

Implementasi Automation Testing End-to-End Menggunakan Cypress pada Web E-Conference Parisada serta Evaluasi Efisiensi Pengujian

Raditya Aria Ramadhani^{1*}, Eri Haryanto², Ryan Ari Setyawan³

^{1,2,3}Program Studi Informatika, Universitas Janabadra

*ariaramadhani72@gmail.com

Abstrak

Pengujian manual pada aplikasi interaktif seperti web e-conference sering menghadapi kendala berupa kesalahan manusia (human error) dan durasi eksekusi yang lama, terutama pada proses regresi yang dilakukan secara berulang. Penelitian ini bertujuan untuk mengimplementasikan automated end-to-end (E2E) testing menggunakan framework Cypress serta mengevaluasi tingkat efisiensi waktu pengujian pada platform Parisada. Metodologi penelitian yang digunakan mencakup perancangan skenario fungsionalitas utama yang meliputi proses autentikasi, navigasi sistem, pembuatan ruang pertemuan, hingga terminasi sesi komunikasi. Pengujian dilakukan melalui tiga kali percobaan (P1, P2, P3) untuk mensimulasikan berbagai variabel penghambat, seperti latensi jaringan yang tidak stabil dan tingkat familiaritas pengguna terhadap antarmuka aplikasi. Hasil penelitian menunjukkan bahwa seluruh skenario pengujian otomatis berhasil mencapai tingkat keberhasilan (pass rate) sebesar 100%. Dari aspek efisiensi, penggunaan Cypress secara signifikan mampu mereduksi durasi eksekusi dari rata-rata 60,0 detik pada pengujian manual menjadi hanya 23,67 detik pada pengujian otomatis. Analisis data menunjukkan adanya peningkatan efisiensi waktu rata-rata sebesar 60,55%. Pengujian otomatis juga terbukti lebih konsisten dan andal dalam menghadapi fluktuasi teknis dibandingkan pengujian manual. Kesimpulannya, transisi ke automation testing menggunakan Cypress terbukti sangat efektif dalam mempercepat siklus quality assurance (QA) dan menjaga stabilitas fungsionalitas perangkat lunak pada aplikasi Parisada secara berkelanjutan.

Kata kunci : Automation Testing, Cypress, Efisiensi Pengujian, End-to-End Testing

Abstract

Manual testing on interactive applications such as web e-conferences often faces constraints in the form of human error and long execution durations, especially in repetitive regression processes. This research aims to implement automated end-to-end (E2E) testing using the Cypress framework and evaluate the level of testing time efficiency on the Parisada platform. The research methodology includes designing core functionality scenarios covering the authentication process, system navigation, meeting room creation, and communication session termination. Testing was conducted through three trials (P1, P2, P3) to simulate various inhibiting variables, such as unstable network latency and user familiarity levels with the application interface. The results showed that all automated testing scenarios successfully achieved a 100% pass rate. In terms of efficiency, the use of Cypress significantly reduced the execution duration from an average of 60.0 seconds in manual testing to only 23.67 seconds in automated testing. Data analysis indicates an average time efficiency increase of 60.55%. Automated testing also proved to be more consistent and reliable in dealing with technical fluctuations compared to manual testing. In conclusion, the transition to automation testing using Cypress has proven highly effective in accelerating the quality assurance (QA) cycle and maintaining continuous software functionality stability on the Parisada application.

Keywords : Automation Testing, Cypress, E-Conference, Efficiency, End-to-End Testing

1. Pendahuluan

Perkembangan teknologi informasi global saat ini mendorong peningkatan kebutuhan akan platform

komunikasi digital yang stabil, terutama pada kategori aplikasi web e-conference yang melayani interaksi pengguna secara real-time. Di

Indonesia, kebutuhan akan sistem komunikasi yang andal menjadi isu nasional seiring dengan percepatan transformasi digital di berbagai sektor pelayanan publik^[1]. Sebagai platform yang krusial, sistem seperti Parisada memerlukan kepastian fungsionalitas pada alur kritis seperti autentikasi, pembuatan ruang pertemuan, dan akses rapat. Kegagalan pada alur utama ini tidak hanya menghambat produktivitas pengguna secara langsung, tetapi juga dapat menurunkan kepercayaan terhadap integritas sistem informasi^[2].

