunakan teknologi OpenAI.

1. Jawaban mahasiswa

Jawaban yang diberikan oleh mahasiswa melalui sistem pembelajaran merupakan data primer yang dikumpulkan secara otomatis. Jawaban ini digunakan sebagai input untuk proses evaluasi, dengan cara sistem membandingkan jawaban mahasiswa dengan jawaban yang benar untuk mengukur tingkat pemahaman mereka.

* 1. Data Sekunder
     + 1. Data OpenAI

Data sekunder diperoleh dari model OpenAI yang diakses melalui API. Data ini digunakan untuk menghasilkan pertanyaan otomatis dan melakukan *embedding* teks dalam proses evaluasi jawaban mahasiswa. Model OpenAI seperti GPT digunakan untuk *prompting* dalam generasi soal, sementara model *embedding* OpenAI (Ada 002) digunakan untuk mengukur kesamaan konteks jawaban mahasiswa dengan jawaban yang benar.

## Sistematika Penyajian

Laporan ini disusun dengan sistematika sebagai berikut:

1. Bab I Pendahuluan

Bab ini berisi latar belakang penelitian, rumusan masalah, tujuan penelitian, dan ruang lingkup. Dalam bab ini dijelaskan pentingnya pengembangan sistem pembelajaran interaktif berbasis *web* menggunakan teknologi OpenAI, serta permasalahan yang dihadapi dalam proses pembuatan soal dan evaluasi. Diharapkan, pembaca dapat memahami konteks dan tujuan dari penelitian yang dilakukan.

1. Bab II Kajian Teori

Pada bab ini, akan dibahas landasan teori yang mendukung penelitian ini, termasuk konsep sistem pembelajaran interaktif, *framework* Laravel dan Flask, serta teknologi OpenAI. Selain itu, akan diulas mengenai metode penilaian otomatis dan pengukuran kesamaan konteks menggunakan *cosine similarity*. Bab ini memberikan kerangka teoritis yang diperlukan untuk memahami pengembangan sistem.

1. Bab III Analisis Dan Rancangan Sistem

Bab ini menjelaskan metodologi yang digunakan dalam pengembangan sistem, termasuk analisis kebutuhan pengguna, perancangan arsitektur aplikasi, dan integrasi dengan API OpenAI. Dijelaskan juga langkah-langkah dalam merancang antarmuka pengguna, serta metode pengumpulan data primer dan sekunder, seperti dokumen pembelajaran dan jawaban mahasiswa. Perancangan sistem dilakukan dengan mengutamakan kemudahan penggunaan dan efektivitas dalam mendukung proses pembelajaran.

1. Bab IV Implementasi

Dalam bab ini, akan disajikan hasil pengembangan sistem, termasuk tampilan antarmuka aplikasi, proses generasi pertanyaan otomatis, dan evaluasi jawaban mahasiswa. Hasil analisis data akan dibahas secara mendetail, termasuk efektivitas sistem dalam menghasilkan soal dan memberikan umpan balik kepada mahasiswa. Pembahasan ini bertujuan untuk menginterpretasikan hasil yang diperoleh dan memberikan wawasan mengenai kinerja sistem.

1. Bab V Pengujian

Bab ini membahas hasil pengujian sistem berdasarkan serangkaian tes yang telah dilakukan untuk mengukur kinerja, akurasi, dan keandalan sistem. Pengujian dilakukan melalui simulasi nyata dengan melibatkan pengguna akhir, serta analisis kuantitatif dan kualitatif terhadap efektivitas sistem dalam konteks pembelajaran. Bab ini juga membahas temuan selama pengujian dan evaluasi sistem, serta perbaikan yang telah atau perlu dilakukan berdasarkan hasil pengujian.

1. Bab VI Simpulan dan Saran

Bab terakhir ini menyimpulkan temuan dari penelitian dan memberikan rekomendasi untuk pengembangan lebih lanjut. Kesimpulan yang diambil dari analisis hasil akan disajikan, dan saran untuk peningkatan sistem serta penelitian di masa mendatang akan diberikan. Bab ini bertujuan untuk merangkum hasil penelitian dan memberikan panduan untuk langkah selanjutnya.

