EDUTECH : Jurnal Inovasi Pendidikan Berbantuan Teknologi
Vol. 6, No. 4, September-November 2026
e-ISSN: 2797-0140 | p-ISSN: 2797-0590
Online Journal System: https://jurnalp4i.com/index.php/edutech

OTOMASI PEMANTAUAN MAHASISWA LATIHAN PADA PLATFORM BEECROWD

Utari Wijayanti[1], Dewi Rachmatin[2], Raihan Dzaky Antariksa Putra[3], Amalia Purwa Safitri[4], Cici Isma Hanifah[5], Nabilah Azahra[6], Syaira Salsabila[7], Anastasia Dwi Febrianty[8], Muhammad Fajri Nizam Afnaya[9], Muhammad Zaky Al Fajri[10]
Universitas Pendidikan Indonesia[1,2,3,4,5,6,7,8,9,10]
e-mail: utari@upi.edu

Diterima: *25/10/2026*; Direvisi: *03/11/2026*; Diterbitkan: *16/11/2026*

ABSTRAK

Salah satu kunci dalam mempelajari pemrograman adalah banyak berlatih. Di pihak lain, dosen memiliki kendala untuk memeriksa seluruh latihan soal pemrograman mahasiswa secara manual, sehingga salah satu alternatif untuk masalah ini adalah menggunakan platform daring yang mendukung sistem online judge seperti Beecrowd, yakni sistem yang dapat menilai apakah program yang dikirimkan oleh pengguna benar atau salah. Platform ini gratis dan dapat mengompilasi sekitar 30 bahasa pemrograman. Selain itu, platform ini juga dapat menampilkan statistik dari soal-soal yang telah diselesaikan pengguna. Meskipun telah memanfaatkan platform judge online, dosen tetap perlu memantau perkembangan mahasiswa agar mereka terus berlatih. Namun, proses pemantauan itu sendiri menyita waktu karena jumlah mahasiswa 42 orang. Oleh karena itu, kami membuat sebuah skrip yang dapat mengumpulkan jumlah soal yang telah diselesaikan oleh mahasiswa secara semi-otomatis, sehingga pekerjaan yang membutuhkan waktu 12 menit dapat diselesaikan hanya dalam dua menit serta memastikan tidak ada data mahasiswa yang terlewat. Penelitian ini bertujuan untuk mendeskripsikan proses pemantauan secara semi-otomatis tersebut di dalam konsol Chrome serta beberapa keuntungan yang dapat diperoleh dari penerapan metode ini di kelas. Teknik ini menggunakan skrip yang berjalan di peramban Chrome untuk mengevaluasi halaman profil yang telah dikumpulkan dan merangkum total jumlah soal yang telah diselesaikan oleh mahasiswa. Metode ini membantu pengajar mengevaluasi sebaran kemampuan mahasiswa, menemukan mahasiswa berprestasi yang berpotensi (*hidden high achievers*), serta membuat pemantauan dapat dilakukan secara sistematis.

Kata Kunci: *Beecrowd, Online Judge, Otomasi Pemantauan*

ABSTRACT

One key in learning programming is training a lot. On the other hand, the lecturer of the programming course may have time limitations in reviewing the practice problems of the students. One alternative to solve this problem uses the online platform that support the online judge whether the program that is submitted by the user is right or wrong like Beecrowd. This platform is free, and it can compile about 30 programming languages. The platform can also show the statistics of the problems that has been done by the user. Nevertheless, the lecturer still has to monitor the progress of the students so that they keep exercising. However, the monitoring progress itself spends time due to the number of students that are about 40. Therefore, we created a script that can collect the number of solved problems by the students semi-automatically, such that the work that needs 12 minutes and done manually can be done in two minutes and we can ensure there is no student's data that is left. This research aims to describe the monitoring process that is semi automatically in Chrome's console and several advantages that can be gotten from applying this method in the class. This technique is using a

Copyright (c) 2026 EDUTECH: Jurnal Inovasi Pendidikan Berbantuan Teknologi
https://doi.org/10.51878/edutech.v6i4.16682

EDUTECH : Jurnal Inovasi Pendidikan Berbantuan Teknologi
Vol. 6, No. 4, September-November 2026
e-ISSN: 2797-0140 | p-ISSN: 2797-0590
Online Journal System: https://jurnalp4i.com/index.php/edutech

script that is running on Chrome's browser to evaluate the collected profiles' pages and summarize the total number of problems solved by the students. This method help the teacher evaluate the distribution of the students capabilities and finding the hidden high achiever students and the monitoring can be done systematically.
Keywords: *Beecrowd, Online Judge, Monitoring Automation*

PENDAHULUAN

Perkembangan teknologi yang semakin pesat mendorong kebutuhan akan sumber daya manusia yang memiliki kemampuan dalam bidang komputasi dan teknologi. Salah satu keterampilan yang berperan dalam mendukung kebutuhan tersebut adalah pemrograman, terutama dalam bidang pendidikan rekayasa. Pemrograman merupakan bagian penting dari pendidikan rekayasa karena melatih kemampuan berpikir komputasional dan pemecahan masalah (Wang et al., 2023). Abstrak, algoritma, dan melakukan debugging, sehingga mereka berlatih secara ekstensif juga sering dialami mahasiswa yang kesulitan memahami konsep tersebut (Wang et al., 2023). Kesulitan tersebut harus diatasi melalui latihan yang ekstensif, sebab pengalaman langsung merupakan jalan bagi mahasiswa untuk benar-benar menguasai keterampilan pemrograman dan membangun pola pikir komputasional (Wang et al., 2023). Namun, Wang et al. (2023) berfokus pada desain urutan soal dan pengalaman pembelajar dan tidak berfokus pada proses monitoring seberapa banyak siswa telah mengerjakan latihan soal.

