

ANALISIS PERBANDINGAN MANUAL TESTING DAN AUTOMATION TESTING PADA SISITEM INFORMASI HUMAN RESOURCE DEVELOPMENT

Maulida Riski¹⁾, Medhanita Dewi Renanti²⁾

^{1,2)} Teknologi Rekayasa Perangkat Lunak Sekolah Vokasi IPB, Jalan Kumbang 14, Bogor, Indonesia
email : maulidariski@apps.ipb.ac.id¹⁾

Abstrak

Pengujian sistem informasi Human Resource Development (HRD) di PT Asta Protek Jiarsi saat ini dilakukan secara manual. Meskipun metode ini memberikan pemahaman mendalam tentang aplikasi, pengujian manual seringkali memakan waktu. Seiring dengan perkembangan teknologi, metode pengujian otomatis muncul sebagai alternatif untuk mempercepat proses pengujian. Tujuan penelitian ini adalah melakukan pengujian manual dan otomatis pada sistem informasi HRD menggunakan kerangka kerja Software Testing Life Cycle (STLC) dan menganalisis perbandingan antara kedua metode tersebut. Cypress dipilih untuk pengujian otomatis karena keunggulannya dalam melakukan pengujian aplikasi berbasis web secara langsung dalam browser. Penelitian ini menguji 10 fitur HRD dengan tiga iterasi. Skenario uji dibuat dengan 64 skenario. Dari 64 skenario, 39 berhasil dan 25 gagal. Hasil menunjukkan pengujian otomatis lebih efisien dalam hal waktu, tetapi hanya mampu menjalankan 92% skenario uji, sedangkan pengujian manual dapat menguji 100% skenario. Pengujian manual melibatkan lebih sedikit tahap persiapan namun lebih banyak langkah implementasi dibandingkan pengujian otomatis.

Kata Kunci :

cypress, pengujian, pengujian otomatis, perbandingan pengujian, software testing life cycle.

Abstract

The testing of the Human Resource Development (HRD) information system at PT Asta Protek Jiarsi is currently conducted manually. While this method provides a deep understanding of the application, manual testing is often time-consuming. With technological advancements, automated testing methods have emerged as an alternative to speed up the testing process. The purpose of this study is to conduct both manual and automated testing on the HRD information system using the Software Testing Life Cycle (STLC) framework and to analyze the comparison between the two methods. Cypress was selected for automated testing due to its advantages in directly testing web-based applications within the browser. This study tested 10 HRD features across three iterations. A total of 64 test scenarios were created, of which 39 passed and 25 failed. The results indicate that automated testing is more time-efficient but was only able to execute 92% of the test scenarios, whereas manual testing could cover 100% of the scenarios. Manual testing involved fewer preparation stages but more implementation steps compared to automated testing.

Keywords :

cypress, testing, automated testing, testing comparison, software testing life cycle.

1. PENDAHULUAN

Saat ini, pengujian sistem informasi Human Resource Development (HRD) di PT Asta Protek Jiarsi hanya dilakukan secara manual. Meskipun metode ini dapat memberikan pemahaman mendalam tentang aplikasi, pengujian manual seringkali memakan waktu dan sumber daya yang besar sehingga tidak tepat untuk pengujian berulang. Seiring dengan perkembangan dunia pengetahuan, metode pengujian otomatis atau *automation testing* muncul sebagai alternatif untuk mempercepat proses pengujian [1].

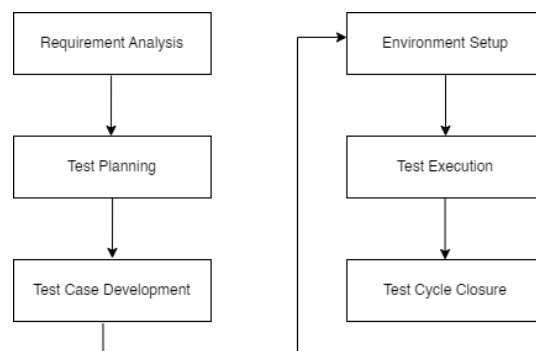
Penelitian yang dilakukan oleh [2] yang membandingkan pengujian manual dan otomatis dari segi keefektifan dan keefisienan. Simpulan dari penelitian tersebut adalah pengujian otomatis memerlukan investasi awal yang lebih tinggi karena proses pengembangan skrip. Meskipun

