Journal of Applied Informatics and Computing (JAIC)
Vol.9, No.6, December 2025, pp. 3356~3362
e-ISSN: 2548-6861

Programming Assessment in E-Learning through Rule-Based Automatic Question Generation with Large Language Models

Halim Teguh Saputro [1]*, Usman Nurhasan [2]*, Vivi Nur Wijayaningrum [3]*
* Teknik Informatika, Politeknik Negeri Malang
2131710122@student.polinema.ac.id [1], usman.nurhasan@polinema.ac.id [2], vivinurw@polinema.ac.id [3]

Article Info

Article history:

Received 2025-08-25
Revised 2025-09-29
Accepted 2025-11-08

Keyword:

Automatic Question Generation,
E-learning,
LLMs,
Programming Rule-based,
Taxonomy Bloom.

ABSTRACT

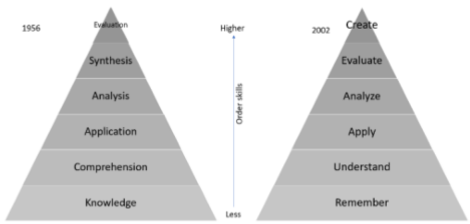
This study develops an evaluation instrument for Python programming using a Rule-Based Automatic Question Generation (AQG) system integrated with Large Language Models (LLMs), designed based on the Revised Bloom's Taxonomy. The urgency of this research stems from the limitations of conventional programming evaluations, which are often time-consuming, less objective, and insufficiently aligned with cognitive learning levels. The proposed method applies assessment terms as rule-based constraints to guide LLM-generated questions, ensuring both pedagogical validity and structural consistency in JSON format. A total of 91 questions were produced, consisting of multiple-choice and coding items, which were then validated by three programming experts and tested on 32 vocational students. The findings indicate that the instrument achieved an overall validity of 77.66% (valid category), with the highest accuracy at the Apply (96.30%) and Create (100%) levels. The reliability test using Cronbach's Alpha yielded 0.721, showing acceptable internal consistency. Item difficulty analysis revealed a strong dominance of easy questions (97.78%), with only 2.22% classified as moderate and none as difficult. Student performance also showed a fluctuating pattern: high in Remember (94.79%), Understand (95.83%), and Create (95.60%), but lower in Apply (86.11%), Analyze (90.97%), and Evaluate (87.15%). These results confirm that integrating Rule-Based AQG with LLMs can produce valid, reliable, and adaptive evaluation instruments that not only capture basic programming competencies but also partially address higher-order cognitive skills. This research contributes both practically by providing educators with an efficient tool for generating evaluation items and academically by enriching the growing body of literature on AI-assisted assessment in programming education.

I. PENDAHULUAN

Perkembangan teknologi yang pesat menuntut kompetensi pemrograman menjadi salah satu keterampilan yang sangat penting dikuasai, khususnya bagi siswa yang ingin berkarir di industri teknologi [1]. Namun, pendidikan di bidang pemrograman masih menghadapi kendala berupa materi yang terkadang kurang terstruktur hingga keterbatasan instrumen evaluasi yang objektif [2], [3]. Hal tersebut menimbulkan kesulitan untuk mengukur keterampilan tingkat lanjut seperti analisis, evaluasi, dan kreativitas. Selain itu, penyusunan soal secara manual membutuhkan waktu lebih panjang, kurang adaptif, dan rentan terhadap subjektivitas pengajar sehingga evaluasi tidak konsisten[4], [5].

Salah satu pendekatan yang dapat mendukung proses evaluasi adalah *Automatic Question Generation* (AQG)[6]. AQG memungkinkan pembuatan soal evaluasi secara otomatis dan adaptif terhadap konten yang diberikan [7]. Penelitian terdahulu menunjukkan bahwa pendekatan *rule-base* AQG ini efektif dalam memetakan kompetensi siswa ke dalam level-level kognitif dari konsep Taksonomi Bloom berdasarkan analisis kode program [8]. Di sisi lain, pemanfaatan *Large Language Models* (LLMs) seperti GPT-4 atau Llama 3.3 memberikan kemampuan generatif teks yang fleksibel, namun masih sering menghasilkan soal yang kurang sesuai dengan pedagogis [7].

Dalam menjaga validitas soal, penelitian ini mengadopsi *assesment term* pada konsep Taksonomi Bloom yang direvisi untuk mendefinisikan kata kerja operasional seperti *recall, describe, implement, analyze, justify,* dan *design* [9]. Penggunaan *assessment term* selain menjaga validitas pedagogis soal, juga penting untuk mengklasifikasikan soal secara lebih konsisten oleh pendidik [9], [10]. Taksonomi Bloom revisi pada Gambar 1 sebagai kerangka komprehensif dalam merancang tujuan pembelajaran dengan instrumen evaluasi [10]. Dengan begitu, AQG yang mengadopsi Taksonomi Bloom tidak hanya mengukur tingkat kognitif, tetapi juga memperhatikan kedalaman pengetahuan yang dibutuhkan siswa dalam pemrograman.



Gambar 1. Taksonomi Bloom asli (1956) dan revisi (2002) pada domain kognitif. penyesuaian urutan level kognitif dari *Knowledge–Evaluation* menjadi *Remember–Create* [10].