Selain itu, pemberian umpan balik dan intervensi pada proses belajar mahasiswa menunjukkan adanya dampak pada perilaku belajar mahasiswa (Utamachant et al., 2023). Namun, penelitian Utamachant et al. (2023) berfokus pada automated judge yang menilai kode saat dikumpulkan, dan belum ada penelitian yang berfokus pada pemantauan *submission* dan perkembangan mahasiswa dalam mengerjakan latihan soal. Serta sistem penelitian Utamachant et al. (2023) menuntut dosen membuat kelas, mengunggah soal, atau mengelola submission di dalam platform. Belum ada penelitian berbasis daftar profil Beecrowd yang dapat langsung dipakai dosen tanpa migrasi penuh kelas ke sistem baru untuk memantau perkembangan mahasiswa dalam mengerjakan latihan soal.

Dalam pembelajaran pemrograman, mahasiswa perlu banyak berlatih agar dapat memahami materi yang telah dipelajari. Salah satu cara yang dapat dilakukan adalah menggunakan sistem online judge (OJ), yaitu platform berbasis web yang menyediakan berbagai soal latihan pemrograman untuk membantu mahasiswa menerapkan teori yang telah dipelajari ke dalam penyelesaian masalah (Wang et al., 2023). Messer et al. (2024) menegaskan bahwa penilaian dan umpan balik otomatis dalam pemrograman , termasuk online judge, telah banyak digunakan untuk menilai kebenaran solusi. Selain itu, Alemán (2010) menunjukkan bahwa penggunaan sistem penilaian otomatis dalam kuliah pemrograman dapat meningkatkan minat dan menghasilkan perbedaan nilai yang signifikan dibandingkan dengan kelompok kontrol. Cavalcanti et al. (2021) mendukung argumen bahwa umpan balik otomatis berperan penting dalam lingkungan belajar daring, termasuk pembelajaran pemrograman. Keuning et al (2018) menyatakan bahwa alat umpan balik otomatis untuk pembelajaran pemrograman; pendapat ini mendukung dasar penggunaan *online judge* sebagai sarana latihan dan evaluasi mandiri.

Mušac et al (2026) dalam penelitiannya membahas alat *automatic assessment* untuk menilai tugas coding, dan memberikan dasar *online judge* merupakan salah satu bentuk penilaian otomatis. Selain itu, Mušac et al. juga membahas bahwa alat penilaian otomatis

Copyright (c) 2026 EDUTECH: Jurnal Inovasi Pendidikan Berbantuan Teknologi
https://doi.org/10.51878/edutech.v6i4.16682

EDUTECH : Jurnal Inovasi Pendidikan Berbantuan Teknologi
Vol. 6, No. 4, September-November 2026
e-ISSN: 2797-0140 | p-ISSN: 2797-0590
Online Journal System: https://jurnalp4i.com/index.php/edutech

membantu mengurangi ketergantungan pada penilaian manual oleh dosen, yang relevan dengan permasalahan keterbatasan waktu dosen dalam memeriksa latihan sekitar 40 mahasiswa.

Salah satu platform yang dapat digunakan adalah Beecrowd. Beecrowd merupakan sebuah platform pembelajaran pemrograman yang menyediakan kumpulan soal yang dikelompokkan berdasarkan tingkat kesulitannya serta umpan balik otomatis secara langsung untuk mengevaluasi hasil pembelajaran yang telah dilakukan (Souza et al., 2026). Selain itu, Beecrowd yang dulunya bernama URI Online Judge menurut Tonin et al. (2013) merupakan platform yang dirancang untuk membantu dosen mengelola latihan dan aktivitas pemrograman mahasiswa. Selain itu, dalam penelitian setahun sebelumnya, Bez et al. (2012) menunjukkan bahwa *online judge* dapat memperkuat pembelajaran algoritma tradisional melalui latihan pemrograman dan evaluasi otomatis.

Meskipun demikian, ketersediaan platform pembelajaran belum sepenuhnya menjamin bahwa aktivitas latihan pemrograman mahasiswa dapat berlangsung secara optimal. Maulana dan Syafrudin (2026) menjelaskan bahwa aktivitas praktikum dalam pembelajaran daring sering kali tidak dilakukan secara optimal sehingga dapat berdampak pada rendahnya keterampilan praktis mahasiswa. Selain itu, pemantauan aktivitas praktikum setiap mahasiswa secara individual masih menjadi tantangan bagi instruktur dalam lingkungan pembelajaran daring (Maulana & Syafrudin, 2026). Oleh sebab itu, dosen kesulitan dalam memantau kemajuan mahasiswa selama mengerjakan latihan dan memastikan apakah mahasiswa memperoleh hasil pembelajaran yang bermakna (Hmelo-Silver, 2004 dalam Yang et al., 2025).

