International Conference on Social Science and Humanity
Homepage : https://proceeding.antispublisher.or.id/index.php/icossh/index
Email : admin@antispublisher.com

e-ISSN : 3063-9611
ICOSSH, Vol. 3, No. 5, 2026
Page 120-129
© 2026 ICOSSH

Implementation of an API Key-Based Large Language Model for ICD Code Recommendation in Hospitals

Budi Fajar Supriyanto[1]*, Muhammad Hilmy Haidar Fawwazie[2], Brian Rizky Solehuddin Bachtiar[3], Syafa Alviatur Sudarto Putri[4], Wafiq Nawa Sabila[5], Dzulfiqor Aflahi Hasan Mumtaz[6]
[1,4,5,6] Politeknik Negeri Jember, Jember, Indonesia,
[2]Universitas Brawijaya, Malang, Indonesia,
[3]Universitas Bina Nusantara, Jakarta, Indonesia

DOI : https://doi.org/10.61796/icossh.v3i4.660

Sections Info

Article history:
Submitted: August 23, 2026
Final Revised: September 11, 2026
Accepted: October 16, 2026
Published: November 23, 2026

Keywords:
Artificial intelligence
Hospital information system
Icd coding
Large language model
Medical record

ABSTRACT

Objective: *This study aimed to design, implement, and evaluate the accuracy and usability of a Large Language Model (LLM)-based ICD code recommendation feature, integrated into a hospital information system (SIMRS) through an Application Programming Interface (API) key, to support coders during the abstracting and coding process.* ***Method:*** *This was a quantitative Research and Development (R&D) study with a cross-sectional evaluation design, conducted as an early proof-of-concept using synthetic data rather than real patient records. A total sample of 25 synthetic diagnosis narratives, co-developed with health information management (RMIK) students and lecturers to represent Chapter I of ICD-10 (Certain Infectious and Parasitic Diseases, A00–B99), was used to test a prototype architecture connecting a coding-support module to a commercial LLM via a secured API key, guided by a retrieval-augmented prompt referencing the ICD-10 index. LLM-generated recommendations were compared against the codes assigned by RMIK lecturers acting as expert validators (reference standard). Accuracy, precision, recall, F1-score, and Cohen's Kappa were calculated, and 20 RMIK students and lecturers assessed the prototype's usability using the System Usability Scale (SUS).* ***Results:*** *The prototype produced a top-1 accuracy of 64.0% (16/25) and a top-3 accuracy of 76.0% (19/25) against the expert validators' codes, with moderate agreement (Kappa = 0.58) and a mean SUS score of 74.5 (category: Good). Response time per case averaged 3.2 seconds.* ***Novelty:*** *Integrating an LLM through an API key into a coding-support prototype is technically feasible and shows promising early accuracy and usability for ICD code recommendation on a limited, single-chapter synthetic dataset.*

INTRODUCTION

Clinical coding using the International Classification of Diseases (ICD) is one of the core responsibilities of Medical Record and Health Information Officers (*Perekam Medis dan Informasi Kesehatan*/PMIK). ICD-10 codes for diagnoses and ICD-9-CM codes for medical procedures serve as the foundation for calculating Indonesian Case Base Groups (INA-CBG) tariffs under the National Health Insurance (*Jaminan Kesehatan Nasional*) scheme, as well as a primary source for epidemiological data and health policy planning at both facility and national levels. Coding inaccuracies directly affect reimbursement claim amounts, the validity of morbidity statistics, and the quality of hospital reporting to various stakeholders.

In practice, manual coding faces several challenges: high volumes of medical records, time constraints faced by coders, unstructured physician diagnosis narratives or the use of local abbreviations, and the complexity of coding rules (such as combination codes and primary-secondary diagnosis sequencing) that require simultaneous clinical and administrative expertise. Recent systematic reviews on the application of LLMs for ICD coding confirm that these challenges drive the exploration of artificial intelligence-based automation, while emphasizing the necessity of a verification stage because model outputs are not entirely free from errors [5], [6].

