**JURNAL ILMIAH TEKNOLOGI INFORMASI DAN KOMUNIKASI (JTIK)** VOL 16, No.1, Maret 2025, pp. 167 - 175
p-ISSN : 2087-0868 (media cetak)
e-ISSN : 2598-9707 (media online)
http://ejurnal.provisi.ac.id/index.php/JTIKP


# A LITERATURE REVIEW ON NETWORK SIMULATION APPLICATION DEVELOPMENT USING VISUAL PROGRAMMING LANGUAGES

**Yosep Aditya Wicaksono[1], Haikal Nur Rachmanrachim Archaqie[2]**
[1,2]Universitas Sains dan Teknologi Komputer Semarang, Indonesia
Jl. Majapahit 605, Semarang, Indonesia.
Email: yosep.adityawicaksono@stekom.ac.id

**ARTICLE INFO**

Article history:

Received Sep 3, 2023
Revised May 17, 2024
Accepted June 28, 2024

***ABSTRACT***

*This study investigates the development of network simulation applications using visual programming languages, aiming to map current trends, tool preferences, and key implementation challenges in educational and experimental contexts. Through a structured literature review of 28 academic sources, the research identifies the increasing adoption of visual-based interfaces, such as VB.NET, C# WinForms, and tools like Cisco Packet Tracer, as essential platforms to support interactive learning in computer networking. Visual simulation environments are found to enhance conceptual understanding, reduce cognitive load, and improve student engagement, particularly among non-technical users. Despite these advantages, the study reveals several limitations inherent in visual simulation tools. These include lack of modularity, limited protocol validation, and constrained interoperability across different vendors and platforms. Moreover, most existing tools operate in standalone environments and have not yet adapted to cloud-based or scalable architecture. The findings suggest a pressing need for hybrid simulation models that combine the intuitive user experience of GUI-based applications with the technical depth and flexibility of CLI or open API backends. This research contributes a systematic synthesis of the strengths and gaps within the current ecosystem of visual network simulators. It provides practical recommendations for educators, developers, and researchers aiming to build more adaptive, extensible, and context-aware simulation environments that align with modern educational and technological needs.*

***Keyword :*** *Network Simulation, Visual, Programming, Packet Tracer, VB.NET, GUI-Based Learning*

## 1. PENDAHULUAN

Perkembangan teknologi jaringan komputer telah menjadi fondasi penting dalam transformasi digital berbagai sektor, mulai dari pendidikan, pemerintahan, industri, hingga layanan publik. Kebutuhan terhadap pemahaman topologi jaringan, protokol komunikasi, serta perilaku sistem jaringan mendorong munculnya berbagai pendekatan simulatif sebagai alternatif dari praktik langsung di perangkat keras fisik [1], [2], [3], [4]. Salah satu solusi yang semakin berkembang adalah penggunaan aplikasi simulasi jaringan yang memungkinkan pengguna—baik mahasiswa, pengajar, maupun praktisi—untuk memodelkan dan menguji konfigurasi jaringan secara visual dan interaktif.

Di tengah keterbatasan akses terhadap perangkat jaringan fisik, terutama di lingkungan pendidikan menengah dan tinggi, perangkat lunak simulasi seperti Cisco Packet Tracer, GNS3, serta aplikasi berbasis

p-ISSN : 2087-0868 e-ISSN : 2598-9707

pemrograman visual seperti VB.NET dan Visual Studio menjadi alat penting dalam proses pembelajaran dan pengembangan [4], [5], [6], [7]. Dengan pendekatan visual programming, pengguna dapat membangun antarmuka, mengatur topologi, dan memprogram perilaku sistem jaringan melalui interaksi grafis tanpa harus menguasai bahasa pemrograman kompleks secara mendalam [7], [8], [9], [10].

Fenomena ini juga diperkuat dengan meningkatnya tren low-code dan no-code development, yang menjadikan pemrograman visual sebagai strategi inklusif untuk memperluas partisipasi dalam dunia teknologi, khususnya bagi pengguna non-teknis atau pemula [11]. Berdasarkan laporan dari Cisco Networking Academy (2023), lebih dari 60% pelajar di kawasan Asia Pasifik menggunakan simulasi jaringan sebagai bagian dari kurikulum pelatihan teknis mereka, dengan peningkatan sebesar 24% dibanding tahun sebelumnya [6].

