**BULLETIN OF COMPUTER SCIENCE RESEARCH**
**ISSN 2774-3659 (Media Online)**
Vol 5, No 4, June 2025 | Hal 434-446
https://hostjournals.com/bulletincsr
DOI: 10.47065/bulletincsr.v5i4.558

# Implementasi Langchain dan Large Language Models Dalam Automatic Question Generation Untuk Computer Assisted Test

**Novri Rahman, Nazruddin Safaat Harahap*, Muhammad Affandes, Pizaini**

Fakultas Sains dan Teknologi, Teknik Informatika, Universitas Islam Negeri Sultan Syarif Kasim Riau, Pekanbaru, Indonesia
Email: [1]12150110009@students.uin-suska.ac.id, [2,*]nazruddin.safaat@uin-suska.ac.id, [3]affandes@uin-suska.ac.id, [4]pizaini@uin-suska.ac.id
Email Penulis Korespondensi: nazruddin.safaat@uin-suska.ac.id

**Abstrak**−Perkembangan teknologi *Artificial Intelligence* (AI), khususnya *Large Language Models* (LLM), membuka peluang baru dalam transformasi sistem evaluasi pendidikan. Penelitian ini bertujuan mengimplementasikan kerangka kerja *LangChain* yang terintegrasi dengan LLM untuk sistem *Automatic Question Generation* (AQG) pada *Computer Assisted Test* (CAT), dengan studi kasus pada materi Biologi kelas XI. Metode yang digunakan mencakup pengumpulan data dari dokumen PDF materi ajar, proses *embedding* menggunakan *Facebook AI Similarity Search* (FAISS) sebagai basis pengetahuan, serta pembuatan soal otomatis melalui model GPT-4o. Sistem dirancang dengan arsitektur *microservices* yang terdiri atas layanan *frontend* dan *backend* menggunakan *framework Next.js, FastAPI,* dan *Express.js*. Evaluasi dilakukan melalui pendekatan *User Acceptance Test* dan *framework DeepEval*. Hasil evaluasi menunjukkan tingkat kepuasan guru sebesar 92,7% dan respons positif dari mahasiswa sebesar 67,5%. Sementara itu, hasil *DeepEval* menunjukkan rata-rata metrik *hallucination* sebesar 3,69%, *contextual precision* 97,44%, *contextual relevancy* 83,30%, *answer relevancy* 70,63%, dan *prompt alignment* 92,47%. Temuan ini menunjukkan bahwa integrasi *LangChain* dan LLM efektif dalam menghasilkan soal yang relevan dan kontekstual, meskipun aspek ketepatan jawaban masih perlu ditingkatkan. Penelitian ini diharapkan menjadi solusi efisien dalam penyusunan soal evaluasi digital serta membuka arah baru bagi pengembangan AI dalam dunia pendidikan.

**Kata Kunci:** Automatic Question Generation; Computer Assisted Test; Large Language Models; LangChain; GPT-4o

**Abstract**−The advancement of Artificial Intelligence (AI), particularly Large Language Models (LLM), presents new opportunities in transforming educational assessment systems. This study aims to implement the LangChain framework integrated with LLM for an Automatic Question Generation (AQG) system within a Computer Assisted Test (CAT) platform, using eleventh-grade Biology subject matter as a case study. The methodology includes data collection from PDF-based instructional materials, text embedding using Facebook AI Similarity Search (FAISS) as the knowledge base, and automatic question generation through the GPT-4o model. The system is developed using a microservices architecture comprising frontend and backend services built with the Next.js, FastAPI, and Express.js frameworks. System evaluation was conducted using the User Acceptance Test (UAT) and the DeepEval framework. The evaluation results show a teacher satisfaction rate of 92.7% and a positive response from students at 67.5%. Meanwhile, the DeepEval assessment reported average scores of 3,69% for hallucination, 97,44% for contextual precision, 83,30% for contextual relevancy, 70,63% for answer relevancy, and 92,47% for prompt alignment. These findings indicate that the integration of LangChain and LLM is effective in generating contextually accurate and relevant questions, although improvements are still needed in answer relevancy. This study is expected to provide an efficient solution for digital-based educational assessment and contribute to future developments in educational AI.

**Keywords**: Automatic Question Generation; Computer Assisted Test; Large Language Models; LangChain; GPT-4o

## 1. PENDAHULUAN

Kemajuan teknologi telah mengubah lanskap pendidikan secara signifikan, membawa revolusi dalam cara kita belajar, mengajar, dan mengevaluasi kemampuan siswa melalui pemanfaatan *Artificial Intelligence* (AI) dan *Machine Learning* (ML) [1]. Transformasi ini tidak hanya merupakan evolusi bertahap, tetapi juga sebuah lompatan besar yang memungkinkan pendekatan baru dalam proses pendidikan, mulai dari pengelolaan administrasi hingga penyusunan materi evaluasi yang lebih efisien. Salah satu inovasi utama yang menjadi pendorong perubahan ini adalah *Large Language Models* (LLM), seperti *Generative Pre-Trained Transformer 4* (GPT-4) yang menawarkan kemampuan luar biasa dalam memahami bahasa manusia, menagkap konteks, dan menghasilkan teks yang mendekati kualitas tulisan manusia [2], [3]. Dalam konteks Pendidikan, LLM memiliki potensi besar untuk mendukung pengembangan *Computer Assisted Test* (CAT), sebuah sistem evaluasi berbasis komputer yang dirancang untuk memberikan pengalaman ujian yang lebih adaptif, cepat, dan akurat dibandingkan metode tradisional berbasis kertas dan pensil [4].

Di Indonesia, sistem CAT mulai diterapkan pada seleksi Calon Pegawai Negeri Sipil (CPNS) sejak tahun 2013 oleh Badan Kepegawaian Negara (BKN), dengan tujuan meningkatkan transparansi dan akuntabilitas proses seleksi [5]. Pelaksanaan CAT umumnya dilakukan secara *online* melalui jaringan komputer lokal ataupun internet, sehingga soal dapat diakses oleh banyak peserta sekaligus dan evaluasi menjadi lebih cepat [6]. Penggunaan CAT juga dinilai meningkatkan efisiensi serta keadilan penilaian karena mengurangi potensi manipulasi nilai dan penggunaan kertas dalam evaluasi pembelajaran [7]. Namun, penerapan CAT menghadapi tantangan seperti kesenjangan akses teknologi dan risiko kecurangan, sehingga diperlukan infrastruktur memadai dan pengawasan ketat untuk menjaga integritas ujian [8]. Berdasarkan penelitian oleh [9], ujian berbasis komputer (CAT) efektif mengurangi beban administratif serta waktu dan biaya pelaksanaan tes secara signifikan. Dengan semakin meluasnya implementasi CAT, muncul kebutuhan untuk mengembangkan sistem yang lebih cerdas dan adaptif, salah satunya melalui integrasi teknologi mutakhir seperti LLM.

Fokus utama penelitian tentang LLM selama ini lebih banyak tertuju pada aplikasi umum seperti *chatbot*, sistem rekomendasi, atau penerjemahan otomatis, sedangkan eksplorasi spesifik dalam *Automatic Question Generation* (AQG)

BULLETIN OF COMPUTER SCIENCE RESEARCH
ISSN 2774-3659 (Media Online)
Vol 5, No 4, June 2025 | Hal 434-446
https://hostjournals.com/bulletincsr
DOI: 10.47065/bulletincsr.v5i4.558

untuk mendukung CAT masih relatif terbatas [10], [11]. Contohnya, penelitian oleh [12] menganalisis dokumen *chatbot* dari basis data *Scopus* untuk memetakan tren perkembangan *chatbot* dari waktu ke waktu, dan menemukan peningkatan signifikan penggunaan LLM dalam implementasi *chatbot* praktis di berbagai bidang, termasuk pendidikan. Berdasarkan penelitian [12] integrasi LLM pada *chatbot* terbukti efektif dan penggunaannya semakin meluas. Dengan pendekatan konseptual yang serupa, AQG berpotensi memanfaatkan LLM untuk merevolusi proses penyusunan soal dengan menghasilkan pertanyaan yang relevan, berkualitas akademik, dan sesuai dengan kebutuhan siswa secara otomatis, sehingga mengurangi beban kerja tenaga pendidik dalam menyusun evaluasi secara manual.