Advances in generative artificial intelligence, particularly Large Language Models (LLMs), create new opportunities as decision-support tools in clinical coding. Unlike conventional Natural Language Processing (NLP) approaches that rely on statistical classification models or pretrained/fine-tuned language models requiring retraining whenever coding governance updates occur [12], [13], off-the-shelf commercial LLMs accessed via an Application Programming Interface (API) key have been shown to perform clinical coding tasks directly without additional model training [7]. This capability enables seamless integration into existing Hospital Information Systems (*Sistem Informasi Manajemen Rumah Sakit*/SIMRS) without building machine learning models from scratch.

A. State of the Art

Previous studies have explored automated clinical coding along two primary directions. The first direction focuses on language models specifically trained or fine-tuned for ICD classification tasks, such as PLM-ICD, which utilizes pretrained language models for automated multi-label coding [12], and domain-specific fine-tuned LLMs designed to enhance medical coding efficiency [8]. While this approach generally reports high accuracy, it requires substantial computational infrastructure and large labeled datasets, including in self-hosted LLM implementations for automated ICD coding of tumor pathology reports [4].

The second direction investigates the use of off-the-shelf generative LLMs through zero-shot or few-shot prompting without model retraining. Studies evaluating the reasoning capabilities of LLMs for medical coding report promising performance under zero-shot prompting schemes [3], while other research demonstrates that off-the-shelf LLMs can perform automated clinical coding competitively compared to specialized models, provided adequate output verification mechanisms are in place [6], [7]. Domain-specific LLM applications, such as ICD coding for retina cases [1], alongside investigations into knowledge prompt injection strategies and model scale effects on ICD coding performance [2], further reinforce that prompt quality and supplementary context heavily influence LLM output accuracy. More broadly, LLMs have also been evaluated to support healthcare data interoperability [9] and deployed across diverse real-world clinical workflows beyond ICD coding [10], including as medical code selectors in other classification systems such as the International Classification of Primary Care (ICPC), demonstrating the generalizability of LLM approaches beyond ICD-10 [11]. A recent systematic review summarizes that LLM performance in ICD coding varies considerably depending on the prompting strategy, clinical domain, and verification mechanisms employed [5].

B. Gap Analysis

Most prior studies reporting high accuracy rely on custom-trained or fine-tuned models [12] or self-hosted LLMs [4], [8], which demand computational infrastructure investments and technical expertise that are not always available in educational institutions or small-to-medium-sized hospitals in Indonesia. Conversely, studies utilizing off-the-shelf commercial LLMs via API keys without model retraining [3], [6], [7] have largely been evaluated on English-language datasets within developed healthcare systems. Research examining the direct integration of this approach into Indonesian hospital information system workflows—complete with multi-layered human-in-the-loop verification mechanisms and user-centered usability evaluation—remains highly limited.

This study addresses this gap by presenting the design and preliminary evaluation of an API key-based LLM integration architecture within a coding-support module, combining accuracy assessment, inter-rater agreement with expert validators, and user acceptance evaluation within the context of Health Information Management (*Rekam Medis dan Informasi Kesehatan*/RMIK) education in Indonesia. Specifically, this study aims to design an LLM-based ICD code recommendation prototype integrated via an API key, measure its accuracy and agreement against codes assigned by expert validators (RMIK lecturers) as the reference standard in a synthetic data proof-of-concept, and evaluate the perceived usability of the system from the perspective of RMIK students and lecturers as end users.

RESEARCH METHODS

A. Research Design

This study employed a quantitative approach using a modified three-stage Research and Development (R&D) method: (1) requirements analysis and system architecture design, (2) prototype development integrating the LLM into a coding-support module, and (3) cross-sectional evaluation of accuracy and usability. Due to restricted access to real patient medical records, this study was positioned as an early proof-of-concept utilizing synthetic data co-developed and validated with students and lecturers of the RMIK Study Program, serving as a technical feasibility step prior to testing on real clinical data in a hospital environment in future research.

