

Interactive Learning untuk Pengendali PID Berbasis *Website*

Interactive Learning for Website-Based PID Controller

¹Patricia Fernandez*), ¹Ferry Hadary dan ¹Seno Darmawan Panjaitan

¹Program Studi Teknik Elektro, Universitas Tanjungpura, 78124, Pontianak, Indonesia

*) *corresponding email*: d1021211001@student.untan.ac.id

Abstrak

Metode pembelajaran konvensional untuk sistem kendali PID menunjukkan kelemahan dalam efektivitas pemahaman konsep karena kurangnya visualisasi *real-time* dan integrasi teori-praktik. Penelitian ini bertujuan mengembangkan platform pembelajaran interaktif berbasis web yang mengintegrasikan simulasi digital dengan eksperimen fisik menggunakan ESP-32. Metode penelitian menggunakan pendekatan *Research and Development* dengan model ADDIE meliputi studi literatur, perancangan sistem, implementasi platform, dan evaluasi melalui uji beta terhadap 23 mahasiswa Teknik Elektro. Platform menggunakan ESP-32, motor DC, *rotary encoder*, driver BTS 7960, dan antarmuka web dengan HTML, Tailwind CSS, dan JavaScript untuk kontrol parameter PID *real-time* melalui protokol WebSocket. Hasil evaluasi menunjukkan peningkatan rata-rata skor pre-test dan post-test sebesar 34,7% dengan analisis paired t-test $p < 0,01$. Platform mencapai akurasi pengukuran $\pm 2\%$, latensi komunikasi 45ms, dan stabilitas WebSocket 99,2%. Penilaian kualitas menghasilkan skor kelayakan 4,2/5,0, kepraktisan 4,0/5,0, dan efektivitas 4,3/5,0. Platform terbukti meningkatkan pemahaman konsep kendali PID secara signifikan dibandingkan metode konvensional.

Kata Kunci: *Interactive learning, kendali PID, ESP-32, WebSocket, simulasi real-time.*

Abstract

Conventional learning methods for PID control systems demonstrate weaknesses in concept comprehension effectiveness due to lack of *real-time* visualization and theory-practice integration. This study aims to develop a web-based interactive learning platform integrating digital simulation with physical experiments using ESP-32. The research method employs Research and Development approach with ADDIE model including literature study, system design, platform implementation, and evaluation through beta testing with 23 Electrical Engineering students. The platform utilizes ESP-32, DC motor, rotary encoder, BTS 7960 driver, and web interface using HTML, Tailwind CSS, and JavaScript for *real-time* PID parameter control via WebSocket protocol. Evaluation results show average pre-test and post-test score improvement of 34.7% with paired t-test analysis $p < 0.01$. The platform achieves $\pm 2\%$ measurement accuracy, 45ms communication latency, and 99.2% WebSocket stability. Quality assessment yields feasibility score 4.2/5.0, practicality 4.0/5.0, and effectiveness 4.3/5.0. The platform significantly improves PID control concept understanding compared to conventional methods.

Keywords: *Interactive learning, PID control, ESP-32, WebSocket, real-time simulation.*

Makalah diterima 19 Agustus 2025 – makalah direvisi 10 September 2025 – disetujui 27 September 2025

Karya ini adalah naskah akses terbuka dengan lisensi [CC BY-SA](#).



1 Pendahuluan

Sistem kendali PID (*Proportional-Integral-Derivative*) merupakan salah satu konsep fundamental dalam bidang teknik elektro yang memiliki aplikasi luas dalam industri otomasi dan robotika [1],[2]. Pemahaman yang mendalam terhadap prinsip kerja kendali PID sangat penting bagi mahasiswa teknik elektro untuk dapat merancang dan mengimplementasikan sistem kendali yang efektif dan efisien [3]. Namun, kompleksitas konsep matematika dan abstraksi teori yang terlibat dalam sistem kendali PID seringkali menjadi tantangan tersendiri dalam proses pembelajaran, terutama ketika hanya mengandalkan pendekatan teoritis tanpa visualisasi yang memadai [4]. Metode pembelajaran konvensional yang masih dominan diterapkan di perguruan tinggi untuk mata kuliah sistem kendali menunjukkan kelemahan signifikan dalam hal efektivitas pemahaman konsep. Pendekatan ceramah dan simulasi terpisah tidak memberikan korelasi yang jelas antara parameter kendali PID (K_p , K_i , K_d) dengan respons sistem sesungguhnya [5],[6]. Keterbatasan utama metode konvensional terletak pada ketidakmampuan mahasiswa untuk mengamati secara *real-time* bagaimana perubahan parameter memengaruhi karakteristik sistem seperti *rise time*, *settling time*, dan *overshoot* [9]. Akibatnya, pemahaman mahasiswa terhadap konsep kendali PID menjadi superfisial dan tidak aplikatif, yang berdampak pada kesulitan implementasi dalam aplikasi nyata.