# KAJIAN TEORI

#### 2.1 ****Sistem Pembelajaran Berbasis Web****

Sistem pembelajaran berbasis web telah menjadi pilar utama dalam dunia pendidikan, khususnya dengan meningkatnya penggunaan teknologi informasi dan komunikasi. Sistem ini memungkinkan proses belajar-mengajar dilakukan secara fleksibel, memungkinkan pengajar dan mahasiswa untuk berinteraksi tanpa dibatasi ruang dan waktu ​[2], [6]. Salah satu framework yang sering digunakan dalam pengembangan aplikasi pembelajaran berbasis *web* adalah **Laravel**, sebuah *framework* PHP yang menawarkan struktur Model-View-Controller (MVC). Laravel mendukung pengembangan aplikasi yang terstruktur dengan baik, memudahkan pembuatan, pengelolaan, serta pemeliharaan aplikasi berbasis *web*​[6].

Selain Laravel, **Flask** yang berbasis Python juga digunakan untuk membangun sistem *backend* ringan yang memungkinkan integrasi dengan teknologi kecerdasan buatan seperti OpenAI ​[2]. Integrasi ini penting dalam konteks pembelajaran modern karena memungkinkan pengembangan fitur otomatis seperti penilaian jawaban dan generasi pertanyaan otomatis.

#### 2.2 ****Framework Laravel dan Flask****

Framework Laravel menyediakan berbagai fitur canggih yang mendukung pengembangan aplikasi berbasis web, termasuk sistem routing yang fleksibel, ORM (*Object-Relational Mapping*) dengan *Eloquent*, serta dukungan terhadap berbagai basis data. Laravel juga mendukung pengembangan aplikasi berbasis REST API, yang sangat penting dalam pengembangan aplikasi berbasis *web* modern [5].

Di sisi lain, Flask adalah *framework* Python minimalis yang sangat berguna dalam integrasi dengan kecerdasan buatan. Flask memberikan fleksibilitas tinggi dalam membangun layanan backend yang dapat diintegrasikan dengan model AI berbasis Python, seperti OpenAI ​[6], [7]. Flask sering digunakan dalam proyek-proyek yang melibatkan otomatisasi, terutama pada pembelajaran mesin dan sistem pembelajaran berbasis AI ​[4].

#### 2.3 ****OpenAI dan Cipta Pertanyaan Otomatis****

Penggunaan OpenAI dalam generasi pertanyaan otomatis (*Automatic Question Generation* atau AQG) telah menjadi solusi unggul dalam dunia pendidikan, khususnya dalam mempercepat dan mempermudah proses evaluasi. Model bahasa besar seperti GPT-3 dan GPT-4 dari OpenAI mampu menghasilkan pertanyaan berdasarkan teks yang disediakan oleh dosen dalam format PDF [8], [9].

Penelitian oleh Kurdi et al. menunjukkan bahwa generasi pertanyaan otomatis tidak hanya menghemat waktu dan tenaga, tetapi juga memastikan relevansi pertanyaan terhadap materi pembelajaran yang disajikan ​[8]. Lebih lanjut, penelitian oleh Bulathwela et al. memperkenalkan model EduQG, yang berfokus pada pengembangan pertanyaan pendidikan menggunakan teks ilmiah dan dataset soal sains, memperlihatkan hasil yang superior dalam menghasilkan pertanyaan pendidikan berkualitas tinggi​ [10]. EduQG memungkinkan otomatisasi pembuatan soal yang relevan dengan materi pembelajaran, yang sangat penting dalam skenario pembelajaran daring dan kelas berskala besar ​[8], [10].

#### 2.4 *****Cosine Similarity* dan Penilaian Otomatis****

Metode *Cosine Similarity* merupakan salah satu teknik yang paling sering digunakan untuk membandingkan kesamaan konteks antara jawaban mahasiswa dan jawaban yang benar. Algoritma ini bekerja dengan menghitung sudut kosinus antara dua vektor teks, yang dihasilkan dari embedding teks. Model *embedding* seperti text-embedding-ada-002 dari OpenAI digunakan untuk merepresentasikan teks dalam ruang vektor dan menghitung kesamaan konteks ​[12], [13].

Menurut penelitian oleh Steck et al. meskipun *cosine similarity* sering digunakan untuk mengukur kesamaan semantik dalam berbagai aplikasi praktis, hasilnya bisa bervariasi tergantung pada metode regulasi yang digunakan dalam pembelajaran model ​[11]. Penelitian ini juga menunjukkan bahwa penggunaan *cosine similarity* harus diimbangi dengan pemahaman yang baik tentang bagaimana *embedding* yang dihasilkan oleh model bekerja. Penelitian Subecz (2021) juga menekankan pentingnya pemahaman ini dalam membangun aplikasi *web* yang melibatkan otomatisasi proses penilaian ​[6].