B. Population and Sample

The study population consisted of synthetic diagnosis narratives modeled after discharge summary narratives, developed collaboratively with RMIK students and lecturers based on common cases in Chapter I of ICD-10 (Certain Infectious and Parasitic Diseases, codes A00–B99) as the scope for initial prototype testing. The sample was determined using total sampling of all synthetic narratives successfully compiled and agreed upon by the development team as representative, totaling 25 diagnosis narratives. Because the data were synthetic rather than a probabilistic sample from a real patient population, sample size determination did not employ statistical formulas such as Slovin's formula, but instead followed case-sufficiency guidelines for technical feasibility in pilot testing, as is standard in small-scale systems engineering studies.

- **Inclusion criteria:** Single diagnosis narratives as well as narratives with comorbidities classified under Chapter I of ICD-10, written using standard clinical terminology commonly found in discharge summaries.
- **Exclusion criteria:** Grammatically incomplete or ambiguous narratives that could not be validated by RMIK lecturers.

C. System Architecture and API Key Integration

The system was developed by adding a code recommendation module to an existing SIMRS coding interface. The workflow operates as follows:

1. The PMIK inputs or selects a diagnosis narrative from the discharge summary in the SIMRS.
2. The system transmits a request containing the anonymized diagnosis text to the commercial LLM API endpoint via an encrypted connection (HTTPS/TLS), accompanied by an API key stored securely in server-side environment variables (rather than in source code or client-side storage), following an off-the-shelf LLM deployment approach without additional model training [7].

3. The transmitted prompt is constructed by embedding relevant excerpts from the ICD-10 index as supplementary context (knowledge prompt), adhering to findings that knowledge injection strategies and prompt structure significantly impact LLM accuracy in ICD coding tasks [2], alongside structured instructions requiring the output to return candidate codes with confidence scores in JSON format.
4. The API response is parsed by the SIMRS backend and displayed to the coder as a list of the top three recommended codes (top-3) accompanied by a brief rationale.
5. The coder performs final verification and may accept, modify, or reject the recommendations before saving the code as final data (human-in-the-loop mechanism), aligning with recommendations that human verification is an essential component when LLMs are applied to zero-shot clinical coding [6].

(See Fig. 1: Architecture of API Key-Based LLM Integration in SIMRS)

To safeguard patient confidentiality, all diagnosis narratives undergo automated de-identification prior to transmission to the LLM provider, and API requests are configured using enterprise-tier provider options specifying that transmitted data are neither retained nor used for model retraining (zero data retention). This configuration forms a key component of privacy risk mitigation, as further elaborated in the Discussion section.

D. Instruments and Data Collection

The primary instrument was a comparison worksheet recording the synthetic diagnosis narrative, the reference standard code established by expert validators (RMIK lecturers), the system's top-1 and top-3 recommended codes, system response time (in seconds), and concordance notes. The second instrument was the Indonesian version of the System Usability Scale (SUS) questionnaire using a 5-point Likert scale (1 = Strongly Disagree to 5 = Strongly Agree), completed by 20 respondents (RMIK students and lecturers) after using the prototype during the trial period.

E. Data Analysis

Code concordance was analyzed using accuracy, precision, recall, and F1-score metrics with expert validator codes serving as the reference standard, alongside Cohen's Kappa coefficient to measure inter-rater agreement beyond chance [13], following standard machine learning and deep learning evaluation metric definitions and calculation procedures [14]. SUS responses were scaled to a 0–100 score range and interpreted using the adjective rating scale (Poor–OK–Good–Excellent) [15]. All statistical analyses were performed using [name of statistical software, e.g., SPSS/Microsoft Excel].