Secara teknis, aplikasi web *e-conference* memiliki tingkat kompleksitas yang lebih tinggi dibandingkan aplikasi web konvensional karena tidak hanya memuat proses navigasi halaman dan pengelolaan data, tetapi juga melibatkan interaksi real-time, perubahan status antarmuka secara asinkron, sinkronisasi sesi pengguna, serta ketergantungan terhadap kestabilan koneksi jaringan. Pada aplikasi *e-conference*, fitur seperti pembuatan ruang pertemuan, proses masuk ke ruang rapat, aktivasi perangkat kamera atau mikrofon, serta terminasi sesi komunikasi harus berjalan secara berurutan dan responsif. Kondisi tersebut menyebabkan pengujian tidak cukup hanya memastikan halaman dapat diakses, tetapi juga harus memvalidasi kesinambungan alur pengguna dari awal hingga akhir dalam berbagai kondisi teknis. Oleh karena itu, pengujian end-to-end menjadi penting untuk memastikan bahwa

seluruh komponen utama pada aplikasi *e-conference* dapat berjalan stabil, konsisten, dan sesuai dengan kebutuhan pengguna.

Menanggapi tantangan tersebut, penjaminan kualitas perangkat lunak melalui aktivitas pengujian menjadi aspek yang sangat fundamental. Pendapat pakar dalam tren pengujian web terbaru menekankan bahwa penggunaan alat pengujian otomatis kini menjadi keharusan di industri untuk menghadapi kompleksitas aplikasi modern^[3]. Selama ini, proses validasi fitur pada platform Parisada masih sangat bergantung pada metode manual. Namun, ketergantungan pada metode manual sering kali menjadi hambatan dalam mencapai efisiensi pengiriman produk karena durasi eksekusi yang lama dan tingginya potensi kesalahan manusia (*human error*)^[4]. Penelitian terdahulu telah membuktikan bahwa transisi dari metode manual ke otomatis mampu secara signifikan menekan biaya operasional dan waktu pengujian repetitif^[5]. Berbagai penelitian sebelumnya telah mengevaluasi penggunaan alat pengujian otomatis pada beragam sektor, mulai dari sistem *Enterprise Resource Planning* (ERP)^[6], Platform *E-Commerce*^[7], sistem informasi sumber daya manusia^[4], hingga portal lowongan kerja^[8]. Namun, terdapat celah penelitian (*gap*) di mana sebagian besar studi tersebut berfokus pada aplikasi dengan karakteristik antarmuka yang cenderung statis. Belum ditemukan penelitian

yang secara khusus mengevaluasi efisiensi pengujian otomatis pada aplikasi *web e-conference* yang memiliki tingkat dinamika elemen asinkron dan latensi interaksi *real-time* yang tinggi.

Kebaruan dalam penelitian ini terletak pada penerapan *automation testing* dengan pendekatan *end-to-end* (E2E) menggunakan *framework* Cypress. Berbeda dengan alat pengujian konvensional, Cypress menawarkan kebaruan melalui arsitektur yang berjalan langsung di dalam peramban, memungkinkan akses penuh terhadap elemen aplikasi secara asinkron tanpa masalah sinkronisasi^{[9],[10]}. Hal ini didukung oleh teori pengujian *black-box* modern yang menyatakan bahwa skrip otomasi yang stabil mampu meningkatkan kualitas aplikasi secara berkelanjutan^[11].

Penelitian ini berfokus pada implementasi skrip otomasi menggunakan Cypress untuk memvalidasi fitur utama pada platform Parisada, yaitu *Login*, *Create Room*, dan *Join Room Meeting*. Selain aspek teknis, penelitian ini secara khusus melakukan evaluasi matematis terhadap efisiensi waktu yang dihasilkan melalui komparasi durasi waktu eksekusi antara pengujian manual dan otomatis^[12]. Dengan demikian, penelitian ini diarahkan untuk menjawab kebutuhan pengujian otomatis pada aplikasi *e-conference* yang memiliki karakteristik interaksi *real-time*, elemen antarmuka asinkron, serta ketergantungan tinggi

terhadap stabilitas alur pengguna. Pendekatan ini diharapkan dapat memberikan kontribusi nyata dalam meningkatkan kualitas sistem informasi dan mendukung kinerja tim pengembangan dalam menjaga stabilitas aplikasi secara berkelanjutan^[13].

2. Tinjauan Pustaka

2.1. Penelitian Terkait

Bagian ini memaparkan sejumlah penelitian relevan dalam kurun waktu lima tahun terakhir yang menjadi rujukan dalam penelitian ini. Penelitian-penelitian tersebut digunakan untuk memperkuat dasar teoritis mengenai pengujian manual, pengujian otomatis, pengujian end-to-end, serta validasi fungsional pada sistem informasi berbasis web.