Penelitian terdahulu telah mengembangkan berbagai alat bantu untuk memantau aktivitas latihan mahasiswa dalam menyelesaikan permasalahan pemrograman yang kompleks, seperti dashboard untuk memantau kemajuan latihan pemrograman secara real-time (Yang et al., 2025). Groher et al. (2024) mengembangkan learning analytics dashboard untuk menganalisis hasil tugas dan ujian mahasiswa pada mata kuliah pengantar pemrograman Python yang mempertimbangkan aspek keberagaman gender untuk memberikan pemahaman baru dalam mengevaluasi pembelajaran. Chen et al (2017) menekankan pentingnya dosen memantau kemajuan mahasiswa; dashboard instruktur membantu mengidentifikasi mahasiswa yang membutuhkan bantuan dan memfasilitasi pendampingan sejawat. Selain itu, Kaliisa et al. (2023) menyatakan bahwa dashboard atau ringkasan data aktivitas mahasiswa membantu dosen memantau kemajuan, mengidentifikasi kesenjangan belajar, dan mengambil tindakan intervensi.

Selain itu, Rahmatulloh & Putra (2020) merancang sistem online judge menggunakan Judge0 API untuk mengevaluasi dan mendeteksi hasil pemrograman mahasiswa secara otomatis. Namun, pemanfaatan data aktivitas yang ada pada sistem Beecrowd serta evaluasi pengalaman langsung yang dirasakan oleh mahasiswa masih terbatas. Catatan yang perlu diperhatikan terkait otomasi memantau hasil pemrograman mahasiswa dibahas Mušac et al (2026) Dimana penulis menekankan penilaian otomatis tugas pemrograman diperlukan, tetapi implementasi otomatisasi harus mempertimbangkan keterbatasan platform dan kebijakan penggunaan data.

Azad et al. (2020) juga membahas mengenai mekanisme anti-bot: CAPTCHA, *rate limiting*, dan pemblokiran alamat IP ketika pengunjung diduga sebagai bot. Serupa dengan pembahasan oleh Azad et al., Turk et al. (2020) juga membahas *adversarial scraping*, yakni metode untuk mendeteksi perilaku otomatis, CAPTCHA, serta risiko pemblokiran akun sementara atau permanen. Sehingga, dari artikel Turk et al. ini dapat disimpulkan bawha otomatisasi dapat memicu pembatasan akses. Oleh karena itu, penelitian ini bertujuan untuk

Copyright (c) 2026 EDUTECH: Jurnal Inovasi Pendidikan Berbantuan Teknologi
https://doi.org/10.51878/edutech.v6i4.16682

EDUTECH : Jurnal Inovasi Pendidikan Berbantuan Teknologi
Vol. 6, No. 4, September-November 2026
e-ISSN: 2797-0140 | p-ISSN: 2797-0590
Online Journal System: https://jurnalp4i.com/index.php/edutech

meningkatkan akurasi dosen dengan memanfaatkan otomatisasi pemantauan data aktivitas latihan pemrograman mahasiswa menggunakan Beecrowd, namun dengan tetap memperhatikan batasan-batasan yang ada dari platform Beecrowd untuk mengantisipasi kemungkinan mekanisme anti-bot atau bahkan pemblokiran akun sementara atau permanen. Otomatisasi pemantauan ini berguna untuk membantu dosen agar dapat mengidentifikasi pola latihan mahasiswa secara lebih sistematis.

METODE PENELITIAN

Penelitian ini menerapkan metode kualitatif deskriptif guna menggambarkan fitur otomasi pemantauan aktivitas latihan pemrograman pada platform Beecrowd serta beberapa manfaat pembelajaran di kelas. Data fitur diperoleh melalui eksplorasi langsung pada platform oleh peneliti. Setiap fitur yang berkaitan dengan pemantauan aktivitas latihan didokumentasikan melalui tangkapan layar dan catatan lapangan, kemudian dikelompokkan berdasarkan fungsinya. Teknik pemantauan semi-otomatis berupa skrip peramban yang mengambil jumlah soal yang terselesaikan (*solved*) dari sejumlah profil sekaligus berdasarkan daftar tautan profil, lalu menyimpan hasilnya sebagai berkas CSV berisi ID, jumlah *solved*, dan tautan profil.

Untuk menjaga kerahasiaan identitas mahasiswa di kelas, pendemonstrasian metode ini menggunakan lima top profil publik pada menu *ranking based on country* (diambil dari data negara Indonesia). Namun untuk pengujian pengukuran waktu menggunakan data 42 mahasiswa yang merupakan data pengujian sesungguhnya. Skrip mengambil halaman profil melalui sesi masuk peneliti. Peneliti mengkaji manfaat dan keterbatasan teknik dengan pendekatan deskriptif berdasarkan tiga kriteria: jumlah langkah dibandingkan dengan pencatatan manual, bentuk keluaran, dan jumlah profil yang dapat diproses sekaligus. Manfaat dan keterbatasan teknik dikaji secara deskriptif dengan membandingkan jumlah langkah yang dibutuhkan untuk mencatat jumlah *solved* dari lima profil secara manual, yakni membuka profil, membaca angka, hingga mencatat ke tabel, dengan jumlah langkah menggunakan skrip yang telah dibuat.

