Terbit *online* pada laman web jurnal : www.journal.unbara.ac.id/index.php/INTECH
INFORMATIKA DAN TEKNOLOGI (INTECH)
ISSN (Online) : 2722-7367



Analysis of Industrial Informatics Engineering Technology Students' Understanding of the JavaScript Programming Language

Analisis Tingkat Pemahaman Mahasiswa Teknologi Rekayasa Informatika Industri terhadap Penggunaan Bahasa Pemrograman *JavaScript*

Rangga Rizky Radityo[1], Ihsan Kamaludin[2], M. Dandi Al Idrus[3], Dzulhaqqi Nur Fadhilah[4], Adam Nibros Lausarodan[5], Ihsan Tanama Sitio[6]
1,2,3,4,5 Teknologi Rekayasa Informatika Industri, Politeknik Manufaktur Bandung, Bandung, Indonesia Jl. Kanayakan No. 21, Dago, Bandung
E-mail : *222443019@mhs.polman-bandung.ac.id [1], 222443014@mhs.polman-bandung.ac.id [2], 222443015@mhs.polman-bandung.ac.id [3], 222443011@mhs.polman-bandung.ac.id [4], 222443001@mhs.polman-bandung.ac.id [5], ihsan.sitio@polman-bandung.ac.id [6]

ABSTRACT

The rapid development of web technology demands high competency in the JavaScript programming language. However, students often experience a gap between understanding basic programming theories and applying complex logic in practical scenarios. This study aims to analyze students' understanding levels of JavaScript using a descriptive quantitative method through a survey approach. Data were collected using questionnaires and technical tests (snippet code) distributed to 99 fourth-semester students in the Industrial Informatics Engineering Technology Study Program. The results showed that students have a strong foundation in fundamental concepts such as variables and control structures. However, their understanding significantly decreased in dynamic topics like Document Object Model (DOM) manipulation and asynchronous programming. Furthermore, although students actively reuse code, the implementation of industry-standard practices like the Agile method and teamwork remains minimal. The main obstacles encountered are debugging difficulties and a lack of independent practice. Therefore, it is recommended that educational institutions increase the use of real-world projects and code review sessions to enhance students' mastery of advanced programming logic and industry readiness.

Keywords: JavaScript, student comprehension, web programming, descriptive survey, software engineering

ABSTRAK

Perkembangan teknologi web menuntut kompetensi tinggi dalam penggunaan bahasa pemrograman JavaScript. Namun, mahasiswa sering kali mengalami kesenjangan antara pemahaman teori dasar dan kemampuan penerapan logika yang kompleks di lapangan. Penelitian ini bertujuan untuk menganalisis tingkat pemahaman mahasiswa terhadap JavaScript menggunakan metode kuantitatif deskriptif melalui pendekatan survei. Data dikumpulkan menggunakan kuesioner dan uji teknikal (snippet code) yang disebarkan kepada 99 mahasiswa semester 4 di Program Studi Teknologi Rekayasa Informatika Industri. Hasil penelitian menunjukkan bahwa mahasiswa memiliki pemahaman dasar yang kuat pada materi fundamental seperti variabel dan struktur kontrol. Akan tetapi, tingkat pemahaman menurun signifikan pada materi dinamis seperti manipulasi Document Object Model (DOM) dan pemrograman asinkron (asynchronous). Selain itu, meskipun mahasiswa aktif melakukan penggunaan kembali kode (reuse code), penerapan praktik standar industri seperti metode Agile dan kerja tim masih sangat minim. Kendala utama yang dihadapi adalah kesulitan dalam proses debugging dan kurangnya latihan mandiri. Oleh karena itu, disarankan agar institusi pendidikan memperbanyak porsi studi kasus nyata (real-world projects) dan sesi telaah kode (code review) untuk meningkatkan penguasaan logika pemrograman tingkat lanjut serta kesiapan industri mahasiswa.