1 Fauzan dkk. (2023) dalam penelitian berjudul “Perbandingan Pengujian Manual dan Terotomasi pada Software Enterprise Resource Planning” membandingkan pengujian manual dan terotomasi pada perangkat lunak ERP menggunakan Selenium IDE. Hasil penelitian menunjukkan bahwa pengujian terotomasi mampu meningkatkan efisiensi waktu secara signifikan pada sistem dengan tingkat kompleksitas tinggi^[6].

2 Bhanuwa dan Jannah (2025) dalam penelitian berjudul “Analisis Perbandingan Automation Testing Katalon Studio dan Selenium IDE pada Website E-Commerce Menggunakan Metode

End-to-End” menganalisis performa dua alat otomasi, yaitu Katalon Studio dan Selenium IDE, pada situs *e-commerce*. Pengujian dilakukan menggunakan pendekatan *end-to-end* dengan beberapa kali pengulangan untuk memastikan validitas data eksekusi. Hasil penelitian menunjukkan bahwa *automation testing* dapat membantu mempercepat proses validasi pada sistem *e-commerce* [7].

3 Rakly dan Andriyani (2024) dalam penelitian berjudul “Pengujian Perangkat Lunak E-Commerce: Perbandingan Antara Pengujian Manual dan Otomatis dengan Katalon Studio dan Python” mengevaluasi efektivitas pengujian manual dan otomatis pada platform belanja daring. Hasil penelitian menunjukkan bahwa pengujian manual masih memiliki keunggulan dalam fleksibilitas penilaian manusia, tetapi pengujian otomatis lebih efisien untuk pengujian rutin karena mampu mempercepat siklus rilis perangkat lunak[5].

4 Azhari dan Suyatno (2024) dalam penelitian berjudul “Pengujian Otomatis GUI dengan Katalon Studio pada Situs Lowongan Kerja Jobstreet dan Glints” menerapkan Software Testing Life Cycle (STLC) untuk pengujian antarmuka pengguna secara otomatis pada portal lowongan kerja. Hasil penelitian menunjukkan bahwa skrip otomasi efektif dalam mereduksi kesalahan operasional yang disebabkan oleh faktor kelelahan atau ketidaktepatan pengujian[8].

5 Riski dan Renanti (2024) dalam penelitian berjudul “Analisis Perbandingan Manual Testing dan Automation Testing pada Sistem Informasi Human Resource Development” membandingkan efektivitas pengujian manual dan pengujian otomatis pada sistem informasi HRD. Hasil penelitian mempertegas bahwa transisi ke pengujian otomatis mampu memangkas durasi eksekusi secara signifikan dibandingkan metode konvensional[4].

6 Nugroho (2023) dalam penelitian berjudul “Penerapan Sistem Informasi Pendaftaran Peserta Didik Baru Berbasis WEB untuk Peningkatan Mutu dan Jumlah Pendaftar” mengembangkan sistem informasi pendaftaran peserta didik baru berbasis web. Hasil penelitian menunjukkan bahwa sistem berbasis web dapat membantu proses pendaftaran dan pengelolaan data menjadi lebih terstruktur, sehingga relevan sebagai penguat pentingnya validasi fungsional pada aplikasi web[14].

7 Amareta dkk. (2026) dalam penelitian berjudul “Sistem Informasi Klasifikasi Atlet Ju-Jitsu Berbasis Web Menggunakan Metode Agile” mengembangkan sistem informasi klasifikasi atlet berbasis web. Hasil penelitian menunjukkan bahwa pengembangan aplikasi web membutuhkan evaluasi dan validasi agar fungsi utama sistem sesuai dengan kebutuhan pengguna akhir[15].

8 Berdasarkan beberapa penelitian tersebut, dapat disimpulkan bahwa pengujian otomatis telah banyak diterapkan pada berbagai jenis sistem informasi, seperti ERP, *e-commerce*, portal lowongan kerja, sistem HRD, dan aplikasi web lainnya. Namun, sebagian besar penelitian terdahulu masih berfokus pada aplikasi dengan karakteristik antarmuka yang relatif statis. Kebaruan penelitian ini terletak pada penerapan automation testing menggunakan Cypress untuk aplikasi web *e-conference* Parisada yang memiliki dinamika elemen asinkron, interaksi real-time, dan ketergantungan terhadap stabilitas alur komunikasi pengguna

2.2. Landasan Teori

1. Pengujian Perangkat Lunak

Proses sistematis untuk menganalisis entitas perangkat lunak guna mendeteksi perbedaan antara kondisi aktual dengan kondisi yang diharapkan serta mengevaluasi fitur guna menemukan kecacatan (*defect*).

2. Pengujian *End-to-End* (E2E)

Teknik pengujian yang memvalidasi seluruh alur kerja perangkat lunak dari awal hingga akhir, termasuk integrasinya dengan sistem eksternal, guna menyimulasikan skenario perjalanan pengguna di dunia nyata.