HASIL DAN PEMBAHASAN

Hasil

Fitur Pemantauan Bawaan Beecrowd

Eksplorasi pada platform Beecrowd menemukan empat fitur yang berkaitan dengan pemantauan aktivitas latihan, ditunjukkan pada Tabel 1. Halaman profil pengguna menampilkan jumlah submission, jumlah soal terselesaikan (*solved*), dan soal tersulit yang terselesaikan (*hardest solved*). Data ini menjadi sumber bagi teknik pemantauan dalam penelitian ini. Fitur Rank menyediakan beberapa tampilan: seluruh pengguna, universitas, negara, dan perusahaan. Tampilan utama memuat peringkat, nama pengguna, universitas, poin, dan status aktif. Jumlah solved tidak muncul di sana. Peneliti harus membuka profil satu per satu untuk membacanya.

Tabel 1. Fitur Bawaan Beecrowd yang Berkaitan dengan Pemantauan

Fitur	Informasi yang ditunjukkan
Halaman profil	Peringkat, poin, tanggal bergabung, *Submissions*, *Solved*, *Hardest Solved*

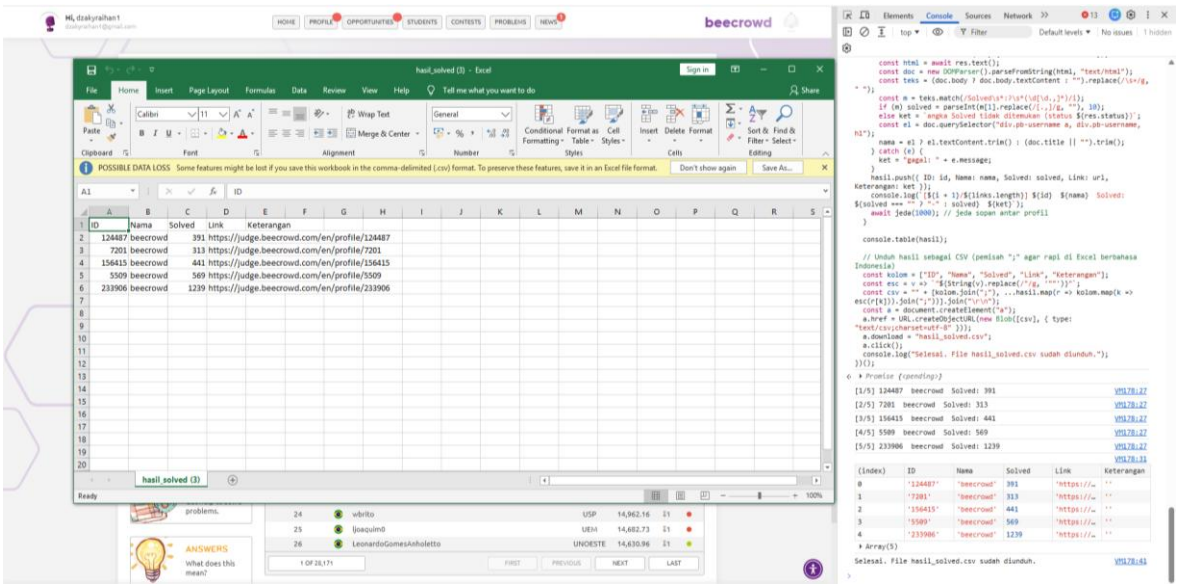
Copyright (c) 2026 EDUTECH: Jurnal Inovasi Pendidikan Berbantuan Teknologi
https://doi.org/10.51878/edutech.v6i4.16682

EDUTECH : Jurnal Inovasi Pendidikan Berbantuan Teknologi
Vol. 6, No. 4, September-November 2026
e-ISSN: 2797-0140 | p-ISSN: 2797-0590
Online Journal System: https://jurnalp4i.com/index.php/edutech

Rank	Daftar pengguna berurutan menurut poin, berisi peringkat, nama pengguna, universitas, poin, dan status aktif

Teknik Pemantauan Semi-Otomatis

Teknik yang dikembangkan oleh tim penulis bekerja dalam tiga tahap. Pertama, tautan profil mahasiswa yang akan dipantau dikumpulkan. Kedua, skrip disalin ke konsol peramban Chrome pada saat halaman judge.Beecrowd.com terbuka, kemudian skrip menampilkan kotak isian tempat seluruh tautan ditempel sekaligus. Skrip menyaring tautan berformat profil dan menghapus tautan ganda. Untuk setiap profil, skrip mengambil halaman profil melalui sesi masuk peneliti, mencari angka *solved* pada teks halaman, dan memberi jeda satu detik sebelum memproses profil berikutnya sebagai bentuk kesopanan terhadap server. Ketiga, skrip menampilkan hasil sebagai tabel pada konsol dan mengunduh berkas CSV bernama hasil_solved.csv yang memuat ID, nama, jumlah *solved*, tautan, dan keterangan apabila angka tidak ditemukan atau pengambilan halaman gagal. Berkas CSV memakai pemisah titik koma agar dapat dibuka langsung di Excel berbahasa Indonesia. Contoh penggunaan skrip dan hasil keluarannya ditunjukkan pada Gambar 1.



Gambar 1. Penggunaan Skrip dan Hasil Keluaran

Tabel 2 merangkum perbandingan jumlah langkah antara pencatatan manual dan penggunaan skrip, dengan asumsi tautan profil sudah terkumpul. Pada pencatatan manual, peneliti membuka tautan, membaca angka solved, dan mencatatnya untuk setiap profil, sehingga jumlah langkah naik seiring jumlah profil. Penggunaan skrip memerlukan empat langkah tetap berapa pun jumlah profilnya: membuka konsol, menempel skrip, menempel seluruh tautan, dan membuka berkas CSV. Skrip lebih ringkas mulai dari dua profil, dan selisih langkahnya melebar pada jumlah profil yang lebih besar. Skrip berjalan minimal sebanyak jumlah profil dalam detik karena ada jeda satu detik antarprofil, dan selama itu pengguna tidak perlu melakukan apa pun.