F. Ethical Statement and Data Source

This study did not use real patient medical record data and therefore did not require Ethical Clearance approval from a health research ethics committee. All diagnosis narratives consisted of synthetic data collaboratively created by the research team alongside students and lecturers of the RMIK Study Program, representing typical clinical case patterns without containing any real patient identities or protected health information. Nevertheless, third-party API services were configured in accordance with recommended data security practices (see System Architecture section) to simulate production conditions required for future deployment using real clinical data under appropriate ethical approval.

RESULTS

The study included 25 synthetic diagnosis narratives meeting the inclusion criteria, compiled and agreed upon by RMIK students and lecturers. Sample characteristics are presented in Table I.

Table 1. Characteristics of Synthetic Diagnosis Narrative Samples ($n = 25$)

Characteristic	Category	n (%)
Type of Care	Inpatient	15 (60%)
	Outpatient	10 (40%)
Diagnosis Complexity	Single Diagnosis	16 (64%)
	Diagnosis with Comorbidities (≥ 2 diagnoses)	9 (36%)
ICD-10 Chapter Group	Chapter I – Certain Infectious and Parasitic Diseases (A00–B99)	25 (100%)

A comparison between the LLM-based system recommendations and the expert validators' reference standard codes is presented in Table II. Overall, the system achieved higher top-3 accuracy (76.0%) than top-1 accuracy (64.0%), indicating that the correct code frequently appeared among the top candidate recommendations even when not ranked first.

Table 2. Performance of Llm-Based Icd Code Recommendations Against Expert Validator Reference Standards ($n = 25$)

Metric	Value	Interpretation
Top-1 Accuracy	64.0% (16/25)	Fair
Top-3 Accuracy	76.0% (19/25)	Good
Precision	0.66	—
Recall (Sensitivity)	0.64	—
F1-score	0.65	—
Cohen's Kappa	0.58	Moderate
Mean Response Time	3.2 seconds	—

Because this initial trial was restricted to a single ICD-10 chapter (Chapter I), accuracy variations were analyzed across case complexity rather than across chapters. System performance was higher for single-diagnosis narratives utilizing explicit,

standardized terminology, whereas performance declined on narratives involving multiple comorbidities or non-standard abbreviations. This pattern aligns with previous findings showing that LLM performance in ICD coding is heavily influenced by case complexity and prompting strategies [2], [5], indicating that performance across other ICD-10 chapters with more intricate coding structures (such as chapters with extensive combination codes) requires evaluation in subsequent studies.

(See Fig. 2: User Interface of the LLM-Based ICD Code Recommendation Module)

Usability evaluation results from the 20 respondents (RMIK students and lecturers) using the SUS are presented in Table III. The overall mean SUS score fell within the "Good" category, with the highest score recorded in the ease-of-use dimension and a comparatively lower score in trust toward system recommendations. This suggests that respondents still felt the need to verify system outputs rather than relying on them unconditionally, aligning with the human-in-the-loop principle embedded in the system design.

Table 3. System Usability Scale (Sus) Evaluation Results ($n = 20$)

Dimension	Mean Score	Category
Ease of Use	78.0	Good
Trust in Recommendations	70.5	Good
Work Time Efficiency	75.0	Good
Total SUS Score (Mean)	**74.5**	**Good**

DISCUSSION

A. Interpretation of Findings and Technical Feasibility of API Key Integration

The findings in Table II demonstrate that integrating an LLM via an API key can generate ICD code recommendations with 64.0% top-1 accuracy and 76.0% top-3 accuracy, albeit within a limited trial of 25 synthetic diagnosis narratives from a single ICD-10 chapter. The moderate agreement level (Kappa = 0.58) indicates that system recommendations do not yet fully align with expert validator assessments, leaving substantial room for improvement before autonomous operation could be considered. However, the 12-percentage-point gap between top-1 and top-3 accuracy holds practical significance: in daily coding workflows, coders do not need to rely on the system as a decision replacement, but rather as a candidate shortlist that narrows the search space within an ICD-10 index containing thousands of codes. Thus, higher top-3 accuracy offers meaningful practical utility even when top-1 accuracy is suboptimal. In other words, the primary value of the LLM lies not in full automation, but in accelerating the initial abstracting stage prior to manual verification.