3. *Framework* Cypress

Kerangka kerja otomatisasi berbasis JavaScript yang beroperasi langsung di dalam peramban,

memberikan akses penuh ke siklus hidup aplikasi serta memiliki fitur *automatic waiting* yang stabil untuk menangani elemen dinamis tanpa masalah sinkronisasi.

4. *Software Testing Life Cycle* (STLC)

Urutan kegiatan sistematis yang mencakup analisis kebutuhan, perencanaan, desain skenario, implementasi, hingga pelaporan hasil guna menjamin stabilitas fungsionalitas sistem secara terukur.

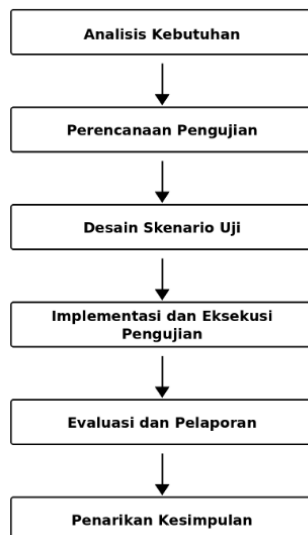
3. Metode Penelitian

3.1. Desain dan Tahapan Penelitian

Penelitian ini menggunakan desain penelitian kuantitatif deskriptif dengan pendekatan eksperimen laboratorium. Pemilihan desain ini didasarkan pada kebutuhan peneliti untuk mengukur metrik performa secara presisi melalui variabel waktu eksekusi dan tingkat keberhasilan skenario uji. Eksperimen dilakukan dengan membandingkan secara langsung efisiensi antara metode pengujian manual dengan pengujian otomatis menggunakan *framework* Cypress.

Tahapan penelitian disusun untuk menggambarkan alur pelaksanaan pengujian otomatis *end-to-end* menggunakan Cypress pada aplikasi web *e-conference* Parisada. Alur penelitian dibuat secara berurutan mulai dari identifikasi kebutuhan pengujian hingga penarikan kesimpulan berdasarkan hasil

evaluasi. Tahapan penelitian yang digunakan ditunjukkan pada Gambar 1.



Gambar 1. Tahapan Penelitian

Tahap pertama adalah analisis kebutuhan, yaitu proses mengidentifikasi fitur utama pada aplikasi Parisada yang memiliki peran penting dalam alur penggunaan sistem. Fitur yang menjadi fokus pengujian meliputi *Login*, *Create Room*, *Join Room Meeting*, dan *End Meet*. Tahap ini dilakukan untuk memastikan bahwa skenario pengujian diarahkan pada fungsi yang paling berpengaruh terhadap stabilitas aplikasi.

Tahap kedua adalah perencanaan pengujian, yaitu menentukan pendekatan pengujian yang digunakan, lingkungan pengujian, perangkat lunak pendukung, serta metrik evaluasi. Pada tahap ini ditetapkan bahwa pengujian dilakukan menggunakan pendekatan *end-to-end* dengan *framework* Cypress, sedangkan metrik yang

diamati meliputi status keberhasilan pengujian dan durasi waktu eksekusi.

Tahap ketiga adalah desain skenario uji, yaitu penyusunan alur pengujian berdasarkan aktivitas pengguna dari awal hingga akhir. Skenario disusun untuk merepresentasikan proses nyata yang dilakukan pengguna, mulai dari autentikasi, pembuatan ruang pertemuan, masuk ke ruang rapat, hingga mengakhiri sesi komunikasi.

Tahap keempat adalah implementasi dan eksekusi pengujian, yaitu menerjemahkan skenario uji ke dalam skrip otomatis berbasis JavaScript menggunakan Cypress. Pengujian kemudian dijalankan untuk memperoleh data hasil eksekusi otomatis. Selain itu, pengujian manual juga dilakukan sebagai pembanding untuk mengukur efisiensi waktu yang dihasilkan oleh metode otomatis.