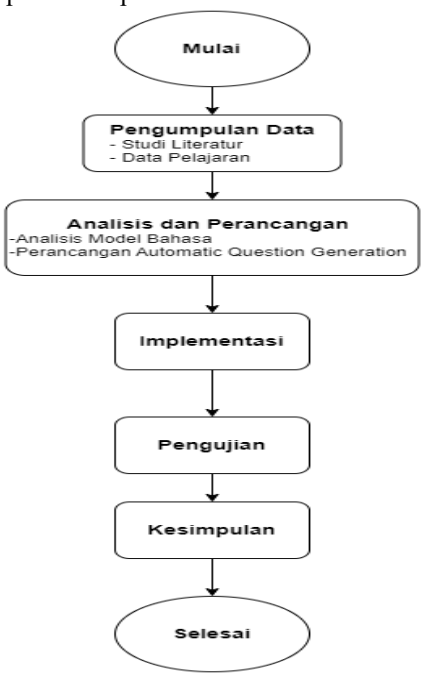
Penerapan AQG yang didukung oleh LLM dapat ditingkatkan lebih lanjut melalui kerangka kerja seperti *LangChain* yang telah mempermudah integrasi LLM dengan sumber data yang kompleks [13]. *LangChain* memungkinkan penggabungan LLM dengan berbagai sumber data, baik terstruktur maupun tidak terstruktur, untuk meningkatkan akurasi dan relevansi *output*. Penelitian sebelumnya oleh [14], menunjukkan bahwa *LangChain* jika digabungkan dengan model *Large Language Model Application 3* (*LLaMA3)* memungkinkan pengembangan aplikasi AI yang lebih cepat dan efisien, terutama dalam menyederhanakan alur kerja penghubungan model bahasa dengan berbagai sumber data. Selain itu, penerapan AQG yang didukung oleh LLM dapat ditingkatkan dengan *Retrieval Augmented Generation* (RAG), seperti yang diterapkan dalam sistem tanya jawab hadis berbasis AI menggunakan *LangChain* untuk menghubungkan model bahasa besar dengan *retrieval* berbasis dokumen guna meningkatkan relevansi dan akurasi jawaban [15]. Hal ini juga didukung oleh [16] yang berhasil mengurangi halusinasi dalam sistem *Question Answering* (QA) biomedis dengan menggabungkan LLM dan *knowledge graph* melalui *LangChain*. Pendekatan lain yang dilakukan oleh [17] adalah *In-Context Learning* (ICL) memungkinkan LLM seperti GPT-4 menghasilkan pertanyaan hanya dengan *few-shot* contoh, dan ketika dikombinasikan dengan *Retrieval-Augmented Generation* (RAG) dalam model *hybrid*, dapat meningkatkan kompleksitas dan relevansi pertanyaan. Meskipun demikian, terdapat tantangan yang harus diatasi dalam implementasi teknologi ini. Soal yang dihasilkan harus memenuhi standar akademik dari segi kualitas, tingkat kesulitan, dan relevansi dengan kurikulum yang berlaku, agar dapat digunakan secara efektif dalam berbagai tingkat pembelajaran.

Berdasarkan latar belakang tersebut, penelitian ini muncul sebagai respons terhadap kesenjangan yang ada, dengan tujuan mengimplementasikan *LangChain* dan LLM dalam sistem AQG untuk CAT, dengan studi kasus mata pelajaran biologi kelas 11. Fokus utama penelitian ini adalah mengembangkan sistem pembuatan soal otomatis berbasis teks dari dokumen PDF guna membantu tenaga pendidik dalam menyusun soal ujian secara efisien. Selain itu, penelitian ini mengeksplorasi strategi untuk meminimalkan bias dan ketidakakuratan dalam generasi soal, serta memastikan transparasi dan keandalan sistem. Dengan demikian, penelitian ini diharapkan tidak hanya memberikan solusi praktis bagi dunia pendidikan, tetapi juga berkontribusi pada pengembangan ilmiah terkait integrasi AI dalam sistem evaluasi pendidikan berbasis digital.

# 2. METODOLOGI PENELITIAN

## 2.1 Tahapan Penelitian

Tahapan-tahapan dalam penelitian sistem *Automatic Question Generation* berbasis *LangChain* dan *Large Language Models* untuk *Computer Assisted Test* dapat dilihat pada Gambar 1.




**Gambar 1.** Tahapan Penelitian

BULLETIN OF COMPUTER SCIENCE RESEARCH
ISSN 2774-3659 (Media Online)
Vol 5, No 4, June 2025 | Hal 434-446
https://hostjournals.com/bulletincsr
DOI: 10.47065/bulletincsr.v5i4.558

Gambar 1 menampilkan rangkaian tahapan penelitian, mulai dari pengumpulan data hingga kesimpulan. Berikut penjelasan tiap tahap penelitian ini:

### 2.1.1 Pengumpulan Data

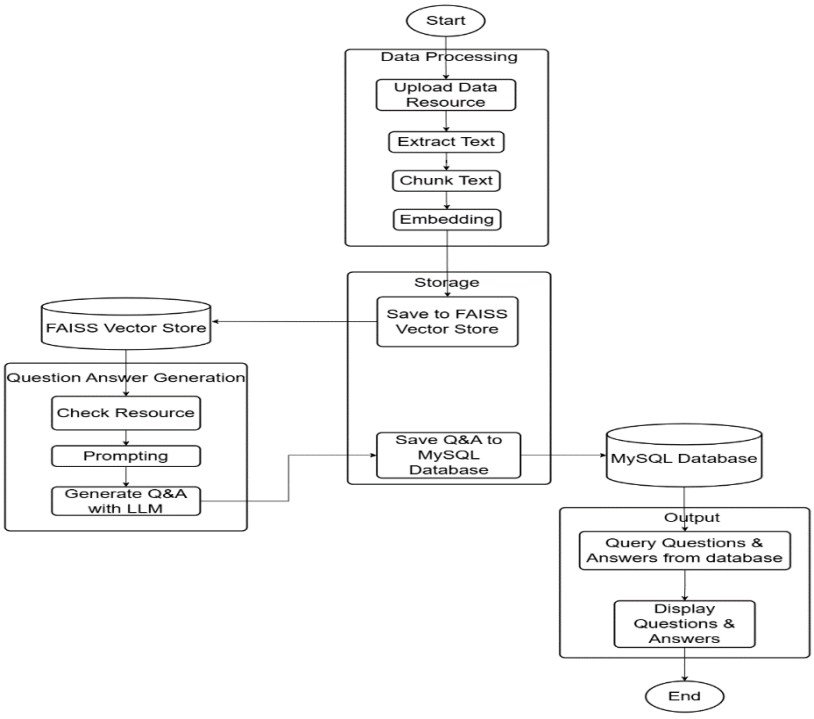
Tahap awal adalah pengumpulan data yang menjadi dasar penelitian. Berikut penjelasan lebih lanjut mengenai tahapan ini:

a. Studi Literatur
   Pencarian studi literatur ini bertujuan untuk menambah wawasan penulis mengenai penelitian yang akan dilakukan. Beberapa topik yang perlu dicari oleh peneliti terkait penelitian ini meliputi *Automatic Question Generation System, LangChain, Large Language Models* (LLM), *Computer Assisted Test* (CAT), *Natural Language Processing* (NLP), serta materi pelajaran biologi kelas 11 yang akan digunakan dalam penelitian.
b. Data Pelajaran
   Materi pelajaran biologi kelas 11 yang digunakan dalam penelitian ini berbentuk dokumen PDF yang diterbitkan oleh Kementrian Pendidikan, Kebudayaan, Riset, dan Teknologi. Dokumen ini mencakup delapan bab pembahasan yang akan dijadikan sebagai sumber utama dalam pengujian sistem untuk menghasilkan pertanyaan secara otomatis.

### 2.1.2 Analisis dan Perancangan

Tahapan ini terdiri dari analisis model bahasa dan perancangan sistem AQG. Berikut penjelasan lebih lanjut mengenai tahapan ini:

a. Analisis Model Bahasa
   Model bahasa yang digunakan dalam penelitian ini adalah GPT-4o dari OpenAI. Pemilihan model ini didasarkan pada kemampuannya dalam menghasilkan teks yang alami, akurat, dan kontekstual [18]. Selain itu, GPT-4o memiliki keunggulan dalam memahami serta menyusun pertanyaan dengan struktur yang sesuai berdasarkan materi yang diberikan, sehingga mendukung proses otomatisasi pembuatan soal secara efektif.
b. Perancangan Sistem *Automatic Question Generation*
   Perancangan sistem *Automatic Question Generation* (AQG) bertujuan untuk mendefinisikan alur kerja sistem secara menyeluruh, mulai dari pemrosesan data hingga menghasilkan pertanyaan yang sesuai. Proses ini dapat dilihat pada Gambar 2.