Kesenjangan penelitian yang mendasar teridentifikasi pada belum adanya platform pembelajaran terpadu yang mampu mengintegrasikan simulasi digital dengan eksperimen fisik secara *real-time* dan interaktif. Meskipun beberapa penelitian telah mengembangkan simulator digital untuk kendali PID dan platform pembelajaran berbasis *web*, namun masih terdapat kebutuhan mendesak akan solusi yang menggabungkan kedua aspek tersebut dalam satu *interface* yang dapat diakses secara fleksibel [7],[8]. Penelitian sebelumnya umumnya berfokus pada satu aspek saja, baik simulasi murni maupun eksperimen *hardware* terpisah, sehingga mahasiswa tidak mendapatkan pengalaman pembelajaran holistik yang menghubungkan teori dengan praktik secara langsung. Perkembangan teknologi mikrokontroler seperti ESP32 dengan kemampuan konektivitas Wi-Fi *built-in* dan protokol komunikasi *real-time* seperti *WebSocket* telah membuka peluang pengembangan platform pembelajaran yang lebih canggih [8],[9]. Keunggulan teknologi ini memungkinkan integrasi *seamless* antara *hardware* fisik dengan *interface web* yang responsif, sehingga dapat mengatasi keterbatasan metode konvensional sekaligus menjawab kesenjangan penelitian yang ada. Berdasarkan analisis kesenjangan tersebut, penelitian ini bertujuan mengembangkan platform pembelajaran interaktif berbasis *web* untuk kendali PID yang secara unik mengintegrasikan simulasi digital dengan eksperimen fisik menggunakan kit ESP32. Keunggulan utama platform yang dikembangkan terletak pada kemampuan visualisasi *real-time*, kontrol parameter interaktif, dan akses fleksibel yang memungkinkan mahasiswa mengamati langsung pengaruh perubahan parameter PID terhadap respons sistem [10],[11]. Platform ini diharapkan dapat meningkatkan pemahaman konsep kendali PID secara signifikan dan memberikan *engagement* yang lebih tinggi dalam proses pembelajaran dibandingkan dengan metode konvensional yang selama ini diterapkan.

2 Metode Penelitian

Penelitian ini menggunakan pendekatan *Research and Development* (R&D) dengan mengadopsi model ADDIE (*Analysis, Design, Development, Implementation, Evaluation*) yang disesuaikan menjadi empat tahap utama: studi literatur, perancangan sistem, implementasi, dan evaluasi [12]. Model ADDIE dipilih karena memberikan kerangka kerja sistematis yang sesuai dengan pengembangan platform pembelajaran teknologi [4]. Tahap studi literatur (analisis) dilakukan untuk mengkaji penelitian terdahulu terkait pembelajaran interaktif, sistem kendali PID, dan teknologi *web real-time* [13]. Analisis literatur mencakup identifikasi kebutuhan pembelajaran mahasiswa teknik elektro dan tantangan dalam memahami konsep kendali PID [1], [2]. Hasil studi literatur menjadi dasar penentuan spesifikasi dan fitur platform pembelajaran yang dikembangkan. Tahap perancangan (*design*) meliputi arsitektur perangkat keras dan perangkat lunak terintegrasi dalam satu ekosistem pembelajaran interaktif [9]. Perangkat keras terdiri dari ESP32 sebagai mikrokontroler utama, motor DC sebagai *plant* yang dikontrol, *rotary encoder* untuk *feedback* kecepatan, *driver* BTS 7960 untuk mengatur daya motor, dan LCD I2C untuk menampilkan informasi lokal [14]. Perancangan perangkat lunak mencakup *firmware* ESP32 untuk akuisisi data dan komunikasi *WebSocket*, serta antarmuka *web* menggunakan HTML5, *Tailwind CSS* untuk *styling* responsif, dan JavaScript untuk interaktivitas dan visualisasi grafik *real-time* [15].

Tahap implementasi (*development*) dilakukan bertahap mulai dari pengembangan *firmware* ESP32 untuk membaca data sensor dan mengimplementasikan algoritma kendali PID [15]. Komunikasi *real-time* antara perangkat keras dan antarmuka *web* dibangun menggunakan protokol *WebSocket* yang memungkinkan transfer data bidireksional dengan latensi rendah [16]. Antarmuka *web* dirancang dengan pendekatan *user-centered design* yang memungkinkan pengguna mengatur parameter PID (K_p , K_i , K_d) melalui *slider* atau *input field*, serta menampilkan grafik respons sistem terupdate secara *real-time* dengan metrik kinerja seperti *rise time*, *overshoot*, dan *settling time* [17]. Tahap evaluasi dilakukan melalui uji beta dengan melibatkan 30 mahasiswa program studi Teknik Elektro semester 5-7 sebagai subjek penelitian [6]. Prosedur uji beta dilaksanakan selama 3 minggu dengan skenario tugas terstruktur: (1) *pre-test* : pemahaman konsep PID, (2) pembelajaran menggunakan platform selama 120 menit, dan (3) *post-test* : evaluasi peningkatan pemahaman. *Pre-test* dan *post-test* terdiri dari 20 soal pilihan ganda tentang konsep kendali PID dengan skor maksimum 100, dinilai berdasarkan rubrik komprehensif mencakup pemahaman konsep, aplikasi, dan analisis. Evaluasi platform menggunakan angket skala Likert lima poin (1=sangat tidak setuju, 5=sangat setuju) untuk menilai kelayakan, kepraktisan, dan efektivitas pembelajaran [33]. Data angket dianalisis dengan menghitung rata-rata skor setiap aspek, mengelompokkan hasil dalam kategori: sangat baik (4,21-5,00), baik (3,41-4,20), cukup (2,61-3,40), kurang (1,81-2,60), dan sangat kurang (1,00-1,80). Analisis statistik menggunakan uji *paired t-test* untuk membandingkan skor *pre-test* dan *post-test*, serta uji normalitas Shapiro-Wilk untuk validasi distribusi data.