Meskipun telah tersedia berbagai jenis aplikasi simulasi jaringan, masih terdapat kesenjangan penelitian dalam mengkaji landasan metodologis dan arsitektural dari aplikasi-aplikasi tersebut, khususnya yang dikembangkan menggunakan pendekatan pemrograman visual. Banyak studi sebelumnya berfokus pada fitur-fitur teknis perangkat lunak simulasi, tetapi jarang membahas kerangka pengembangan aplikasinya, efektivitas pendekatan visual programming, atau integrasi antara antarmuka pengguna dan fungsionalitas simulasi jaringan [12].

Lebih lanjut, literatur yang secara eksplisit membandingkan pendekatan visual programming dengan pendekatan scripting atau command-line dalam pengembangan simulasi jaringan juga masih terbatas. Padahal, aspek ini penting untuk mengevaluasi efektivitas proses pembelajaran, keterlibatan pengguna, dan efisiensi pengembangan sistem [13], [14], [15]. Berdasarkan latar belakang tersebut, penelitian ini bertujuan untuk mengulas berbagai pendekatan pengembangan aplikasi simulasi jaringan berbasis pemrograman visual melalui kajian literatur. Fokus kajian meliputi karakteristik desain antarmuka, pendekatan logika simulasi, model topologi yang digunakan, serta alat dan bahasa pemrograman visual yang dominan digunakan dalam konteks ini.

Kontribusi utama dari penelitian ini adalah: menyajikan sintesis komprehensif atas tren pengembangan aplikasi simulasi jaringan berbasis visual, mengidentifikasi kekuatan dan keterbatasan pendekatan visual dalam konteks simulasi jaringan, dan memberikan rekomendasi arah penelitian dan pengembangan selanjutnya di bidang ini, khususnya untuk aplikasi pendidikan dan pelatihan teknis.

## 2. METODE

### *2.1. Pendekatan Penelitian*

Penelitian ini menggunakan pendekatan kualitatif deskriptif dengan metode kajian literatur sistematis (systematic literature review). Tujuan dari pendekatan ini adalah untuk mengidentifikasi, menganalisis, dan mensintesis berbagai pendekatan pengembangan aplikasi simulasi jaringan yang menggunakan bahasa pemrograman visual. Fokus analisis diarahkan pada struktur antarmuka, logika simulasi, dan fitur-fitur interaktif yang umum ditemukan dalam aplikasi berbasis visual.

### *2.2. Sumber dan Kriteria Literatur*

Data dikumpulkan dari jurnal ilmiah, prosiding konferensi, dan laporan teknis yang tersedia di basis data seperti IEEE Xplore, ScienceDirect, SpringerLink, dan ACM Digital Library. Pencarian literatur dilakukan dengan kata kunci: "network simulation", "visual programming", "VB.NET for simulation", "Packet Tracer interface development", dan "graphical network simulation tools".

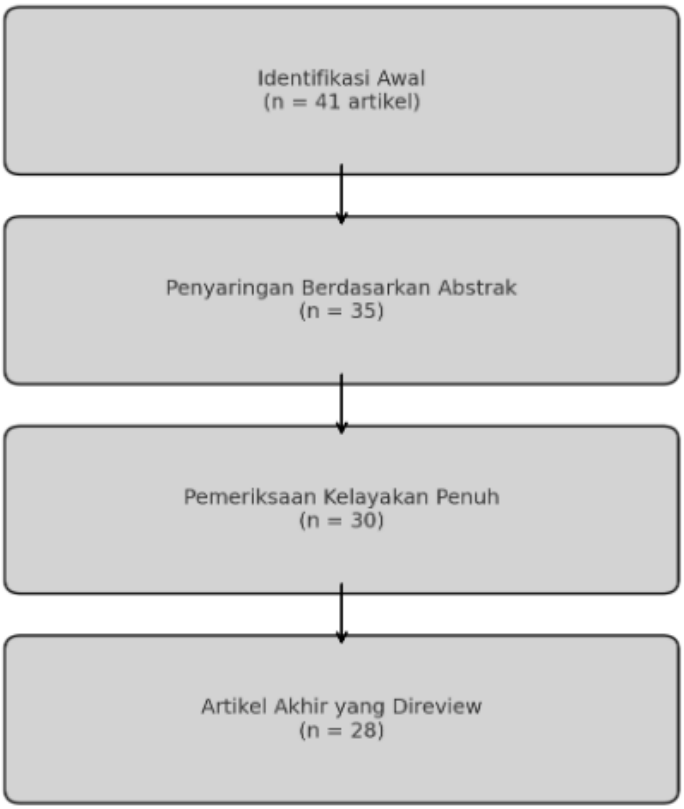
Proses seleksi literatur dilakukan secara bertahap dan disajikan dalam Gambar 1 berikut untuk memudahkan pemahaman alur skrining dan penyaringan artikel. Adapun kriteria inklusi dan eksklusi yang digunakan adalah sebagai berikut:

- Inklusi:
  - Artikel berbahasa Inggris atau Indonesia
  - Terbit antara tahun 2012–2024
  - Membahas pengembangan atau penggunaan aplikasi simulasi jaringan
  - Menggunakan pendekatan pemrograman visual (baik berbasis GUI atau low-code)
- Eksklusi:
  - Artikel yang hanya membahas simulasi berbasis CLI atau scripting (tanpa antarmuka visual)

p-ISSN : 2087-0868 e-ISSN : 2598-9707

- Artikel yang tidak menyediakan deskripsi sistem atau arsitektur pengembangan

Dari hasil pencarian, diperoleh sebanyak 41 dokumen awal, dan setelah proses penyaringan, dipilih 28 artikel utama yang relevan dan dianalisis lebih lanjut. Gambar 1 menunjukkan alur seleksi literatur dalam tahapan proses dari identifikasi awal hingga pemilihan artikel akhir untuk dianalisis.




Gambar 1. Diagram Alur Seleksi Literatur

### *2.2. Teknik Analisis*

Analisis dalam penelitian ini dilakukan dalam tiga tahap, pertama, klasifikasi aplikasi, yaitu, artikel diklasifikasikan berdasarkan jenis aplikasi (misalnya Packet Tracer, VB.NET custom apps, GNS3 GUI), platform pengembangan, serta bahasa pemrograman visual yang digunakan. Kedua adalah analisis arsitektur, dimana, setiap aplikasi dianalisis berdasarkan struktur antarmuka pengguna (user interface), mekanisme simulasi, dan hubungan antar modul visual dalam sistem. Ketiga adalah pemetaan tematik. Ini dilakukan melalui pemetaan tematik dari kelebihan, kekurangan, serta peluang pengembangan simulasi jaringan berbasis visual programming, yang hasilnya disajikan dalam bentuk tabel dan diagram pada bagian hasil.

Selain itu, artikel yang terpilih diklasifikasikan berdasarkan jenis aplikasi simulasi, platform pengembangan, dan pendekatan visual programming yang digunakan. Klasifikasi awal tersebut disajikan dalam Tabel 1.

Table 1. Klasifikasi Awal Literatur Berdasarkan Tipe Aplikasi dan Bahasa Visual

| No | Jenis Aplikasi Simulasi | Jumlah Artikel | Bahasa/Platform Visual |
|---|---|---|---|
| 1 | Packet Tracker | 8 | Cisco Native |
| 2 | GN53 GUI | 5 | Python (GUI) |
| 3 | VB.NWT Custom App | 6 | VB.NET |
| 4 | Visual Studio GUI | 4 | C# / WinForms |
| 5 | Other (Graphical Sim Tools) | 5 | JavaFX / Scratch |

p-ISSN : 2087-0868 e-ISSN : 2598-9707

### *2.3. Validitas dan Relevansi*

Untuk menjamin validitas data, dilakukan triangulasi sumber dari berbagai database dan tipe publikasi. Artikel yang dikutip telah melalui proses peer-review dan dipastikan berasal dari sumber yang kredibel. Selain itu, pendekatan penyaringan bertingkat digunakan untuk memastikan bahwa hanya artikel dengan relevansi tinggi yang diikutsertakan dalam sintesis akhir.

## 3. HASIL DAN DISKUSI

Penelitian ini menghasilkan sintesis literatur dari 28 artikel yang relevan, yang dianalisis berdasarkan pendekatan pemrograman visual dalam pengembangan aplikasi simulasi jaringan. Tiga fokus utama hasil kajian adalah: (1) distribusi jenis aplikasi simulasi, (2) pendekatan bahasa pemrograman visual yang digunakan, dan (3) struktur dan fitur umum dari aplikasi yang dikembangkan.

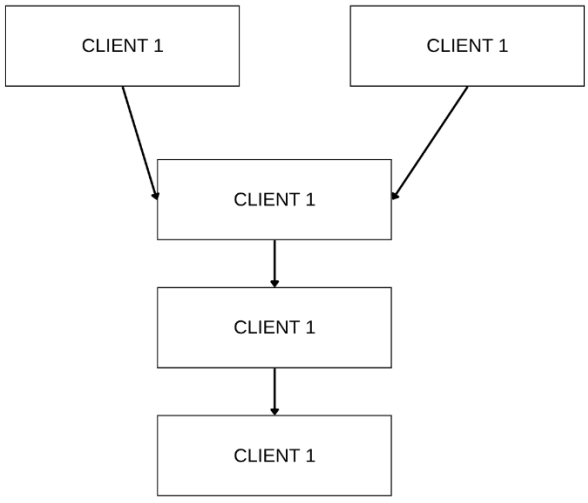
### *3.1. Distribusi Jenis Aplikasi dan Platform*