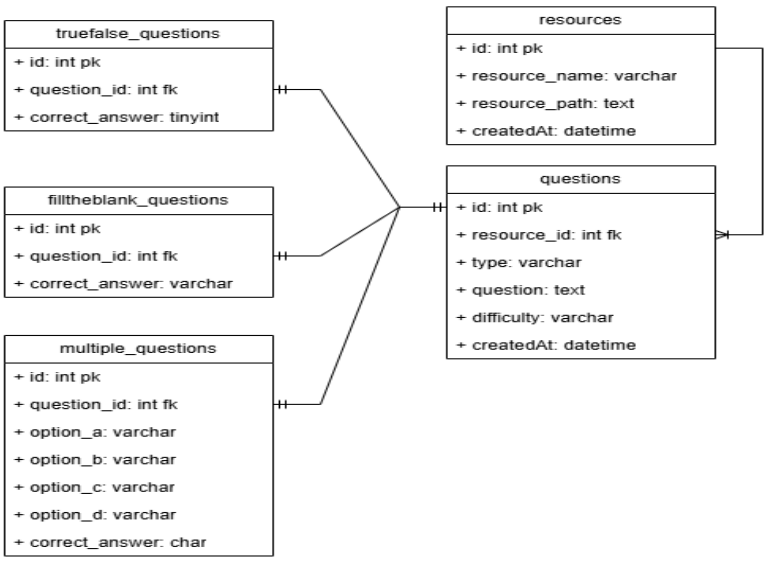

**Gambar 2.** Alur *Automatic Question Generation*

Gambar 2 menunjukkan diagram alur kerja sistem AQG yang meliputi proses dari pengolahan data hingga *output* berupa pertanyaan dan jawaban. Penjelasan dari masing-masing alur kerja tersebut adalah sebagai berikut:

1. *Upload Data Resource*
   Sumber data pada penelitian ini berupa dokumen PDF yang berisi materi pelajaran diunggah ke sistem.
2. *Extract Text*
   Setiap file PDF diolah menggunakan *PyPDFLoader* dari pustaka *LangChain* untuk mengekstrak seluruh konten teks sebagai *input* utama bagi sistem.

BULLETIN OF COMPUTER SCIENCE RESEARCH
ISSN 2774-3659 (Media Online)
Vol 5, No 4, June 2025 | Hal 434-446
https://hostjournals.com/bulletincsr
DOI: 10.47065/bulletincsr.v5i4.558

3. *Chunk Text*
   Teks hasil ekstraksi kemudian dipecah menjadi potongan–potongan kecil (*chunks*) menggunakan *RecursiveCharacterTextSplitter* dari pustaka *langchain_text_splitters*. Setiap *chunk* memiliki panjang maksimal dan *overlap* yang telah ditentukan agar memudahkan analisis, *embedding*, dan pemrosesan lebih lanjut oleh model LLM.
4. *Embedding*
   Setelah teks dipecah menjadi potongan–potongan kecil, setiap *chunk* dikonversi menjadi representasi vektor numerik menggunakan *OpenAIEmbeddings* dari pustaka *langchain_openai*.
5. *Save to FAISS Vector Store*
   Setelah proses menjadi representasi vektor (*embedding*), selanjutnya data tersebut disimpan dalam *vector store*, yang bertugas mengindeks dan menyimpan embedding sehingga dapat diakses kembali dengan cepat. Proses *embedding* dan penyimpanan dilakukan sekali saat *setup*, sehingga tahap pencarian dokumen relevan (*retrieval*) dapat mengandalkan operasi *similarity search* pada vektor yang sudah ada.
6. *FAISS (Vector Store)*
   Tempat Representasi vektor yang dihasilkan melalui proses *embedding* disimpan dalam *vector store* menggunakan *FAISS* dari pustaka *langchain_community.vectorstores*. *Facebook AI Similarity Search* (FAISS) menyediakan indeks berbasis *Cosine distance* yang sangat efisien untuk menyimpan dan melakukan pencarian (*similarity search*) pada jutaan vektor secara cepat.
7. *Check Resource*
   Saat ingin membuat soal, sistem melakukan pencarian vektor menggunakan fungsi *similarity_search_with_score* dari FAISS untuk menemukan *chunk–chunk* paling relevan dengan konteks yang dimasukkan.
8. *Prompting*
   Pada tahap *prompting*, sistem menggunakan pendekatan *few-shot prompting* untuk memastikan keluaran model sesuai spesifikasi penelitian. Instruksi disusun mencakup konteks materi, tipe soal (pilihan ganda, benar–salah, atau isian), kategori materi, level kognitif Taksonomi Bloom (C1-C3), dan pedoman format *JavaScript Object Notation* (JSON) yang dihasilkan oleh *JsonOutputParser*. Sebagai bagian dari *few-shot prompting*, disertakan beberapa contoh skema JSON (*MultipleQuestion, TrueFalseQuestion, FillTheBlankQuestion*) sehingga model memperoleh referensi konkret tentang struktur data yang diharapkan. Selain itu, *prompt* juga mengharuskan penggunaan bahasa Indonesia, output JSON tanpa field atau penjelasan tambahan, format jawaban, serta penyesuaian pada level soal yang dipilih. *Prompt* lengkap kemudian dikirim melalui metode *generate_question* pada objek LLM, yang berfungsi sebagai penghubung ke GPT-4o.
9. *Generate Questions and Answers with LLM*
   Pada tahap ini, *prompt* yang telah diformat dikirimkan ke LLM melalui metode *generate_question* untuk menghasilkan keluaran JSON. Keluaran tersebut kemudian diparsing oleh *JsonOutputParser* menjadi struktur data pertanyaan dan jawaban yang mencakup tipe soal, kategori, level kognitif, opsi, dan jawaban benar. Data tersebut langsung disimpan dalam *database* MySQL untuk keperluan manajemen soal dan evaluasi.
10. *Save Questions and Answers to MySQL Database*
    Hasil pertanyaan dan jawaban yang dihasilkan oleh sistem disimpan pada *database* MySQL sebagai bagian dari manajemen soal dan evaluasi. Struktur basis data dirancang untuk memfasilitasi penyimpanan berbagai jenis soal yang terhubung dengan tabel utama *questions*. Desain relasional ini divisualisasikan dalam *Entity Relationship Diagram* (ERD) yang dapat dilihat pada Gambar 3.




**Gambar 3.** ERD *Question Management*

BULLETIN OF COMPUTER SCIENCE RESEARCH
ISSN 2774-3659 (Media Online)
Vol 5, No 4, June 2025 | Hal 434-446
https://hostjournals.com/bulletincsr
DOI: 10.47065/bulletincsr.v5i4.558

Gambar 3 menunjukkan ERD untuk manajemen soal pada sistem. Tabel utama *questions* berfungsi sebagai pusat data soal, yang mencakup kolom seperti *id*, *resource_id*, *type*, *question, difficulty*, dan *createdAt* yang mengacu pada sumber materi dari tabel *resources*. Setiap soal yang disimpan dapat berupa salah satu dari tiga jenis, yaitu pilihan ganda, benar/salah, atau isian singkat, yang masing-masing direpresentasikan dalam tabel *multiple_questions, truefalse_questions, dan filltheblank_questions.*

11. *Query Questions and Answers from Database*
    Pada tahap ini sistem mengirimkan kueri SQL ke *database* MySQL untuk mengambil informasi soal dan jawaban secara lengkap. Pertama sistem memperoleh nomor identifikasi, teks pertanyaan, kategori materi, dan tingkat kesulitan dari tabel *questions* yang terhubung dengan tabel *resources* untuk keterangan kategori. Selanjutnya sistem menyesuaikan jenis soal, lalu menjalankan kueri ke tabel *multiple_questions, truefalse_questions,* atau *filltheblank_questions* untuk mendapatkan opsi jawaban dan kunci jawaban.
12. *Display Question and Answer*
    Pada tahap ini sistem mengambil data soal dan jawaban yang telah diproses dan disimpan sebelumnya, kemudian menyajikannya melalui antarmuka pengguna.

### 2.1.3 Implementasi

Pada tahap implementasi, Sistem AQG diimplementasikan menggunakan arsitektur *microservices*, dengan pemisahan layanan *frontend* dan *backend* untuk mendukung modularitas dan skalabilitas. *Frontend* bertugas sebagai antarmuka pengguna, sedangkan *backend* mengelola logika bisnis, penyimpanan, dan pengambilan data melalui REST API. Setiap layanan dikembangkan, diuji, dan dipelihara secara independen sehingga memudahkan perluasan fungsionalitas dan pemeliharaan di masa depan.

a. Implementasi *Frontend*
   *Frontend* dikembangkan menggunakan *Next.js*, sebuah *framework React* yang memungkinkan pengembangan antarmuka pengguna yang dinamis dan efisien. Menurut [19], *Next.js* menyediakan serangkaian teknik optimasi komprehensif seperti *server-side rendering* (SSR), *Incremental Static Regeneration* (ISR), *code splitting,* dan dukungan multi-bahasa, yang secara keseluruhan berkontribusi pada waktu muat lebih cepat dan pengalaman pengguna yang lebih baik. Hal ini ditegaskan juga oleh penelitian [20] yang menjelaskan bahwa kemampuan SSR pada *Next.js* secara efektif mengatasi kendala pemuatan halaman yang lambat dan keterbatasan *Search Engine Optimization* (SEO) yang sering dihadapi oleh aplikasi *React* murni yang mengandalkan *client-side rendering.*
b. Implementasi *Backend*
   *Backend* dikembangkan menggunakan *Python* dengan *framework FastAPI* untuk menangani *question-service,* serta *Express.js* untuk *service* lainnya seperti *cat-service. FastAPI* dipilih karena kemampuannya dalam membangun *web service* yang mendukung integrasi sistem informasi *multiplatform* dengan mudah dan efisien. Penelitian oleh [21] menunjukkan bahwa *FastAPI* mampu menyediakan dokumentasi otomatis yang memudahkan klien dalam menggunakan *endpoint* yang disediakan, serta memiliki performa yang baik di bawah beban tinggi, dengan waktu respons rata-rata 6.198 ms saat diakses oleh 1.000 pengguna. Sementara itu, *Express.js* digunakan karena fleksibilitasnya dalam membangun layanan berbasis *Node.js* dan waktu eksekusi yang cepat. Penelitian oleh [22] menunjukkan bahwa *Express.js* memiliki waktu eksekusi rata-rata 26,85% lebih cepat dibandingkan *framework Hono,* menjadikannya efisien untuk pengembangan *service* dalam arsitektur *microservices.*