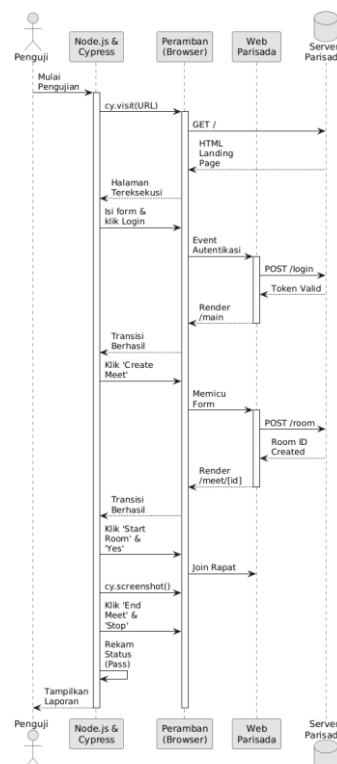
Tahap kelima adalah evaluasi dan pelaporan, yaitu menganalisis hasil pengujian berdasarkan status *pass / fail*, kesesuaian antara hasil aktual dan hasil yang diharapkan, serta perbandingan durasi antara pengujian manual dan otomatis. Hasil evaluasi tersebut digunakan untuk mengetahui tingkat keberhasilan fungsionalitas sistem dan efisiensi pengujian.

Tahap terakhir adalah penarikan kesimpulan, yaitu merumuskan temuan penelitian berdasarkan hasil evaluasi yang telah dilakukan. Kesimpulan digunakan untuk menjelaskan efektivitas implementasi Cypress dalam

mempercepat proses pengujian dan menjaga stabilitas fungsionalitas aplikasi web *e-conference* Parisada.

3.2. Instrumen, Lokasi, dan Arsitektur Pengujian

Instrumen utama dalam penelitian ini adalah perangkat lunak otomasi Cypress versi terbaru yang berjalan di atas lingkungan Node.js. Perangkat keras yang digunakan meliputi komputer dengan spesifikasi yang memadai untuk menjalankan peramban *Google Chrome* secara stabil. Lokasi penelitian dilakukan pada lingkungan *staging* aplikasi web *e-conference* Parisada untuk menjaga integritas data produksi selama proses eksperimen berlangsung. Objek penelitian difokuskan pada empat fitur utama yaitu *Login*, *Create Room*, *Join Room Meeting*, dan *End Meet*. Mekanisme interaksi antara instrumen pengujian dan sistem aplikasi dijelaskan melalui arsitektur pengujian pada Gambar 2 berikut:



Gambar 2 Arsitektur Pengujian Sistem

Alur arsitektur pengujian *End-to-End* pada aplikasi *e-conference* Parisada dimulai dengan eksekusi skrip Cypress melalui lingkungan Node.js di komputer penguji (*client*). Skrip Cypress berjalan secara langsung di dalam peramban yang sama dengan aplikasi Parisada yang sedang diuji, sehingga memungkinkan interaksi *real-time* dengan antarmuka web. Skrip otomasi yang dijalankan akan mensimulasikan tindakan pengguna, dan sistem Cypress secara otomatis melakukan proses *assertion* (validasi) terhadap respons antarmuka server untuk mencocokkan hasil aktual dengan hasil yang diharapkan.

3.3. Teknik Pengumpulan dan Analisis Data

Teknik pengumpulan data dilakukan melalui observasi langsung terhadap jalannya pengujian dan pengambilan log hasil eksekusi dari Cypress *Test Runner*. Untuk pengujian manual, waktu diukur menggunakan alat pengukur waktu (*stopwatch*) sejak aksi pertama dilakukan hingga sistem memberikan respons akhir. Teknik analisis data bersifat komparatif, di mana peneliti membandingkan rata-rata durasi eksekusi dari kedua metode. Efisiensi pengujian diukur menggunakan perhitungan persentase efisiensi relatif untuk menentukan tingkat reduksi beban kerja yang dihasilkan oleh skrip otomasi

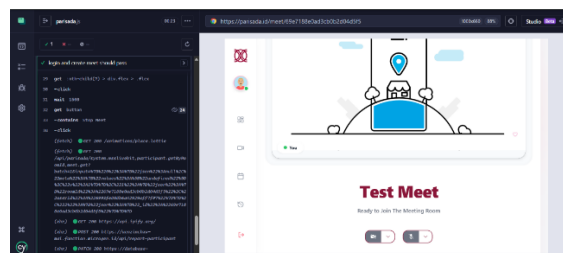
4. Hasil dan Pembahasan

Bagian ini memaparkan hasil dari implementasi skrip pengujian *End-to-End* (E2E) menggunakan *framework* Cypress pada aplikasi web *e-conference* Parisada. Pembahasan difokuskan pada dua metrik utama: hasil validasi fungsionalitas sistem guna memastikan akurasi deteksi kecacatan dan pengukuran perbandingan durasi waktu eksekusi untuk mengevaluasi efisiensi pengujian. Integrasi alat pengujian yang adaptif dalam lingkungan pengembangan modern terbukti krusial untuk menjaga stabilitas sistem secara berkelanjutan.