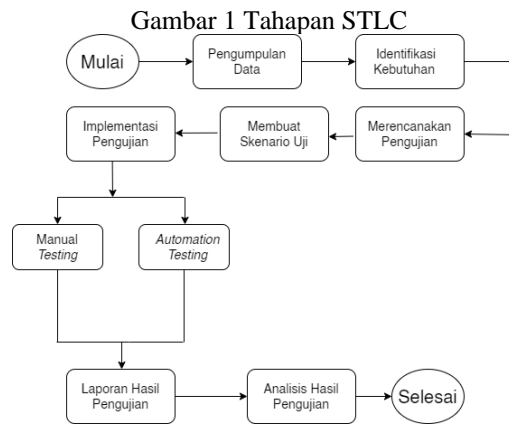
demikian, pengujian otomatis mempunyai peranan untuk mempersingkat waktu ketika dilakukan pengujian regresi. Menurut [3], *automation testing* dapat memudahkan dalam pengujian fitur secara berulang dengan waktu yang lebih sedikit. Hal itu dikarenakan skrip pengujian dapat digunakan kembali jika diperlukan. Dalam penelitian yang dilakukan [4] terkait perbandingan *automation testing* dengan *manual testing* pada situs *e-commerce*, didapatkan hasil bahwa rata-rata waktu untuk melakukan *automation testing* lebih cepat dibandingkan dengan pengujian manual. Pengujian otomatis menjadi alat yang efisien dalam menjalankan sejumlah besar pengujian dengan waktu singkat, menghemat sumber daya tenaga kerja, dan meningkatkan kualitas operasi pengujian [5]. Meskipun pengujian otomatis memiliki beberapa keunggulan dibandingkan dengan pengujian manual, tetapi untuk melaksanakannya dibutuhkan pengetahuan tentang alat uji otomatis yang digunakan. Selain itu, hasil penelitian yang dilakukan oleh [1] menyebutkan bahwa tahap persiapan menggunakan *automation testing* membutuhkan waktu yang lebih besar dibandingkan dengan pengujian manual.

Berdasarkan studi literatur yang telah dilakukan, dikemukakan hipotesis bahwa pengujian otomatis akan meningkatkan kecepatan pengujian. Oleh karena itu, diperlukan analisis terhadap perbandingan kedua metode tersebut guna membuktikan hipotesis ini. Tujuan dari penelitian ini adalah melakukan pengujian manual dan otomatis pada sistem informasi HRD dengan kerangka kerja *Software Testing Life Cycle* (STLC) dan menganalisis perbandingan antara kedua metode pengujian tersebut. Analisis perbandingan yang akan dilakukan meliputi perbandingan efisiensi, efektivitas, serta perbandingan jumlah tahapan kedua metode meliputi tahap persiapan dan tahap implementasi. Alat uji yang akan digunakan dalam pengujian otomatis pada penelitian ini adalah *Cypress*. *Cypress* dipilih karena memiliki keunggulan dalam melakukan pengujian aplikasi berbasis web secara langsung dalam *browser*. Hal ini tidak dimiliki oleh alat uji otomatis lain seperti Selenium atau Katalon Studio. Menurut [6] penerapan STLC pada proses pengujian dapat membuat alur pengujian lebih terstruktur dan menyediakan fokus yang lebih baik pada setiap tahap pengujian yang dilalui. STLC dipilih karena langkah-langkahnya sistematis untuk melakukan proses pengujian perangkat lunak sehingga dapat membantu dalam mencapai tujuan penelitian.

2. METODE / ALGORITMA

Metode pengujian yang digunakan pada penelitian ini ialah *Software Testing Life Cycle* (STLC). Tahapan STLC diantaranya adalah *requirement analysis*, *test planning*, *test case development*, *environment setup*, *test execution*, dan *test cycle closure* [7]. Tahapan STLC ditunjukkan oleh Gambar 1. Metode penelitian ditunjukkan oleh Gambar 2. Penelitian ini berlangsung selama empat bulan, yakni dimulai dari bulan Januari 2024 sampai dengan bulan April 2024. Lokasi penelitian dilakukan di Sekolah Vokasi IPB yang beralamat di Jl. Kumbang No.14, Kota Bogor, Jawa Barat.