### 2.1.4 Pengujian

Tahap pengujian merupakan langkah akhir dalam penelitian ini, yang dilakukan setelah implementasi sistem selesai. Berikut penjelasan dari tahap ini:

a. *User Acceptance Test* (UAT)
   *User Acceptance Test* (UAT) adalah tahap akhir dalam pengembangan aplikasi yang bertujuan untuk mengevaluasi apakah sistem beroperasi dengan baik serta memenuhi kebutuhan pengguna sebelum digunakan secara umum [23]. Selain itu skala *Likert* juga diterapkan untuk menilai persepsi, sikap, dan opini responden [24]. Dalam penelitian ini, UAT melibatkan enam guru untuk mengevaluasi fitur sistem, termasuk generator pertanyaan, manajemen *course, modul, exam*, dan antarmuka pengguna. Guru melakukan evaluasi kualitas konten soal yang dihasilkan dengan mengisi formulir evaluasi sebagaimana tercantum pada Tabel 2. Dalam formulir tersebut, pertanyaan kunci terkait kualitas soal adalah “Apakah kualitas pertanyaan dan jawaban yang dihasilkan oleh fitur generator otomatis telah sesuai dengan materi pembelajaran?”, pertanyaan ini secara implisit telah mencakup tiga aspek utama, yaitu keakuratan, relevansi, dan tingkat kesulitan. Setiap guru merespon pertanyaan tersebut menggunakan skala *Likert* 1 sampai 5, sehingga persepsi guru terhadap ketiga aspek tersebut dapat terukur melalui satu indikator. Selain itu, dua belas mahasiswa teknik informatika menguji alur penggunaan aplikasi secara menyeluruh, mencakup proses *enrollment course*, pelaksanaan ujian, hingga pengiriman hasil. Selama pelaksanaan UAT, sistem dievaluasi berdasarkan beberapa metrik, yaitu tingkat kepuasan pengguna terhadap kualitas soal yang dihasilkan oleh generator otomatis, kemudahan penggunaan sistem oleh guru dan mahasiswa, dan relevansi soal dengan materi pembelajaran. Masing-masing metrik dinilai melalui pertanyaan dan pernyataan dalam formulir evaluasi pada Tabel 2 dan Tabel 3 yang direspon menggunakan skala *Likert*. Data dari tiap metrik digunakan untuk menilai sejauh mana sistem memenuhi kebutuhan pengguna dan sebagai dasar perbaikan sistem pada tahap berikutnya.

BULLETIN OF COMPUTER SCIENCE RESEARCH
ISSN 2774-3659 (Media Online)
Vol 5, No 4, June 2025 | Hal 434-446
https://hostjournals.com/bulletincsr
DOI: 10.47065/bulletincsr.v5i4.558

b. *DeepEval*
*DeepEval* merupakan *framework open-source* yang digunakan untuk mengevaluasi performa *Large Language Models* (LLM) [25]. Evaluasi ini mencakup beberapa aspek utama, yaitu tingkat *hallucination, contextual precision, contextual relevancy, answer relevancy,* dan *prompt alignment.*

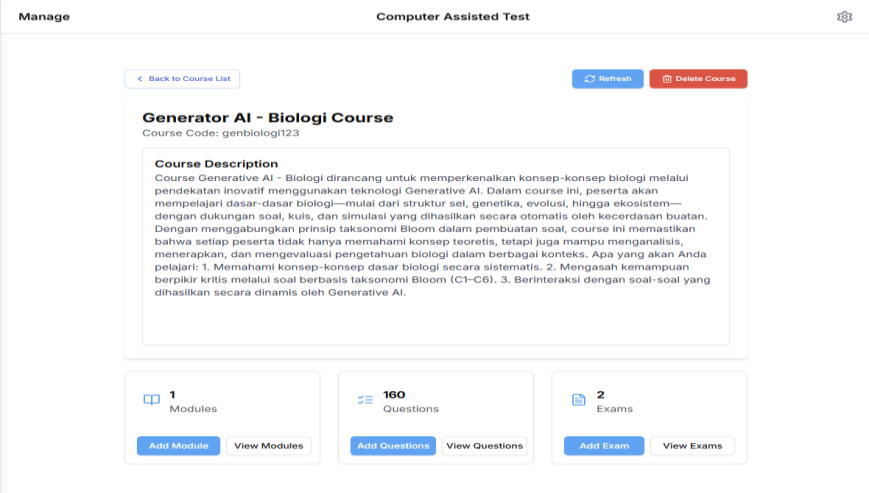
# 3. HASIL DAN PEMBAHASAN

## 3.1 Implementasi Sistem

Implementasi sistem ini dilakukan dengan arsitektur *microservices* yang terdiri dari *frontend* dan *backend.* Berikut penjelasan dari implementasi ini:

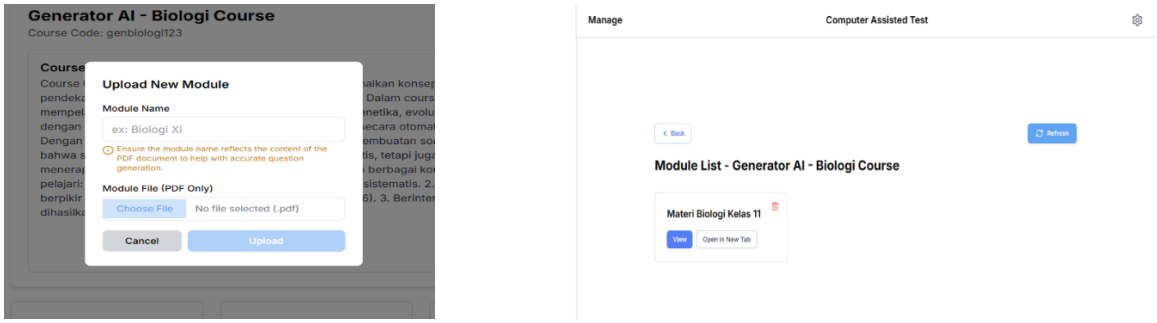
### 3.1.1 Implementasi *Frontend*

Frontend dikembangkan menggunakan *Next.js*, sebuah *framework React* yang memungkinkan pembuatan antarmuka pengguna yang dinamis. Antarmuka ini memungkinkan pengguna untuk mengunggah dokumen PDF, mengatur parameter pembuatan soal, dan melihat hasil soal yang dihasilkan.




**Gambar 4.** Tampilan *Manage Course*

Gambar 4 menampilkan halaman *Manage Course* pada sistem CAT, pengguna dapat melihat deskripsi kursus dan ringkasan jumlah modul, soal, dan ujian yang telah dibuat, sekaligus menggunakan tombol *Refresh*, *Delete Course, Add Module, View Modules, Add Questions, View Questions, Add Exam dan View Exams* untuk mengelola konten dan pengaturan kursus.

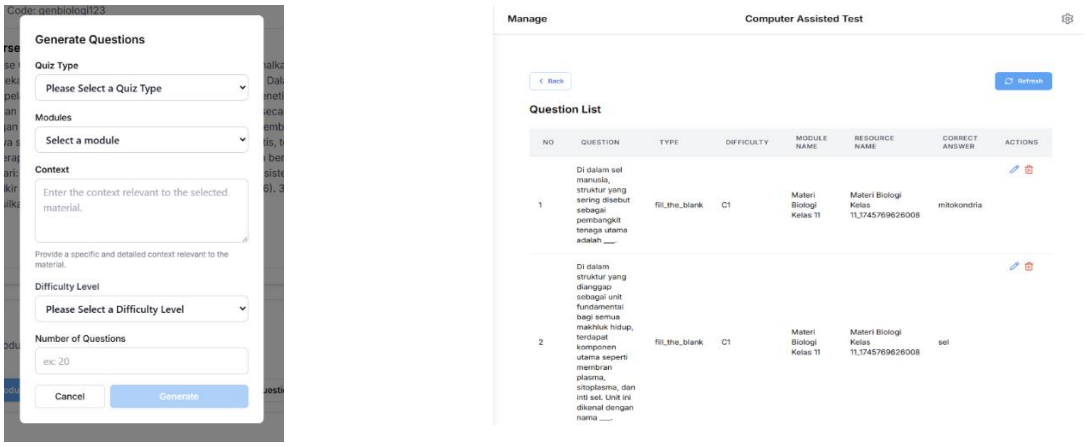



(a) (b)

**Gambar 5.** (a) Tampilan *Upload Modules* dan (b) *Module List*

Gambar 5 merupakan dua tampilan pada sistem CAT, yaitu layar *pop-up Upload New Module* dan halaman *Module List.* Pada Gambar 5(a) adalah layar *pop-up Upload New Module* di mana pengguna memasukkan nama modul, memilih *file* PDF materi lewat tombol *Choose File* lalu mengunggahnya dengan menekan tombol *Upload* atau membatalkan proses dengan tombol *Cancel*. Pada Gambar 5(b) adalah halaman *Module List* pada sistem CAT. Pengguna dapat kembali ke halaman sebelumnya dengan tombol *Back*, memuat ulang daftar modul dengan tombol *Refresh*, serta melihat daftar modul. Setiap modul ditampilkan sebagai kartu berisi judul modul, ikon *Delete*, tombol *View* dan *Open in New Tab* untuk mengakses materi.