Berdasarkan beberapa penelitian sebelumnya yang telah dibahas, terdapat topik ini dapat terus di sempurnakan. Penelitian ini menawarkan kontribusi berupa sistem *Rule-Based* AQG dengan memanfaatkan LLMs dari Llama 3.3 70B untuk pembuatan soal dan *assessment term* pada Taksonomi Bloom sebagai aturan dasar pembuatan soal evaluasi pemrograman siswa. Penelitian ini menguji efektivitas instrumen melalui analisis validitas, reliabilitas, serta kesukaran soal [11], [12]. Fokus penelitian ini diarahkan pada keakuratan dan pemerataan soal pada setiap level Taksonomi Bloom, sehingga instrumen yang dihasilkan mampu memberikan evaluasi yang objektif, konsisten dan sesuai dengan tujuan pembelajaran[13].

Dengan permasalahan dan beberapa solusi penelitian sebelumnya, tujuan dari penelitian ini adalah: (1) mengevaluasi efektivitas *Rule-based* AQG dengan menggunakan *LLMs* dan Taksonomi Bloom dalam pembelajaran Python, (2) Menguji validitas, reliabilitas, dan kesukaran soal *Rule-based* AQG oleh *LLMs*.

II. METODE

A. Desain Penelitian

Penelitian ini menggunakan pendekatan pengembangan serta mengevaluasi instrumen dengan uji coba lapangan, dengan tujuan untuk menguji kualitas soal yang dihasilkan dari *Rule-based* AQG dengan *LLMs* dalam mengukur kemampuan pemrograman Python siswa [8], [14]. Soal sebelum diujikan ke siswa akan di validasi oleh para ahli/dosen, lalu analisis instrumen dilakukan setelah semua data didapatkan. Tahapan penelitian juga jelaskan pada Gambar 2.

Gambar 2. Tahapan penelitian meliputi *data preparation*, proses AQG, validasi ahli, uji coba instrumen pada siswa, serta analisis instrumen melalui uji validitas, reliabilitas, dan analisis kesukaran butir soal.

Tahapan penelitian pada Gambar 2 dilakukan guna menguji kualitas soal yang dihasilkan dari proses AQG. Sebelum proses AQG dilakukan, perlu mempersiapkan berbagai data materi Python tertentu dan juga *assessment term* Taksonomi Bloom sebagai dasar penyusunan soal [9]. Proses AQG mengolah data tersebut dan menghasilkan butir soal yang sesuai dengan tipe soal dan level kognitif. Soal tersebut divalidasi oleh dosen untuk memastikan kesesuaian konten dan indikator kognitif dengan memberikan label pada tiap soal [15]. Kemudian, dilakukan uji coba pengerjaan soal oleh 32 mahasiswa vokasi yang berada di semester 6 dari jurusan Teknologi Informasi, yang hasilnya dianalisis dengan uji reliabilitas dan analisis kesukaran butir soal guna menilai konsistensi serta kualitas instrumen[12], [16].

B. Instrumen Penelitian

Instrumen penelitian berupa tes pemrograman Python yang dihasilkan secara otomatis melalui sistem *Rule-Based* AQG terintegrasi dengan *LLMs* [8], [16]. Model Llama 3.3 70B dipilih sebab model open-source yang terbukti kompetitif dengan model besar lain seperti GPT-3, Chinchilla-70B, dan PaLM-540B [17]. Kemampuan ini mendukung aksesibilitas, efisiensi, dan penerapan yang berkelanjutan dalam penelitian. Penerapan AQG berbasis LLMs semakin sering digunakan sebagai instrumen evaluasi pembelajaran dengan menghasilkan soal yang sesuai dengan Taksonomi Bloom [18], [19]. Berdasarkan penelitian sebelumnya, penelitian ini, menekankan pengembangan instrumen evaluasi di bidang pemrograman pada materi Python dasar dengan memanfaatkan model *open-source* yaitu Llama 3.3 70B untuk menghasilkan soal yang dapat mengevaluasi kognitif siswa.

Terdapat 10 modul pembelajaran pemrograman Python dasar yang digunakan dengan 9 modul yang masing-masing menghasilkan 10 soal evaluasi level kognitif dari *remember* hingga *evaluate* dan modul pembelajaran terakhir menghasilkan 1 soal untuk menilai level kognitif *create* siswa sebagai *final test,* sehingga terdapat 91 soal yang dihasilkan oleh LLMs. Dalam penyusunan soal pada proses AQG menggunakan *assessment term* sebagai aturan dasar memberikan label soal untuk tiap level Bloom, hal ini guna memastikan setiap soal merepresentasikan indikator kognitif sesuai dengan tingkatannya [8], [9].

Kualitas instrumen dijamin melalui validasi ahli oleh dosen berkompeten di bidang pemrograman dan pedagogi [15]. Validasi mencakup kesesuaian soal dengan level kognitif Taksonomi Bloom, konten materi pemrograman Python, dan kejelasan bahasa yang digunakan. Kemudian instrumen diujikan pada 32 siswa vokasi untuk memperolah data empiris [12], [14]. Data tersebut dianalisis melalui uji reliabilitas (Cronbach Alpha) serta analisis kesukaran butir soal [20], [21]. Dengan demikian, instrumen ini tidak hanya dihasilkan melalui pendekatan teknologi, tetapi juga diverifikasi secara akademis dan empiris, sehingga dapat menilai kelayakan instrumen dalam mengukur kemampuan pemrograman siswa.