Kata Kunci : JavaScript, pemahaman mahasiswa, pemrograman web, survei deskriptif, rekayasa perangkat lunak

Informasi Artikel
Submit 12 Mei 2026 | Diterima 3 November 2026 | Publish Online 30 November 2026.
This is an open access article under the CC BY-NC 4.0 license.
Published by Universitas Baturaja.

PENDAHULUAN

Perkembangan teknologi informasi yang semakin pesat menuntut sumber daya manusia yang memiliki kemampuan di bidang pemrograman, khususnya dalam pengembangan aplikasi berbasis web. Saat ini, aplikasi web tidak hanya digunakan sebagai media informasi, tetapi juga sebagai platform layanan, transaksi, serta berbagai kebutuhan industri digital. Oleh karena itu, kemampuan dalam pengembangan web menjadi kompetensi penting bagi mahasiswa di bidang teknologi informasi. Salah satu bahasa pemrograman yang memiliki peran penting dalam pengembangan web modern adalah JavaScript. Bahasa pemrograman ini memungkinkan pengembang untuk membuat halaman web menjadi interaktif dan dinamis, seperti validasi formulir, manipulasi tampilan halaman, komunikasi data secara real-time, hingga pengembangan aplikasi berbasis web skala besar.

Pada Program Studi Teknologi Rekayasa Informatika, JavaScript telah menjadi bagian dari materi pembelajaran pemrograman web. Namun, tingkat pemahaman mahasiswa terhadap penggunaan JavaScript dapat berbeda-beda, dipengaruhi oleh pengalaman belajar, metode pengajaran, intensitas praktik, serta keterlibatan mahasiswa dalam proyek pemrograman. Sejalan dengan hal tersebut, beberapa penelitian terdahulu dalam rentang lima tahun terakhir telah mengkaji tantangan dan tingkat pemahaman dalam pembelajaran pemrograman. Penelitian sebelumnya telah menggunakan pendekatan klasifikasi algoritma C4.5 untuk mengukur tingkat pemahaman mahasiswa pada mata kuliah bahasa pemrograman guna menemukan pola kesulitan belajar mahasiswa [11]. Pendekatan clustering menggunakan algoritma K-Means juga telah digunakan untuk mengelompokkan tingkat pemahaman mahasiswa terhadap bahasa pemrograman guna membantu institusi merancang strategi pembelajaran yang lebih terarah [12]. Dalam konteks internasional, penelitian mengenai tingkat pemahaman berbasis kerangka kerja edukasi khusus menunjukkan bahwa praktik langsung menggunakan bahasa pemrograman mampu meningkatkan pemahaman konsep abstrak secara signifikan [13]. Dari sisi tinjauan teori yang lebih mendasar. Pemahaman logika informatika menjadi salah satu faktor penting dalam membantu mahasiswa menghadapi kompleksitas bahasa pemrograman tingkat lanjut [14]. Penelitian lain juga menunjukkan bahwa tingkat pemahaman mahasiswa terhadap bahasa pemrograman tertentu masih tergolong rendah sehingga diperlukan pendekatan pembelajaran interaktif berbasis studi kasus [15]. Dari berbagai literatur yang ada, kajian yang secara spesifik mengevaluasi pemahaman JavaScript melalui kacamata pengalaman praktis mahasiswa tingkat menengah masih terbilang minim. Di sinilah letak kebaruan (*novelty*) penelitian ini, yaitu dengan mengerucutkan fokus pada bahasa JavaScript dan mengambil sampel spesifik dari mahasiswa Tingkat 2 Program Studi Teknologi Rekayasa Informatika Industri Jurusan Otomasi Manufaktur dan Mekatronika, Politeknik Manufaktur Bandung. Alih-alih menggunakan algoritma klasifikasi seperti beberapa penelitian terdahulu, kajian ini lebih memilih metode survei deskriptif. Pendekatan ini sengaja digunakan untuk memotret kondisi riil di lapangan—baik dari segi tingkat pemahaman maupun kendala teknis yang benar-benar dihadapi mahasiswa. Pada akhirnya, temuan dari penelitian ini diharapkan dapat menjadi landasan evaluasi yang konkret untuk menyempurnakan kualitas dan strategi pembelajaran ke depannya.