Gambar 2 Langkah Kerja Penelitian

Pengumpulan Data

Pengumpulan data dilakukan dengan melakukan observasi terhadap sistem yang akan diuji, yaitu sistem informasi *Human Resource Development* (HRD). Observasi dilakukan pada lingkungan *development*. Berikut merupakan hasil observasi pada sistem informasi HRD.

- a. Sistem informasi HRD memiliki tiga pengguna, yaitu HRD, karyawan, dan admin. HRD adalah pengguna yang bertanggung jawab untuk mengelola data karyawan. Karyawan memiliki hak untuk mengakses sistem dan melakukan beberapa aksi, sedangkan admin berperan sebagai manajemen sistem informasi HRD.
- b. Cakupan pengujian diambil dari salah satu pengguna sistem, yaitu HRD.

Identifikasi Kebutuhan (Requirement Analysis)

Deskripsi mengenai kebutuhan fungsional HRD pada sistem informasi HRD ditunjukkan oleh Tabel 1.

Tabel 1 Kebutuhan fungsional HRD

No.	Nama Fungsi	Deskripsi
1.	<i>Login</i>	Pengguna dapat masuk ke sistem
2.	<i>Approval profile</i>	Pengguna dapat melakukan validasi terhadap perubahan profil yang dilakukan oleh karyawan
3.	<i>Completed task</i>	Pengguna dapat melihat tugas karyawan yang sudah selesai dan melakukan pencarian
4.	<i>To do list</i>	Pengguna dapat melihat daftar tugas dari masing-masing karyawan dan melakukan pencarian
5.	<i>Employee</i>	Pengguna dapat mengelola data karyawan
6.	<i>Reimbursement</i>	Pengguna dapat melihat daftar <i>reimbursement</i> karyawan dan melakukan pencarian
7.	<i>Medical claim</i>	Pengguna dapat melihat data <i>medical claim</i> yang diajukan karyawan dan melakukan pencarian
8.	<i>Leave</i>	Pengguna dapat mengelola data karyawan yang cuti
9.	<i>Attendance</i>	Pengguna dapat melihat data kehadiran karyawan
10.	<i>Overtime detail</i>	Pengguna dapat melihat data lembur karyawan

Pada setiap fitur terdapat komponen penting yaitu *search box* untuk mencari data karyawan juga fitur *export* untuk melakukan *generate* data menjadi format Microsoft Excel atau *Comma Separated Values* (CSV). Dalam hal ini, komponen-komponen tersebut juga masuk ke dalam lingkup pengujian karena merupakan kebutuhan HRD yang harus bisa dilakukan oleh sistem.

Perencanaan Pengujian (Testing Planning)

ANALISIS PERBANDINGAN MANUAL TESTING DAN AUTOMATION TESTING PADA SISITEM INFORMASI HUMAN RESOURCE DEVELOPMENT

Pengujian akan dilakukan secara manual dan otomatis. Pengujian manual langsung dilakukan pada *browser* dan hasilnya dicatat menggunakan Google Sheets. *Automation testing* akan memanfaatkan alat pengujian otomatis. Sebelum menentukan alat yang akan digunakan untuk pengujian otomatis, dilakukan analisis terhadap beberapa alat uji otomatis. Analisis alat uji otomatis ditunjukkan oleh Tabel 2.

Analisis dilakukan dengan membaca literatur terkait alat uji otomatis yang dapat digunakan. Menurut [8] *Playwright* lebih ramah pengguna baru karena kemudahannya, sedangkan *Selenium* lebih sesuai untuk pengguna berpengalaman dengan kemampuan pemrograman yang handal. Masing-masing memiliki kelebihan dan kelemahan, sehingga pemilihan alat uji dapat disesuaikan dengan kebutuhan proyek. Menurut [9] yang membandingkan alat uji otomatis antara *Cypress*, *Selenium*, dan *Katalon Studio* dihasilkan simpulan bahwa *Selenium* mendukung berbagai sistem operasi dan bahasa serta pengujian lintas *browser*, tetapi membutuhkan konfigurasi awal yang rumit. *Cypress* cepat dan andal namun terbatas pada beberapa *browser*, sedangkan *Katalon Studio* mudah digunakan tanpa keterampilan pemrograman tingkat lanjut, tetapi terbatas dalam pilihan bahasa *scripting*.