#### 2.5 *****Embedding* dalam OpenAI (text-embedding-ada-002)****

*Embedding* OpenAI memainkan peran penting dalam proses penilaian otomatis. *Embedding* adalah representasi numerik dari teks yang memungkinkan sistem untuk memahami hubungan semantik antara kata atau frasa dalam konteks yang lebih luas. Salah satu model terbaru yang digunakan adalah text-embedding-ada-002, yang menghasilkan *embedding* berkualitas tinggi dengan 1536 dimensi, memungkinkan penangkapan informasi semantik yang kaya dan mendetail [13].

Model embedding seperti text-embedding-ada-002 digunakan untuk mengukur kesamaan konteks antara jawaban mahasiswa dan jawaban referensi dengan presisi tinggi [2], [9]. Ghader et al. dalam tinjauannya menyatakan bahwa *embedding* semantik menjadi solusi penting dalam *Automatic Question Generation* (AQG) dan penilaian otomatis karena kemampuannya untuk memetakan hubungan antar-konsep dalam teks, membantu sistem pembelajaran dalam mengukur tingkat pemahaman mahasiswa [8].

Selain performa unggul dalam berbagai tugas pemrosesan bahasa alami, text-embedding-ada-002 dirancang agar efisien, menjadikannya solusi yang terjangkau bahkan untuk aplikasi dengan sumber daya terbatas. Dengan teknik ini, penilaian otomatis dapat dilakukan lebih efisien dan akurat, karena sistem mampu menilai jawaban siswa secara fleksibel berdasarkan kesamaan semantik, meskipun kalimat atau strukturnya berbeda dengan jawaban referensi [13].

#### 2.6 ****Penggunaan jQuery Ajax dan JSON untuk Komunikasi API****

REST API (*Representational State Transfer*) memungkinkan komunikasi yang efisien antara server dan klien melalui protokol HTTP, memanfaatkan metode standar seperti GET, POST, PUT, dan DELETE. REST API sering diintegrasikan dengan AJAX (Asynchronous JavaScript and XML) untuk memungkinkan pengiriman data secara asinkron antara *frontend* dan *backend* tanpa harus me-refresh halaman secara keseluruhan. jQuery AJAX menyederhanakan proses ini, memudahkan pengembang untuk membuat panggilan API secara asinkron dan menangani data dalam format JSON, yang ringan dan mudah dibaca.

Dalam studi terbaru, penggunaan jQuery AJAX dengan REST API telah terbukti efektif dalam pengembangan aplikasi web yang dinamis dan responsif. Misalnya, dengan menerapkan AJAX, pengembang dapat membuat sistem yang memungkinkan pengguna untuk mengirimkan atau mengambil data secara real-time dari server, seperti pada pengelolaan data kimia dalam proyek *open-source* berbasis REST API. JSON digunakan untuk memastikan data dapat diproses dengan cepat di kedua sisi aplikasi, memastikan efisiensi dan interoperabilitas yang tinggi [14], [15].

#### 2.7 ****REST API dalam Pengembangan Aplikasi Web****

REST API telah menjadi standar arsitektur utama dalam pengembangan aplikasi *web* modern. Penggunaan REST API tidak hanya mempermudah komunikasi antara server dan klien, tetapi juga memungkinkan penggunaan teknologi pendukung seperti Swagger untuk mendokumentasikan API dan memfasilitasi pengujian langsung melalui antarmuka web. Dalam proyek open-source di bidang kimia, REST API digunakan untuk mengakses data yang tersimpan di server dalam berbagai format, seperti JSON, dengan mekanisme otentikasi OAuth2 untuk keamanan tambahan.

Studi lebih lanjut juga menunjukkan bagaimana REST API dapat diterapkan untuk menangani berbagai tugas, mulai dari manajemen berkas hingga integrasi dengan layanan cloud seperti Amazon S3. Penggunaan JSON sebagai format transfer data semakin mendukung fleksibilitas dalam pengembangan aplikasi yang membutuhkan skalabilitas dan pengelolaan data yang komplek [14], [15].

Dengan pendekatan ini, pengembang dapat membangun aplikasi yang tidak hanya dinamis dan interaktif, tetapi juga aman dan efisien dalam hal komunikasi data, yang sangat penting dalam skenario pengembangan *web* modern.