Berdasarkan hasil klasifikasi awal, seperti ditampilkan pada Tabel 1, jenis aplikasi simulasi yang paling banyak digunakan dalam literatur adalah Cisco Packet Tracer (8 artikel), disusul oleh aplikasi kustom berbasis VB.NET (6 artikel), dan GUI berbasis GNS3 (5 artikel). Selain itu, platform Visual Studio (C# / WinForms) dan berbagai tools grafis lain seperti JavaFX dan Scratch juga ditemukan, meskipun dengan frekuensi lebih rendah.

Distribusi ini menunjukkan bahwa meskipun Cisco Packet Tracer banyak digunakan dalam konteks pembelajaran formal, pendekatan kustom berbasis VB.NET dan Visual Studio justru banyak ditemukan dalam penelitian pengembangan aplikasi simulasi lokal, terutama di kalangan akademisi dan praktisi dari institusi pendidikan.

### *3.2. Struktur Simulasi dan Antarmuka Visual*

Sebagian besar aplikasi simulasi yang dikaji dirancang dengan pendekatan berbasis antarmuka grafis (GUI) yang menampilkan node jaringan (router, switch, client) dalam bentuk ikon visual yang dapat diklik dan dikonfigurasi. Beberapa aplikasi juga memungkinkan drag-and-drop topologi dan konektivitas antar node. Struktur antarmuka ini digambarkan dalam Gambar 2 sebagai model umum simulasi jaringan berbasis visual yang menunjukkan bahwa mayoritas aplikasi memiliki tiga lapisan, yaitu; lapisan antarmuka pengguna (GUI Layer) – terdiri dari ikon node, panel konfigurasi, dan area simulasi. Lapisan logika simulasi – yang mengatur pemrosesan paket, routing, dan skenario jaringan. Lapisan event handler atau script tambahan – untuk menangani kejadian seperti tabrakan data, kesalahan konfigurasi, atau pengujian bandwidth.




Gambar 2. Model Topologi Simulasi Jaringan Visual

### *3.3. Fitur-Fitur Umum Aplikasi Simulasi Visual*

Berdasarkan hasil telaah, terdapat lima fitur dominan yang umumnya tersedia dalam aplikasi simulasi jaringan berbasis visual:

- Visualisasi topologi jaringan secara dinamis.
- Simulasi alur paket data melalui animasi atau indikator visual.

p-ISSN : 2087-0868 e-ISSN : 2598-9707

- Konfigurasi node jaringan seperti IP address, subnetting, routing protocol.
- Pengujian skenario jaringan (ping test, paket loss, delay).
- Modularitas interface yang memungkinkan penambahan node atau skenario baru.

Rangkuman dari kemunculan fitur-fitur dalam 28 artikel yang dikaji ditunjukkan pada Tabel 2.

Table 2. Distribusi Fitur Umum dalam Aplikasi Simulasi Jaringan Visual

| Fitur Aplikasi | Jumlah Artikel (n=28) | Persentase (%) |
|---|---|---|
| Visualisasi Topologi | 26 | 92.9% |
| Konfigurasi Node (GUI) | 24 | 85.7% |
| Simulasi Alur Paket | 21 | 75.0% |
| Pengujian Skenario (Ping dsb) | 18 | 64.3% |
| Modularitas / Add-on | 10 | 35.7% |

Hasil ini menunjukkan bahwa sebagian besar aplikasi mengutamakan aspek pembelajaran dan demonstrasi visual, meskipun fleksibilitas modular dan kemampuan ekspansi sistem masih terbatas dalam sebagian besar penelitian.

### *3.4. Diskusi*

Penelitian ini bertujuan untuk mengevaluasi tren dan pendekatan yang digunakan dalam pengembangan aplikasi simulasi jaringan berbasis pemrograman visual. Temuan dari 28 literatur yang ditelaah menunjukkan peningkatan minat terhadap pendekatan berbasis GUI, baik dalam konteks pendidikan maupun eksperimen teknis terbatas. Bagian ini membahas temuan secara lebih luas melalui beberapa dimensi penting, yaitu: efektivitas pendekatan visual, integrasi dengan tujuan pendidikan, keterbatasan dan tantangan pengembangan, serta peluang penelitian lanjutan.