This API key-based approach also offers practical advantages over developing or deploying custom-trained/fine-tuned classification models or self-hosted LLMs as investigated in prior literature [4], [8], [12]: institutions do not need to provision computational infrastructure for model training, do not require massive labeled datasets, and can leverage ongoing language capability updates from LLM providers without internal retraining. Furthermore, the pay-per-use API pricing model based on token

volume substantially lowers upfront implementation costs compared to procuring bespoke AI systems, making it potentially more accessible for small-to-medium-sized hospitals in Indonesia.

B. Comparison with Previous Studies

Compared to studies relying on conventional classification models custom-trained or fine-tuned specifically for ICD coding [8], [12], our findings indicate that an off-the-shelf generative LLM utilizing zero-shot/few-shot prompting offers greater flexibility across varied diagnosis narratives without requiring model retraining. This aligns with research evaluating zero-shot LLM reasoning capabilities for medical coding [3] and studies demonstrating that off-the-shelf LLMs can perform competitively alongside specialized models when paired with robust verification mechanisms [6], [7]. Nevertheless, the observed decline in accuracy on cases with multiple comorbidities mirrors limitations documented in systematic reviews on LLM-based ICD coding [5], suggesting that complex multi-condition cases pose cross-methodological challenges rather than a flaw unique to a single model type. Similarly, our top-3 accuracy aligns with benchmark findings evaluating LLMs as medical code selectors in other classification systems (ICPC), which reported significant performance gains when evaluation criteria expanded from a single output to top-ranked candidates [11], reinforcing the positioning of LLMs as candidate-filtering aids rather than standalone final decision-makers.

C. Prompt Engineering and Recommendation Quality Governance

LLM output quality is heavily governed by prompt design. In this study, prompt construction followed three core principles:

1. Embedding explicit context consisting of relevant ICD-10 index excerpts (retrieval-augmented generation) so the model does not rely solely on internal parametric knowledge that may contain outdated code versions.
2. Enforcing structured output (JSON) containing the code, official code description, and confidence score to enable programmatic backend validation before user display.
3. Including explicit instructions directing the model to express uncertainty ("uncertain") rather than forcing an answer when diagnosis narratives are ambiguous, serving as a hallucination mitigation strategy consistent with literature establishing verification as mandatory in zero-shot clinical coding [6].

Without such grounding mechanisms, LLMs risk generating linguistically plausible codes that do not exist in the current ICD-10 edition or violate combination coding rules, as demonstrated in research examining LLM sensitivity to knowledge injection strategies [2].

D. Practical Implications for Medical Record Units and the PMIK Profession

For medical record departments, API key-based LLM integration shifts the PMIK role from "code lookup" toward "code verification," potentially allowing coders to dedicate deeper clinical reasoning to complex cases while accelerating routine coding tasks. This aligns with the usability results in Table III, which reflect positive perceptions of work-time efficiency. For hospital management, faster coding cycles may expedite INA-CBG claim submissions and reduce uncollectible receivables caused by verification delays. At the professional policy level, these findings inform discussions on emerging digital competencies, suggesting that generative AI literacy—including foundational knowledge of API workflows, prompting, and model limitations—should be integrated into RMIK curricula, echoing systematic review findings that successful clinical LLM adoption requires user readiness and workflow alignment [10]. Expanding LLM

utilization to support healthcare data interoperability across information systems [9] also opens avenues for adjacent modules, such as automated discharge summarization or clinical data exchange.