BULLETIN OF COMPUTER SCIENCE RESEARCH
ISSN 2774-3659 (Media Online)
Vol 5, No 4, June 2025 | Hal 434-446
https://hostjournals.com/bulletincsr
DOI: 10.47065/bulletincsr.v5i4.558




(a) (b)

**Gambar 6.** (a) Tampilan *Generate Questions* dan (b) *Question List*

Gambar 6 merupakan dua tampilan pada sistem CAT, yaitu layar *pop-up Generate Questions* dan halaman *Question List*. Pada Gambar 6(a) adalah layar *pop-up Generate Questions.* Pengguna dapat memilih jenis kuis melalui *dropdown Quiz Type*, memilih modul melalui *dropdown Modules*, memasukkan konteks soal yang relevan pada kolom *Context*, menentukan tingkat kesulitan lewat *dropdown Difficulty Level*, memasukkan jumlah soal pada kolom *Number of Questions* lalu memulai pembuatan soal dengan tombol *Generate* atau membatalkan proses dengan tombol *Cancel*. Pada Gambar 6(b) adalah halaman *Question List* pada sistem CAT. Pengguna dapat kembali ke halaman sebelumnya dengan tombol *Back*, memuat ulang daftar soal dengan tombol *Refresh*, serta melihat tabel yang menampilkan nomor urut, teks soal, tipe soal, tingkat kesulitan, nama modul, nama sumber, jawaban benar, dan ikon aksi *Edit* dan *Delete* untuk setiap soal.

### 3.1.2 Implementasi *Backend*

Backend terdiri dari beberapa *service* yang dikembangkan menggunakan *FastAPI* dan *Express.js*:

a. *Generator-Service*
*Service* ini dikembangkan dengan *FastAPI*, *service* ini bertanggung jawab untuk memproses dokumen yang diunggah, membagi teks menjadi potongan kecil (*chunking*), melakukan *embedding*, dan menyimpan representasi vektor menggunakan FAISS sebagai basis pengetahuan. Selain itu, *Generator-Service* juga bertanggung jawab untuk menghasilkan soal secara otomatis berdasarkan konteks materi yang telah diproses, dengan mengirimkan *prompt* ke LLM untuk menghasilkan soal sesuai dengan level dan tipe soal yang diinginkan. *Endpoint* yang digunakan dapat dilihat pada Tabel 1.

**Table 1.** *Generator-Service Endpoint*

| Endpoint | Fungsi |
|---|---|
| /upload_resource | Menerima dan menyimpan dokumen sumber (PDF), melakukan ekstrasi teks, *chunk* teks, *embedding,* dan penyimpanan representasi vektor ke FAISS |
| /generate_question | Menerima permintaan generate soal beserta parameter, memanggil LLM untuk menghasilkan soal otomatis berdasarkan konteks, lalu menyimpan hasil soal ke database |

*b. Cat-Service*
*Service* ini dikembangkan menggunakan *Express.js* dan berperan dalam mengelola informasi pengguna, manajemen kursus, soal, ujian, dan skor ujian. Selain itu, *service* ini juga mendukung pelaksanaan ujian secara langsung oleh siswa melalui antarmuka sistem, serta pencatatan hasil pengerjaan untuk keperluan evaluasi.

## 3.2 Pengujian Sistem

Setelah implementasi, sistem diuji melalui dua metode utama, yaitu *User Acceptance Test* (UAT) dan evaluasi menggunakan *DeepEval*.

### 3.2.1 *User Acceptance Test (UAT)*

Dalam penelitian ini, dilakukan evaluasi terhadap kualitas sistem yang dikembangkan untuk menilai kelayakan serta kesesuaian output berdasarkan persepsi guru dan mahasiswa. Hasil evaluasi disajikan pada Tabel 2 dan Tabel 3, yang menampilkan skor untuk setiap jawaban sesuai dengan kriteria yang telah ditentukan.

BULLETIN OF COMPUTER SCIENCE RESEARCH
ISSN 2774-3659 (Media Online)
Vol 5, No 4, June 2025 | Hal 434-446
https://hostjournals.com/bulletincsr
DOI: 10.47065/bulletincsr.v5i4.558

Evaluasi dilakukan dengan memberikan penilaian terhadap setiap kriteria yang telah ditetapkan untuk menilai kualitas dan kesesuaian sistem. Setiap jawaban yang diberikan dibobotkan berdasarkan tingkat persetujuan, guna memperoleh skor akhir yang mencerminkan indeks kepuasan pengguna terhadap sistem secara keseluruhan. Adapun bobot yang digunakan yaitu jawaban Sangat Setuju (SS) diberi bobot 5, Setuju (S) diberi bobot 4, Ragu-ragu (RG) diberi bobot 3, Tidak Setuju (TS) diberi bobot 2, dan Sangat Tidak Setuju (STS) diberi bobot 1. Skor akhir dihitung berdasarkan akumulasi dari nilai-nilai tersebut.

**Table 2.** Hasil Evaluasi Oleh Guru

| No | Pertanyaan | Jawaban | | | | |
|---|---|---|---|---|---|---|
| | | SS | S | RG | TS | STS |
| 1 | Apakah sistem ini mudah untuk digunakan? | 4 | 2 | 0 | 0 | 0 |
| 2 | Apakah tampilan dari sistem ini menarik? | 1 | 4 | 0 | 1 | 0 |
| 3 | Apakah fitur-fitur dari sistem ini dapat membantu anda? | 5 | 1 | 0 | 0 | 0 |
| 4 | Apakah pada sistem ini, modul dapat ditambahkan dengan mudah melalui tombol *Add Module*? | 6 | 0 | 0 | 0 | 0 |
| 5 | Apakah pada sistem ini, exam atau ujian dapat ditambahkan dengan mudah melalui tombol *Add Exam*? | 6 | 0 | 0 | 0 | 0 |
| 6 | Apakah pengajar dapat melihat seluruh soal yang telah ditambahkan dengan jelas? | 4 | 2 | 0 | 0 | 0 |
| 7 | Apakah kualitas pertanyaan dan jawaban yang dihasilkan oleh fitur generator otomatis telah sesuai dengan materi pembelajaran? | 2 | 4 | 0 | 0 | 0 |
| 8 | Apakah respons sistem ini berjalan dengan semestinya? | 5 | 1 | 0 | 0 | 0 |
| 9 | Apakah fitur dari generator soal otomatis pada sistem *Computer Assisted Test* (CAT) ini dapat membantu meningkatkan kepuasan pengajar terhadap proses evaluasi pembelajaran? | 3 | 3 | 0 | 0 | 0 |
| 10 | Apakah anda akan menggunakan sistem ini lagi? | 4 | 2 | 0 | 0 | 0 |
| | Total | 40 | 19 | 0 | 1 | 0 |

Pada Tabel 2, setelah dibobot skor yang diperoleh adalah 200 untuk SS, 76 untuk S, 0 untuk RG, 2 untuk TS, dan 0 untuk STS, sehingga totalnya 278. Jika dibagi dengan skor maksimal 6 responden × 10 pertanyaan × 5 bobot tertinggi = 300, maka persentase indeks kepuasan adalah 278/300 × 100% ≈ 92,7%. Interval penilaian yang ada pada bobot skala likert sebagai berikut:

a. Indeks 0% – 19,99% : Sangat Tidak Setuju
b. Indeks 20% – 39,99% : Tidak Setuju
c. Indeks 40% – 59,99% : Kurang Setuju
d. Indeks 60% – 79,99% : Setuju
e. Indeks 80% – 100% : Sangat Setuju

Nilai indeks 92,7 % menunjukkan bahwa responden guru "Sangat Setuju" terhadap kinerja sistem, yang memudahkan mereka membuat *course*, mengunggah modul, membuat soal otomatis, manajemen ujian, dan memantau skor siswa. Tingginya persentase ini disebabkan oleh kesesuaian fitur unggah modul dan generator soal otomatis dengan kebutuhan guru, hampir semua guru menilai sistem berjalan andal tanpa kendala teknis sehingga alur pembuatan dan manajemen soal menjadi sangat efisien.