C. Prosedur Penelitian

Prosedur penelitian pada Gambar 2 ini dirancang guna menguji kualitas instrumen soal pemrograman Python yang dihasilkan secara otomatis melalui sistem *Rule-based* AQG dan *LLMs*.

1. Data Preparation

Materi pembelajaran Python disusun ke dalam 10 modul [2]. Sistem AQG menghasilkan total 91 soal sesuai level kognitif Taksonomi Bloom revisi (*remember* hingga *create*) dengan menggunakan *assessment term* dan isi materi dalam tiap modul sebagai aturan dasar pembuatan [9].

2. Proses AQG

Soal dihasilkan secara otomatis melalui sistem yang mengintegrasikan pendekatan Rule-based AQG dan LLMs sehingga memastikan soal terbentuk sesuai isi materi dan level kognitif [6][11]. Pada tahap perancangan prompt, aturan berbasis Taksonomi Bloom dengan memanfaatkan *assessment term* [9] digunakan untuk menentukan level kognitif yang harus dicapai oleh setiap soal, materi yang dipilih juga digunakan agar soal tetap sesuai dengan konteks pembelajaran, dan terdapat aturan format agar memastikan keluaran LLM konsisten dalam strsuktur JSON. Jenis soal yang dihasilkan terdiri dari soal pilihan ganda untuk menilai level *Remember*, *Understand*, *analyze*, dan *Evaluate*, dan soal *coding* yang untuk menilai level *Apply* dan *Create*. Demikian *rule-based* berfungsi tidak hanya mengendalikan format *output*, tetapi juga mekanisme menentukan label level Bloom yang kemudian diproses oleh model Llama 3.3 70B. Tabel 1 memaparkan beberapa soal dari proses AQG berdasarkan level Bloom.

TABEL I
HASIL SOAL DARI PROSES AQG DENGAN LLM

Level Bloom	Pertanyaan
Remember	Manakah dari berikut ini yang merupakan operator *modulus* dalam Python? a. * b. ** c. % d. //
Understand	Mengapa kita menggunakan *keyword* super() dalam pewarisan *class* di Python? a. Untuk menghentikan eksekusi konstruktor b. Untuk memanggil *method* dari *class* anak c. Untuk memanggil *method* dari *class* induk d. Untuk menimpa *method* yang sudah ada
Apply	Tulislah program Python untuk menambahkan pasangan 'kelas':'10A' ke *dictionary* siswa.
Analyze	Perhatikan kode berikut: angka = (10, 20, 30) angka[1] = 25 Apa yang akan terjadi saat kode dijalankan? a. Nilai *tuple* berhasil diubah b. Tidak terjadi apa-apa c. Program akan *error* karena *tuple* tidak dapat diubah d. Nilai 25 ditambahkan ke akhir *tuple*
Evaluate	Evaluasi kode berikut: data = {"a": 1, "b": 2} data.update({"c": 3, "a": 5}) print(data) a. {"a": 1, "b": 2, "c": 3} b. {"a": 5, "b": 2, "c": 3} c. {"a": 5, "b": 2} d. {"b": 2, "c": 3}
Create	Buatlah program Python yang: 1. Menampilkan daftar siswa, jumlah hadirnya, dan status kehadiran: - **"Sangat Baik"** jika hadir ≥ 9 - **"Baik"** jika hadir 7–8 kali - **"Cukup"** jika hadir 5–6 kali - **"Kurang"** jika hadir < 5 2. Menghitung rata-rata kehadiran seluruh siswa 3. Menampilkan nama-nama siswa yang statusnya “Kurang”

3. Validasi Ahli

Soal divalidasi oleh tiga orang dosen yang memiliki keahlian di validasi oleh di bidang *software enginer* dan bidang *data science* yang menggunakan Python. Mekanisme validasi dilakukan untuk menilai:

(a) kesesuaian level kognitif Bloom,
(b) relevansi konten terhadap materi Python, dan
(c) kejelasan bahasa soal.

Jika, poin b dan c telah sesuai, dosen/ahli memberikan label pada tiap soal tersebut dengan level kognitif Taksonomi Bloom, yang kemudian dianalisis kesesuaian label soal hasil AQG dengan label yang diberikan oleh dosen/ahli. Validitas isi instrumen diuji menggunakan metode *proportion agreement* karena sesuai untuk membandingkan kesesuaian label sistem dengan label pakar secara sederhana namun efektif [22].