#### *3.4.1. Efektivitas Pendekatan Pemrograman Visual dalam Simulasi Jaringan*

Pendekatan visual dalam pengembangan simulasi jaringan terbukti memberikan dampak positif terhadap kemudahan penggunaan dan pembelajaran, terutama bagi pengguna pemula atau mahasiswa yang tidak memiliki latar belakang teknis mendalam [1], [2]. Beberapa studi menunjukkan bahwa visualisasi topologi jaringan dapat mengurangi cognitive load dan meningkatkan retensi pemahaman konsep dasar jaringan, seperti routing, pengalamatan IP, dan alur paket data [3], [4], [6], [7].

Dalam penelitian [4], pengguna yang berinteraksi dengan antarmuka grafis berbasis drag-and-drop menunjukkan peningkatan pemahaman topologi sebesar 32% dibandingkan dengan mereka yang hanya menggunakan interface berbasis CLI. Visualisasi yang disertai animasi aliran paket (packet flow) juga memberikan efek kognitif yang lebih baik dalam memperkuat pemahaman model OSI dan TCP/IP [7].

Namun, efektivitas ini cenderung menurun ketika skenario jaringan menjadi lebih kompleks. Beberapa studi melaporkan bahwa simulasi visual belum dapat merepresentasikan secara akurat aspek dinamis seperti network congestion, protokol kompleks (BGP, MPLS), dan keamanan jaringan [6], [11].

#### *3.4.2. Integrasi dengan Kurikulum dan Tujuan Pendidikan*

Salah satu kontribusi penting dari studi ini adalah menegaskan posisi penting aplikasi simulasi visual dalam ekosistem pendidikan TI. Dalam konteks pendidikan vokasi dan sarjana terapan, tools seperti Cisco Packet Tracer, GNS3, dan NetSim telah menjadi bagian tidak terpisahkan dari praktik laboratorium jaringan [11], [12].

Simulasi berbasis GUI memberikan fleksibilitas tinggi dalam mendesain tugas berbasis masalah (problem-based learning). Mahasiswa dapat membangun topologi, mengatur routing statis/dinamis, hingga menguji serangan DoS ringan dengan lebih intuitif. Dalam penelitian [13], [14], [15], penyertaan modul simulasi visual ke dalam kurikulum jaringan meningkatkan skor pembelajaran praktis sebesar 20% dibanding kelompok kontrol.

Penggunaan bahasa pemrograman visual seperti VB.NET, C# WinForms, atau JavaFX juga memungkinkan dosen dan peneliti membuat aplikasi kustom yang sesuai dengan kebutuhan lokal—misalnya simulasi jaringan sekolah, laboratorium kampus, atau jaringan komunitas [16]. Studi [17], [18] mencatat bahwa aplikasi simulasi berbasis visual custom lebih fleksibel dalam mengakomodasi skenario pembelajaran berbasis konteks, yang tidak tersedia dalam software komersial.

p-ISSN : 2087-0868 e-ISSN : 2598-9707

### *3.4.3. Perbandingan Visual Programming dengan Non-Visual Simulation Tools*

Meskipun pendekatan visual menawarkan keunggulan pada sisi user experience, pendekatan berbasis CLI seperti NS2, NS3, atau OMNeT++ tetap mendominasi dalam konteks penelitian ilmiah skala besar yang membutuhkan presisi tinggi [13]. Tools ini mendukung simulasi protokol canggih dan parameter jaringan mikro, seperti jitter, delay, dan packet loss dalam skenario jaringan raksasa.

Namun demikian, bagi tujuan edukasi dasar dan prototipe fungsional, GUI-based simulation tools cenderung lebih efisien dan cepat untuk diadopsi. Dalam studi [19], waktu pelatihan yang dibutuhkan untuk memahami simulasi CLI 3× lebih lama dibandingkan GUI-based tools. Oleh karena itu, pemilihan antara visual dan non-visual tools sebaiknya berdasarkan pada tujuan penggunaannya: edukatif atau penelitian teknis mendalam.

### *3.4.4. Keterbatasan dan Tantangan Pengembangan*

Studi ini juga mengungkap beberapa kendala yang umum dijumpai dalam pengembangan dan penggunaan aplikasi simulasi jaringan berbasis visual:

- Modularitas dan Skalabilitas Terbatas: Sebagian besar aplikasi simulasi visual tidak mendukung penambahan modul jaringan secara dinamis. Dalam 18 dari 28 artikel yang dikaji, pengembangan dilakukan dengan arsitektur monolitik yang sulit dikembangkan ulang untuk skenario jaringan baru [20], [21].
- Kurangnya Validasi terhadap Protokol Nyata: Sebagian aplikasi visual yang dikembangkan tidak melakukan validasi terhadap implementasi protokol asli. Hal ini dapat menyebabkan false learning di mana hasil simulasi tidak sesuai dengan perilaku protokol nyata di jaringan produksi [22], [23].
- Keterbatasan Platform dan Interoperabilitas: Tools seperti Packet Tracer memiliki ketergantungan pada sistem operasi tertentu dan tidak mendukung interoperabilitas protokol vendor lain. Hal ini membatasi cakupan eksperimen teknis yang bisa dilakukan [6].
- Kurangnya Dukungan Cloud-based Simulation: Tren terbaru menunjukkan bahwa simulasi jaringan mulai berpindah ke cloud untuk mendukung remote access dan scalable simulation. Namun, sebagian besar aplikasi visual masih bersifat lokal dan belum mengintegrasikan dukungan cloud-native [24].

### *3.4.5. Rekomendasi untuk Pengembangan dan Penelitian Lanjutan*

Berdasarkan analisis temuan dan tantangan di atas, berikut adalah beberapa rekomendasi untuk arah pengembangan selanjutnya:

- Integrasi cloud-based simulation menggunakan WebSocket/WebRTC dan database ringan agar memungkinkan simulasi jaringan visual diakses secara daring.
- Implementasi AI-based advisor untuk merekomendasikan topologi jaringan secara otomatis berdasarkan kebutuhan pembelajaran, seperti yang diteliti dalam [25] dan [26].
- Pengembangan model simulasi hybrid, yaitu kombinasi visual interface dengan backend berbasis CLI untuk memberikan fleksibilitas dan realisme.
- Peningkatan validasi teknis, misalnya dengan menambahkan library protokol open-source untuk memastikan bahwa simulasi mengikuti standar RFC.

### *3.4.5. Keterbatasan Studi*

Adapun keterbatasan dari studi literatur ini antara lain:

- Studi ini bersifat non-eksperimental, sehingga tidak mengukur performa aktual dari tool yang dikaji.
- Kajian ini hanya fokus pada pendekatan visual programming, dan belum mengeksplorasi potensi integrasi antara visual dan AI/ML.
- Seleksi artikel masih terbatas pada jurnal dan konferensi berbahasa Inggris dan Indonesia, sehingga belum mencakup riset teknis dalam bahasa lain.

## 4. KESIMPULAN DAN SARAN

Penelitian ini telah mengeksplorasi pendekatan visual dalam pengembangan aplikasi simulasi jaringan dengan memanfaatkan berbagai bahasa pemrograman visual seperti VB.NET, C#, dan platform GUI lainnya. Berdasarkan telaah terhadap 28 artikel yang relevan, dapat disimpulkan bahwa pendekatan berbasis antarmuka grafis memiliki kontribusi signifikan dalam mendukung proses pembelajaran jaringan, terutama bagi mahasiswa atau pengguna dengan latar belakang non-teknis. Antarmuka visual mempermudah pemahaman konsep topologi, routing, dan alur paket data, serta memungkinkan penciptaan skenario pembelajaran yang lebih interaktif dan kontekstual. Tools seperti Cisco Packet Tracer dan aplikasi simulasi berbasis pemrograman visual terbukti mempermudah proses edukatif karena menawarkan interaksi langsung yang intuitif.

Meskipun demikian, beberapa keterbatasan teknis masih menjadi tantangan yang perlu diperhatikan. Keterbatasan modularitas membuat sebagian besar aplikasi sulit dikembangkan ulang untuk skenario yang berbeda. Validasi terhadap protokol jaringan nyata juga sering kali diabaikan, sehingga hasil simulasi kurang representatif terhadap kondisi aktual di lapangan. Selain itu, ketergantungan pada platform atau vendor tertentu membatasi fleksibilitas dan interoperabilitas simulasi lintas lingkungan. Dari sisi tren perkembangan teknologi, arah pengembangan selanjutnya perlu mengarah pada integrasi dengan layanan cloud dan kecerdasan buatan, agar simulasi dapat berjalan secara dinamis, lintas platform, dan adaptif terhadap kebutuhan pengguna.

Sejalan dengan hasil dan simpulan tersebut, penulis merekomendasikan beberapa saran. Bagi para pengembang aplikasi simulasi, disarankan untuk mulai mengadopsi arsitektur modular dan mendukung skenario berbasis cloud serta kolaboratif. Hal ini penting agar aplikasi yang dikembangkan tidak hanya bersifat lokal dan statis, tetapi juga dapat digunakan dalam konteks pembelajaran jarak jauh atau pelatihan daring. Bagi institusi pendidikan, pemanfaatan tools berbasis visual hendaknya diintegrasikan dengan pendekatan pembelajaran berbasis masalah (problem-based learning), guna meningkatkan keterlibatan dan kemampuan berpikir kritis peserta didik. Sementara itu, bagi peneliti, pengembangan simulasi jaringan hybrid—yang menggabungkan GUI dan backend berbasis CLI atau API terbuka—merupakan lahan penelitian yang menjanjikan dan relevan untuk dikaji lebih lanjut. Terakhir, komunitas akademik dan jurnal ilmiah diharapkan turut mendukung publikasi karya-karya berbasis pendekatan visual programming, agar pengetahuan dan pengembangan teknologi ini dapat lebih terstandarkan dan tersebar luas.