#### 2.8 ****Penelitian Terkait****

Penelitian mengenai *Automatic Question Generation* (AQG) telah berkembang pesat dalam beberapa tahun terakhir. Das et al. menyajikan survei komprehensif mengenai teknik generasi pertanyaan otomatis dan penilaian jawaban, menggarisbawahi pentingnya otomatisasi dalam mengurangi beban kerja pengajar dan meningkatkan efisiensi proses pembelajaran​ [2].

Bulathwela et al. juga mengembangkan model EduQG yang menggunakan model bahasa besar untuk menghasilkan pertanyaan yang lebih spesifik berdasarkan teks ilmiah, memberikan hasil yang lebih baik dibandingkan pendekatan tradisional ​[10]. Model ini membuktikan bahwa pertanyaan berkualitas tinggi dapat dihasilkan secara otomatis menggunakan pendekatan pre-training dan fine-tuning dengan data ilmiah ​[8], [10].

Penelitian lain oleh Subecz menekankan pentingnya struktur yang baik dalam pengembangan aplikasi *web* berbasis pembelajaran menggunakan framework seperti Laravel, yang mendukung pengintegrasian dengan teknologi seperti OpenAI ​[6]. Ini menunjukkan bahwa teknologi pembelajaran berbasis AI memiliki potensi besar untuk terus berkembang dan diterapkan secara luas di berbagai platform pendidikan daring.

# ANALISIS DAN RANCANGAN SISTEM



## Gambaran Umum

Bab ini memberikan penjelasan terperinci mengenai analisis dan rancangan sistem yang diusulkan untuk mengotomatiskan proses pembuatan soal dan evaluasi, serta interaksi antara dosen, mahasiswa, dan sistem. Pembahasan dalam bab ini dimulai dengan identifikasi *stakeholder* dan peran mereka dalam sistem, diikuti dengan deskripsi mengenai bagaimana proses otomatisasi pembuatan dan evaluasi soal akan diterapkan, mulai dari unggah PDF hingga evaluasi otomatis.

Selanjutnya, akan dibahas desain antarmuka untuk dosen dan mahasiswa, termasuk antarmuka unggahan PDF dan *chatbot*. Rancangan penyimpanan data juga akan dijelaskan secara detail untuk menggambarkan bagaimana informasi disimpan dan dikelola di *database*. Alur sistem yang mencakup seluruh proses dari awal hingga akhir akan dipaparkan dalam bentuk diagram dan langkah-langkah yang mendetail.

Pada bab ini, akan disertakan juga *use case diagram*, *activity diagram*, dan *entity relationship diagram* sebagai representasi visual untuk memperjelas alur kerja sistem dan relasi antar data yang digunakan.

## *Stakeholder*

A diagram of a diagram

Description automatically generated

Gambar . *Use Case Diagram*

Sistem ini dikembangkan dengan sasaran pengguna utama dosen dan mahasiswa. Dosen menggunakan platform ini untuk mengunggah materi pembelajaran dalam format PDF, kemudian sistem secara otomatis menghasilkan pertanyaan berdasarkan materi yang diunggah. Selain itu, dosen akan memperoleh hasil penilaian otomatis terhadap jawaban mahasiswa, yang dapat dijadikan sebagai referensi mengenai tingkat pemahaman siswa (Gambar 3.2.1). *Chatbot* pada sistem ini akan membimbing mahasiswa dengan pertanyaan, dimulai dari yang umum hingga ke yang lebih mendetail, guna membantu mereka memahami materi dengan lebih baik.

## Otomatisasi Proses Pembuatan dan Evaluasi Soal

Saat ini, proses pembuatan dan evaluasi soal dilakukan secara manual yang mengharuskan dosen membuat soal dan menilai jawaban mahasiswa satu per satu. Proses ini memakan waktu yang cukup lama, terutama jika jumlah mahasiswa banyak.