Pengujian dilakukan untuk membandingkan kedua metode pengujian dari sisi rata-rata waktu yang dibutuhkan, jumlah skenario uji yang dapat diimplementasikan, dan perbandingan langkah-langkah pengujian dari kedua metode. Pengujian akan dilakukan sebanyak tiga kali iterasi untuk menjamin keakuratan. Jenis pengujian yang akan dilakukan adalah *black box testing*. *Black box testing* adalah metode pengujian yang mengisyaratkan sistem yang akan diuji seperti kotak hitam sehingga hanya memasukkan *input*-an dan tidak mengetahui apa yang terjadi di dalamnya [10]. Fitur yang diuji adalah fitur yang telah dikumpulkan pada tahap *requirement analysis* sebelumnya.

Tabel 2 Analisis Alat Uji Otomatis

Variabel Pemanding	<i>Cypress</i>	<i>Selenium</i>	<i>Playwright</i>	<i>Katalon Studio</i>
Proses instalasi	Cukup mudah karena dengan satu <i>command line</i> dapat mengunduh seluruh <i>package</i>	Harus menyesuaikan <i>driver</i> sesuai dengan <i>browser</i> yang akan digunakan	Cukup mudah karena dengan satu <i>command line</i> dapat mengunduh seluruh <i>package</i>	Cukup mudah seperti proses instalasi aplikasi pada umumnya
Lisensi	Gratis, <i>open source</i>	Gratis, <i>open source</i>	Gratis, <i>open source</i>	Berbayar
<i>Script test</i>	Menulis kode pemrograman secara manual	Menulis kode pemrograman secara manual	Dibuat otomatis dengan melakukan perekaman	Dibuat otomatis dengan melakukan perekaman
<i>Webdriver</i>	Tidak perlu	Memerlukan	Tidak perlu	Memerlukan
Eksekusi kode	Cepat	Agak lambat, tergantung konfigurasi <i>webdriver</i>	Cepat	Agak lambat, tergantung konfigurasi <i>webdriver</i>
Komunitas	Komunitas luas sehingga dapat memudahkan proses belajar	Komunitas luas sehingga dapat memudahkan proses belajar	Komunitas pendukung belum seluas alat uji lain	Komunitas luas sehingga dapat memudahkan proses belajar

Pembuatan Skenario Uji (Test Case Development)**Test Case Pengujian Manual**

Studi ini menggunakan *template* yang merujuk pada penelitian yang dilakukan oleh [11]. Skenario uji untuk sistem informasi HRD ditunjukkan oleh Tabel 3.

Tabel 3 Test Case Pengujian

Fitur	ID	TC_Description	Expected Result
<i>Login</i>	TC_1.1	Memastikan gagal <i>login</i> dengan mengisi <i>invalid username</i>	Gagal <i>login</i>
	TC_1.2	Memastikan gagal <i>login</i> dengan mengisi <i>invalid password</i>	Gagal <i>login</i>
	TC_1.3	Memastikan gagal <i>login</i> dengan mengisi <i>password</i> saja	Gagal <i>login</i>
	TC_1.4	Memastikan gagal <i>login</i> dengan mengisi <i>username</i> saja	Gagal <i>login</i>
	TC_1.5	Memastikan gagal <i>login</i> dengan mengosongkan <i>password</i> dan <i>username</i>	Gagal <i>login</i>
	TC_1.6	Memastikan berhasil <i>login</i> dengan mengisi valid <i>username</i> dan <i>password</i>	Berhasil <i>login</i>
<i>Approval profile</i>	TC_2.1	Mencari nama yang tersedia	Menampilkan data
	TC_2.2	Mencari nama yang tidak tersedia	Menampilkan <i>alert</i>
	TC_2.3	Melakukan <i>action approve</i>	Data hilang dari tabel
	TC_2.4	Melakukan <i>action reject</i>	Data hilang dari tabel
	TC_2.5	Melakukan <i>export</i> data	Data terunduh
<i>Completed task</i>	TC_3.1	Memilih <i>picklist</i> pada kolom <i>category</i>	<i>Picklist</i> tersedia
	TC_3.2	Memilih kalender pada kolom <i>due date</i>	Bisa pilih tanggal pada kalender
	TC_3.3	Mencari data pada kolom <i>description task</i>	Menampilkan data
	TC_3.4	Mencari data pada kolom <i>description realization</i>	<i>Picklist</i> tersedia
	TC_3.5	Melakukan <i>export</i> data	Data terunduh
<i>To do list</i>	TC_4.1	Memilih <i>picklist</i> pada kolom <i>category</i>	<i>Picklist</i> tersedia
	TC_4.2	Memilih kalender pada kolom <i>due date</i>	Bisa pilih tanggal pada kalender
	TC_4.3	Mencari data pada kolom <i>description realization</i>	Menampilkan data
	TC_4.4	Memilih <i>picklist</i> pada kolom <i>task status</i>	<i>Picklist</i> tersedia
	TC_4.5	Melakukan <i>export</i> data	Data terunduh
<i>Employee</i>	TC_5.1	Mencari nama karyawan yang tersedia	Menampilkan data
	TC_5.2	Mencari nama karyawan yang tidak tersedia	Menampilkan <i>alert</i>
	TC_5.3	Memilih <i>picklist</i> pada kolom <i>organization</i>	<i>Picklist</i> tersedia
	TC_5.4	Memilih <i>picklist</i> pada kolom <i>job level</i>	<i>Picklist</i> tersedia
	TC_5.5	Memilih <i>picklist</i> pada kolom <i>employment status</i>	<i>Picklist</i> tersedia
	TC_5.6	Memilih kalender pada kolom <i>join date</i>	Bisa pilih tanggal pada kalender
	TC_5.7	Memilih kalender pada kolom <i>end date</i>	Bisa pilih tanggal pada kalender
	TC_5.8	Mencari data <i>email private</i>	Menampilkan data