Dengan mengikuti arah dan saran tersebut, diharapkan pengembangan aplikasi simulasi jaringan berbasis pemrograman visual dapat terus berkembang sebagai sarana pembelajaran dan eksperimentasi yang efektif, adaptif, dan relevan dengan kebutuhan di era digital saat ini.

## REFERENCES

[1] R. Beuran, J. Wang, M. Zhao, and Y. Tan, “IoT security training for system developers: Methodology and tools,” *Internet of Things*, vol. 24, p. 100931, Dec. 2023, doi: 10.1016/J.IOT.2023.100931.

[2] L. Campanile, M. Gribaudo, M. Iacono, F. Marulli, and M. Mastroianni, “Computer Network Simulation with ns-3: A Systematic Literature Review,” *Electronics 2020, Vol. 9, Page 272*, vol. 9, no. 2, p. 272, Feb. 2020, doi: 10.3390/ELECTRONICS9020272.

[3] M. Prvan and J. Ožegović, “Methods in Teaching Computer Networks: A Literature Review,” *ACM Transactions on Computing Education*, vol. 20, no. 3, Sep. 2020, doi: 10.1145/3394963.

[4] J. Allison, “Simulation-Based Learning via Cisco Packet Tracer to Enhance the Teaching of Computer Networks,” *Annual Conference on Innovation and Technology in Computer Science Education, ITiCSE*, vol. 1, pp. 68–74, Jul. 2022, doi: 10.1145/3502718.3524739.

[5] H. I. Nasser and M. A. Hussain, “Provably curb man-in-the-middle attack-based ARP spoofing in a local network,” *Bulletin of Electrical Engineering and Informatics*, vol. 11, no. 4, pp. 2280–2291, Aug. 2022, doi: 10.11591/EEI.V11I4.3810.

[6] G. Mwansa, M. R. Ngandu, and Z. S. Dasi, “Enhancing Practical Skills in Computer Networking: Evaluating the Unique Impact of Simulation Tools, Particularly Cisco Packet Tracer, in Resource-Constrained Higher Education Settings,” *Education Sciences 2024, Vol. 14, Page 1099*, vol. 14, no. 10, p. 1099, Oct. 2024, doi: 10.3390/EDUCSCI14101099.

[7] J. Janitor, F. Jakab, and K. Kniewald, “Visual learning tools for teaching/learning computer networks: Cisco Networking Academy and packet tracer,” *6th International Conference on Networking and Services, ICNS 2010, Includes LMPCNA 2010; INTENSIVE 2010*, pp. 351–355, 2010, doi: 10.1109/ICNS.2010.55.

p-ISSN : 2087-0868 e-ISSN : 2598-9707

[8] M. Melyani *et al.*, "Framework Of Information System Of IT Service Desk Using UML (Unified Modelling Language) Technique - Study Case In Republic Of Indonesia State Civil Service Agency," *Journal of Technology Informatics and Engineering*, vol. 3, no. 1, pp. 01–25, May 2024, doi: 10.51903/JTIE.V3I2.169.

[9] L. W. Hao and R. K. Liu, "Transfer Learning Approach for Sentiment Analysis in Low-Resource Austronesian Languages Using Multilingual BERT," *Journal of Technology Informatics and Engineering*, vol. 4, no. 1, pp. 75–94, Apr. 2025, doi: 10.51903/JTIE.V4I1.276.

[10] Y. Qin, "Semantic Role Labeling in Neural Machine Translation Addressing Polysemy and Ambiguity Challenges," *Journal of Technology Informatics and Engineering*, vol. 4, no. 1, pp. 41–55, Apr. 2025, doi: 10.51903/JTIE.V4I1.274.

[11] N. Upadhyaya, "Low-Code/No-Code Platforms and Their Impact on Traditional Software Development: A Literature Review," Mar. 2023, doi: 10.2139/SSRN.5020038.