Sistem yang diusulkan bertujuan untuk mengotomatiskan proses tersebut, mulai dari pembuatan soal hingga penilaian jawaban mahasiswa. Berikut adalah alur proses yang diusulkan:

1. **Unggah PDF**: Dosen mengunggah dokumen pembelajaran dalam format PDF ke sistem.
2. **Konversi PDF ke Teks**: Sistem mengonversi berkas PDF menjadi teks menggunakan Flask.
3. **Cipta Soal Otomatis**: Berdasarkan teks yang dihasilkan, sistem secara otomatis membuat pertanyaan esai beserta jawabannya menggunakan taksonomi Bloom tingkat 1 (*remembering*) dan tingkat 2 (*understanding*). Dari setiap halaman PDF, akan dihasilkan 4 pertanyaan: 2 pertanyaan berbasis taksonomi Bloom tingkat 1 dan 2 pertanyaan berbasis taksonomi Bloom tingkat 2.
4. **Pemilihan Soal**: Dosen dapat memilih dan mengatur pertanyaan serta jawaban yang telah dihasilkan oleh sistem.
5. **Interaksi dengan Mahasiswa**: Mahasiswa menjawab pertanyaan melalui antarmuka *chatbot*, yang membimbing mereka dari pertanyaan umum hingga mendetail.
6. **Evaluasi Otomatis**: Jawaban mahasiswa dievaluasi secara otomatis dengan membandingkannya menggunakan *cosine similarity* terhadap jawaban hasil cipta yang telah dikonfirmasi oleh dosen.

## Rancangan Antarmuka

Antarmuka sistem dibagi menjadi beberapa komponen:

1. **Antarmuka Pengunggahan PDF dan Cipta Soal**

Dosen dapat mengunggah berkas PDF melalui antarmuka *frontend* yang dibangun menggunakan Bootstrap. Berkas PDF yang diunggah kemudian akan diproses di *backend* Laravel dan dikonversi menjadi teks oleh Flask. Setelah dokumen diproses, sistem secara otomatis menghasilkan pertanyaan dan jawaban berdasarkan konten dari PDF yang diunggah. Dosen dapat melihat pertanyaan dan jawaban yang dihasilkan, serta memiliki kemampuan untuk:

* Memilih pertanyaan yang relevan,
* Menyortir pertanyaan sesuai dengan kebutuhan,
* Mengedit pertanyaan dan jawaban jika diperlukan sebelum digunakan dalam evaluasi mahasiswa.

Antarmuka ini mengintegrasikan proses pengunggahan dan cipta soal dalam satu langkah yang efisien dan mudah digunakan.

1. **Antarmuka *Chatbot* Siswa**

Antarmuka *chatbot* dirancang untuk memberikan pengalaman interaktif kepada mahasiswa dalam menjawab pertanyaan yang telah dihasilkan oleh sistem. Berikut adalah fitur utamanya:

* ***Chatbot* Interaktif:** Siswa dapat menjawab pertanyaan yang muncul melalui antarmuka *chatbot*, dibangun menggunakan jQuery dan Bootstrap.
* **Penilaian Otomatis:** Jawaban mahasiswa akan dinilai secara otomatis menggunakan *cosine similarity* untuk mengukur kesesuaian jawaban mereka dengan jawaban yang benar.
* **Umpan Balik *Real-Time*:** Siswa akan menerima pertanyaan tambahan jika jawaban belum mencapai standar/batas nilai *cosine* yang ditentukan oleh dosen.

1. **Antarmuka Mata Kuliah dan Topik**

Antarmuka ini berfungsi untuk mengelola mata kuliah dan topik pembelajaran. Fitur-fitur utamanya mencakup:

* **Manajemen Mahasiswa:** Dosen dapat mengelola daftar mahasiswa yang terdaftar di setiap mata kuliah, termasuk menambah atau menghapus mahasiswa.
* **Pengaturan Penilaian:** Dosen dapat mengatur penilaian (*cosine similarity treshold*), menginput nilai, serta memonitor hasil evaluasi.
* **History *Chatbot*:** Dosen dapat melihat riwayat interaksi chatbot secara detail, termasuk jawaban yang diberikan oleh mahasiswa dan performa mereka.
* **Pengaturan Pertanyaan:** Dosen dapat mengatur ulang pertanyaan dan jawaban, serta menambahkan pertanyaan baru sesuai dengan perkembangan topik yang diajarkan.
* **Pengaturan *Deadline*:** Dosen dapat mengatur tenggat waktu untuk interaksi chatbot dan evaluasi berdasarkan mata kuliah dan topik yang diampu.