Berdasarkan latar belakang serta celah kajian dari kelima penelitian terdahulu tersebut, penelitian ini dilakukan untuk menganalisis tingkat pemahaman mahasiswa

Teknologi Rekayasa Informatika terhadap penggunaan bahasa pemrograman JavaScript melalui metode survei menggunakan kuesioner.

TINJAUAN PUSTAKA

Bahasa Pemrograman JavaScript

JavaScript merupakan bahasa pemrograman yang digunakan untuk membuat halaman web menjadi interaktif dan dinamis. Bahasa ini pertama kali dikembangkan untuk meningkatkan kemampuan halaman web yang sebelumnya hanya bersifat statis. Dengan JavaScript, pengembang dapat menambahkan berbagai fitur seperti validasi formulir secara langsung, manipulasi elemen halaman web melalui *Document Object Model* (*DOM*), pengolahan interaksi pengguna, serta komunikasi data dengan server tanpa perlu memuat ulang halaman.

Dalam perkembangannya, JavaScript tidak hanya digunakan pada sisi klien (*client-side*), tetapi juga dapat digunakan pada sisi server melalui berbagai teknologi modern. Hal ini menjadikan JavaScript sebagai salah satu bahasa pemrograman yang paling banyak digunakan dalam pengembangan aplikasi web modern. Selain itu, berbagai framework dan library berbasis JavaScript turut mempermudah pengembangan aplikasi berskala besar sehingga pemahaman terhadap JavaScript menjadi kompetensi penting bagi mahasiswa di bidang teknologi informasi.

Pemahaman terhadap JavaScript tidak hanya mencakup pengetahuan sintaks dasar, tetapi juga meliputi pemahaman konsep seperti variabel, fungsi, percabangan, perulangan, manipulasi data, event handling, serta pengelolaan interaksi antara pengguna dan aplikasi. Oleh karena itu, kemampuan mahasiswa dalam memahami konsep-konsep tersebut menjadi indikator penting dalam menilai tingkat penguasaan JavaScript. Uraian mengenai JavaScript pada penelitian ini mengacu pada berbagai literatur pemrograman web dan dokumentasi resmi pengembangan web [1]–[4].

Pemahaman Mahasiswa dalam Pembelajaran Pemrograman

Pemahaman mahasiswa dalam pembelajaran pemrograman dapat diartikan sebagai kemampuan dalam memahami konsep dasar pemrograman, mengimplementasikan algoritma ke dalam kode program, serta menyelesaikan permasalahan menggunakan pendekatan komputasional. Tingkat pemahaman ini tidak hanya dipengaruhi oleh kemampuan individu mahasiswa, tetapi juga oleh metode pembelajaran yang diterapkan, ketersediaan praktik laboratorium, serta pengalaman mahasiswa dalam mengerjakan proyek pemrograman.

Metode pembelajaran yang menekankan praktik langsung, diskusi pemecahan masalah, dan pengerjaan proyek nyata cenderung mampu meningkatkan pemahaman mahasiswa dibandingkan metode pembelajaran yang hanya berfokus pada teori. Mahasiswa yang aktif melakukan latihan pemrograman dan eksperimen kode biasanya memiliki kemampuan lebih baik dalam memahami konsep pemrograman.

Selain itu, keterlibatan mahasiswa dalam proyek pengembangan aplikasi juga dapat meningkatkan kemampuan mereka dalam menerapkan konsep pemrograman ke dalam kasus nyata. Oleh karena itu, evaluasi tingkat pemahaman mahasiswa perlu dilakukan untuk mengetahui sejauh mana proses pembelajaran yang telah berlangsung memberikan dampak terhadap kemampuan pemrograman mahasiswa. Kajian mengenai pembelajaran pemrograman pada penelitian ini mengacu pada beberapa penelitian terdahulu terkait pendidikan pemrograman dan kesulitan mahasiswa dalam memahami konsep pemrograman [7], [8].