	TC_5.9	Memilih <i>button action detail</i>	Mengarah ke detail karyawan
	TC_5.10	Memilih <i>button action create resignation</i>	Muncul <i>alert</i> sukses dan data hilang dari tabel
	TC_5.11	Memilih <i>button action send email</i>	Muncul <i>alert</i> sukses
	TC_5.12	Menambahkan karyawan dengan mengosongkan <i>field mandatory</i>	Muncul <i>alert</i> tidak bisa ke page selanjutnya
	TC_5.13	Menambahkan karyawan dengan mengisi lengkap semua <i>field</i>	<i>Button next</i> step berhasil di klik dan data tersimpan
	TC_5.14	<i>Import file</i> karyawan sesuai <i>template</i>	Data bertambah pada tabel
	TC_5.15	<i>Import file</i> karyawan tidak sesuai <i>template</i>	Muncul <i>alert</i>
	TC_5.16	Melakukan <i>export</i> data	Data terunduh
Reimbursement	TC_6.1	Melakukan pencarian data <i>reimbursement desc</i>	Menampilkan data
	TC_6.2	Memilih kalender pada kolom <i>effective date</i>	Bisa pilih tanggal pada kalender
	TC_6.3	Memilih kalender pada kolom <i>date thru</i>	Bisa pilih tanggal pada kalender
	TC_6.4	Memilih <i>picklist</i> pada kolom status	<i>Picklist</i> tersedia
	TC_6.5	Melakukan <i>export</i> data	Data terunduh
Medical claim	TC_7.1	Memilih <i>picklist</i> pada kolom <i>type</i>	<i>Picklist</i> tersedia
	TC_7.2	Melakukan pencarian data <i>deases</i>	Menampilkan data
	TC_7.3	Memilih kalender pada kolom <i>date</i>	Bisa pilih tanggal pada kalender
	TC_7.4	Melakukan <i>export</i> data	Data terunduh
Leave	TC_8.1	Memilih kalender pada kolom <i>leave date</i>	Bisa pilih tanggal pada kalender
	TC_8.2	Memilih kalender pada kolom <i>expired date</i>	Bisa pilih tanggal pada kalender
	TC_8.3	Melakukan pencarian data <i>reason</i>	Menampilkan data
	TC_8.4	Memilih <i>picklist</i> pada kolom <i>approval</i>	<i>Picklist</i> tersedia
	TC_8.5	Melakukan aksi <i>approve</i> validasi	Status menjadi “ <i>Approved</i> ”
	TC_8.6	Melakukan aksi <i>reject</i> validasi	Status menjadi “ <i>Rejected</i> ”
	TC_8.7	Melakukan <i>export</i> data	Data terunduh
Attendance	TC_9.1	Memilih kategori bulan	Bisa pilih bulan pada kalender
	TC_9.2	Memilih <i>picklist</i> pada kolom <i>day</i>	<i>Picklist</i> tersedia
	TC_9.3	Memilih kalender pada kolom <i>date schedule in/out</i>	Bisa pilih tanggal pada kalender
	TC_9.4	Melakukan pencarian data <i>time schedule in/out</i>	Menampilkan data
	TC_9.5	Mengklik <i>button action</i> ke halaman <i>overtime</i>	Pindah Halaman
	TC_9.5	Melakukan <i>export</i> data	Data terunduh
Overtime	TC_10.1	Memilih <i>picklist</i> pada kolom <i>day</i>	<i>Picklist</i> tersedia
	TC_10.2	Melakukan pencarian data <i>address In</i>	Menampilkan data