E. Data Security Governance and Regulatory Compliance

Transmitting diagnosis narratives to third-party LLM providers via API raises critical medical record confidentiality concerns. Under Indonesian Law No. 27 of 2022 on Personal Data Protection (*Undang-Undang Pelindungan Data Pribadi*/UU PDP), health data are classified as specific personal data; thus, their processing—including transmission to third-party API providers—requires a lawful basis, data minimization principles, and formal data processing agreements. In the tested architecture, risks were mitigated through: (a) automated de-identification prior to transmission, (b) selecting API service tiers with zero/limited data retention policies prohibiting customer data use for model retraining, (c) encryption in transit (HTTPS/TLS) and server-side secret manager storage for API keys, and (d) access logging (audit trails) for accountability. For institutions facing stricter data-residency requirements, prior research demonstrates the feasibility of self-hosted LLMs for sensitive clinical coding tasks such as tumor pathology reports, offering a middle ground between commercial API convenience and full data control [4]. Nonetheless, hospitals planning production deployment of commercial APIs must conduct a formal Privacy Impact Assessment (PIA) and execute formal confidentiality agreements with LLM providers given the high risk associated with health data breaches.

F. Study Limitations

Several limitations must be considered when interpreting these findings. First, the dataset consisted of synthetic narratives rather than real hospital medical records, meaning the tested narratives may not fully capture the unstructured documentation, diverse local abbreviations, or ambiguous handwriting encountered in real clinical practice. Second, the sample size was relatively small ($n = 25$) and restricted to a single ICD-10 chapter (Chapter I), limiting generalizability to chapters with more complex coding structures such as combination codes, fifth-character subclassifications, or intricate primary-secondary sequencing rules. Third, reference standard codes were established by RMIK lecturers and students rather than practicing hospital coders experienced in operational INA-CBG claims, meaning concordance rates might differ when evaluated against seasoned clinical coders. Fourth, this evaluation did not assess long-term operational costs under token-based pricing or longitudinal system stability, including potential output shifts caused by provider-side model updates (model drift). Accordingly, these results should be interpreted as early technical feasibility evidence rather than proof of production readiness. Future research should expand across multiple ICD-10 chapters, increase sample sizes, utilize real clinical data under formal ethical approval, and involve certified hospital coders as validators.

CONCLUSION

Fundamental Finding: Integrating a Large Language Model via an API key into a coding-support prototype is technically feasible as an ICD code recommendation aid, yielding promising preliminary results on synthetic data (64.0% top-1 accuracy, 76.0% top-3 accuracy, and a mean SUS score of 74.5 rated as Good). This approach offers implementation efficiency compared to custom-trained classification models by eliminating model training from scratch and lowering upfront adoption costs via pay-per-use pricing. **Implication:** The system should be positioned strictly as a decision-

support tool operating within a human-in-the-loop framework, wherein final code verification remains the authority and responsibility of the coder to mitigate model hallucinations and protect patient privacy. **Limitation:** However, given the small scale of this study (25 synthetic diagnosis narratives from a single ICD-10 chapter validated within an academic setting), these findings cannot yet be generalized as readiness for hospital deployment. **Future Research:** Future studies are recommended to expand coverage across ICD-10 chapters, enlarge sample sizes, evaluate real clinical data with appropriate ethical clearance, and engage certified hospital coders as validators prior to operational hospital adoption.

REFERENCES

[1] J. Ong *et al.*, "Applying large language model artificial intelligence for retina International Classification of Diseases (ICD) coding," *J. Med. Artif. Intell.*, 2023, doi: 10.21037/jmai-23-106.

[2] X. Jiang *et al.*, "The performance boundaries of knowledge prompts: a study on injection strategies and scale effects for ICD coding tasks in large language models," *Digit. Health*, vol. 12, 2026, doi: 10.1177/20552076261434052.