METODE PENELITIAN

Penelitian ini menggunakan jenis penelitian kuantitatif deskriptif dengan pendekatan survei. Pendekatan kuantitatif deskriptif dipilih karena penelitian ini bertujuan untuk mengumpulkan, mengolah, dan menganalisis data berupa angka yang merepresentasikan tingkat pemahaman mahasiswa, lalu mendeskripsikan hasilnya secara objektif. Metode survei digunakan sebagai sarana utama untuk mengumpulkan data secara efisien dari responden dalam waktu yang bersamaan.

Populasi dan Sampel

Populasi dan data penelitian yang digunakan dalam penelitian ini adalah mahasiswa aktif Tingkat 2 Program Studi Teknologi Rekayasa Informatika Industri, Jurusan Teknik Otomasi Manufaktur dan Mekatronika, Politeknik Manufaktur Bandung. Pemilihan kelompok mahasiswa pada semester ini dilakukan dengan pertimbangan bahwa mereka telah atau sedang menempuh mata kuliah pemrograman web. Pada tahap ini, mahasiswa diasumsikan sudah memiliki dasar pemrograman dan terbiasa menghadapi studi kasus nyata dalam praktikum atau persiapan proyek akhir, sehingga data yang diperoleh berasal dari responden dengan pengalaman teknis yang relevan.

Teknik Pengumpulan Data

Pengumpulan data primer dilakukan melalui penyebaran kuesioner secara daring menggunakan instrumen *Google Forms*. Kuesioner disebarkan melalui grup komunikasi mahasiswa. Data sekunder diperoleh melalui studi literatur dari jurnal, buku, dan dokumentasi resmi terkait bahasa pemrograman JavaScript untuk mendukung penyusunan instrumen dan analisis hasil.

Instrumen Penelitian

Pengumpulan data menggunakan kuesioner dengan Skala Likert rentang 1 hingga 5 (1: Sangat Tidak Paham/Sangat Tidak Setuju hingga 5: Sangat Paham/Sangat Setuju). Instrumen kuesioner difokuskan langsung pada pengumpulan data dengan indikator berikut:

1. Data Pemahaman Konsep Dasar: Mengukur pemahaman mahasiswa terhadap sintaks dasar JavaScript, penggunaan variabel, pemahaman tipe data, serta kemampuan manipulasi *Document Object Model (DOM)*.
2. Data Penerapan dalam Praktik: Mengukur intensitas dan kemampuan penggunaan JavaScript pada proyek pengembangan berbasis web, keselarasan praktik dengan

siklus pengembangan perangkat lunak yang cepat (*agile*), serta pengujian teknis langsung melalui tes logika sederhana berbasis cuplikan kode (*snippet code*).

3. Data Kendala Pembelajaran: Mengidentifikasi data terkait tingkat kesulitan mahasiswa saat melakukan *debugging* (pencarian kesalahan kode), hambatan dalam memahami alur logika program, serta pengaruh kurangnya intensitas latihan mandiri.

Teknik Analisis Data

Data kuantitatif yang terkumpul dari kuesioner akan diolah menggunakan perangkat lunak pengolah angka (seperti Microsoft Excel atau SPSS). Teknik analisis yang digunakan adalah statistik deskriptif, yaitu menghitung frekuensi dan persentase dari setiap jawaban responden. Hasil persentase tersebut kemudian dikategorikan untuk menarik kesimpulan mengenai sejauh mana tingkat pemahaman dan apa saja kendala utama mahasiswa Teknologi Rekayasa Informatika Industri terhadap penggunaan JavaScript.

HASIL DAN PEMBAHASAN

Hasil Penelitian

Karakteristik Responden

Penelitian ini melibatkan 99 responden yang merupakan mahasiswa aktif Semester 4 Program Studi Teknologi Rekayasa Informatika Industri. Seluruh responden telah mengikuti pembelajaran dasar pemrograman dan memiliki pengalaman dalam penggunaan JavaScript, baik melalui praktikum, tugas perkuliahan, maupun proyek pengembangan antarmuka web. Karakteristik responden dalam penelitian ini dianalisis berdasarkan jenis kelamin dan usia. Distribusi demografis responden disajikan pada Tabel 4.1 dan Tabel 4.2.