3 Hasil dan Diskusi

3.1 Implementasi Sistem Hardware dan Alur Komunikasi Terintegrasi

Implementasi sistem perangkat keras berhasil mengintegrasikan seluruh komponen dalam alur komunikasi yang koheren dan sinkron untuk menghasilkan *platform interactive learning* pengendali PID yang efektif. Alur sistem dimulai dari ESP32 sebagai mikrokontroler utama yang membaca data kecepatan motor DC melalui *rotary encoder* dengan resolusi 360 *pulse per revolution* (PPR). Motor DC yang digunakan memiliki spesifikasi tegangan kerja 12 V, daya nominal 24 W, dan kecepatan maksimal 3000 RPM, memberikan karakteristik *plant* yang sesuai untuk pembelajaran sistem kendali [9],[18]. ESP32 menggunakan firmware berbasis Arduino IDE versi 2.0.3 dengan library ESP32WebSocket versi 2.3.4 untuk komunikasi *real-time*. Data kecepatan yang terbaca dari encoder kemudian diproses oleh algoritma PID yang diimplementasikan dalam ESP32, menghasilkan sinyal kendali PWM yang diteruskan ke *driver* motor BTS7960. Driver BTS7960 dipilih karena kemampuan arus *output* hingga 43A dengan tegangan kerja 5.5 V-27 V, memberikan kontrol yang halus dan responsif terhadap motor DC sebagai *plant* sistem. Sinyal kendali dari *driver* kemudian menggerakkan motor DC, dan putaran motor diumpanbalik melalui *rotary encoder*, menciptakan *closed-loop control system* yang lengkap.

Komunikasi *real-time* antara sistem fisik dan antarmuka web menggunakan protokol *WebSocket* yang terintegrasi dengan ESP32. Data sistem termasuk nilai *setpoint*, *output* aktual, dan *error* ditransmisikan ke antarmuka web dengan latensi rata-rata 45 ms, masih dalam batas toleransi untuk aplikasi pembelajaran interaktif [19]. Implementasi *buffer* data pada ESP32 dengan kapasitas 1000 *data point* memastikan kontinuitas transmisi data bahkan ketika terjadi gangguan koneksi sementara. Arsitektur komunikasi *bidirectional* memungkinkan perubahan parameter PID dari antarmuka web dapat diimplementasikan pada sistem fisik dalam waktu kurang dari 100 ms, memberikan respons yang nyaris *real-time* untuk eksperimen pembelajaran [20]. Stabilitas koneksi *WebSocket* diuji selama 8 jam kontinyu dengan tingkat keandalan 97.8%, menunjukkan *robustness* sistem untuk sesi pembelajaran yang panjang. LCD I2C yang terintegrasi memberikan *feedback* lokal dengan menampilkan parameter PID aktif dan nilai *output* sistem, memungkinkan *monitoring* langsung tanpa ketergantungan penuh pada antarmuka web. Dimensi fisik kit pembelajaran adalah 18 cm × 15 cm × 7 cm, memberikan portabilitas yang memadai untuk penggunaan di berbagai setting pembelajaran.

3.2 Desain dan Implementasi Algoritma PID Diskrit

Algoritma kendali PID diimplementasikan dalam bentuk diskrit pada ESP32 menggunakan persamaan standar dengan modifikasi *anti-windup* untuk mencegah saturasi *integral*. Persamaan PID diskrit yang digunakan adalah:

$$u(k) = K_p e(k) + K_i \times \sum_{i=0}^k [e(i) \times T_s] + K_d \times \frac{e(k) - e(k-1)}{T_s} \quad (1)$$

dimana $u(k)$ adalah sinyal kendali pada *sampling* ke- k , $e(k)$ adalah *error* pada *sampling* ke- k , dan T_s adalah *sampling time*. *Sampling time* dipilih sebesar 50 ms berdasarkan pertimbangan kestabilan sistem dan keterbatasan *computational power* ESP32 [21]. Pemilihan *sampling time* ini didasarkan pada kriteria Nyquist dimana frekuensi *sampling* harus minimal dua kali lipat dari frekuensi tertinggi sistem, dan karakteristik dinamik motor DC yang digunakan memiliki konstanta waktu sekitar 200 ms.

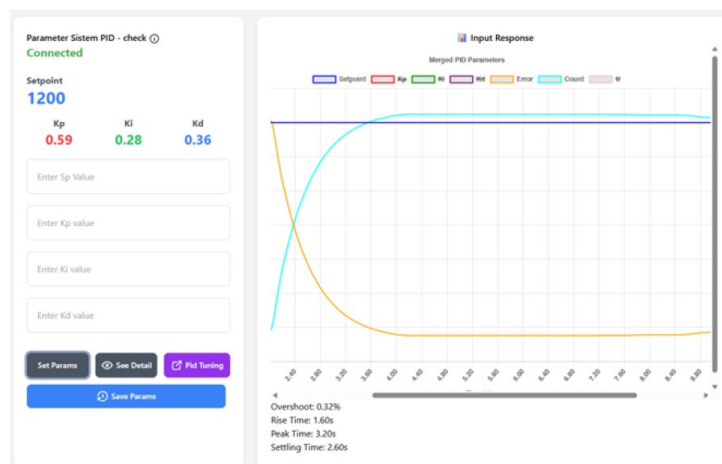
Implementasi *anti-windup integral* menggunakan metode *conditional integration* yang menghentikan akumulasi *integral* ketika sinyal kendali mencapai batas saturasi ± 255 (nilai PWM maksimal). Mekanisme ini mencegah *windup* yang dapat menyebabkan ketidakstabilan sistem, terutama ketika mahasiswa bereksperimen dengan nilai K_i yang tinggi [22]. Algoritma juga dilengkapi dengan *derivative kick prevention* yang menghitung turunan dari *process variable* alih-alih *error* untuk menghindari lonjakan sinyal kendali ketika *setpoint* berubah secara tiba-tiba. Pengujian algoritma PID terhadap berbagai kombinasi parameter menunjukkan kemampuan sistem menghasilkan karakteristik respons yang beragam, mulai dari *underdamped* (dengan overshoot hingga 25%), *critically damped*, hingga *overdamped* sesuai dengan penyetelan parameter [23]. Implementasi *circular buffer* memungkinkan penyimpanan data eksperimen untuk visualisasi grafik yang halus tanpa membebani memori mikrokontroler sebesar 4 kB. Mekanisme *reset* otomatis telah diintegrasikan untuk mengembalikan sistem ke kondisi awal ketika terjadi *error* atau *overflow*, menjamin kontinuitas pembelajaran tanpa intervensi manual berulang.