4. Uji Coba Instrumen

Instrumen diujikan kepada 32 mahasiswa semester 6 jurusan Teknologi Informasi pada pendidikan vokasi. Data yang diperoleh berupa jawaban siswa pada soal pilihan ganda (benar/salah) dan soal *coding* (skor numerik dengan CodeBLEU) dengan dikatakan benar jika skor > 70 [23]. Kemampuan tiap siswa berdasarkan level kognitif Taksonomi Bloom dinilai berdasarkan soal yang dijawab benar dengan rumus sebagai berikut [24]:

$$Level\ X = \left(\frac{\Sigma b}{\Sigma n}\right) \times 100\%$$

Keterangan:

- Level X : Tingkat Taksonomi Bloom (*Remember, Understand, Apply, Analyze, & Evaluate*).
- $\sum$b : Jumlah soal pada tingkat x yang dijawab benar.
- n : Jumlah soal pada tingkat x di dalam modul

5. Analisis Instrumen

Data hasil uji coba dianalisis untuk menilai kualitas instrumen [20]:

- Reliabilitas (Cronbach Alpha) guna menilai konsistensi soal.
- Analisis kesukaran butir soal

D. Analisis Data

Analisis data mencakup tiga tahap utama, yaitu validasi ahli, reliabilitas internal, dan analisis butir soal. Ringkasan analisis, beserta rumus dan hasil yang diperoleh, disajikan pada Tabel 2 berikut [20], [21], [24].

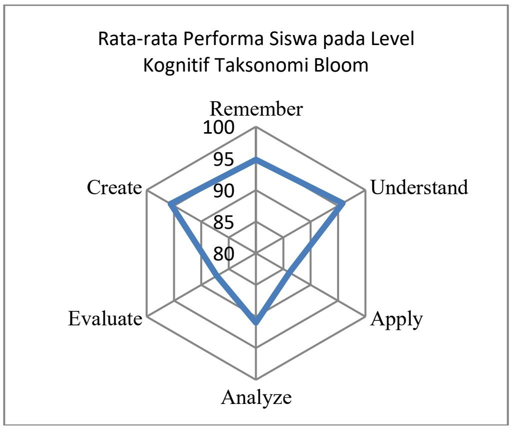
TABEL II
ANALISIS DATA

Tahap Analisis	Keterangan
Validasi ahli	Tingkat validasi dihitung dari persentase kesesuaian label sistem dengan label ahli menggunakan rumus: $$V = \frac{\Sigma K}{N} \times 100\%$$ Hasil yang diperoleh adalah persentase validitas soal per level Bloom.
Reliabilitas (*Cronbach Alpha*)	Mengukur konsistensi instrumen dengan menghitung koefisien reliabilitas berdasarkan data jawaban siswa. $$\alpha = \frac{k}{k-1}\left(1 - \frac{\Sigma\sigma_i^2}{\sigma_i^2}\right)$$ Hasil yang diperoleh berupa nilai α, dengan kriteria α ≥ 0,70 dianggap reliabel.
Analisis kesukaran butir soal	Mengetahui tingkat kesulitan soal berdasarkan jawaban dari seluruh siswa dengan rumus: $$p = \frac{B}{N}$$ Hasil yang diperoleh berupa kategori soal (mudah, sedang, sulit).

III. HASIL DAN PEMBAHASAN

A. Evaluasi Kemampuan Siswa

Evaluasi kemampuan siswa dilakukan untuk memetakan capaian pembelajaran pada setiap tingkat kognitif Taksonomi Bloom (*remember, understand, apply, analyze, evaluate,* dan *create*) [25]. Pengukuran ini diperoleh dari hasil pengerjaan soal dari 32 siswa vokasi terhadap dua tipe soal, yaitu pilihan ganda dan praktik *coding*, yang memiliki label tiap level kognitif saat proses AQG. Soal pilihan ganda dinilai secara konvensional, sedangkan soal praktik coding pada level *apply* dan *create* dinilai menggunakan CodeBLEU [23]. Gambar 3 memetakan nilai rata-rata yang diraih siswa di tiap tingkatan kognitif.



Gambar 3. Rata-rata skor siswa pada enam level kognitif Taksonomi Bloom yang dievaluasi melalui soal proses AQG.

Hasil evaluasi menunjukkan capaian siswa bervariasi di setiap level kognitif Bloom. Persentase tertinggi terdapat pada *Remember* (94,79%) dan *Understand* (95,83%), menandakan siswa mampu mengingat serta memahami konsep dasar Python dengan baik. Namun, pada *Apply* (86,11%) terjadi penurunan, menunjukkan masih ada kesulitan dalam menerapkan konsep ke dalam praktik pemrograman yang benar secara sintak maupun logika.

Pada level *Analyze* (90,97%) dan *Evaluate* (87,15%), capaian relatif lebih rendah dibandingkan level dasar (*Remember & Understand*), mencerminkan tantangan dalam menganalisis masalah dan menilai efektivitas solusi.

Menariknya, capaian pada *Create* (95,60%) justru sangat tinggi, meskipun level ini umumnya menuntut kreativitas dan kemampuan merancang solusi orisinal. Kondisi ini disebabkan oleh struktur instrumen, soal *Apply* muncul berulang pada tiap materi dengan masing-masing memiliki 2 butir soal, sehingga responden memiliki lebih banyak kesempatan melakukan kesalahan kecil yang memengaruhi rata-rata skor. Sedangkan, soal *Create* hanya diberikan satu kali pada materi terakhir sebagai *final test*, sehingga responden cenderung lebih fokus dan mengalokasikan usaha terbaik. Faktor ini menjelaskan mengapa capaian *Create* relatif lebih tinggi dibanding level *Apply* dan *Evaluate*.