Copyright (c) 2026 EDUTECH: Jurnal Inovasi Pendidikan Berbantuan Teknologi
https://doi.org/10.51878/edutech.v6i4.16682

EDUTECH : Jurnal Inovasi Pendidikan Berbantuan Teknologi
Vol. 6, No. 4, September-November 2026
e-ISSN: 2797-0140 | p-ISSN: 2797-0590
Online Journal System: https://jurnalp4i.com/index.php/edutech

Tabel 2. Perbandingan Jumlah Langkah Pencatatan Jumlah *Solved*

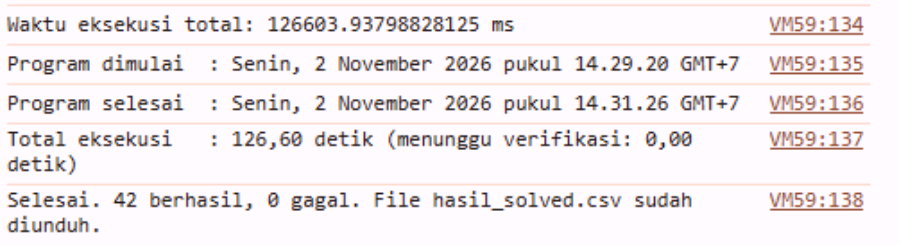
Cara	Langkah per profil	Langkah tetap	Total untuk *n* profil	contoh, *n* = 10
Manual	3	0	3n	30
Skrip	0	4	4	4

Tabel 3. Tabel Validasi Skrip Menggunakan Data 5 Top Profil Publik dari Negara Indonesia

Profil	Angka pada Beecrowd	Hasil Skrip	Kesesuaian
124487	391	391	Sesuai
7201	313	313	Sesuai
156415	441	441	Sesuai
5509	569	569	Sesuai
233906	1239	1239	Sesuai

Teknik ini memiliki tiga keterbatasan. Pertama, skrip hanya mengambil jumlah solved, padahal profil juga menampilkan jumlah submission dan soal tersulit yang terselesaikan. Kedua, keluarannya berupa angka total saat skrip dijalankan dan tidak mencantumkan waktu pengiriman. Ketiga, skrip mencari kata Solved pada teks halaman profil, sehingga hasilnya dapat keliru atau gagal bila tampilan halaman berubah. Pengambilan nama pada uji awal juga belum berfungsi, sehingga kolom nama tidak digunakan.

Pengukuran waktu yang diperlukan untuk mengambil data 42 mahasiswa menggunakan skrip diukur dengan menyelipkan penghitung waktu sebelum proses dan sesudah proses, membutuhkan total waktu 2 menit 6 detik, yakni 126603 ms, dan ditunjukkan pada Gambar 2.



Gambar 2. Waktu eksekusi skrip dalam mengambil data 42 mahasiswa

Di pihak lain, proses pengambilan data secara manual, dengan kondisi sinyal internet lancar tanpa kendala, membutuhkan waktu lebih dari 12 menit 16 detik karena dosen perlu menyalin alamat dari laman profil mahasiswa, menempelkannya pada peramban, dan kemudian mengetikkan data total latihan yang telah dikerjakan oleh mahasiswa ke file Excel secara manual. Total waktu diukur dengan menggunakan *stopwatch* sebagaimana ditunjukkan pada Gambar 3.

Pembahasan

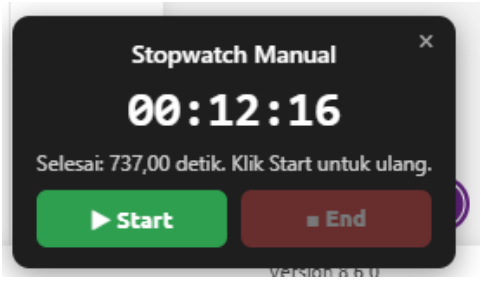
Sistem online judge menilai kode sumber yang dikirim pengguna secara otomatis dalam lingkungan yang seragam (Wasik et al., 2018). Kajian penilaian otomatis dalam pendidikan

Copyright (c) 2026 EDUTECH: Jurnal Inovasi Pendidikan Berbantuan Teknologi
https://doi.org/10.51878/edutech.v6i4.16682

EDUTECH : Jurnal Inovasi Pendidikan Berbantuan Teknologi
Vol. 6, No. 4, September-November 2026
e-ISSN: 2797-0140 | p-ISSN: 2797-0590
Online Journal System: https://jurnalp4i.com/index.php/edutech

ilmu komputer memperhatikan informasi yang disediakan sistem bagi pengajar untuk memahami dan mengoptimalkan pembelajaran (Paiva et al., 2022).

Peneliti sempat mencoba menulis skrip untuk mengekstrak data dari laman profil mahasiswa secara otomatis menggunakan Python dan dipanggil melalui file .bat, namun metode ini mengalami kendala karena sistem Beecrowd mendeteksi bahwa metode ini tidak dilakukan oleh manusia. Kendala dalam menggunakan file .bat ini telah diantisipasi oleh berbagai website (Turk et al., 2020; Azad et al., 2020). Berdasarkan kendala penggunaan bot otomatis ini, penulis mencoba mengembangkan sistem semi-otomatis, di mana peneliti tetap harus secara manual menyalin kode skrip ke konsol peramban serta menyalin daftar laman profil yang perlu dievaluasi.