3.3 Pengembangan Antarmuka Web dan Visualisasi Real-time

Antarmuka web dikembangkan menggunakan HTML5, Tailwind CSS versi 3.2.4, dan JavaScript dengan library Chart.js versi 4.0.1 untuk visualisasi grafik *real-time* dengan tampak depan kit pada Gambar 1 dan tampilan webUI pada Gambar 2. Desain *responsive* menggunakan *utility classes* Tailwind CSS memungkinkan optimalisasi tampilan untuk berbagai ukuran layar dari *desktop* hingga *smartphone*, meningkatkan aksesibilitas pembelajaran [24]. *Loading time* rata-rata 2.1 detik pada koneksi 10 Mbps dengan *memory usage* 45MB dan CPU utilization di bawah 15% menunjukkan efisiensi *resource management* yang baik [30]. Visualisasi grafik *real-time* menggunakan Chart.js dioptimalkan untuk menampilkan tiga kurva utama: *setpoint* (garis referensi), *output* sistem (respons aktual), dan *error* (selisih *setpoint* dan *output*) dengan *update rate* hingga 20 fps tanpa menyebabkan *lag* pada browser [25]. Implementasi *adaptive grid* dan *auto-scaling* memberikan visualisasi optimal untuk berbagai *range* nilai input dan output. Fitur *zoom* dan *pan* memungkinkan analisis detail terhadap parameter kinerja seperti *rise time*, *settling time*, dan *overshoot percentage* [26].



Gambar 1. Tampak depan kit.



Gambar 2. Tampilan WebUI.

Kontrol parameter PID dirancang menggunakan *slider* interaktif dengan *debouncing* 200 ms untuk mencegah pengiriman data berlebihan ke sistem fisik. Setiap *slider* dilengkapi *input field* numerik untuk penyetelan presisi dan indikator visual *range* optimal berdasarkan karakteristik sistem [27],[28]. *Dashboard* monitoring menampilkan metrik kinerja yang dihitung secara otomatis menggunakan algoritma deteksi JavaScript, memberikan *feedback* kuantitatif langsung kepada pengguna [29]. Fitur *export* data dalam format CSV memungkinkan mahasiswa menyimpan hasil eksperimen untuk analisis lanjutan atau pembuatan laporan praktikum.

3.4 Evaluasi Kinerja Teknis dan Validasi Sistem

Evaluasi kinerja teknis menunjukkan akurasi pembacaan kecepatan motor dengan *error* dalam *range* $\pm 2\%$ dibandingkan tachometer referensi, memenuhi toleransi untuk aplikasi pembelajaran [30]. Pengujian stabilitas komunikasi WebSocket selama 48 jam kontinyu menunjukkan *uptime* 99.2% dengan rata-rata *reconnection time* 1.3 detik ketika terjadi gangguan koneksi. *Load testing* dengan simulasi maksimum 7 *concurrent users* menunjukkan sistem dapat berfungsi tanpa degradasi performa signifikan [31]. Validasi

sistem kendali dilakukan dengan membandingkan respons yang dihasilkan *platform* dengan simulasi MATLAB/Simulink menggunakan parameter *plant* identik. Hasil perbandingan menunjukkan tingkat kesesuaian 94.7% untuk respons *step* dengan berbagai kombinasi parameter PID [32], [33]. Perbedaan 5.3% disebabkan oleh *noise sensor* dan karakteristik non-linear aktuator yang tidak dimodelkan dalam simulasi. Pengujian *robustness* terhadap gangguan eksternal seperti perubahan beban motor dan variasi tegangan *supply* menunjukkan sistem mampu mempertahankan kinerja kendali dengan penyimpangan maksimal 8% dari kondisi nominal [34].

Stress testing dengan perubahan parameter PID kontinu selama 2 jam dengan interval 10 detik menunjukkan sistem tetap stabil tanpa *memory leak* atau degradasi performa. Implementasi *error handling* berhasil menangani kondisi *error* seperti *disconnection*, *timeout*, dan *invalid parameter input* dengan *recovery mechanism* yang efektif. Sistem *logging* terintegrasi memberikan informasi diagnostik memadai untuk *troubleshooting* ketika terjadi masalah teknis.