Secara keseluruhan, pola capaian tidak sepenuhnya menurun seiring meningkatnya kompleksitas level Bloom yang ada pada penelitian [8], melainkan fluktuatif. Fenomena ini menunjukkan adanya faktor lain selain kompleksitas kognitif, seperti jenis soal, kualitas soal, dan strategi siswa dalam menyelesaikan tugas. Walaupun begitu, temuan ini berhasil membuat instrumen soal yang dikembangkan melalui *Rule-Based* AQG terintegrasi LLMs memberikan variasi evaluasi yang komprehensif sesuai kerangka Taksonomi Bloom.

B. Hasil Validasi Dosen/Ahli

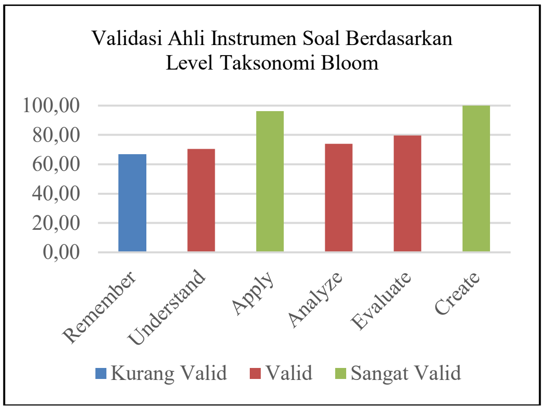
Validasi ahli dilakukan untuk memastikan bahwa soal yang dihasilkan sistem *Rule-Based* AQG terintegrasi LLMs sesuai dengan level kognitif Taksonomi Bloom [16], [20]. Hasil validasi pada Tabel 2 menunjukkan bahwa sebagian besar soal berada pada kategori valid hingga sangat valid.

TABEL III
REKAP VALIDASI AHLI PER LEVEL TAKSONOMI BLOOM

Level	Validasi Ahli (sesuai/total)	Validitas (%)	Keterangan
Remember	36/54	66,67	Kurang Valid
Understand	38/54	70,37	Valid
Apply	52/54	96,30	Sangat Valid
Analyze	40/54	74,07	Valid
Evaluate	43/54	79,63	Valid
Create	3/3	100,00	Sangat Valid
Total	212/273	77,66	Valid

Validasi ini dilakukan oleh 3 dosen/ahli dengan hasil menunjukkan bahwa instrumen soal memiliki tingkat validitas mencapai 77,66% dengan kategori valid. Pada level *Remember* (66,67%) relatif rendah sehingga beberapa soal perlu ditinjau kembali agar lebih tepat mengukur kemampuan mengingat siswa terhadap materi yang diberikan. Level *Understand* (70,37%) termasuk dalam kategori valid namun berada diambang batas kurang valid, sehingga tetap memerlukan perbaikan pada beberapa butir soal. Rendahnya capaian soal Remember dan Understand ini dipengaruhi kecenderungan dosen sering menafsirkan soal Remember sebagai Understand, dan sebaliknya, kedua level ini termasuk level kognitif rendah dengan perbedaan yang tipis. Sebaliknya level *Apply* (96,30%) dan *Create* (100%) berada pada kategori sangat valid, menandakan soal pada level ini sudah sesuai dengan indikator kognitif Taksonomi Bloom. Level *Analyze* (74,07%) dan *Evaluate* (79,63%) berada pada kategori valid yang cukup representatif untuk mengukur keterampilan analisis dan evaluasi siswa.

Secara keseluruhan, instrumen dinilai layak digunakan, dengan catatan revisi diperlukan terutama pada level kognitif rendah agar kualitas instrumen semakin merata di semua level. Grafik batang digunakan guna memperjelas perbedaan tingkat validitas antar level kognitif Bloom, hasil rekapitulasi pada Tabel 3 divisualisasikan dalam bentuk diagram batang pada Gambar 4. Visualisasi ini memudahkan identifikasi level mana yang memiliki validitas tinggi maupun rendah, sekaligus memperlihatkan distribusi kualitas soal secara keseluruhan.



Gambar 4. Grafik yang memperlihatkan persentase validitas instrumen soal. Warna batang menunjukkan kategori validitas: biru untuk *Kurang Valid*, merah untuk *Valid*, dan hijau untuk *Sangat Valid*.

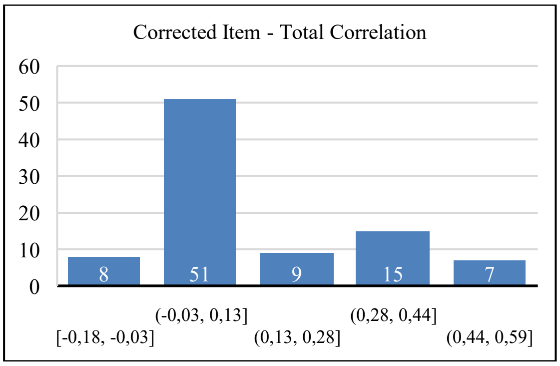
C. *Hasil Reliabilitas Soal (Cronbach Alpha)*

Uji reliabilitas dilakukan menggunakan *Cronbach's Alpha* terhadap 90 butir soal yang diujikan kepada 32 siswa [12]. Hasil uji ditampilkan pada Tabel 4.