Tabel 1. Karakteristik Responden Berdasarkan Jenis Kelamin

Jenis Kelamin	Jumlah	Persentase
Laki-laki	69	70,4%
Perempuan	29	29,6%
Total	98	100%

Tabel 2. Karakteristik Responden Berdasarkan Usia

Usia	Jumlah	Persentase
19 Tahun	28	38%
20 Tahun	45	62%
21 Tahun	24	33%
22 Tahun	1	1%
Total	98	100%

Berdasarkan tabel karakteristik responden, mayoritas responden didominasi oleh mahasiswa laki-laki dengan rentang usia terbanyak berada pada usia 20 tahun. Hal ini menunjukkan bahwa sebagian besar responden berada pada kelompok usia produktif mahasiswa tingkat menengah yang relevan dengan proses pembelajaran dan penggunaan JavaScript dalam pengembangan web.

Tingkat Pemahaman Konsep Dasar JavaScript

Secara umum, tingkat pemahaman mahasiswa terhadap konsep dasar JavaScript berada pada kategori sedang. Berdasarkan Tabel 2, mahasiswa menunjukkan penguasaan yang baik pada materi fundamental, seperti struktur kontrol logika (*mean* 3,37) dan deklarasi variabel (*mean* 3,30). Namun, pemahaman ini cenderung menurun ketika menghadapi materi yang lebih dinamis, seperti manipulasi *Document Object Model* (DOM) yang hanya memperoleh nilai rata-rata 3,10.

Tabel 3. Statistik Deskriptif Tingkat Pemahaman JavaScript

Indikator	Mean	Median	Standar Deviasi	Kategori
Pemahaman var/let/const	3,30	3	1,24	Sedang
Pemahaman tipe data	3,21	3	1,34	Sedang
Struktur kontrol dan loop	3,37	4	1,34	Sedang
Function dan Arrow Function	3,03	3	1,21	Sedang
Manipulasi DOM	3,10	3	1,28	Sedang

Secara umum, mahasiswa telah memiliki pemahaman yang cukup baik terhadap konsep dasar JavaScript, terutama pada aspek sintaks dan logika dasar pemrograman. Akan tetapi, ketika materi mulai memasuki implementasi yang lebih dinamis seperti manipulasi *DOM* dan *asynchronous programming*, tingkat pemahaman mahasiswa cenderung menurun. Hal ini menunjukkan bahwa mahasiswa relatif lebih mudah memahami konsep teoritis dibandingkan implementasi praktis yang membutuhkan integrasi logika dan interaksi antarelemen dalam aplikasi web.

Analisis Penerapan JavaScript dalam Praktik

Dalam hal penerapan pada proyek pengembangan web, kemampuan mahasiswa juga berada pada tingkat sedang. Mahasiswa terlihat cukup mahir dalam memanfaatkan sumber daya yang ada, terbukti dari indikator *reuse code* atau penggunaan *library* pihak ketiga yang mencatat skor rata-rata tertinggi (3,33). Sebaliknya, adaptasi mahasiswa terhadap alur kerja industri perangkat lunak modern masih tergolong rendah. Hal ini terlihat dari rendahnya indikator penerapan metode *Agile* (*mean* 2,96) dan kolaborasi tim (*mean* 3,03).

Tabel 4. Analisis Penerapan JavaScript dalam Praktik

Indikator	Mean	Median	Standar Deviasi	Kategori
Teamwork dalam proyek	3,03	3	1,40	Sedang
Agile dan deadline	2,96	3	1,18	Sedang
Testing kode	3,14	3	1,29	Sedang
Modular programming	3,04	3	1,29	Sedang
Reuse code/library	3,33	4	1,38	Sedang

Secara umum, mahasiswa telah memiliki pengalaman dasar dalam menerapkan JavaScript pada proyek pengembangan berbasis web. Akan tetapi, penerapan praktik

pengembangan perangkat lunak yang lebih terstruktur, seperti *Agile development*, modular *programming*, dan *testing* sistematis, masih belum dilakukan secara optimal. Hal ini menunjukkan bahwa mahasiswa relatif memahami penggunaan JavaScript pada level implementasi sederhana, namun masih memerlukan penguatan dalam aspek pengembangan perangkat lunak kolaboratif dan praktik industri modern.