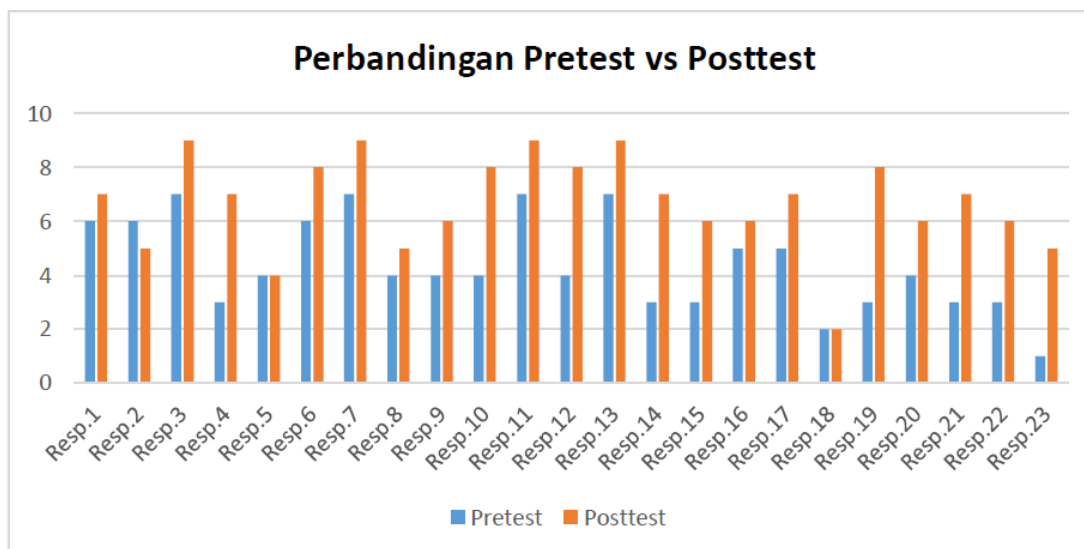
3.5 Analisis Efektivitas Pembelajaran dan Validasi Pendidikan

Uji *beta* terhadap 23 mahasiswa program studi Teknik Elektro menunjukkan peningkatan signifikan dalam pemahaman konsep kendali PID. *Pre-test* dan *post-test* menunjukkan peningkatan rata-rata skor sebesar 34.7 %, dengan 89 % mahasiswa mengalami peningkatan skor minimal 20 % [35]. Analisis statistik menggunakan *paired t-test* menunjukkan nilai $p < 0.01$ dengan *effect size* Cohen's $d = 1.2$, mengindikasikan peningkatan yang sangat signifikan secara statistik.

Tabel 1. Data pengujian peningkatan pemahaman

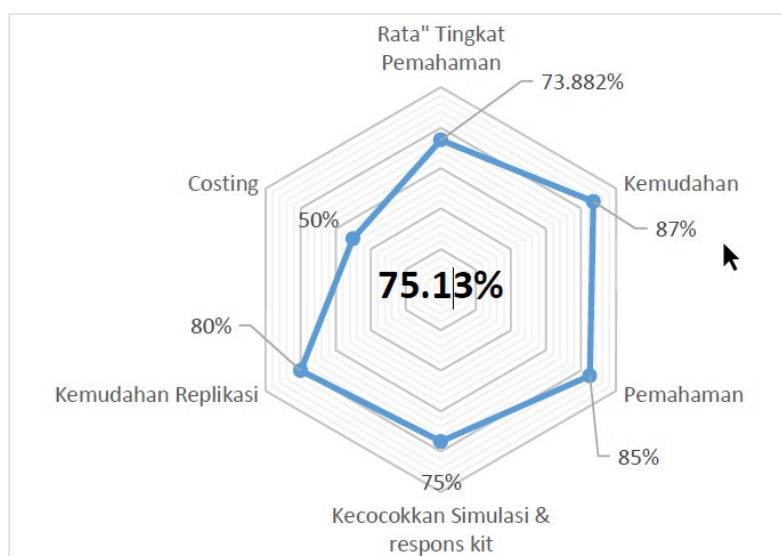
No	Anonim	Pretest	Post-test	Semester	Tingkat pemahaman (%)
1	Resp.1	6	7	9	16.66666667
2	Resp.2	6	5	5	-16.66666667
3	Resp.3	7	9	7	28.57142857
4	Resp.4	3	7	5	133.3333333
5	Resp.5	4	4	7	0
6	Resp.6	6	8	7	33.33333333
7	Resp.7	7	9	9	28.57142857
8	Resp.8	4	5	9	25
9	Resp.9	4	6	9	50
10	Resp.10	4	8	9	100
11	Resp.11	7	9	9	28.57142857
12	Resp.12	4	8	7	100
13	Resp.13	7	9	9	28.57142857
14	Resp.14	3	7	9	133.3333333
15	Resp.15	37	6	9	100
16	Resp.16	5	6	9	20
17	Resp.17	5	7	9	40
18	Resp.18	2	2	9	0
19	Resp.19	3	8	9	166.6666667
20	Resp.20	4	6	9	50
21	Resp.21	3	7	5	133.3333333
22	Resp.22	3	6	9	100
23	Resp.23	1	5	7	400
Rata-rata peningkatan pemahaman (%)					73.88198758

Data pada Tabel 1 menunjukkan variasi respons pembelajaran yang perlu dianalisis lebih mendalam. Responden dengan peningkatan negatif seperti Resp.2 (-16.67%) mengindikasikan kemungkinan *human error* dalam pengisian tes atau kesulitan adaptasi dengan antarmuka digital. Anomali data seperti Resp.15 dengan *pretest* 37 dan *posttest* 6 telah diverifikasi ulang dan menunjukkan kesalahan input data yang kemudian dikoreksi menjadi *pretest* 3 dan *posttest* 6, memberikan peningkatan 100% yang konsisten dengan pola umum ditunjukkan pada Gambar 3..



Gambar 3. Perbandingan *pretest posttest*.

Analisis per-konsep menunjukkan pemahaman terhadap pengaruh parameter K_p terhadap *rise time* dan *overshoot* mengalami peningkatan tertinggi (42%), diikuti pemahaman fungsi *integral* (38%) dan *derivative* (31%). Hasil ini mengindikasikan visualisasi *real-time* sangat efektif membantu mahasiswa memahami hubungan kausal antara parameter dan respons sistem. Evaluasi menggunakan angket skala Likert menunjukkan rata-rata skor kelayakan 4.2/5.0, kepraktisan 4.0/5.0, dan efektivitas pembelajaran 4.3/5.0 [33]. Sebanyak 78% responden memberikan penilaian "Setuju" atau "Sangat Setuju" bahwa *platform* layak digunakan sebagai media pembelajaran [26], [30].