4.1. Hasil Penelitian

1. Implementasi dan Eksekusi Pengujian Otomatis

Eksekusi skrip otomasi dilakukan pada lingkungan lokal dengan target URL produksi (<https://parisada.id/>). Skenario terintegrasi bernama *"login and create meet should pass"* mencakup seluruh alur kritis dari autentikasi hingga pengakhiran sesi rapat. Implementasi ini menggunakan pendekatan *black-box testing* yang memungkinkan verifikasi fungsionalitas fitur utama tanpa perlu mengakses kode internal aplikasi. Penggunaan Cypress memberikan keunggulan teknis melalui arsitektur yang berjalan langsung di peramban, sehingga interaksi elemen dinamis dapat ditangani dengan tingkat akurasi yang tinggi.



Gambar 3 Hasil Eksekusi Pengujian Otomasi pada Cypress *Test Runner*

2. Hasil Validasi Fungsionalitas Sistem

Fokus utama dari pengujian otomasi adalah memastikan alur kerja perangkat lunak berjalan sesuai spesifikasi kebutuhan guna menghindari kesalahan yang merugikan sistem secara keseluruhan. Validasi dilakukan dengan membandingkan hasil yang diharapkan (*expected result*) dengan hasil aktual (*actual result*) yang ditangkap oleh sistem *assertion*.

Rekapitulasi hasil fungsionalitas E2E disajikan pada Tabel 1.

Tabel 1. Hasil pengujian fungsionalitas E2E

ID Skenario	Fitur yang Diuji	Expected Result	Actual Result	Status
TC-01	Login	Sistem menerima kredensial valid dan mengarahkan pengguna ke halaman /main	Pengguna berhasil login dan antarmuka /main termuat sempurna	Pass
TC-02	Create Meet	Sistem berhasil membuat ID Ruang baru dan mengarahkan ke /meet/[id_ruang]	Sistem merender halaman pre-join dengan nama "Test Meet"	Pass
TC-03	Join Meet	Sistem memunculkan modal konfirmasi dan memasukkan pengguna ke ruang rapat.	Pengguna berhasil masuk ke ruang rapat dan elemen kontrol (kamera/mikrofon) termuat.	Pass
TC-04	End Meet	Sistem memunculkan modal Stop/Leave dan memutuskan sesi rapat saat diklik Stop.	Sesi terputus dengan sukses sesuai perintah aksi otomatis	Pass

Hasil pada Tabel 1 menunjukkan tingkat keberhasilan (*pass rate*) sebesar 100%. Keberhasilan skenario pengujian secara konsisten ini mengonfirmasi performa andal instrumen otomatisasi dalam mendeteksi kesalahan

pada sistem informasi yang memiliki kompleksitas tinggi^[7].

3. Analisis Efisiensi Pengujian

Metrik evaluasi efisiensi diukur melalui komparasi waktu eksekusi antara metode manual dan otomatis. Untuk menjamin validitas data, pengujian dilakukan melalui tiga skenario percobaan (P1, P2, P3) yang menyimulasikan berbagai kondisi interaksi pengguna dan latensi jaringan. Data pengukuran dicatat pada Tabel 2.

Tabel 2. Perbandingan Durasi Pengujian

Percobaan	Pengujian Manual (Detik)	Pengujian Otomatis (Detik)	Efisiensi (%)
P1	64	23	64,06
P2	54	23	57,41
P3	62	25	59,68
Rata-rata	60	23,67	60,55

Berdasarkan Tabel 2, waktu pengujian manual menunjukkan variasi lebih tinggi dibandingkan pengujian otomatis. Pada P1, durasi manual mencapai 64 detik karena menyimulasikan kondisi pengguna baru yang masih membutuhkan waktu untuk mengenali antarmuka. Pada P2, durasi manual menurun menjadi 54 detik karena pengguna telah lebih terbiasa dengan alur sistem. Pada P3, durasi manual kembali meningkat menjadi 62 detik akibat simulasi latensi jaringan. Sebaliknya, pengujian otomatis menggunakan Cypress tetap stabil pada rentang 23–25 detik karena mekanisme *automatic waiting* dan *retryability* memungkinkan skrip menunggu elemen tersedia sebelum menjalankan aksi

berikutnya. Tingkat penghematan waktu dihitung menggunakan rumus persentase efisiensi relatif pada Persamaan (1):

$$E = \frac{T_m - T_a}{T_m} \times 100\% \quad (1)$$

Keterangan:

E = persentase efisiensi waktu pengujian (%)

T_m = waktu eksekusi pengujian manual (detik)

T_a = waktu eksekusi pengujian otomatis (detik)

Melalui perhitungan rata-rata data pada Tabel 2, didapatkan hasil efisiensi sebesar:

$$E = \frac{60,0 - 23,67}{60,0} \times 100\% = 60,55\%$$

Hasil eksperimen ini memberikan bukti empiris bahwa otomasi mampu mereduksi durasi eksekusi secara signifikan dibandingkan metode manual dengan rata-rata penghematan waktu sebesar 60,55%^{[4], [6]}.