1. **Antarmuka *Dashboard***

*Dashboard* ini memberikan dosen akses cepat ke rangkuman informasi penting mengenai aktivitas di dalam sistem. Informasi yang ditampilkan meliputi:

* **Daftar materi *chatbot* yang sedang berjalan:** Dosen dapat melihat *chatbot* dengan topik yang masih dalam tenggat waktu.
* **Akses cepat ke mata kuliah yang aktif:** Dosen dan mahasiswa bisa dengan mudah mengakses mata kuliah yang sedang diampu atau diikuti melalui *dashboard*.
* **Statistik Pembelajaran:** Dosen dapat melihat rangkuman performa mahasiswa di mata kuliah yang mereka ampu, sementara mahasiswa bisa memeriksa perkembangan hasil interaksi mereka dengan *chatbot* di mata kuliah yang diikuti.

*Dashboard* ini akan dibuat menggunakan Bootstrap untuk tata letak yang responsif, serta memanfaatkan AJAX untuk memperbarui data secara *real-time*.

## Rancangan Penyimpanan Data

A computer screen shot of a computer

Description automatically generated

Gambar . *Entity Relationship Diagram*

Sistem menggunakan MySQL untuk menyimpan data (Gambar 3.5.1), yang mencakup informasi tentang:

* 1. **Berkas Path:** Lokasi berkas PDF yang diunggah oleh dosen untuk diproses lebih lanjut oleh sistem.
  2. **Pertanyaan:** Pertanyaan yang dihasilkan dari berkas PDF menggunakan OpenAI, termasuk jawaban yang benar.
  3. **Jawaban Mahasiswa:** Jawaban yang diberikan oleh mahasiswa melalui antarmuka *chatbot*, yang kemudian akan dievaluasi menggunakan metode penilaian seperti *cosine similarity*.
  4. **Progres Mahasiswa:** Catatan mengenai progres mahasiswa dalam menjawab pertanyaan, seperti status pengerjaan dan nilai yang diperoleh.
  5. **Data Mahasiswa:** Informasi setiap mahasiswa yang berpartisipasi dalam sistem, termasuk profil dan status akademis mereka.
  6. **Mata Kuliah dan Topik:** Sistem menyimpan berbagai mata kuliah (*course*) dan setiap mata kuliah memiliki beberapa topik. Setiap mahasiswa bisa mengikuti mata kuliah yang berbeda-beda sesuai kurikulum mereka.
  7. ***Role* Pengguna:** MySQL juga berperan dalam menyimpan informasi terkait *role* pengguna, seperti dosen, mahasiswa, dan asisten. *Role* ini menentukan hak akses setiap pengguna di dalam sistem, seperti mengunggah berkas, melihat pertanyaan, atau menjawab soal.

## Alur Sistem

A diagram of a diagram

Description automatically generated with medium confidence

Gambar . *Activity Diagram*

Berikut adalah alur proses dalam sistem berdasarkan Gambar 3.6.1:

1. Dosen mengunggah PDF.
2. Path berkas PDF disimpan di *database*.
3. Path berkas dikirimkan ke Flask untuk konversi PDF ke teks.
4. OpenAI API menghasilkan pertanyaan dan jawaban otomatis dari teks.
5. Pertanyaan dikirimkan kembali ke Laravel untuk ditampilkan kepada dosen.
6. Dosen menyortir pertanyaan dan menyimpannya ke basis data.
7. Pertanyaan dan jawaban yang telah disortir dan disesuaikan disimpan ke dalam *database.*
8. Siswa mengakses fitur *chatbot*.
9. *Chatbot* mengajukan pertanyaan kepada mahasiswa dengan memberikan petunjuk berupa nomor halaman tempat pertanyaan tersebut dihasilkan.
10. Jawaban mahasiswa dibandingkan dengan jawaban sesungguhnya menggunakan *cosine* similarity.
11. Jika jawaban tidak memenuhi ambang batas *cosine similarity*, *chatbot* menghasilkan pertanyaan serupa.
12. Jika semua pertanyaan telah ditanyakan sistem menghitung rata – rata peforma mahasiswa berdasarkan nilai *cosine similarity* dan jumlah pertanyaan.
13. Dosen dapat menyesuaikan nilai akhir performa hasil interaksi mahasiswa dan *chatbot.*
14. Histori interaksi akan disimpan dalam *database*.
15. Histori dapat diakses kembali pada antarmuka mata kuliah dan topik.