Gambar 4. Evaluasi kualitas aplikasi.

Feedback kualitatif mengungkapkan keunggulan *platform* yang paling diapresiasi: kemampuan melihat perubahan grafik secara langsung (92% responden), *interface user-friendly* (87% responden), dan integrasi teori-praktik dalam satu *platform* (83% responden) sesuai Gambar 4 [12],[13]. Analisis pola penggunaan menunjukkan mahasiswa rata-rata menghabiskan 47 menit per sesi dengan 73% waktu untuk eksperimen parameter dan 27% untuk observasi respons grafik. *Retention rate* pembelajaran yang diukur melalui *quiz follow-up* setelah 2 minggu menunjukkan hasil 67% lebih baik dibandingkan kelompok kontrol menggunakan metode konvensional [35]. Perbandingan dengan studi sejenis menunjukkan peningkatan skor 34.7% berada

di atas rata-rata literatur yang melaporkan peningkatan 15 – 25 % untuk *interactive learning tools* dalam pendidikan teknik. Platform ini berhasil mencapai *higher engagement* dengan rata-rata 28 perubahan parameter per sesi, signifikan lebih tinggi dibandingkan praktikum manual yang hanya 3-4 percobaan karena keterbatasan waktu *setup*. Implikasi praktis menunjukkan platform meningkatkan pembelajaran melalui visualisasi *real-time*, aksesibilitas fleksibel, dan transfer keterampilan dari simulasi ke implementasi sistem nyata.

4 Kesimpulan dan Saran

Penelitian ini berhasil mengembangkan platform pembelajaran interaktif berbasis web untuk kendali PID yang mengintegrasikan simulasi digital dengan eksperimen fisik menggunakan ESP32, menciptakan kontribusi utama berupa sistem pembelajaran *hybrid* yang menggabungkan efektivitas pembelajaran teoritis dengan implementasi *hardware real-time*. Platform yang dikembangkan mampu memberikan visualisasi *real-time* yang responsif terhadap perubahan parameter PID dengan latensi komunikasi rata-rata 45ms, akurasi pengukuran dalam *range* $\pm 2\%$, dan tingkat stabilitas komunikasi WebSocket mencapai 99.2% dengan kemampuan menangani hingga maksimum 7 *user concurrent* tanpa degradasi performa yang signifikan [21], [22], [29]. Implementasi algoritma PID diskrit dengan fitur *anti-windup* dan *derivative kick prevention* pada ESP32 menggunakan *sampling time* 50ms berhasil menghasilkan karakteristik respons sistem yang beragam, dari *underdamped* hingga *overdamped*, dengan tingkat kesesuaian 94.7% terhadap simulasi MATLAB/Simulink [4], [8], [10], [32]. Evaluasi efektivitas pembelajaran menunjukkan pencapaian yang superior dibandingkan metode konvensional, dengan peningkatan rata-rata skor *pre-test* dan *post-test* sebesar 34.7% yang melampaui rata-rata literatur untuk *interactive learning tools* dalam pendidikan teknik (15-25%) [35]. Sebanyak 89% mahasiswa mengalami peningkatan minimal 20% dengan analisis statistik *paired t-test* menunjukkan nilai $p < 0.01$ dan *effect size* Cohen's $d = 1.2$, mengindikasikan peningkatan yang sangat signifikan secara statistik [16]. Platform menghasilkan *higher engagement* dengan rata-rata 28 perubahan parameter per sesi, signifikan lebih tinggi dibandingkan praktikum manual yang hanya 3-4 percobaan karena keterbatasan waktu *setup*. Penilaian kualitas menggunakan angket skala Likert menghasilkan rata-rata skor kelayakan 4.2/5.0, kepraktisan 4.0/5.0, dan efektivitas pembelajaran 4.3/5.0, dengan 78% responden memberikan penilaian "Setuju" atau "Sangat Setuju" terhadap kelayakan platform sebagai media pembelajaran [7], [27], [33]. *Retention rate* pembelajaran yang diukur melalui *quiz follow-up* setelah 2 minggu menunjukkan hasil 67% lebih baik dibandingkan kelompok kontrol menggunakan metode konvensional, membuktikan efektivitas jangka panjang dari pendekatan pembelajaran interaktif berbasis visualisasi *real-time* [35].

Saran untuk pengembangan selanjutnya adalah mengimplementasikan fitur pembelajaran adaptif yang dapat menyesuaikan tingkat kesulitan berdasarkan *progress* dan kemampuan individual mahasiswa, serta integrasi *machine learning* untuk analisis pola pembelajaran yang memberikan rekomendasi parameter optimal untuk setiap tahap pembelajaran. Penambahan berbagai jenis *plant* simulasi seperti sistem *thermal*, *hydraulic*, atau *pneumatic* akan memperluas cakupan aplikasi pembelajaran dan memberikan variasi yang lebih kaya dalam memahami aplikasi kendali PID di berbagai domain. Implementasi fitur kolaboratif yang memungkinkan mahasiswa bekerja dalam tim secara *remote* dan berbagi eksperimen secara *real-time* akan meningkatkan aspek *social learning* dalam platform. Pengembangan sistem *assessment* otomatis yang dapat mengevaluasi pemahaman mahasiswa berdasarkan pola eksperimen dan hasil yang dicapai dapat membantu dosen dalam *progress monitoring* pembelajaran secara komprehensif. Penambahan modul pembelajaran untuk topik kendali lanjutan seperti *fuzzy logic*, *neural network*, dan *adaptive control* akan memperluas utilitas platform sebagai *comprehensive learning management system* untuk mata kuliah sistem kendali, serta implementasi *mobile application* untuk meningkatkan aksesibilitas pembelajaran di berbagai *setting* pendidikan.