Pengujian otomatis terbukti lebih dari 2,5 kali lipat lebih cepat karena menghilangkan ketergantungan pada proses kognitif manusia yang rentan terhadap hambatan fisik dan kelelahan operasional. Selain efisiensi durasi, transisi ke metode otomatis sangat efektif untuk pengujian rutin guna menekan biaya operasional repetitif dalam siklus penjaminan kualitas produk digital. Stabilitas waktu Cypress yang konstan di rentang 23 hingga 25 detik menunjukkan keunggulan fitur *built-in retryability* dalam menjaga kualitas interaksi antarmuka web

meskipun terjadi fluktuasi kondisi teknis dan latensi jaringan^{[5], [8]}.

4.2. Pembahasan

Hasil penelitian menunjukkan bahwa implementasi *automation testing* menggunakan Cypress mampu memvalidasi alur kritis aplikasi web *e-conference* Parisada secara konsisten, mulai dari *login*, pembuatan ruang pertemuan, masuk ke ruang rapat, hingga pengakhiran sesi. Keberhasilan seluruh skenario dengan status *pass* menunjukkan bahwa pendekatan *end-to-end* sesuai digunakan untuk menguji kesinambungan proses pada aplikasi *e-conference*, bukan hanya memeriksa fungsi secara terpisah.

Dari sisi efisiensi, pengujian otomatis mampu menurunkan rata-rata durasi eksekusi dari 60,0 detik menjadi 23,67 detik, dengan efisiensi waktu sebesar 60,55%. Hal ini menunjukkan bahwa Cypress dapat mengurangi ketergantungan terhadap aktivitas manual yang dipengaruhi oleh kecepatan mengetik, ketelitian penguji, serta waktu reaksi terhadap perubahan antarmuka. Stabilitas waktu otomatis pada rentang 23–25 detik juga menunjukkan bahwa mekanisme *automatic waiting* dan *retryability* efektif dalam menangani elemen asinkron pada aplikasi *e-conference*. Jika dibandingkan dengan penelitian terdahulu, hasil ini sejalan dengan temuan bahwa *automation testing* lebih efisien dibandingkan

pengujian manual pada sistem ERP, e-commerce, dan sistem informasi HRD [4]. [5]. [6] . Namun, penelitian ini memiliki penekanan berbeda karena objek yang diuji adalah aplikasi web *e-conference* yang memiliki interaksi *real-time*, elemen antarmuka dinamis, dan ketergantungan terhadap stabilitas sesi komunikasi. Keterbatasan penelitian ini terletak pada cakupan pengujian yang masih berfokus pada aspek fungsional, sehingga penelitian lanjutan disarankan mengintegrasikan *load testing*, *performance testing*, dan *security testing*.

5. Kesimpulan

Implementasi pengujian otomatis menggunakan framework Cypress pada aplikasi web *e-conference* Parisada terbukti mampu memvalidasi seluruh alur kritis pengguna, yaitu *Login*, *Create Room*, *Join Room Meeting*, dan *End Meet*, dengan tingkat keberhasilan 100%. Dari sisi efisiensi, pengujian otomatis menghasilkan rata-rata durasi eksekusi 23,67 detik, lebih cepat dibandingkan pengujian manual sebesar 60,0 detik, sehingga diperoleh efisiensi waktu sebesar 60,55%. Hasil ini menunjukkan bahwa Cypress efektif dalam mempercepat proses pengujian, menjaga konsistensi eksekusi, serta menangani elemen antarmuka asinkron pada aplikasi *e-conference*. Keterbatasan penelitian ini terletak pada cakupan pengujian yang masih berfokus pada aspek fungsional,

sehingga penelitian selanjutnya disarankan mengintegrasikan pengujian non-fungsional seperti *load testing*, *performance testing*, dan *security testing*.