TABEL IV
HASIL UJI RELIABILITAS INSTRUMEN

Jumlah Responden	Jumlah Item	Cronbach's Alpha	Kategori
32	90	0,721	Reliabel

Nilai *Croncbach's Alpha* sebesar 0,721 menunjukkan bahwa instrumen soal memiliki konsistensi internal yang dapat diterima (Acceptable). Hasil ini sejalan dengan temuan Marar dkk. (2023) yang menggunakan nilai ambang ≥ 0,70 sebagai indikator reliabilitas yang memadai untuk instrumen penelitian di bidang pendidikan dan kesehatan [26]. Dengan demikian, soal yang dihasilkan sistem *Rule-Based AQG* terintegrasi *LLMs* dapat dikatakan reliabel dan layak digunakan untuk mengukur kemampuan pemrograman siswa. Analisis ini diperkuat dengan grafik distribusi nilai *Corrected Item-Total Correlation* (CITC) dari tiap soal yang ditampilkan pada Gambar 5 [11]. Grafik ini akan memperlihatkan butir soal mana yang memiliki kontribusi tinggi terhadap reliabilitas.

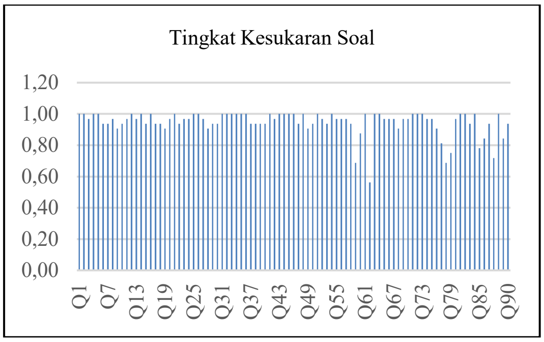


Gambar 5. Distribusi *Corrected Item-Total Correlation* untuk 90 Butir Soal. Grafik menunjukkan variasi nilai korelasi tiap butir soal terhadap total skor.

Namun, analisis lebih lanjut pada CITC menunjukkan bahwa terdapat variasi kualitas antar butir soal. Berdasarkan grafik pada Gambar 5, mayoritas soal (56,7%) berada di kisaran yang rendah (0,00-0,13), bahkan 8,9% soal memiliki nilai negatif, hal ini mengindikasikan bahwa butir-butir soal tersebut tidak mendukung instrumen [11]. Hanya 16,7% soal yang mendekati ambang batas 0,30 dan 7,8% soal yang kuat dengan nilai di atas 0,44. Kondisi ini disebabkan oleh dominasi soal dengan tingkat kesukaran yang rendah, kecenderungan LLM menghasilkan soal homogen pada tingkat kognitif dasar juga menjadi salah satu faktor yang perlu di evaluasi lebih lanjut. Berdasarkan temuan ini, menunjukkan bahwa sebagian besar soal perlu ditinjau ulang atau diperbaiki, terutama pada level dengan kontribusi rendah, agar reliabilitas instrumen secara keseluruhan dapat meningkat.

D. Hasil Analisis Kesukaran Butir Soal

Analisis tingkat kesukaran butir soal digunakan untuk mengetahui seberapa sulit siswa dalam menjawab soal tersebut dengan benar [12], [20]. Indeks kesukaran soal terbagi menjadi tiga kategori, yaitu mudah (p > 0,7), sedang (0,3 ≤ p ≤ 0,7), dan sukar (p < 0,3) [27]. Gambar 6 memvisualisasikan distribusi indeks kesukaran dari butir soal Q1-Q90.



Gambar 6. Visualisasi distribusi indeks kesukaran setiap soal (Q1–Q90). Skor kesukaran mendekati 1,00 menunjukkan soal yang mudah, sedangkan skor mendekati 0,30 atau lebih rendah menunjukkan soal yang relatif sukar .

Untuk memperjelas pola distribusi tersebut, Tabel 4 berikut menyajikan ringkasan jumlah soal dalam kategori mudah, sedang, dan sukar.

TABEL V
DISTRIBUSI TINGKAT KESUKARAN SOAL

Kategori	Jumlah Soal	Persentase (%)
Mudah	88	97,78
Sedang	2	2,22
Sukar	0	0

Distribusi tingkat kesukaran soal pada Gambar 6 dan Tabel 5 menunjukkan bahwa mayoritas butir soal (97,8%) berada pada kategori mudah, dengan indeks kesukaran di atas 0,70. Hal ini tampak jelas pada Q1–Q50, di mana seluruh soal termasuk mudah dengan nilai mendekati 1,00. Dominasi soal dengan kategori mudah ini terindikasi karena beberapa hal. Proses AQG yang dilakukan cenderung menghasilkan pertanyaan konseptual dasar apabila *prompt* tidak secara eksplisit meminta tingkat kesulitan atau melibatkan pengecoh soal. Akibatnya, variasi soal dengan tingkat kesulitan lebih tinggi menjadi terbatas. Faktor lain yang mungkin berpengaruh adalah profil responden atau mahasiswa, yaitu mahasiswa vokasi yang telah menempuh enam semester perkuliahan di bidang Teknologi Informasi, sehingga soal cenderung mudah bagi mereka.

Walaupun mayoritas soal berkategori mudah, Pada Q51–Q90 mulai muncul sedikit variasi, khususnya dua soal yaitu Q59 (0,69) dan Q62 (0,56) yang termasuk kategori sedang. Kedua soal ini menuntut siswa untuk berpikir lebih mendalam, kemungkinan besar berada pada level *Analyze* atau *Evaluate*, sehingga tidak seluruh siswa mampu menjawab dengan benar. Meski demikian, jumlah soal kategori sedang masih sangat sedikit (hanya 2,2% dari total soal), dan tidak ada soal yang termasuk kategori sukar (<0,30).