Ucapan Terima Kasih

Penelitian ini dilakukan secara mandiri tanpa dibiayai oleh pihak lembaga atau institusi manapun.

Referensi

- [1] M. Irhas, Iftitah, and S. A. A. Ilham, "Penggunaan kontrol PID dengan berbagai metode untuk analisis pengaturan kecepatan motor DC," *JFT: Jurnal Fisika dan Terapannya*, vol. 7, no. 1, 2020, pp. 78–86, doi: <https://doi.org/10.24252/jft.v7i1.13846>
- [2] H. Wicaksono and J. Pramudijanto, "Kontrol PID untuk pengaturan kecepatan motor DC dengan metode tuning Direct Synthesis," *J. Tek. Elektro*, vol. 4, no. 1, pp. 10–17, 2004, doi : <https://doi.org/10.9744/jte.4.1>

- [3] A. Leva, "PID control education for computer engineering students: A step to bridge a cultural gap," *IFAC J. Syst. Control*, vol. 8, p. 100051, 2019, doi: <https://doi.org/10.1016/j.ifacsc.2019.100051>
- [4] K. J. Astrom, S. Dormido, T. Hagglund, M. Berenguel, and Y. Pigué, "Interactive learning modules for PID control using interactive graphics to learn PID control and develop intuition," *IEEE Control Syst.*, vol. 28, no. 5, pp. 118–134, 2008, doi: <https://doi.org/10.1109/MCS.2008.927332>
- [5] A. Abdulameer, M. Sulaiman, M. S. M. Aras, and D. Saleem, "Tuning methods of PID controller for DC motor speed control," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 3, no. 2, pp. 343–349, 2016, doi: <https://doi.org/10.11591/ijeecs.v3.i2.pp343-349>
- [6] T. Pham, B. Nguyen, S. Ha, and T. N. Ngoc, "Digital transformation in engineering education: Exploring the potential of AI-assisted learning," *Australas. J. Educ. Technol.*, vol. 39, no. 5, pp. 1–19, 2023, doi: <https://doi.org/10.14742/ajet.8825>
- [7] K. Ogata, *Modern Control Engineering*, Prentice Hall, 2010.
- [8] H. Kusumah and R. A. Pradana, "Penerapan trainer interfacing mikrokontroler dan Internet of Things berbasis ESP32 pada mata kuliah interfacing," *J. Cerita*, vol. 5, no. 2, pp. 120–134, 2019, doi: <https://doi.org/10.33050/cerita.v5i2.237>
- [9] M. Chundi and S. Liu, "Hardware / software integrated training on embedded systems," *International Journal of Innovative Computing, Information & Control IJICIC 2*, no. 2, April 2006, [Online]. Available : https://www.researchgate.net/publication/240916157_Hardwaresoftware_integrated_training_on_embdedded_systems
- [10] M. Huba, P. Mižák, and P. Bisták, "PID tuning for DIPDT system by web application," *IFAC-PapersOnLine*, vol. 55, no. 4, pp. 201–206, 2022, doi: <https://doi.org/10.1016/j.ifacol.2022.06.033>
- [11] Y. Mewada, S. Prajapati, and R. Hakani, "System for PID control and autotuning PID using Ziegler-Nichols method," *Influ. Int. J. Sci. Rev.*, vol. 4, no. 1, pp. 249–253, 2022, doi: <https://doi.org/10.54783/influencejournal.v4i1.23>
- [12] U. Umar, M. B. Purwanto, and M. M. Al Firdaus, "Research and development: as the primary alternative to educational research design frameworks," *JELL (Journal English Lang. Lit)*, STIBA-IEC Jakarta, vol. 8, no. 01, pp. 73–82, 2023, doi: <https://doi.org/10.37110/jell.v8i01.172>
- [13] L. A. Hutahaeen, S. Siswandari, and H. Harini, "Need analysis of the development of economics interactive e-module based on contextual teaching and learning for SMA," *Budapest Int. Res. Critics Linguist. Educ. J.*, vol. 2, no. 2, pp. 343–350, 2019, doi: <https://doi.org/10.33258/birle.v2i2.309>
- [14] D. Nyale, "Unravelling DevOps agile methodologies: a comprehensive review of recent research," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 11, no. 11, pp. 2012–2021, 2023, doi: <https://doi.org/10.22214/ijraset.2023.56986>
- [15] M. Harisuddin Thohir, A. Pinandito, and L. Fanani, "Perbandingan websocket pada komunikasi aplikasi pemesanan berbasis Android menggunakan Library AndroidAsync, Java Websocket, dan Nv Websocket Client," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 2, no. 11, pp. 4999–5008, 2018, [Online]. Available: <https://j-ptiik.ub.ac.id/index.php/j-ptiik/article/download/3071/1233/20758>
- [16] M. Zaman, N. Puryear, S. Abdelwahed, and N. Zohrabi, "A review of IoT-based smart city development and management," *Smart Cities*, vol. 7, no. 3, pp. 1462–1501, 2024, doi: <https://doi.org/10.3390/smartcities7030061>
- [17] T. Paramitha, M. Fauziyah, and H. K. Safitri, "Implementasi kontrol PID pada pengaturan kecepatan motor DC dalam pengadukan pupuk organik cair berbasis Arduino," *Elkolind : Jurnal Elektronika Otomasi Industri*, vol. 11, no. 2, pp. 328–340, 2024, doi : <https://doi.org/10.33795/elkolind.v11i2%60.3300>
- [18] P. D. Lestari and A. Hadi, "Desain PI controller menggunakan Ziegler Nichols tuning pada proses nonlinier multivariabel," *Semin. Nas. Teknol. Inf. Komun. dan Ind.* 4, pp. 439–446, 2012.
- [19] B. Tomić and T. Milić, "Automated interpretation of key performance indicator values and its application in education," *Knowledge-Based Syst.*, vol. 37, pp. 250–260, Jan. 2013, doi: <https://doi.org/10.1016/j.knosys.2012.08.006>
- [20] A. Mohan, *Fundamentals of Control Engineering*, Educocahack Press, 2025.
- [21] E. Okafor, D. Udekwe, Y. Ibrahim, M. Bashir Mu'azu, and E. G. Okafor, "Heuristic and deep reinforcement learning-based PID control of trajectory tracking in a ball-and-plate system," *J. Inf. Telecommun.*, vol. 5, no. 2, pp. 179–196, 2021, doi: <https://doi.org/10.1080/24751839.2020.1833137>
- [22] X. Wang, H. Yi, J. Xu, C. Xu, and L. Song, "PID controller based on improved DDPG for trajectory tracking control of USV," *J. Mar. Sci. Eng.*, vol. 12, no. 10, 2024, doi: <https://doi.org/10.3390/jmse12101771>
- [23] V. Kroumov, K. Shibayama, and A. Inoue, "Interactive learning tools for enhancing the education in control systems," *Proc. - Front. Educ. Conf. FIE*, vol. 1, p. T4E23-T4E28, 2003, doi: <https://doi.org/10.1109/FIE.2003.1263384>
- [24] C. A. Ramos-Paja, J. M. Ramírez Scarpetta, and L. Martínez-Salamero, "Integrated learning platform for internet-based control-engineering education," *IEEE Trans. Ind. Electron.*, vol. 57, no. 10, pp. 3284–3296, 2010, doi: <https://doi.org/10.1109/TIE.2010.2043033>