**Tabel 3.** Hasil Evaluasi Oleh Mahasiswa

| No | Pernyataan | Jawaban | | | | |
|---|---|---|---|---|---|---|
| | | SS | S | RG | TS | STS |
| 1 | Saya berpikir akan menggunakan sistem ini lagi | 3 | 6 | 3 | 0 | 0 |
| 2 | Saya merasa sistem ini tidak rumit untuk digunakan | 0 | 3 | 4 | 5 | 0 |
| 3 | Saya merasa sistem ini mudah digunakan | 0 | 9 | 3 | 0 | 0 |
| 4 | Saya tidak membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini | 1 | 4 | 5 | 2 | 0 |
| 5 | Saya merasa fitur-fitur sistem ini berjalan dengan semestinya | 1 | 8 | 1 | 2 | 0 |
| 6 | Saya tidak merasa ada banyak hal yang tidak konsisten (tidak serasi pada sistem ini) | 1 | 4 | 3 | 3 | 1 |
| 7 | Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat | 0 | 7 | 4 | 1 | 0 |
| 8 | Saya merasa sistem ini tidak membingungkan | 0 | 3 | 2 | 6 | 1 |
| 9 | Saya merasa tidak ada hambatan dalam menggunakan sistem ini | 2 | 8 | 2 | 0 | 0 |
| 10 | Saya tidak perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini | 0 | 6 | 2 | 3 | 0 |
| | Total | 8 | 58 | 29 | 22 | 2 |

BULLETIN OF COMPUTER SCIENCE RESEARCH
ISSN 2774-3659 (Media Online)
Vol 5, No 4, June 2025 | Hal 434-446
https://hostjournals.com/bulletincsr
DOI: 10.47065/bulletincsr.v5i4.558

Pada Tabel 3, setelah dibobot skor yang diperoleh adalah 40 untuk SS, 232 untuk S, 87 untuk RG, 44 untuk TS, dan 2 untuk STS, sehingga totalnya 405. Jika dibagi dengan skor maksimal 12 responden × 10 pertanyaan × 5 bobot tertinggi = 600, maka persentase indeks kepuasan adalah 405/600 × 100% ≈ 67,5%. Interval penilaian yang ada pada bobot skala likert sebagai berikut:

a. Indeks 0% – 19,99% : Sangat Tidak Setuju
b. Indeks 20% – 39,99% : Tidak Setuju
c. Indeks 40% – 59,99% : Kurang Setuju
d. Indeks 60% – 79,99% : Setuju
e. Indeks 80% – 100% : Sangat Setuju

Nilai indeks sebesar 67,5% menunjukkan bahwa responden mahasiswa menyatakan "Setuju" terhadap kinerja sistem. Hal ini berarti mahasiswa dapat menggunakan sistem untuk membuat akun, bergabung ke dalam *course*, mempelajari modul, mengerjakan ujian, dan melihat skor dengan lancar. Namun, akses yang dimiliki mahasiswa berbeda dengan guru karena mereka hanya dapat menggunakan fitur yang berkaitan dengan pembelajaran tanpa dapat mengunggah materi atau membuat soal. Meskipun secara umum mahasiswa merasa sistem ini berfungsi dengan baik, beberapa masih mengalami kendala pada tahap awal penggunaan dan menemukan tampilan yang kurang konsisten, seperti tombol yang membingungkan atau fitur yang tidak sesuai dengan peran mereka. Oleh karena itu, sebagian mahasiswa merasa perlu adanya panduan tambahan agar lebih mudah memahami cara kerja sistem.

### 3.2.2 *DeepEval*

Evaluasi selanjutnya adalah evaluasi menggunakan *framework DeepEval.* Nilai yang akan dicari pada evaluasi ini adalah *hallucination, contextual precision, contextual relevancy, answer relevancy,* dan *prompt alignment.*

a. *Hallucination*
Nilai yang digunakan untuk mengukur seberapa sering model membuat klaim atau menambahkan informasi yang tidak didukung oleh konteks. Misalnya, menyebutkan data statistik yang tidak ada di dokumen sumber.
b. *Contextual Precision*
Nilai yang digunakan untuk mengukur kemampuan *Retrieval Augmented Generation* (RAG) dalam menempatkan potongan teks relevan pada peringkat lebih tinggi daripada yang tidak relevan. Dengan metrik ini, model mengambil informasi yang tepat dari dokumen sebelum menjawab, sehingga dapat mengurangi risiko jawaban melantur ke hal tidak relevan.
c. *Contextual Relevancy*
Nilai yang digunakan untuk mengukur keseluruhan seberapa cocok seluruh informasi terhadap pertanyaan atau tugas pengguna, tanpa membandingkan keluaran model. Metrik ini memastikan bahwa sebelum model mulai menghasilkan jawaban, data yang disediakan memang memadai dan tepat sasaran untuk menjawab pertanyaan.
d. *Answer Relevancy*
Nilai yang digunakan untuk mengukur seberapa baik keluaran model sesuai dengan pertanyaan atau instruksi pengguna. Dengan metrik ini, pengguna dapat mengetahui apakah jawaban yang diberikan benar-benar menjawab pertanyaan yang diminta, sehingga dapat meningkatkan kejelasan dan kegunaan respons model.
e. *Prompt Alignment*
Nilai yang digunakan untuk mengukur tingkat kepatuhan keluaran model terhadap instruksi eksplisit dalam *prompt*, misalnya jumlah soal yang diminta atau format jawaban.

**Table 4.** Hasil Evaluasi Menggunakan *DeepEval*

| No | Konteks | Tipe | Level | Hallucination | Contextual Precision | Contextual Relevancy | Answer Relevancy | Prompt Alignment |
|---|---|---|---|---|---|---|---|---|
| 1 | Terkait menjelajah sel | True False | C1 | 0 | 1 | 1 | 0,5 | 0,818 |
| 2 | Terkait menjelajah sel | True False | C2 | 0 | 1 | 1 | 0,5 | 0,8 |
| 3 | Terkait menjelajah sel | True False | C3 | 0 | 1 | 1 | 0,5 | 1 |
| 4 | Terkait menjelajah sel | Fill The Blank | C1 | 0 | 1 | 1 | 1 | 1 |
| 5 | Terkait menjelajah sel | Fill The Blank | C2 | 0 | 1 | 0,66 | 1 | 1 |
| 6 | Terkait menjelajah sel | Fill The Blank | C3 | 1 | 1 | 1 | 1 | 0,86 |
| 7 | Terkait menjelajah sel | Multiple Choice | C1 | 0 | 1 | 1 | 1 | 0,95 |
| 8 | Terkait menjelajah sel | Multiple Choice | C2 | 0 | 1 | 1 | 0,66 | 0,45 |

BULLETIN OF COMPUTER SCIENCE RESEARCH
ISSN 2774-3659 (Media Online)
Vol 5, No 4, June 2025 | Hal 434-446
https://hostjournals.com/bulletincsr
DOI: 10.47065/bulletincsr.v5i4.558

| No | Konteks | Tipe | Level | Hallucination | Contextual Precision | Contextual Relevancy | Answer Relevancy | Prompt Alignment |
|---|---|---|---|---|---|---|---|---|
| 9 | Terkait menjelajah sel | Multiple Choice | C3 | 0 | 1 | 1 | 0,66 | 1 |
| 10 | Terkait pergerakan zat melalui membran sel | True False | C1 | 0 | 1 | 0,66 | 0,5 | 0,81 |
| : | : | : | : | : | : | : | : | : |
| 18 | Terkait pergerakan zat melalui membran sel | Multiple Choices | C3 | 0 | 1 | 1 | 0,66 | 1 |
| 19 | Terkait proses pengaturan pada tumbuhan | True False | C1 | 0 | 1 | 1 | 0,5 | 1 |
| : | : | : | : | : | : | : | : | : |
| 27 | Terkait proses pengaturan pada tumbuhan | Multiple Choices | C3 | 0 | 1 | 1 | 0,66 | 1 |
| 28 | Terkait transport dan pertukaran zat pada manusia | True False | C1 | 0 | 1 | 1 | 0,5 | 1 |
| : | : | : | : | : | : | : | : | : |
| 36 | Terkait transport dan pertukaran zat pada manusia | Multiple Choices | C3 | 0 | 1 | 1 | 0,66 | 1 |
| 37 | Terkait sistem pertahanan tubuh terhadap penyakit | True False | C1 | 0 | 1 | 1 | 0,5 | 1 |
| : | : | : | : | : | : | : | : | : |
| 45 | Terkait sistem pertahanan tubuh terhadap penyakit | Multiple Choices | C3 | 0 | 1 | 1 | 0,66 | 1 |
| 46 | Terkait mobilitas pada manusia | True False | C1 | 0 | 1 | 0,66 | 0,5 | 1 |
| : | : | : | : | : | : | : | : | : |
| 54 | Terkait mobilitas pada manusia | Multiple Choices | C3 | 0 | 1 | 0,33 | 0,66 | 1 |
| 55 | Terkait hormon dalam reproduksi manusia | True False | C1 | 0 | 1 | 0,66 | 0,5 | 1 |
| : | : | : | : | : | : | : | : | : |
| 63 | Terkait hormon dalam reproduksi manusia | Multiple Choices | C3 | 0 | 1 | 0,66 | 0,66 | 1 |
| 64 | Terkait tumbuh kembang makhluk hidup | True False | C1 | 0 | 1 | 0,3 | 0,5 | 1 |
| : | : | : | : | : | : | : | : | : |
| 72 | Terkait tumbuh kembang makhluk hidup | Multiple Choices | C3 | 0 | 1 | 0,66 | 0,66 | 1 |