TC_10.3	Melakukan <i>export</i> data	Data terunduh
---------	------------------------------	---------------

Test Case Pengujian Otomatis

Pembuatan *test case* pada *Cypress* ditulis pada *file* spesifikasi dengan ekstensi *cy.js*. Gambar 3 merupakan contoh *file* spesifikasi untuk *test case overtime*.

```
import MenuElement from "../Pages/Menu";
import OvertimePage from "../Pages/Overtime";

describe('Overtime', () => {
  let DataOvertime;
  let Overtime = new OvertimePage

  beforeEach(() => {
    cy.fixture("Overtime.json").then((data) => {
      DataOvertime = data;
    });
    cy.viewport(1280, 720);
    cy.login('hrd.qa.1', 'hrd12@')
  });

  const openOvertimeMenu = () => {
    const Menu = new MenuElement();
```

Gambar 3 File spesifikasi *overtime*

Implementasi Pengujian (Test Execution)

Pada tahap ini akan dijalankan *test case* yang sudah dibuat sebelumnya. Hasil implementasi pengujian manual ditunjukkan oleh Tabel 4.

Tabel 4 Hasil implementasi pengujian

Fitur	Jumlah Skenario	Jumlah Berhasil	Persentase Lulus Uji
<i>Login</i>	6	6	100%
<i>Approval Profile</i>	5	5	100%
<i>Completed Task</i>	4	2	50%
<i>To do list</i>	5	2	40%
<i>Employee</i>	16	11	68,75%
<i>Reimbursement</i>	5	2	40%
<i>Medical Claim</i>	4	2	50%
<i>Leave</i>	7	4	57%
<i>Attendance</i>	6	4	66%
<i>Overtime</i>	3	2	66,6%

3. HASIL DAN PEMBAHASAN

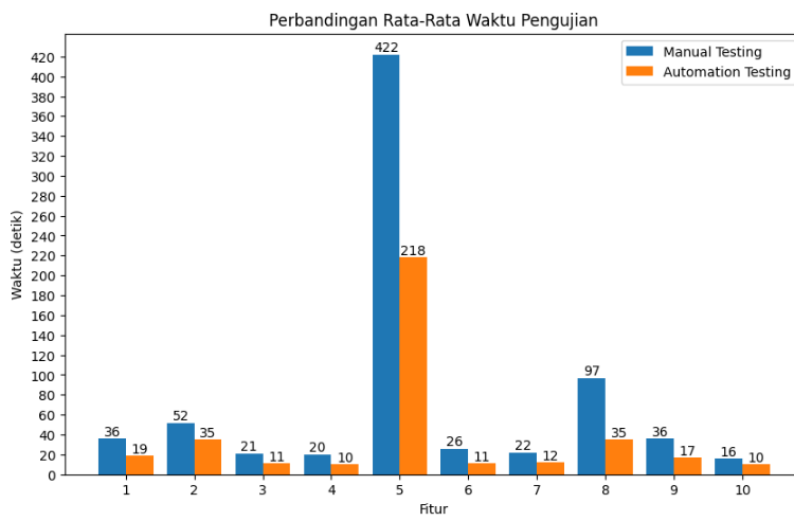
Perbandingan Efisiensi Waktu Pengujian

Hasil pengujian yang pada Tabel 5 menunjukkan waktu antara pengujian manual dan otomatis. Pengujian dilakukan sebanyak tiga kali iterasi karena waktu yang didapat dari pengujian bervariasi. Selisih waktu dari setiap iterasi pengujian sangat kecil sehingga hasilnya bisa dianggap valid. Perbedaan waktu ini terjadi karena beberapa faktor, diantaranya koneksi internet, kecepatan mengetik penguji, dan pengalaman penguji saat melakukan percobaan. Oleh karena itu diambil rata-rata dari tiga kali iterasi.