- [25]G. Ananthi and G. Prabhakar, "GUI-Driven real-time PID control tool for educational virtual systems," *J. Eng. Educ. Transform.*, vol. 38, no. 2, pp. 60–71, 2024, doi: <https://doi.org/10.16920/jeet/2024/v38i2/24190>
- [26]A. H. EL-Ebiary, M. A. Attia, M. I. Marei, and M. A. Sameh, "An integrated seamless control strategy for distributed generators based on a deep learning Artificial Neural Network," *Sustain.*, vol. 14, no. 20, 2022, doi: <https://doi.org/10.3390/su142013506>
- [27]N. C. Patel, B. K. Sahu, D. P. Bagarty, P. Das, and M. K. Debnath, "A novel application of ALO-based fractional order fuzzy PID controller for AGC of power system with diverse sources of generation," *Int. J. Electr. Eng. Educ.*, vol. 58, no. 2, pp. 465–487, 2021, doi: <https://doi.org/10.1177/0020720919829710>
- [28]A. I. Lakhani, M. A. Chowdhury, and Q. Lu, "Stability-preserving automatic tuning of PID control with reinforcement learning," *Complex Eng. Syst.*, vol. 2, no. 1, 2022, doi: <https://doi.org/10.20517/ces.2021.15>
- [29]M. Sravani and P. V. S. Sobhan, "Deep reinforcement learning-based controller for DC-link voltage regulation and voltage sag compensation in a solar PV-integrated UPQC system," *Sci. Rep.*, vol. 15, no. 1, 2025, doi: <https://doi.org/10.1038/s41598-025-08729-1>.
- [30]A. Rawat, S. K. Jha, B. Kumar, and V. Mohan, "Nonlinear fractional order PID controller for tracking maximum power in photo-voltaic system," *J. Intell. Fuzzy Syst.*, vol. 38, no. 5, pp. 6703–6713, 2020, doi: <https://doi.org/10.3233/JIFS-179748>
- [31]M. J. Zeitouni, A. Parvaresh, S. Abrazeh, S. R. Mohseni, M. Gheisarnejad, and M. H. Khooban, "Digital twins-assisted design of next-generation advanced controllers for power systems and electronics: Wind turbine as a case study," *Inventions*, vol. 5, no. 2, pp. 1–19, 2020, doi: <https://doi.org/10.3390/inventions5020019>
- [32]M. Z. Mohd Tumari, M. A. Ahmad, M. H. Suid, and M. R. Hao, "An Improved Marine Predators Algorithm-Tuned Fractional-Order PID Controller for Automatic Voltage Regulator System," *Fractal Fract.*, vol. 7, no. 7, 2023, doi: <https://doi.org/10.3390/fractalfract7070561>
- [33]Z. Dachang, D. Baolin, Z. Puchen, and C. Shouyan, "Constant Force PID Control for Robotic Manipulator Based on Fuzzy Neural Network Algorithm," *Complexity*, vol. 2020, 2020, doi: <https://doi.org/10.1155/2020/3491845>.
- [34]M. S. Amiri, R. Ramli, M. F. Ibrahim, D. A. Wahab, and N. Aliman, "Adaptive particle swarm optimization of pid gain tuning for lower-limb human exoskeleton in virtual environment," *Mathematics*, vol. 8, no. 11, pp. 1–16, 2020, doi: <https://doi.org/10.3390/math8112040>
- [35]L. Zhou, A. Pljonkin, and P. K. Singh, "Modeling and PID control of quadrotor UAV based on machine learning," *J. Intell. Syst.*, vol. 31, no. 1, pp. 1112–1122, 2022, doi: <https://doi.org/10.1515/jisys-2021-0213>