## Rancangan Sistem

Sistem ini dibangun dengan menggunakan:

1. **Laravel *Framework*:** Digunakan untuk mengelola *frontend* dan *backend*, termasuk otentikasi, *routing*, dan manajemen data.
2. **Flask *Framework*:** Digunakan untuk pemrosesan berkas PDF, termasuk konversi teks menggunakan pdflumber, serta mengintegrasikan layanan eksternal.
3. **pdflumber:** Sebuah *library* Python yang digunakan dalam proses ekstraksi teks dari PDF dengan akurasi tinggi, memungkinkan pengambilan konten yang akan digunakan untuk menghasilkan pertanyaan.
4. **MySQL:** Digunakan sebagai basis data untuk menyimpan informasi pengguna, materi, dan hasil evaluasi.
5. **Bootstrap:** Membuat antarmuka pengguna yang responsif dan dapat diakses melalui berbagai perangkat.
6. **jQuery AJAX:** Memungkinkan pemanggilan API secara *asynchronous* untuk memperbarui data secara *real-time* tanpa me-refresh halaman.
7. **OpenAI:** Teknologi ini digunakan untuk memproses teks hasil ekstraksi PDF dan menghasilkan pertanyaan esai serta jawaban berdasarkan taksonomi Bloom, yang digunakan untuk evaluasi pembelajara

# DAFTAR PUSTAKA

[1] K. Hwang, S. Challagundla, M. M. Alomair, L. Karen Chen, and F.-S. Choa, “Towards AI-Assisted Multiple Choice Question Generation and Quality Evaluation at Scale: Aligning with Bloom’s Taxonomy,” 2023. [Online]. Available: https://tinyurl.com/35am6sah

[2] B. Das, M. Majumder, S. Phadikar, and A. A. Sekh, “Automatic question generation and answer assessment: a survey,” *Res Pract Technol Enhanc Learn*, vol. 16, no. 1, p. 5, Dec. 2021, doi: 10.1186/s41039-021-00151-1.

[3] C. Grévisse, “LLM-based automatic short answer grading in undergraduate medical education,” *BMC Med Educ*, vol. 24, no. 1, p. 1060, Sep. 2024, doi: 10.1186/s12909-024-06026-5.

[4] A. K. Bhowmick *et al.*, “Automating question generation from educational text,” Sep. 2023.

[5] R. Greene, Shanders Ted, L. Weng, and A. Neelakantan, “New and improved embedding model.” Accessed: Oct. 07, 2024. [Online]. Available: https://openai.com/index/new-and-improved-embedding-model/

[6] Z. Subecz, “Web-development with Laravel framework,” *Gradus*, vol. 8, no. 1, pp. 211–218, 2021, doi: 10.47833/2021.1.CSC.006.

[7] J. Mozafari, A. Abdallah, B. Piryani, and A. Jatowt, “Exploring Hint Generation Approaches in Open-Domain Question Answering,” Sep. 2024.

[8] G. Kurdi, J. Leo, B. Parsia, U. Sattler, and S. Al-Emari, “A Systematic Review of Automatic Question Generation for Educational Purposes,” *Int J Artif Intell Educ*, vol. 30, no. 1, pp. 121–204, Mar. 2020, doi: 10.1007/s40593-019-00186-y.

[9] N.-T. Le, T. Kojiri, and N. Pinkwart, “Automatic Question Generation for Educational Applications – The State of Art,” 2014, pp. 325–338. doi: 10.1007/978-3-319-06569-4\_24.

[10] S. Bulathwela, H. Muse, and E. Yilmaz, “Scalable Educational Question Generation with Pre-trained Language Models,” May 2023.

[11] H. Steck, C. Ekanadham, and N. Kallus, “Is Cosine-Similarity of Embeddings Really About Similarity?,” in *Companion Proceedings of the ACM Web Conference 2024*, New York, NY, USA: ACM, May 2024, pp. 887–890. doi: 10.1145/3589335.3651526.

[12] M. S. Bonney *et al.*, “Development of a digital twin operational platform using Python Flask,” *Data-Centric Engineering*, vol. 3, p. e1, Jan. 2022, doi: 10.1017/dce.2022.1.

[13] S. Ghosh, “A Deep Dive into OpenAI’s text-embedding-ada-002: The Power of Semantic Understanding.” Accessed: Oct. 15, 2024. [Online]. Available: https://medium.com/@siladityaghosh/a-deep-dive-into-openais-text-embedding-ada-002-the-power-of-semantic-understanding-7072c0386f83

[14] S.-P. Ma, M.-J. Hsu, H.-J. Chen, and C.-J. Lin, “RESTful API Analysis, Recommendation, and Client Code Retrieval,” *Electronics (Basel)*, vol. 12, no. 5, p. 1252, Mar. 2023, doi: 10.3390/electronics12051252.