Analisis Ujian Teknikal JavaScript

Analisis ujian teknikal dilakukan untuk mengukur kemampuan nyata mahasiswa dalam memahami perilaku dan logika kode JavaScript melalui beberapa studi kasus berbentuk cuplikan kode (*snippet code*). Pengujian ini mencakup beberapa materi penting, seperti manipulasi array dan perulangan, *asynchronous programming*, fungsi dan parameter, filter array, serta manipulasi *Document Object Model* (*DOM*).

Berdasarkan hasil analisis, tingkat ketepatan jawaban mahasiswa pada ujian teknikal masih menunjukkan hasil yang relatif bervariasi. Beberapa mahasiswa telah mampu memahami logika dasar JavaScript dengan baik, khususnya pada materi yang berkaitan dengan array dan struktur kontrol sederhana. Namun demikian, pada materi yang lebih kompleks seperti *asynchronous programming* dan manipulasi *DOM*, tingkat kesalahan jawaban masih cukup tinggi.

Hasil ujian teknikal menunjukkan bahwa tingkat ketepatan jawaban mahasiswa masih tergolong rendah pada beberapa materi tertentu. Soal terkait *asynchronous programming* memperoleh tingkat ketepatan paling rendah, yaitu hanya sebesar 30,30%, dengan 69 responden menjawab salah. Selain itu, soal manipulasi *DOM* juga menunjukkan tingkat ketepatan yang rendah, yaitu sebesar 35,35%.

Sebaliknya, materi yang berkaitan dengan analisis *array* dan *loop* memperoleh tingkat ketepatan tertinggi, yaitu sebesar 55,56%, diikuti oleh filter *array* sebesar 52,53%. Temuan ini menunjukkan bahwa mahasiswa relatif lebih memahami konsep dasar logika dan manipulasi data sederhana dibandingkan konsep JavaScript yang melibatkan *asynchronous execution* dan interaksi dinamis dengan elemen web.

Hasil ini mengindikasikan bahwa mahasiswa masih mengalami kesulitan dalam memahami perilaku eksekusi JavaScript modern, khususnya pada aspek *asynchronous programming* dan manipulasi *DOM* yang membutuhkan pemahaman logika program yang lebih kompleks.

Tabel 5. Hasil Ujian Teknikal JavaScript

Soal Teknikal	Jawaban Benar	Jawaban Salah	Ketepatan(%)
Analisis Array dan Loop	55	44	55,56%
Asynchronous Programming	30	69	30,30%
Filter Array	52	47	52,53%
Function dan Parameter	36	63	36,36%
Manipulasi DOM	35	64	35,35%

Pembahasan

Perbandingan Penguasaan Konsep Dasar dan Logika Lanjutan

Berdasarkan hasil analisis data, terlihat adanya pola menarik pada tingkat pemahaman mahasiswa. Secara konseptual, mahasiswa merasa cukup percaya diri dengan pengetahuan dasar mereka, yang dibuktikan dengan nilai rata-rata (*mean*) pada indikator variabel dan struktur kontrol yang mencapai angka 3,30 ke atas. Namun, kepercayaan diri ini tidak sepenuhnya selaras dengan kemampuan teknis saat menghadapi studi kasus nyata.

Fenomena ini paling terlihat pada hasil ujian teknikal, di mana tingkat ketepatan jawaban pada materi *asynchronous programming* hanya menyentuh angka 30,30%, dan manipulasi *DOM* sebesar 35,35%. Kesenjangan ini mengindikasikan bahwa mahasiswa cenderung lebih mudah menyerap materi yang bersifat statis dan teoretis daripada materi yang menuntut pemahaman alur eksekusi program yang dinamis. Dalam konteks pengembangan web modern, kelemahan pada sisi logika lanjutan ini menjadi hambatan besar karena JavaScript saat ini sangat bergantung pada interaksi antarelemen dan pengelolaan data secara *asynchronous*.