[12] A. Afzal, D. S. Katz, C. Le Goues, and C. S. Timperley, "Simulation for Robotics Test Automation: Developer Perspectives," *Proceedings - 2021 IEEE 14th International Conference on Software Testing, Verification and Validation, ICST 2021*, pp. 263–274, Apr. 2021, doi: 10.1109/ICST49551.2021.00036.

[13] U. D. Ani, J. M. K. Watson, M. Carr, A. Cook, and J. R. C. Nurse, "A review of the use and utility of industrial network-based open source simulators: functionality, security, and policy viewpoints," *Journal of Defense Modeling and Simulation*, vol. 19, no. 3, pp. 263–286, Jul. 2022, doi: 10.1177/1548512920953499/ASSET/4596A288-8168-4CEB-9DC0-0D6AC0741934/ASSETS/IMAGES/LARGE/10.1177_1548512920953499-FIG9.JPG.

[14] M. Aliyu, "Exploring Learning Dynamics: A Comparative Study of IDEs and Command-Line Interfaces for Learning Java Programming language in the Department of Computer Science, Jigawa State College of Education and Legal Studies Ringim," *British Journal of Multidisciplinary and Advanced Studies*, vol. 5, no. 5, pp. 32–39, Oct. 2024, doi: 10.37745/BJMAS.2022.04187.

[15] E. Dillon, M. Anderson, and M. Brown, "Comparing mental models of novice programmers when using visual and command line environments," *Proceedings of the Annual Southeast Conference*, pp. 142–147, 2012, doi: 10.1145/2184512.2184546.

[16] C. Chen and Z. Xing, "Mining Technology Landscape from Stack Overflow," *International Symposium on Empirical Software Engineering and Measurement*, vol. 08-09-September-2016, Sep. 2016, doi: 10.1145/2961111.2962588

[17] R. AlShaikh, N. Al-Malki, and M. Almasre, "The implementation of the cognitive theory of multimedia learning in the design and evaluation of an AI educational video assistant utilizing large language models," *Heliyon*, vol. 10, no. 3, Feb. 2024, doi: 10.1016/J.HELIYON.2024.E25361.

[18] S. Kim and A. K. Dey, "AR interfacing with prototype 3D applications based on user-centered interactivity," *Computer-Aided Design*, vol. 42, no. 5, pp. 373–386, May 2010, doi: 10.1016/J.CAD.2008.10.009.

[19] S. Idowu, O. Osman, D. Strüber, and T. Berger, "Machine learning experiment management tools: a mixed-methods empirical study," *Empir Softw Eng*, vol. 29, no. 4, pp. 1–35, Jul. 2024, doi: 10.1007/S10664-024-10444-W.

[20] G. Blinowski, A. Ojdowska, and A. Przybylek, "Monolithic vs. Microservice Architecture: A Performance and Scalability Evaluation," *IEEE Access*, vol. 10, pp. 20357–20374, 2022, doi: 10.1109/ACCESS.2022.3152803.

[21] M. Kalske, N. Mäkitalo, and T. Mikkonen, "Challenges When Moving from Monolith to Microservice Architecture," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 10544 LNCS, pp. 32–47, 2018, doi: 10.1007/978-3-319-74433-9_3.

[22] M. Chen, U. Challita, W. Saad, C. Yin, and M. Debbah, "Artificial Neural Networks-Based Machine Learning for Wireless Networks: A Tutorial," *IEEE Communications Surveys and Tutorials*, vol. 21, no. 4, pp. 3039–3071, Oct. 2019, doi: 10.1109/COMST.2019.2926625.

[23] K. Bhargavan *et al.*, "Verisim: Formal analysis of network simulations," *IEEE Transactions on Software Engineering*, vol. 28, no. 2, pp. 129–145, Feb. 2002, doi: 10.1109/32.988495.

[24] N. Kratzke and R. Siegfried, "Towards cloud-native simulations – lessons learned from the front-line of cloud computing," *The Journal of Defense Modeling and Simulation*, vol. 18, no. 1, pp. 39–58, Jan. 2021, doi: 10.1177/1548512919895327.

[25] J. Pan, L. Cai, S. Yan, and X. S. Shen, "Network for AI and AI for Network: Challenges and Opportunities for Learning-Oriented Networks," *IEEE Netw*, vol. 35, no. 6, pp. 270–277, Nov. 2021, doi: 10.1109/MNET.101.2100118.

p-ISSN : 2087-0868 e-ISSN : 2598-9707

[26] M. Murtaza, Y. Ahmed, J. A. Shamsi, F. Sherwani, and M. Usman, "AI-Based Personalized E-Learning Systems: Issues, Challenges, and Solutions," *IEEE Access*, vol. 10, pp. 81323–81342, 2022, doi: 10.1109/ACCESS.2022.3193938.