Dengan demikian, instrumen soal yang dihasilkan sistem *Rule-Based* AQG terintegrasi *LLMs* didominasi oleh soal mudah dan hanya memiliki sedikit variasi soal menengah. Pola ini bermanfaat untuk mengukur penguasaan dasar siswa, tetapi belum cukup seimbang untuk menguji kemampuan berpikir tingkat lanjut. Oleh karena itu, pengembangan instrumen lebih lanjut sebaiknya memperbanyak butir soal dengan tingkat kesukaran sedang hingga sukar agar kualitas instrumen lebih komprehensif.

IV. KESIMPULAN

Instrumen evaluasi pembelajaran pemrograman saat ini masih terbatas karena dominan menilai aspek dasar, membutuhkan waktu lama dalam penyusunan, dan sering kurang objektif. Penelitian ini menawarkan solusi dengan mengembangkan instrumen evaluasi otomatis menggunakan *Rule-based* AQG yang diintegrasikan dengan LLMs (Llama 3.3 70b) dan *assessment term* dari Taksonomi Bloom sebagai dasar aturannya. Proses ini menghasilkan 91 soal, yang terdiri dari soal pilihan ganda dan *coding*, yang kemudian divalidasi oleh tiga dosen ahli melalui kesesuaian level Bloom, konten, dan kejelasan bahasa. Validasi ini dilakukan dengan pendekatan *proportion agreement* untuk mengukur kecocokan label sistem dan ahli, serta instrumen diujikan kepada 32 responden yaitu mahasiswa vokasi semester 6 program studi Teknologi Informasi.

Hasil analisis menunjukkan instrumen yang dihasilkan memiliki validitas 77,66% (kategori valid), reliabilitas dengan *Cronbach's Alpha* 0,721 yang berarti dapat diterima (*acceptable*), serta distribusi tingkat kesukaran yang masih didominasi soal mudah (97,8%). Sementara itu, distribusi capaian siswa memperlihatkan pola fluktuatif: cukup tinggi pada level *Remember* (94,79%), *Understand* (95,83%), *Create* (95,60%), tetapi menurun pada *Apply* (86,11%), *Analyze* (90,97%), dan *Evaluate* (87,15%). Pola ini menunjukkan bahwa meskipun instrumen mampu memetakan keterampilan dasar dan Sebagian keterampilan tingkat lanjut, soal pada level rendah masih sering tertukar oleh ahli (misalnya antara *Remember* dan *Understand*) sehingga menurunkan konsistensi validitas. Secara keseluruhan, penelitian ini membuktikan bahwa integrasi *Rule-based* AQG dan LLMs efektif dalam menghasilkan evaluasi yang objektif, konsisten, serta relevan dengan pembelajaran pemrograman Python dasar. Namun, masih perlu adanya perbaikan guna menyempurnakan penelitian ini, seperti adanya personalisasi latar belakang kemampuan siswa sehingga soal bisa lebih komprehensif, adaptif, dan sesuai dengan kemampuan siswa untuk terus berkembang.

DAFTAR PUSTAKA

[1] L. Kyung Choi, N. Iftitah, and P. Angela, "Developing Technopreneur Skills to Face Future Challenges," *IAIC Transactions on Sustainable Digital Innovation (ITSDI)*, vol. 5, no. 2, pp. 127–135, 2024.

[2] E. Mehmood, A. Abid, M. S. Farooq, and N. A. Nawaz, "Curriculum, Teaching and Learning, and Assessments for Introductory Programming Course," *IEEE Access*, vol. 8, pp. 125961–125981, 2020, doi: 10.1109/ACCESS.2020.3008321.

[3] K. Ishaq and A. Alvi, "Personalization, Cognition, and Gamification-based Programming Language Learning: A State-of-the-Art Systematic Literature Review," Sep. 2023.

[4] Z. Ullah, A. Lajis, M. Jamjoom, A. H. Altalhi, J. Shah, and F. Saleem, "A rule-based method for cognitive competency assessment in computer programming using bloom's taxonomy," *IEEE Access*, vol. 7, pp. 64663–64675, 2019, doi: 10.1109/ACCESS.2019.2916979.

[5] O. Keklik, T. Tuglular, and S. Tekir, "Rule-based automatic question generation using semantic role labeling," *IEICE Trans Inf Syst*, vol. E102D, no. 7, pp. 1362–1373, 2019, doi: 10.1587/transinf.2018EDP7199.

[6] M. F. Naufal and S. F. Kusuma, "Otomatisasi Pembangkitan Pertanyaan untuk Bahasa Indonesia (Systematic Literature Review)," *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 10, no. 1, pp. 185–192, Feb. 2023, doi: 10.25126/jtiik.2023106455.

[7] S. Maity, A. Deroy, and S. Sarkar, "How Effective is GPT-4 Turbo in Generating School-Level Questions from Textbooks Based on Bloom's Revised Taxonomy?," 2024.