Tabel 5 Perbandingan rata-rata waktu pengujian

No.	Fitur	Waktu (s)	
		Manual	Otomatis

1	Login	36	19
2	Approval profile	52	35
3	Completed task	21	11
4	To do list	20	10
5	Employee	422	218
6	Reimbursement	26	11
7	Medical claim	22	12
8	Leave	97	35
9	Attendance	36	17
10	Overtime	16	10
Total		750	381



Gambar 4 Grafik perbandingan rata-rata waktu pengujian

Gambar 4 menunjukkan grafik rata-rata waktu hasil pengujian sistem informasi HRD. Berdasarkan grafik, disimpulkan bahwa *automation testing* memerlukan waktu yang lebih singkat daripada *manual testing*. Hal ini menunjukkan bahwa *automation testing* lebih efisien daripada *manual testing*. Titik puncak tertinggi terjadi pada fitur lima atau fitur *employee* dalam grafik *manual testing* dan *automation testing*. Hal Ini disebabkan oleh jumlah skenario tes terbanyak pada fitur *employee*, mencapai 16 skenario pengujian. Salah satu skenario pada fitur *employee* adalah menambah data karyawan, yang melibatkan pengisian beberapa formulir. *Automation testing* memiliki keunggulan dalam manajemen data uji pada tahap persiapan pengujian dengan menggunakan fitur *fixture*. *File fixture* ini dapat dimuat dalam pengujian dan disesuaikan dengan kebutuhan.

Perbandingan Efektivitas Metode Pengujian

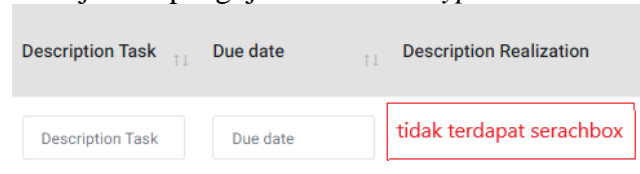
Terdapat perbedaan keefektifan antara kedua metode. Pengujian manual dapat menjalankan semua skenario yang sudah dibuat sedangkan terdapat beberapa skenario yang tidak bisa dijalankan pada pengujian otomatis. Tabel 6 merupakan tabel *test case* yang tidak bisa dijalankan oleh pengujian otomatis.

Tabel 6 Daftar *test case* yang tidak bisa diujikan otomatis

Fitur	TC_ID	TC_Description
Completed task	TC_3.4	Mencari data pada kolom <i>description realization</i>
To do list	TC_4.4	Memilih <i>picklist</i> pada kolom <i>description realization</i>
Reimbursement	TC_6.4	Memilih <i>picklist</i> pada kolom status

Medical claim	TC_7.3	Memilih kalender pada kolom <i>date</i>
Leave	TC_8.4	Memilih <i>picklist</i> pada kolom <i>approval</i>

Pengujian otomatis tidak dapat menjalankan semua skenario pengujian karena ketergantungan pada elemen aplikasi web. *Cypress* memerlukan ketersediaan dan aksesibilitas elemen sebelum pengujian dilakukan. Gambar 5 merupakan salah satu skenario TC_3.4 yang tidak bisa diuji oleh pengujian otomatis *Cypress*.



Gambar 5 Tampilan skenario TC_3.4

Tabel 6 merupakan perbandingan keefektifan dari kedua metode yang dilihat dari banyaknya *test case* yang dapat diujikan oleh masing-masing metode pengujian. Pengujian manual dapat menjalankan 100% *test case* yang sudah dirancang sedangkan pengujian otomatis hanya bisa menjalankan 92% dari total *test case*.

Tabel 4 Perbandingan efektivitas

No.	Komparasi	Manual	Otomatis
1.	Jumlah skenario yang dikembangkan	63	63
2.	Jumlah skenario pengujian yang dapat dijalankan	63	58
Total		100%	92%

Perbandingan Jumlah Tahapan

Analisis selanjutnya ialah membandingkan langkah-langkah antara *manual testing* dan *automation testing*. Perbandingan ini merujuk pada penelitian yang dilakukan oleh [1] yang membandingkan kedua metode berdasarkan tahapan pengujian saat persiapan dan tahap implementasi pengujian. Perbandingan tahapan persiapan ditunjukkan oleh Tabel 7. Perbandingan tahapan implementasi ditunjukkan oleh Tabel 8.