6. Daftar Pustaka

- [1] D. Aprilianti and I. Hermadi, "Implementasi Pengujian Otomatis dan Pengukuran Kualitas Website Employee Self-Service PT Asta Protek Jiarsi," *JOURNAL CERITA: Creative Education of Research in Information Technology and Artificial Informatics*, vol. 8, no. 225, pp. 73–82, 2026.
- [2] B. P. JIFTI and A. S. JIFTI, "Perancangan Automated Testing Pada Studi Kasus Website Indicar," *Jurnal Ilmiah Teknologi Informasi dan Robotika*, vol. 4, no. 1, pp. 29–32, 2022, doi: 10.33005/jifti.v4i1.69.
- [3] I. Kertusha, G. Assress, O. Duman, and A. Arcuri, "A Survey on Web Testing: On the Rise of AI and Applications in Industry," 2025, [Online]. Available: <http://arxiv.org/abs/2503.05378>
- [4] M. Riski and M. D. Renanti, "Analisis Perbandingan Manual Testing Dan Automation Testing Pada Sisitem Informasi Human Resource Development," *Kurawal - Jurnal Teknologi, Informasi dan Industri*, vol. 7, no. 2, pp. 1–11, 2024, doi: 10.33479/kurawal.v7i2.1096.
- [5] B. D. Rakly and W. Andriyani, "Pengujian Perangkat Lunak E-Commerce : Perbandingan Antara Pengujian Manual dan Otomatis dengan Katalon Studio dan Python usulan penelitian diantaranya Analisis Perbandingan Pengujian Manual Dan Automation Testing Pada," *COMSERVA : Jurnal Penelitian dan Pengabdian Masyarakat*, vol. 4, no. 8, pp. 2534–2540, 2024, doi: 10.59141/comserva.v4i8.2737.

- [6] R. Fauzan, F. P. Soedjono, A. A. Permadani, and M. Ainul, "Perbandingan Pengujian Manual dan Terotomasi pada Software Enterprise Resource Planning," *JAIIT (Journal of Advanced in Information and Industrial Technology)*, vol. 5, no. 1, pp. 23–30, 2023.
- [7] M. Andhika *et al.*, "Analisis Perbandingan Automation Testing Katalon Studio Dan Selenium Ide Pada Website E-Commerce Menggunakan Metode End-To-End (STUDI KASUS : ENTERKOMPUTER . COM," *Positif: Jurnal Sistem dan Teknologi Informasi*, pp. 1–11, 2025.
- [8] F. W. Azhari and D. F. Suyatno, "Pengujian Otomatis GUI dengan Katalon Studio pada Situs Lowongan Kerja Jobstreet dan Glints," *JEISBI: (Journal of Emerging Information Systems and Business Intelligence)*, vol. 05, no. 03, pp. 205–213, 2024.
- [9] M. Aidil and M. Jibril, "Implementasi Pengujian Otomatis End-to-End pada Website Konseling Mahasiswa (SiKosa) Menggunakan Cypress," *Jurnal Sistem Informasi (TEKNOFILE)*, vol. 3, no. 12, pp. 929–939, 2025.
- [10] S. Wijanarko, "Analisis Pengujian Otomatis Antarmuka Web Pada Formulir Jotform Menggunakan Cypress," *Jurnal Sosial Teknologi*, vol. 6, no. 1, pp. 132–141, 2026, doi: 10.59188/jurnalsostech.v6i1.32662.
- [11] A. F. Rina Septiani, Budi Jejen Zaenal Abidin, "Implementasi Pengujian Otomatis End-To-End Menggunakan Cypress dengan Metode Black Box Testing untuk Meningkatkan Kualitas Aplikasi EdTech XYZ Berbasis Website," *Ilmiah, Jurnal Komputer, Ilmu Informasi, Teknologi*, vol. 2, no. 2, 2025.
- [12] Rahayu Amaliyah Kamisr, "Studi Komparatif Pengujian Manual Dan Pengujian Otomatis Dengan Cypress Pada Website," *SAINTEKES: Jurnal Sains, Teknologi Dan Kesehatan*, vol. 2, no. 4, pp. 476–485, 2023.
- [13] S. Sri Rahayu, D. Aris Firmansyah, S. Susanti, and U. Adhirajasa Reswara Sanjaya Bandung, "Analisis Penggunaan Tools Automation Testing pada Aplikasi : Systematic Literature Review," *Remik: Riset dan E-Jurnal Manajemen Informatika Komputer*, vol. 8, no. 1, pp. 106–116, 2024, [Online]. Available: <http://doi.org/10.33395/remik.v8i1.13241>
- [14] W. Nugroho, "Penerapan Sistem Informasi Pendaftaran Peserta Didik Baru Berbasis WEB Untuk Peningkatan Mutu dan Jumlah Pendaftar" *Infotek : Jurnal Informatika dan Teknologi*, vol. 6, no. 1, pp. 21–29, 2023.
- [15] U. M. Kharisma Dinda Amareta , Zunita Wulansari, "Sistem Informasi Klasifikasi Atlet Ju-Jitsu Berbasis Web Menggunakan Metode Agile," *Infotek : Jurnal Informatika dan Teknologi*, vol. 9, no. 1, 2026.