Tantangan dalam Praktik

Temuan lain yang menonjol adalah tingginya angka penggunaan kembali kode (*reuse code*) atau pemanfaatan *library* pihak ketiga yang memiliki skor rata-rata tertinggi, yaitu 3,33. Hal ini menunjukkan bahwa mahasiswa memiliki inisiatif yang baik dalam mencari solusi instan untuk menyelesaikan masalah pemrograman. Namun, tingginya angka *reuse code* ini tidak dibarengi dengan pemahaman terhadap standar industri yang lebih luas, seperti metode pengembangan *Agile* yang justru mencatat skor terendah sebesar 2,96.

Kondisi ini menggambarkan profil mahasiswa yang mahir secara pragmatis dalam menyelesaikan tugas individu, namun belum terbiasa bekerja dalam ekosistem pengembangan perangkat lunak profesional yang menuntut struktur, kecepatan iterasi, dan kolaborasi tim yang ketat. Rendahnya kesadaran akan proses *testing* dan penulisan kode yang modular menunjukkan bahwa fokus utama mahasiswa masih terbatas pada "bagaimana program dapat berjalan", bukan pada "bagaimana program dibangun dengan standar kualitas yang baik".

Implikasi terhadap Metode Pembelajaran

Melihat berbagai kendala teknis dan hambatan praktis di atas, hasil penelitian ini memberikan sinyal kuat perlunya penyesuaian dalam metode pengajaran. Tingginya tingkat kesalahan pada proses *debugging* dan penyusunan alur logika yang kompleks merupakan indikasi bahwa jam praktikum di kelas belum cukup untuk membangun jam terbang mahasiswa secara mandiri.

Sebagai langkah perbaikan, disarankan agar kurikulum lebih menekankan pada pemberian studi kasus nyata yang mensimulasikan lingkungan kerja industri. Pendekatan seperti *live-coding* yang interaktif atau sesi bedah kode (*code review*) secara berkelompok dapat menjadi solusi untuk melatih logika komputasional mahasiswa. Dengan demikian, mahasiswa tidak hanya sekadar menghafal sintaks JavaScript, tetapi juga mampu memahami perilaku kode secara mendalam dan siap beradaptasi dengan dinamika pengembangan perangkat lunak di dunia profesional.

KESIMPULAN DAN SARAN

Secara keseluruhan, penelitian ini menyimpulkan bahwa mahasiswa telah memiliki fondasi konsep dasar JavaScript yang cukup kuat, namun masih mengalami kesulitan signifikan saat dihadapkan pada logika pemrograman lanjutan yang dinamis, seperti manipulasi *Document Object Model* (*DOM*) dan *asynchronous programming*. Tantangan teknis tersebut semakin nyata dengan minimnya kebiasaan mahasiswa dalam menerapkan standar industri modern—seperti metode *Agile*, pengujian kode secara sistematis, dan kerja tim kolaboratif—serta tingginya hambatan dalam melakukan *debugging* akibat kurangnya intensitas latihan mandiri. Untuk menjembatani kesenjangan ini, disarankan agar institusi dan pengajar mulai memperbanyak porsi studi kasus berbasis proyek nyata (*real-world project*), sesi *live-coding*, dan bedah kode (*code review*) guna membiasakan mahasiswa dengan alur kerja profesional. Sejalan dengan itu, mahasiswa diharapkan lebih proaktif meningkatkan latihan pemecahan masalah di luar jam perkuliahan, sementara penelitian mendatang dapat difokuskan pada pengujian efektivitas metode pembelajaran yang lebih spesifik untuk mempercepat penguasaan logika pemrograman tingkat lanjut.

DECLARATIONS

Statement on the Use of Artificial Intelligence. Menyatakan apakah AI digunakan dalam artikel tersebut atau tidak.