[3] P. Naliyatthaliyazchayil, R. Muthyala, J. Gichoya, and S. Purkayastha, "Evaluating the reasoning capabilities of large language models for medical coding and hospital readmission risk stratification: zero-shot prompting approach," *J. Med. Internet Res.*, vol. 27, 2025, doi: 10.2196/74142.

[4] K. Arzideh *et al.*, "Automated tumor International Classification of Diseases coding of real-world pathology reports using self-hosted large language models," *JCO Clin. Cancer Inform.*, vol. 10, Art. no. e2500254, 2026, doi: 10.1200/cci-25-00254.

[5] A. Gershon, S. Soffer, G. N. Nadkarni, and E. Klang, "Automatic ICD coding using LLMs: a systematic review," *medRxiv*, preprint, 2025, doi: 10.1101/2025.07.30.25330916.

[6] S. Li *et al.*, "Verification is all you need: prompting large language models for zero-shot clinical coding," *IEEE J. Biomed. Health Inform.*, vol. 29, pp. 8536–8549, 2025, doi: 10.1109/jbhi.2025.3593028.

[7] J. S. Boyle, A. Kascenas, P. Lok, M. Liakata, and A. Q. O'Neil, "Automated clinical coding using off-the-shelf large language models," *arXiv*, preprint arXiv:2310.06552, 2023, doi: 10.48550/arxiv.2310.06552.

[8] Z. Hou, H. Liu, J. Bian, X. He, and Z. Yan, "Enhancing medical coding efficiency through domain-specific fine-tuned large language models," *NPJ Health Syst.*, vol. 2, 2025, doi: 10.1038/s44401-025-00018-3.

[9] D. Yoon *et al.*, "Redefining health care data interoperability: empirical exploration of large language models in information exchange," *J. Med. Internet Res.*, vol. 26, 2024, doi: 10.2196/56614.

[10] Y. Artsi *et al.*, "Large language models in real-world clinical workflows: a systematic review of applications and implementation," *Front. Digit. Health*, vol. 7, 2025, doi: 10.3389/fdgth.2025.1659134.

[11] V. A. De Almeida *et al.*, "Large language models as medical code selectors: a benchmark using the International Classification of Primary Care," *JAMIA Open*, vol. 9, no. 1, Art. no. ooag017, 2026, doi: 10.1093/jamiaopen/ooag017.

[12] C.-W. Huang, S.-C. Tsai, and Y.-N. Chen, "PLM-ICD: automatic ICD coding with pretrained language models," *arXiv*, preprint arXiv:2207.05289, 2022, doi: 10.48550/arxiv.2207.05289.

[13] J. Brooke, "SUS: A 'quick and dirty' usability scale," in *Usability Evaluation in Industry*, P. W. Jordan, B. Thomas, B. A. Weerdmeester, and I. L. McClelland, Eds. London, U.K.: Taylor & Francis, 1996, pp. 189–194.

[14] D. Fauzi, M. Isnan, K. S. Nugroho, and B. Pardamean, *Machine Learning & Deep Learning*. Padang, Indonesia: Media Scitech, 2025.

[15] J. R. Landis and G. G. Koch, "The measurement of observer agreement for categorical data," *Biometrics*, vol. 33, no. 1, pp. 159–174, Mar. 1977, doi: 10.2307/2529310.

*__Budi Fajar Supriyanto__ (Corresponding author)
Politeknik Negeri Jember, Jember, Indonesia
Email: fajar.deroom@gmail.com

Muhammad Hilmy Haidar Fawwazie
Universitas Brawijaya, Malang, Indonesia

Brian Rizky Solehuddin Bachtiar
Universitas Bina Nusantara, Jakarta, Indonesia

Syafa Alviatur Sudarto Putri
Politeknik Negeri Jember, Jember, Indonesia

Wafiq Nawa Sabila
Politeknik Negeri Jember, Jember, Indonesia

Dzulfiqor Aflahi Hasan Mumtaz
Politeknik Negeri Jember, Jember, Indonesia