BULLETIN OF COMPUTER SCIENCE RESEARCH
ISSN 2774-3659 (Media Online)
Vol 5, No 4, June 2025 | Hal 434-446
https://hostjournals.com/bulletincsr
DOI: 10.47065/bulletincsr.v5i4.558

| No | Konteks | Tipe | Level | Hallucination | Contextual Precision | Contextual Relevancy | Answer Relevancy | Prompt Alignment |
|---|---|---|---|---|---|---|---|---|
| Jumlah | | | | 2,66 | 70,16 | 59,98 | 50,86 | 66,585 |
| Rata-rata | | | | 0,0369 | 0,9744 | 0,833 | 0,7063 | 0,9247 |

Pada Tabel 4, perhitungan dimulai dengan menjumlahkan skor setiap metrik dari 72 percobaan (8 konteks soal × 3 tipe soal × 3 level) yang diambil dari delapan bab buku biologi, lalu membaginya dengan jumlah percobaan untuk memperoleh rata rata dalam rentang 0,0 hingga 1,0 dan mengonversi hasil tersebut ke presentase dengan mengalikan dengan 100, sehingga menghasilkan 3,69% untuk *hallucination,* 97,44% untuk *contextual precision*, 83,30% untuk *contextual relevancy*, 70,63% untuk *answer relevancy*, dan 92,47% untuk *prompt alignment.* Nilai *hallucination* yang sangat rendah menunjukkan bahwa model jarang menghasilkan informasi yang tidak sesuai dengan konteks sumber, yang berarti performa model sangat baik dalam hal akurasi informasi, sehingga soal-soal yang dihasilkan minim kesalahan fakta dan tetap selaras dengan materi. Sementara itu, nilai tinggi pada *contextual precision* dan *contextual relevancy* menandakan bahwa sistem *retrieval* dan pemanfaatan konteks berjalan secara optimal. Hal ini berimplikasi pada kualitas soal yang lebih baik, karena sumber informasi yang diolah adalah yang paling relevan. *Answer relevancy* yang mencapai 70,63% menunjukkan bahwa sebagian besar jawaban model relevan dengan pertanyaan. Namun, nilai ini juga menunjukkan adanya ruang untuk peningkatan, terutama mengingat keterbatasan metrik *answer relevancy* yang secara desain lebih cocok untuk evaluasi tanya-jawab sederhana dibandingkan *assessment* pendidikan dengan format multi-level. Hal ini menunjukkan adanya *trade-off* antara kualitas isi dengan struktur atau format jawaban yang dihasilkan model. Nilai *prompt alignment* yang tinggi menunjukkan bahwa model mampu mengikuti instruksi secara konsisten sesuai dengan format dan jumlah soal yang diminta. Berdasarkan standar evaluasi yang merujuk pada penelitian [26] dan [27], nilai *hallucination* dinilai baik jika berada di bawah 0,5 (50%), sementara metrik lainnya dianggap baik jika berada di atas 0,5. Oleh karena itu, dapat disimpulkan bahwa sistem yang dikembangkan telah menunjukkan kinerja yang baik dan efektif dalam menghasilkan soal otomatis yang relevan dan kontekstual.

**Table 5.** Contoh Soal Dan Jawaban Yang Dihasilkan

| No | Pertanyaan | Tipe | Level | Jawaban |
|---|---|---|---|---|
| 1 | Membran inti adalah bagian terpenting yang memisahkan inti sel dari sitoplasma dalam sel eukariotik | True False | C1 | True |
| 2 | Transportasi melalui membran sel dapat dibedakan menjadi aktif dan pasif dengan mengamati kebutuhan energi. Benara tau salah, transport aktif memerlukan energi ATP untuk memindahkan molekul melawan gradien konsentrasi, sedangkan transpor pasif memerlukan perbedaan gradien konsentrasi saja tanpa energi tambahan? | True False | C2 | True |
| 3 | Saraf kranial dapat mempengaruhi gerakan otot rangka yang terlibat dalam proses bicara | True False | C3 | True |
| 4 | Hormon yang berperan penting dalam perkembangan ciri-ciri fisik pria dewasa, termasuk karakteristik seksual sekunder, adalah ___ | Fill The Blank | C1 | Testosteron |
| 5 | Dalam sel prokariotik, seperti bakteri, materi genetik tidak terlindungi oleh struktur yang dikenal sebagai ___, yang membedakannya dari sel eukariotik | Fill The Blank | C2 | Membran inti |
| 6 | Dalam keadaan normal, paru-paru manusia menggunakan mekanisme ___ untuk menukarkan oksigen dan karbon dioksida selama proses pernapasan | Fill The Blank | C3 | Difusi |
| 7 | Dalam dunia botani, bagaimana Anda menamai jaringan pada tumbuhan yang berfungsi untuk pertumbuhan yang tidak terbatas? | Multiple Choices | C1 | Jaringan meristem |
| 8 | Bisakah Anda menjelaskan bagaimana proses pertukaran gas berlangsung di paru-paru manusia dan peran komponen-komponennya? | Multiple Choices | C2 | Gas seperti oksigen dan karbon dioksida berdifusi melalui membran alveolus dengan bantuan sel darah merah dan plasma darah |
| 9 | Bagaimana jaringan meristem berperan dalam pertumbuhan tinggi tanaman? | Multiple Choices | C3 | Dengan membelah diri untuk menambah jumlah sel di bagian ujung batang |

Tabel 5 menyajikan sembilan contoh soal beserta jawaban yang dihasilkan sistem secara otomatis dari materi biologi kelas 11. Jenis soal mencakup *true-false*, *fill the blank*, dan *multiple choices* pada level kognitif C1 hingga C3.

Contoh ini menunjukkan sistem bahwa sistem mampu menghasilkan soal yang bervariasi dan relevan sesuai yang diharapkan.

# 4. KESIMPULAN

Berdasarkan hasil penelitian ini, dapat disimpulkan bahwa penerapan kerangka kerja *LangChain* yang terintegrasi dengan *Large Language Models* (LLM), khususnya GPT-4o dalam sistem *Automatic Question Generation* (AQG) untuk *Computer Assisted Test* (CAT) terbukti mampu menghasilkan soal otomatis yang secara persepsi pengguna mudah digunakan dan secara teknis memiliki tingkat halusinasi yang rendah dan presisi kontekstual yang tinggi. Evaluasi *User Acceptance Test* (UAT) menunjukkan tingkat kepuasan tinggi dari guru dengan skor rata-rata 92,7%, serta respons positif dari mahasiswa dengan skor rata-rata 67,5%, menegaskan bahwa antarmuka dan alur kerja sistem mudah digunakan dan efektif dalam membantu proses penyusunan soal dan pengerjaan ujian. Hasil metrik *DeepEval* memperlihatkan tingkat *hallucination* sebesar 3,69%, *contextual precision* mencapai 97,44%, dan *contextual relevancy* sebesar 83,30%, menegaskan bahwa mekanisme *retrieval* dan pemanfaatan konteks sumber berjalan dengan baik. Selain itu, metrik *prompt alignment* sebesar 92,47% menunjukkan bahwa output model cukup selaras dengan instruksi dan struktur yang ditetapkan dalam *prompt*. Adapun *answer relevancy* yang mencapai 70,63% menggambarkan ketepatan jawaban terhadap pertanyaan yang dihasilkan. Keterbatasan penelitian ini meliputi kajian domain yang hanya terfokus pada materi biologi kelas 11, sehingga generalisasi ke mata pelajaran lain masih perlu diuji, serta sampel UAT dan evaluasi *DeepEval* yang relatif terbatas, berpotensi dapat memunculkan hasil yang bervariasi bila diterapkan dalam skenario ujian nyata dengan populasi yang lebih besar. Untuk penelitian selanjutnya, disarankan melakukan ekstensi ke berbagai bidang studi, pengembangan soal bergambar untuk mendukung mata pelajaran yang bersifat visual, melibatkan lebih banyak responden dan skenario ujian nyata, serta mengoptimalisasi teknik RAG dengan *fine-tuning* model *embedding* dan eksplorasi teknik *prompting* guna meningkatkan hasil generasi yang lebih baik. Dengan demikian, sistem AQG berbasis *LangChain* dan LLM ini memiliki potensi besar untuk mendukung transformasi digital dalam pendidikan, sekaligus membuka peluang riset lanjutan dalam pengembangan AI edukatif.