Author contribution statement. Menjelaskan kontribusi masing-masing penulis terhadap pelaksanaan penelitian dan penyusunan artikel, sesuai dengan konvensi bidang studi terkait.

Funding statement. Mencantumkan sumber pendanaan dan nomor hibah, jika ada, atau menyatakan bahwa tidak ada pendanaan yang diterima.

Conflict of interest. Mengungkapkan ada atau tidaknya konflik kepentingan terkait penerbitan artikel tersebut.

DAFTAR PUSTAKA

[1] D. Flanagan, *JavaScript: The Definitive Guide,* 7th ed. Sebastopol, CA, USA: O'Reilly Media, 2020.
[2] J. Duckett, *JavaScript and jQuery: Interactive Front-End Web Development*. Indianapolis, IN, USA: Wiley, 2014.
[3] Mozilla Developer Network, "JavaScript Guide," *MDN Web Docs*. [Online]. Available: https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide. [Accessed: May 11, 2026].
[4] W3Schools, "JavaScript Tutorial," *W3Schools*. [Online]. Available: https://www.w3schools.com/js/. [Accessed: May 11, 2026].
[5] I. Sommerville, *Software Engineering*, 10th ed. Boston, MA, USA: Pearson, 2016.
[6] R. S. Pressman and B. R. Maxim, *Software Engineering: A Practitioner's Approach*, 9th ed. New York, NY, USA: McGraw-Hill Education, 2020.
[7] A. Robins, J. Rountree, and N. Rountree, "Learning and teaching programming: A review and discussion," *Computer Science Education*, vol. 13, no. 2, pp. 137–172, 2003.

[8] E. Lahtinen, K. Ala-Mutka, and H.-M. Järvinen, "A study of the difficulties of novice programmers," in *Proc. 10th Annu. SIGCSE Conf. Innovation and Technology in Computer Science Education*, 2005, pp. 14–18.

[9] B. Shneiderman, C. Plaisant, M. Cohen, S. Jacobs, N. Elmqvist, and N. Diakopoulos, *Designing the User Interface: Strategies for Effective Human-Computer Interaction,* 6th ed. Boston, MA, USA: Pearson, 2016.

[10] A. Nugroho, *Rekayasa Perangkat Lunak Berorientasi Objek dengan Metode USDP.* Yogyakarta, Indonesia: Andi, 2018.

[11] D. Halimah, M. R. Lubis, and W. Saputra, "Algoritma C4.5 untuk menentukan klasifikasi tingkat pemahaman mahasiswa pada mata kuliah bahasa pemrograman," *Jurnal Teknik Mesin, Industri, Elektro dan Informatika (JTMEI)*, vol. 1, no. 3, pp. 28–36, 2022.

[12] N. Ramadana, Y. Maulita, and H. Sembiring, "Clustering students level of understanding of programming language courses using the K-Means algorithm," *Journal of Artificial Intelligence and Engineering Applications (JAIEA)*, vol. 4, no. 1, pp. 22–30, 2024.

[13] K. Tiengyoo, S. Sotaro, and S. Thaithae, "A study of mathematical understanding levels in set theory based on the APOS framework by using Python programming language for secondary school students," *Eurasia Journal of Mathematics, Science and Technology Education*, vol. 20, no. 2, 2024.

[14] I. W. Y. Pradnyana, I. W. A. Juliantara, I. M. A. W. Putra, W. E. Ariawan, I. N. A. Guna, and I. W. M. Sarjana, "Study literatur pemahaman mahasiswa tentang logika informatika dalam pemrograman," *Jurnal Ilmiah Sains Sosial, Kewirausahaan dan Kebudayaan,* vol. 3, no. 2, pp. 16–22, 2025.

[15] D. M. Sari, "Peningkatan kompetensi mahasiswa dalam pemrograman Swift melalui seminar dan sharing session interaktif," *E-Jurnal Nobel (Nobel Community Service Journal)*, vol. 4, no. 1, 2025.