Gambar 3. Total waktu yang diperlukan untuk memperoleh data secara manual

Pada Beecrowd, informasi per pengguna tersedia di halaman profil, sedangkan tabel peringkat utama tidak menampilkan jumlah solved. Pengajar yang memantau banyak mahasiswa harus membuka profil satu per satu. Tim penulis mengembangkan teknik skrip ini untuk memantau banyak profil sekaligus.

Penelitian tentang dashboard pembelajaran berupaya menentukan data yang bermakna bagi berbagai pemangku kepentingan dan cara menyajikannya, dan sebagian besar studinya masih eksploratif serta berupa proof-of-concept (Schwendimann et al., 2017). Teknik dalam penelitian ini berada pada tahap yang sama: keluarannya berupa berkas CSV tanpa visualisasi, dan datanya hanya jumlah solved. Penelitian ini menunjukkan tiga manfaat prosedur, yaitu pengurangan langkah pencatatan pada banyak profil, keluaran yang terstruktur, dan pemrosesan banyak profil sekaligus. Klaim efisiensi yang lebih luas memerlukan pengukuran pada pengguna sebenarnya.

Pengambilan data otomatis dari halaman web menimbulkan tantangan teknis, hukum, dan etika (Luscombe et al., 2022). Demonstrasi dalam penelitian ini hanya memakai profil yang tercantum pada menu peringkat, dengan identitas disamarkan dan jeda antarpermintaan. Pengguna teknik ini harus memastikan kepatuhan terhadap syarat penggunaan platform. Untuk data mahasiswa, peneliti perlu meminta persetujuan mahasiswa dan menjelaskan data yang dikumpulkan beserta tujuannya.

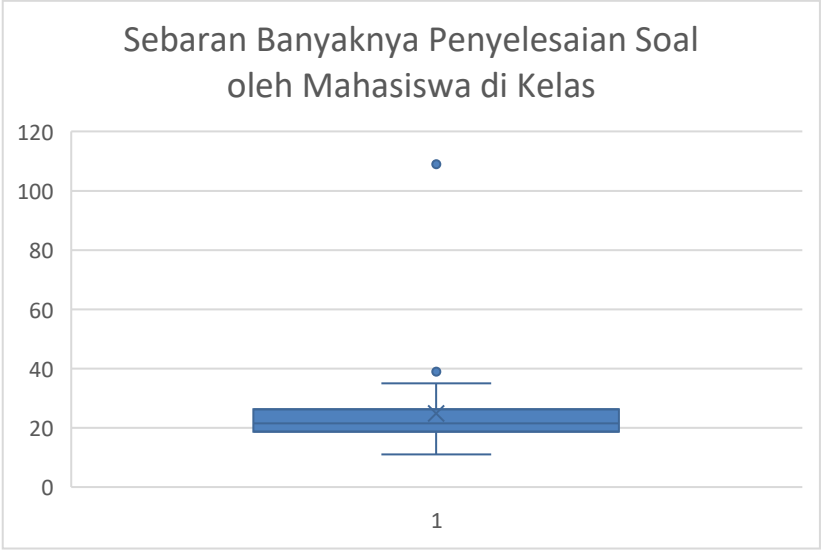
Penelitian ini memiliki beberapa batasan. Peneliti belum menguji teknik pada data mahasiswa dan memakai profil non-mahasiswa dalam demonstrasi. Data hanya berupa jumlah solved pada satu waktu. Peneliti menilai manfaat dari pengamatan sendiri tanpa melibatkan pengguna. Keluaran skrip bergantung pada pola pencarian teks, sehingga perlu dibandingkan dengan halaman profil. Akun peneliti tidak dapat mengakses modul Academic yang ditujukan bagi dosen (Bez et al., 2013), sehingga modul itu tidak dieksplorasi. Penelitian lanjutan dapat menguji teknik ini pada kelas nyata dengan persetujuan mahasiswa, menambahkan data submission dan hardest solved serta visualisasi, dan membandingkannya dengan modul tersebut.

Copyright (c) 2026 EDUTECH: Jurnal Inovasi Pendidikan Berbantuan Teknologi
https://doi.org/10.51878/edutech.v6i4.16682

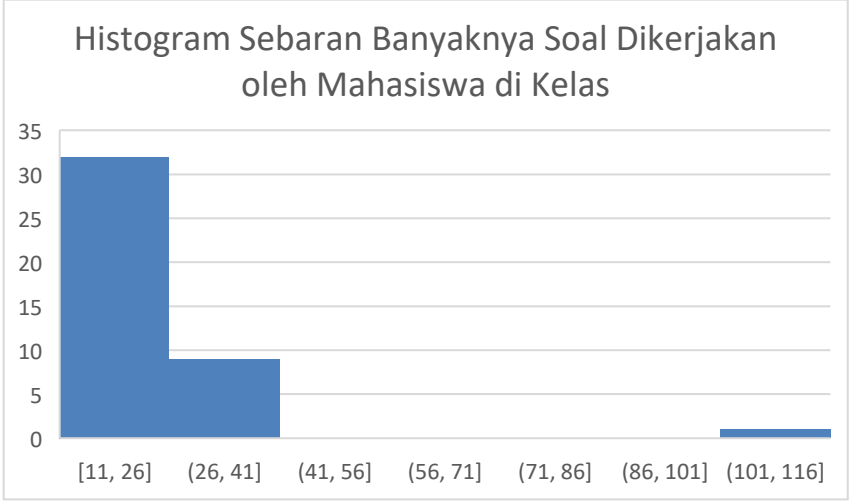
EDUTECH : Jurnal Inovasi Pendidikan Berbantuan Teknologi
Vol. 6, No. 4, September-November 2026
e-ISSN: 2797-0140 | p-ISSN: 2797-0590
Online Journal System: https://jurnalp4i.com/index.php/edutech