# REFERENCES

[1] L. Chen, P. Chen, and Z. Lin, "Artificial Intelligence in Education: A Review," *IEEE Access*, vol. 8, pp. 75264–75278, 2020, doi: 10.1109/ACCESS.2020.2988510.

[2] M. Holland and K. Chaudhari, "Large language model based agent for process planning of fiber composite structures," *Manuf Lett*, vol. 40, pp. 100–103, Jul. 2024, doi: 10.1016/j.mfglet.2024.03.010.

[3] F. Hans-Georg, F. Peter, and K. Julius, "Conceptual Modeling and Large Language Models: Impressions From First ExperimentsWith ChatGPT," *Enterprise Modelling and Information Systems Architectures*, vol. 18, Jan. 2023, doi: 10.18417/emisa.18.3.

[4] S. A. M. Hogenboom, F. F. J. Hermans, and H. L. J. Van der Maas, "Computerized adaptive assessment of understanding of programming concepts in primary school children," *Computer Science Education*, vol. 32, no. 4, pp. 418–448, 2022, doi: 10.1080/08993408.2021.1914461.

[5] A. Maharani, R. Habib Adibarata, T. Anggara, and Y. Hanoselina, "Efektivitas Penggunaan Sistem Cat Dalam Penerimaan Pegawai Negeri Sipil Di Upt Bkn Padang," *Jurnal Ilmu Manajemen, Bisnis dan Ekonomi*, vol. 2, no. 3, 2024, doi: doi.org/10.59971/jimbe.v2i3.359.

[6] K. B. Utomo, A. Azizah, and M. A. Pangestu, "Peran Computer Assited Test dalam Implementasi Penilaian di SD Negeri 005 Palaran," *Jurnal Ilmu Siber dan Teknologi Digital*, vol. 1, no. 1, pp. 29–39, Nov. 2022, doi: 10.35912/jisted.v1i1.1529.

[7] E. P. Saputra, R. N. Alfiyah, and I. Indriyanti, "Computer Assessment Test at the Association of Indonesian Independent Housing Experts with Waterfall Model," *Jurnal CoreIT: Jurnal Hasil Penelitian Ilmu Komputer dan Teknologi Informasi*, vol. 9, no. 1, p. 29, Jun. 2023, doi: 10.24014/coreit.v9i1.11483.

[8] R. Setiawan, "Optimasi Pengalaman Pengguna Dan Prototyping Untuk Penilaian Otomatis Dan Pencegahan Kecurangan," *bit-Tech*, vol. 7, no. 2, pp. 299–306, Dec. 2024, doi: 10.32877/bt.v7i2.1758.

[9] I. A. Buana, M. Yunus, and S. Suratman, "Implementasi Sistem Computer-Based Test (CBT) Dalam Pengelolaan Ujian di MAN Insan Cendekia Paser," *Jurnal Tarbiyah dan Ilmu Keguruan Borneo*, vol. 5, no. 2, pp. 219–228, Mar. 2024, doi: 10.21093/jtikborneo.v5i2.7822.

[10] S. Izadi and M. Forouzanfar, "Error Correction and Adaptation in Conversational AI: A Review of Techniques and Applications in Chatbots," *AI (Switzerland)*, vol. 5, no. 2, pp. 803–841, Jun. 2024, doi: 10.3390/ai5020041.

[11] B. Ogunleye, K. I. Zakariyyah, O. Ajao, O. Olayinka, and H. Sharma, "A Systematic Review of Generative AI for Teaching and Learning Practice," *Educ Sci (Basel)*, vol. 14, no. 6, Jun. 2024, doi: 10.3390/educsci14060636.

[12] N. S. Harahap, A. Saad, and H. Ubaidullah, "Comprehensive Bibliometric Literature Review of Chatbot Research: Trends, Frameworks, and Emerging Applications," *IJACSA) International Journal of Advanced Computer Science and Applications*, vol. 16, no. 1, p. 2025, doi: 10.14569/IJACSA.2025.0160185.

[13] G. Roffo, "Exploring Advanced Large Language Models with LLMsuite," Arxiv, Jul. 2024, doi: 10.13140/RG.2.2.11774.80963.

[14] R. P. Kiran, S. Khaiyum, A. R. Palandye, and A. S. D, "Leveraging LLaMA3 and LangChain for Rapid AI Application Development," *J. Electrical Systems*, vol. 20, no. 10, pp. 2146–2153, 2024, doi: 10.52783/jes.5539.

[15] M. I. Syah, "Penerapan Retrieval Augemented Generation Menggunakan Langchain Dalam Pengembangan Sistem Tanya Jawab Hadis Berbasis Web," Zonasi, vol. 6, no. 2, 2024, doi: https://doi.org/10.31849/zn.v6i2.19940.

[16] L. Pusch and T. O. F. Conrad, "Combining LLMs and Knowledge Graphs to Reduce Hallucinations in Question Answering," *ArXiv*, Nov. 2024, doi: doi.org/10.48550/arXiv.2409.04181.
[17] S. Maity, A. Deroy, and S. Sarkar, "Leveraging In-Context Learning and Retrieval-Augmented Generation for Automatic Question Generation in Educational," *Proceedings of ACM Conference*, 2025, doi: 10.48550/arXiv.2501.17397.
[18] S. Shahriar *et al.*, "Putting GPT-4o to the Sword: A Comprehensive Evaluation of Language, Vision, Speech, and Multimodal Proficiency," *Applied Sciences (Switzerland)*, vol. 14, no. 17, Sep. 2024, doi: 10.3390/app14177782.
[19] V. Patel, "Analyzing the Impact of Next.JS on Site Performance and SEO," *International Journal of Computer Applications Technology and Research*, vol. 12, no. 10, pp. 24–27, 2023, doi: 10.7753/ijcatr1210.1004.
[20] H. A. Jartarghar, G. R. Salanke, A. K. A.R, and S. Dalali, "React Apps with Server-Side Rendering: Next.js," *Journal of Telecommunication, Electronic and Computer Engineering*, vol. 14, no. 4, Dec. 2022, doi: 10.54554/jtec.2022.14.04.005.
[21] A. N. Safitri and I. Harkespan, "Pengembangan Web Service Menggunakan Framework Fastapi Untuk Meningkatkan Kemudahan Integrasi Sistem Informasi Akademik Multiplatform," *Jurnal Teknoif Teknik Informatika Institut Teknologi Padang*, vol. 12, no. 2, pp. 149–157, Oct. 2024, doi: 10.21063/jtif.2024.V12.2.149-157.
[22] A. T. Saputro and M. Novita, "Comparative Analysis of Express and Hono Framework Performance in Simple Registration Application," *sinkron*, vol. 9, no. 1, pp. 406–412, Jan. 2025, doi: 10.33395/sinkron.v9i1.14333.
[23] P. Pujianto, M. Mujito, D. Prabowo, and B. H. Prasetyo, "Pemilihan Warga Penerima Bantuan Program Keluarga Harapan (PKH) Menggunakan Metode Simple Additive Weighting (SAW) dan User Acceptance Testing (UAT)," *Jurnal Informatika Universitas Pamulang*, vol. 5, no. 3, p. 379, Sep. 2020, doi: 10.32493/informatika.v5i3.6671.
[24] B. Simamora, "Skala Likert, Bias Penggunaan dan Jalan Keluarnya," *Jurnal Manajemen*, vol. 12, no. 1, pp. 84–93, Nov. 2022, doi: 10.46806/jman.v12i1.978.
[25] T. Dharmawan and A. Witanti, "Evaluasi Llama3.2 3b Untuk Menghasilkan Soal Otomatis Dengan Deepeval Berdasarkan Metrik Answer Relevancy Dan Hallucination," *Jurnal Informatika Teknologi dan Sains*, vol. 7, no. 1, pp. 242–248, 2025, doi: 10.51401/jinteks.v7i1.5423.
[26] A. B. Permadi, N. H Safaat, L. Handayani, and Yusra, "Implementasi Question Answering System Tafsir Al-Azhar Menggunakan Langchain Dan Large Language Model Berbasis Chatbot Telegram," *Jurnal Teknoif Teknik Informatika Institut Teknologi Padang*, vol. 12, no. 1, pp. 62–69, Apr. 2024, doi: 10.21063/jtif.2024.v12.1.62-69.
[27] T. Dharmawan and A. Witanti, "Evaluasi Llama3.2 3b Untuk Menghasilkan Soal Otomatis Dengan Deepeval Berdasarkan Metrik Answer Relevancy Dan Hallucination," *Jurnal Informatika Teknologi dan Sains*, vol. 7, no. 1, pp. 242–248, 2025, doi: 10.51401/jinteks.v7i1.5423.