[8] Z. Ullah, A. Lajis, M. Jamjoom, A. H. Altalhi, J. Shah, and F. Saleem, "A rule-based method for cognitive competency assessment in computer programming using bloom's taxonomy," *IEEE Access*, vol. 7, pp. 64663–64675, 2019, doi: 10.1109/ACCESS.2019.2916979.

[9] R. Kadar, S. A. Mohamed Yusoff, S. N. Warris, and M. S. Abu Bakar, "Students' Assessments in Learning Programming based on Bloom's Taxonomy," *Journal of Computing Research and Innovation*, vol. 6, no. 3, pp. 13–21, Sep. 2021, doi: 10.24191/jcrinn.v6i3.223.

[10] S. R. Sobral, "Bloom's taxonomy to improve teaching-learning in introduction to programming," *International Journal of Information and Education Technology*, vol. 11, no. 3, pp. 148–153, Mar. 2021, doi: 10.18178/ijiet.2021.11.3.1504.

[11] E. A. O. Zijlmans, J. Tijmstra, L. A. van der Ark, and K. Sijtsma, "Item-score reliability as a selection tool in test construction," *Front Psychol*, vol. 9, no. JAN, Jan. 2019, doi: 10.3389/fpsyg.2018.02298.

[12] N. Z. Zuhri, S. Syihabuddin, and T. Tatang, "Analisis Validitas, Reliabilitas, dan Tingkat Kesukaran Soal Bahasa Arab Tingkat SMP Berbasis Artificial Intelligence (AI) melalui Platform QuestionWell," *Jurnal Pendidikan dan Pembelajaran Indonesia (JPPI)*, vol. 4, no. 2, pp. 693–704, Jul. 2024, doi: 10.53299/jppi.v4i2.576.

[13] P. Blessy Paul and C. Kurian, "Generation of Bloom's taxonomy-based complex-level questions using knowledge graph," in *Proceedings - 2024 IEEE International Conference on Signal Processing, Informatics, Communication and Energy Systems: Harmonizing Signals, Data, and Energy: Bridging the Digital Future, SPICES 2024*, Institute of Electrical and Electronics Engineers Inc., 2024. doi: 10.1109/SPICES62143.2024.10779773.

[14] D. Kusumaningrum and A. Muslihasari, "Pengembangan Instrumen Literasi Lingkungan Ranah Kognitif Untuk Siswa Sekolah Dasar di Kabupaten Malang," Sep. 2020.

[15] A. Yaacoub, J. Da-Rugna, and Z. Assaghir, "Assessing AI-Generated Questions' Alignment with Cognitive Frameworks in Educational Assessment," 2025.

[16] S. Al Faraby, A. Romadhony, and Adiwijaya, "Analysis of LLMs for educational question classification and generation," *Computers and Education: Artificial Intelligence*, vol. 7, Dec. 2024, doi: 10.1016/j.caeai.2024.100298.

[17] H. Touvron *et al.*, "LLaMA: Open and Efficient Foundation Language Models," Feb. 2023, [Online]. Available: http://arxiv.org/abs/2302.13971

[18] N. Scaria, S. Dharani Chenna, and D. Subramani, "Automated Educational Question Generation at Different Bloom's Skill Levels Using Large Language Models: Strategies and Evaluation," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Springer Science and Business Media Deutschland GmbH, 2024, pp. 165–179. doi: 10.1007/978-3-031-64299-9_12.

[19] 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

[20] L. A. Son, "Instrumentasi Kemampuan Pemecahan Masalah matematis analisis reliabilitas, validitas, tingkat kesukaran dan daya beda butir soal," 2019.

[21] M. Erfan *et al.*, "Analisis Kualitas Soal Kemampuan Membedakan Rangkaian Seri Dan Paralel Melalui Teori Tes Klasik Dan Model Rasch," *Indonesian Journal of Educational Research and Review*, vol. 3, no. 1, p. 11, 2020.

[22] K. Singhal *et al.*, "Toward expert-level medical question answering with large language models," *Nat Med*, vol. 31, no. 3, pp. 943–950, Mar. 2025, doi: 10.1038/s41591-024-03423-7.

[23] S. Ren *et al.*, "CodeBLEU: a Method for Automatic Evaluation of Code Synthesis," Sep. 2020, [Online]. Available: http://arxiv.org/abs/2009.10297

[24] J. Cressa and M. Mukhlis, "Level Kognitif Taksonomi Bloom pada Soal Mata Pelajaran Bahasa Indonesia," vol. 3, 2023, [Online]. Available: https://journal.uir.ac.id/index.php/j-lelc

[25] A. Gomes and F. B. Correia, *Bloom's Taxonomy Based Approach to Learn Basic Programming Loops*. IEEE, 2018.

[26] S. Marar, M. A. Hamza, M. Ayyash, and A. Abu-Shaheen, "Development and validation of an instrument to assess the knowledge and perceptions of predatory journals," *Heliyon*, vol. 9, no. 11, Nov. 2023, doi: 10.1016/j.heliyon.2023.e22270.

[27] M. W. Gebremichael, B. Baraki, M. A. Mehari, and B. Assalfew, "Item analysis of multiple choice questions from assessment of health sciences students, Tigray, Ethiopia," *BMC Med Educ*, vol. 25, no. 1, Dec. 2025, doi: 10.1186/s12909-025-06904-6.