Penerapan metode ini di kelas memberi manfaat selain efisiensi waktu, memastikan proses pemantauan sistematis, tidak ada mahasiswa yang terlewat karena saat proses pemantauan dilakukan secara manual tidak jarang dosen memantau beberapa mahasiswa lebih sering dibandingkan mahasiswa-mahasiswa lainnya, serta dosen dapat menemukan *the high achiever students* yang sebelumnya tidak terdeteksi sehingga memungkinkan dosen melakukan proses pendistribusian kelompok belajar secara lebih adil dan merata.

Implementasi di kelas menunjukkan bahwa dosen menemukan ada mahasiswa yang berhasil menyelesaikan hingga 109 soal, sementara rata-rata kelas menyelesaikan 24,93 soal dan hanya terdapat 10 mahasiswa yang menyelesaikan lebih dari 26 soal. Gambar 4 menunjukkan tampilan boxplot sebaran banyaknya soal yang diselesaikan oleh 42 mahasiswa di kelas dan Gambar 5 menunjukkan tampilan histogram sebaran banyaknya soal yang diselesaikan oleh mahasiswa. Berdasarkan data banyaknya soal latihan pada Beecrowd yang telah dikerjakan dan dikombinasikan dengan pertimbangan dari nilai UTS, dosen memiliki data untuk menyebar mahasiswa *high achiever* ke dalam kelompok-kelompok berbeda untuk tugas besar presentasi proyek di akhir semester.



Gambar 4. Boxplot Sebaran Banyaknya Penyelesaian Soal oleh Mahasiswa di Kelas



Gambar 5. Histogram Sebaran Banyaknya Soal Dikerjakan oleh Mahasiswa di Kelas

Copyright (c) 2026 EDUTECH: Jurnal Inovasi Pendidikan Berbantuan Teknologi
https://doi.org/10.51878/edutech.v6i4.16682

EDUTECH : Jurnal Inovasi Pendidikan Berbantuan Teknologi
Vol. 6, No. 4, September-November 2026
e-ISSN: 2797-0140 | p-ISSN: 2797-0590
Online Journal System: https://jurnalp4i.com/index.php/edutech

KESIMPULAN

Penelitian ini menunjukkan bahwa metode pemantauan semi-otomatis pada platform Beecrowd dapat mempermudah dosen dalam memantau aktivitas latihan pemrograman mahasiswa. Dengan menggunakan skrip pada konsol Chrome, proses pengambilan data jumlah soal yang diselesaikan (*solved*) dapat dilakukan secara otomatis dan hasilnya disajikan dalam tabel yang diunduh dalam format CSV. Hasil pengujian menunjukkan bahwa dengan langkah yang sama, metode skrip memerlukan empat langkah, terlepas dari jumlah profil mahasiswa yang diperiksa. Sebaliknya, metode manual memerlukan 3 langkah untuk setiap profil, sehingga untuk n mahasiswa diperlukan $3n$ langkah. Dengan demikian, metode skrip lebih efisien dibandingkan metode manual karena jumlah langkah yang diperlukan tetap sama seiring dengan meningkatnya jumlah profil mahasiswa.

Meskipun demikian, metode ini masih memiliki beberapa keterbatasan seperti data yang dikumpulkan belum lengkap, nama mahasiswa belum teridentifikasi secara otomatis, dan pengambilan data yang masih bergantung pada struktur halaman pada profil Beecrowd. Oleh karena itu, pengembangan selanjutnya dapat difokuskan pada penambahan jenis data, otomatisasi identifikasi mahasiswa, dan peningkatan ketahanan skrip terhadap perubahan tampilan halaman.

UCAPAN TERIMA KASIH

Penelitian ini merupakan luaran tambahan dari penelitian dengan sumber pendanaan Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Pendidikan Indonesia dengan nomor kontrak 989/UN40.F7/PT.01.01/2026.

DAFTAR PUSTAKA

Alemán, J. L. F. (2010). Automated assessment in a programming tools course. *IEEE transactions on education*, *54*(4), 576-581. https://doi.org/10.1109/TE.2010.2098442

Azad, A. B., Starov, O., Laperdrix, P., & Nikiforakis, N. (2020). Web runner 2049: Evaluating third-party anti-bot services. In *International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment,* 135-159. Springer, Cham. https://doi.org/10.1007/978-3-030-52683-2_7

Bez, J. L., Ferreira, C. E., & Tonin, N. A. (2013). URI Online Judge Academic: A tool for professors. In Proceedings of the International Conference on Advanced Information and Communication Technology for Education (ICAICTE 2013) (pp. 763–766). Atlantis Press. https://www.atlantis-press.com/article/8933.pdf