[15] M. D. Hanwell, W. A. de Jong, and C. J. Harris, “Open chemistry: RESTful web APIs, JSON, NWChem and the modern web application,” *J Cheminform*, vol. 9, no. 1, p. 55, Dec. 2017, doi: 10.1186/s13321-017-0241-z.

**NAMA LAMPIRAN**

**NAMA LAMPIRAN**

**NAMA LAMPIRAN**

**RIWAYAT HIDUP PENULIS**

A person in a blue suit

Description automatically generatedLaurentius Gusti Ontoseno Panata Yudha merupakan seorang mahasiswa ilmu komputer yang berfokus pada pengembangan AI dan *backend web*. Ia memiliki keahlian dalam merancang *database* dan mengimplementasikan kerangka kerja *backend*. Laurentius sedang menempuh pendidikan di Universitas Kristen Maranatha di Bandung, Indonesia, di jurusan Teknik Informatika sejak tahun 2021 dengan IPK 3.9/4.0. Selama studinya, ia menerima Beasiswa Internal Maranatha dalam Kategori Prestasi Akademik untuk Semester Ganjil Tahun Akademik 2023/2024, Semester Genap Tahun Akademik 2022/2023, dan Semester Ganjil Tahun Akademik 2022/2023.

Laurentius memiliki pengalaman magang sebagai *backend developer* di WIT.ID Bandung pada periode September hingga Desember 2023, di mana ia membuat *use-case*, laporan, diagram hubungan entitas, dan diagram kelas untuk pencatatan inventaris kantor. Ia juga mengembangkan *database* untuk pencatatan inventaris, melakukan perhitungan depresiasi dan nilai buku pada inventaris yang ada, serta mengimplementasikan sistem CRUD dengan MVC Laravel.

Selain itu, Laurentius juga aktif dalam organisasi di kampusnya, yaitu SEMAFIT Maranatha, di mana ia menjabat sebagai Koordinator Bidang Akademik masa jabatam 2023/2024. Dalam perannya ini, ia mengorganisir program tutorial untuk mahasiswa IT, khususnya mahasiswa baru, serta mengorganisir seminar dengan mengundang pembicara dari industri IT.

Laurentius juga terlibat dalam berbagai proyek, seperti proyek *backend web* Perwalian yang merupakan tugas akhir untuk mata kuliah Pemrograman Web Lanjut di Universitas Kristen Maranatha dengan nilai sempurna 100/100. Ia juga berpartisipasi dalam kompetisi pengembangan *web Fullstack* di Universitas Kristen Satya Wacana pada Agustus 2023, di mana ia meraih peringkat ke-4 dengan mengembangkan aplikasi web UnivAssist yang menawarkan layanan penyewaan kost, laptop, dan buku kuliah sebagai solusi berarti bagi mahasiswa dalam waktu 12 jam. Selain itu, Laurentius juga mengerjakan proyek sistem inventaris web untuk WIT.ID dari September hingga Desember 2023, di mana ia mengembangkan aplikasi web yang menyediakan laporan aset di seluruh kantor WIT, termasuk riwayat penggunaan dan perbaikan.

Laurentius memiliki berbagai sertifikasi, termasuk AWS *Academy Graduate untuk Introduction to Cloud* Semester 2 pada Juni 2023 dan *Introduction to Cloud* Semester 1 pada Juni 2022. Ia menguasai beberapa bahasa pemrograman seperti JavaScript, Python, dan PHP, serta *markup language* seperti HTML dan CSS. Laurentius juga memiliki keterampilan dalam menggunakan *database* MS SQL dan MySQL, serta kerangka kerja Bootstrap, Laravel, dan Django. Selain itu, ia juga familiar dengan teknologi seperti Git, Postman, MySQL Workbench, dan Anaconda.

Dengan kemampuan berbahasa Indonesia sebagai bahasa ibu dan bahasa Inggris dengan tingkat kemahiran profesional, serta *soft skills* seperti pemecahan masalah, manajemen waktu, dan berpikir kritis, Laurentius siap menghadapi tantangan dalam industri IT dan memberikan kontribusi yang signifikan.