Cavalcanti, A. P., Barbosa, A., Carvalho, R., Freitas, F., Tsai, Y.-S., Gašević, D., & Mello, R. F. (2021). Automatic feedback in online learning environments: A systematic literature review. *Computers and Education: Artificial Intelligence, 2*, 100027. https://doi.org/10.1016/j.caeai.2021.100027

Groher, I., Vierhauser, M., & Hartl, E. (2024). A Learning Analytics Dashboard for Improved Learning Outcomes and Diversity in Programming Classes. In *CSEDU* (2) (pp. 618–625). https://doi.org/10.5220/0012735000003693

Copyright (c) 2026 EDUTECH: Jurnal Inovasi Pendidikan Berbantuan Teknologi
https://doi.org/10.51878/edutech.v6i4.16682

Keuning, H., Jeuring, J., & Heeren, B. (2018). A systematic literature review of automated feedback generation for programming exercises. *ACM Transactions on Computing Education (TOCE)*, *19*(1), 1-43. https://doi.org/10.1080/15332269.2018.1436422

Livaja Mušac, P., Nakić, J., & Sović Kržić, A. (2026). Automatic assessment tools for grading coding assignments: A systematic literature review. *Applied Sciences, 16*(11), 5658. https://doi.org/10.3390/app16115658

Luscombe, A., Dick, K., & Walby, K. (2022). Algorithmic thinking in the public interest: Navigating technical, legal, and ethical hurdles to web scraping in the social sciences. *Quality & Quantity*, 56(3). https://doi.org/10.1007/s11135-021-01164-0

Maulana, I., & Syafrudin, T. (2026). Evaluasi hasil belajar berbasis logfile pada pembelajaran pemrograman melalui arsitektur e-learning terintegrasi. *Jurnal Pendidikan dan Teknologi Indonesia*, 6(1), 143–155. https://doi.org/10.52436/1.jpti.1466

Messer, M., Brown, N. C. C., Kölling, M., & Shi, M. (2024). Automated grading and feedback tools for programming education: A systematic review. *ACM Transactions on Computing Education, 24*(1), https://dl.acm.org/doi/full/10.1145/3636515

Paiva, J. C., Leal, J. P., & Figueira, Á. (2022). Automated assessment in computer science education: A state-of-the-art review. *ACM Transactions on Computing Education*, 22, 1–40. https://doi.org/10.1145/3513140

Rahmatulloh, R., & Putra, R. E. (2020). Rancang Bangun Sistem Online Judge dan Pendeteksian Plagiarisme Menggunakan Arsitektur Serverless. *JIEET* (*Journal of Information Engineering and Educational Technology*), 4(1), 20–28. https://doi.org/10.26740/jieet.v4n1.p20-28

Schwendimann, B. A., Rodríguez-Triana, M. J., Vozniuk, A., Prieto, L. P., Boroujeni, M. S., Holzer, A., Gillet, D., & Dillenbourg, P. (2017). Perceiving learning at a glance: A systematic literature review of learning dashboard research. *IEEE Transactions on Learning Technologies*, 10(1), 30–41. https://doi.org/10.1109/TLT.2016.2599522

Souza, O. M. D., Boscarioli, C., & Peres, L. M. (2026). Algorithm X: Competitive Programming as a Teaching Methodology for Algorithms and Data Structures in High School. In 7th International Computer Programming Education Conference (ICPEC 2026) (pp. 6-1). Schloss Dagstuhl–Leibniz-Zentrum für Informatik. https://doi.org/10.4230/OASIcs.ICPEC.2026.6

Tonin, N. A., Zanin, F. A., & Bez, J. L. (2012, September). Enhancing traditional algorithms classes using URI online judge. In *2012 International Conference on e-Learning and e-Technologies in Education (ICEEE)* (pp. 110-113). IEEE. https://doi.org/10.1109/ICeLeTE.2012.6333402

Turk, K., Pastrana, S., & Collier, B. (2020). A tight scrape: methodological approaches to cybercrime research data collection in adversarial environments. In *2020 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)* (pp. 428-437). IEEE. https://doi.org/10.1109/EuroSPW51379.2020.00064

Copyright (c) 2026 EDUTECH: Jurnal Inovasi Pendidikan Berbantuan Teknologi
https://doi.org/10.51878/edutech.v6i4.16682

Utamachant, P., Anutariya, C. & Pongnumkul, S. i-Ntervene: applying an evidence-based learning analytics intervention to support computer programming instruction. *Smart Learn. Environ.* **10**, 37 (2023). https://doi.org/10.1186/s40561-023-00257-7

Wang, J., Lin, P., Tang, Z., & Chen, S. (2023). How problem difficulty and order influence programming education outcomes in online judge systems. *Heliyon,* 9, e20947. https://doi.org/10.1016/j.heliyon.2023.e20947

Wasik, S., Antczak, M., Badura, J., Laskowski, A., & Sternal, T. (2018). A survey on online judge systems and their applications. *ACM Computing Surveys,* 51(1), Article 3. https://doi.org/10.1145/3143560

Yang, Y., Zhang, A. G., Oney, S., & Wang, A. Y. (2025, October). SPARK: Real-Time Monitoring of Multi-Faceted Programming Exercises. In 2025 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC) (pp. 81–92). IEEE. https://doi.org/10.1109/VL-HCC65237.2025.00018

Copyright (c) 2026 EDUTECH: Jurnal Inovasi Pendidikan Berbantuan Teknologi
https://doi.org/10.51878/edutech.v6i4.16682