Tabel 5 Perbandingan tahap persiapan

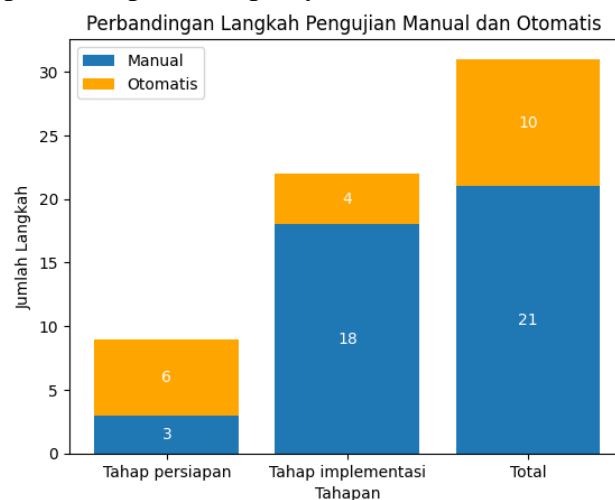
No.	Pengujian Manual	No.	Pengujian Otomatis
1.	Memahami aplikasi	1.	Memahami aplikasi
2.	Membuat skenario uji coba	2.	Menginstall <i>Cypress</i>
3.	Menyiapkan data uji	3.	Membuat <i>project</i> pada <i>Cypress</i>
		4.	Membuat dan menulis <i>class page</i>
		5.	Membuat dan menulis <i>class step</i>
		6.	Membuat dan menulis <i>class fixture</i>

Tabel 6 Perbandingan tahap implementasi

No.	Pengujian Manual	No.	Pengujian Otomatis
1.	Membuka situs web yang diuji	1.	Membuka IDE
2.	Melakukan <i>login</i>	2.	Menjalankan <i>Cypress</i>
3.	Memilih menu <i>employee</i>	3.	Membuka dan menjalankan <i>project automation testing</i>
4.	Melakukan pencarian pegawai berdasarkan nama	4.	Laporan hasil uji coba terbuat secara otomatis
5.	Melakukan pencarian pegawai		

	berdasarkan divisi		
6.	Melakukan pencarian pegawai berdasarkan jabatan		
7.	Melakukan pencarian pegawai berdasarkan status kerja		
8.	Melakukan pencarian pegawai berdasarkan tanggal diterima sebagai karyawan		
9.	Melakukan pencarian pegawai berdasarkan email pegawai		
10.	Mengklik <i>button action</i>		
11.	Memilih <i>action detail</i>		
12.	Memilih <i>action delete</i>		
13.	Memilih <i>action send email reminder</i>		
14.	Mengklik <i>button add</i> untuk menambahkan pegawai		
15.	Memasukkan <i>value</i> pada <i>form</i> di <i>section</i> personal data		
16.	Memasukkan <i>value</i> pada <i>form</i> di <i>section</i> <i>employment</i> data		
17.	Memasukkan <i>value</i> pada <i>form</i> di <i>section</i> <i>payroll</i>		
18.	Klik <i>button submit</i>		

Gambar 6 menunjukkan bahwa langkah-langkah yang diperlukan untuk melakukan *manual testing* lebih sedikit dibandingkan *automation testing*. Artinya dibutuhkan usaha yang lebih besar untuk mempersiapkan *automation testing*. Pada tahap implementasi menunjukkan bahwa *automation testing* hanya membutuhkan empat tahap sedangkan *manual testing* membutuhkan 18 tahap atau empat kali lipat dari *automation testing*.



Gambar 6 Grafik perbandingan langkah pengujian

4. KESIMPULAN

Tingkat ketercapaian target kegiatan penelitian mengenai analisis perbandingan pengujian manual dan otomatis menunjukkan bahwa pengujian otomatis lebih efisien dalam hal waktu, meskipun tidak seefektif pengujian manual dalam hal cakupan skenario uji yang dijalankan. Pengujian manual mampu menguji 100% skenario uji sedangkan pengujian otomatis hanya

92%. Pengujian otomatis memerlukan lebih banyak tahapan dalam persiapan dibandingkan pengujian manual, namun lebih sederhana dalam tahap implementasi. Dampak dari penelitian ini adalah pengetahuan yang lebih mendalam mengenai efisiensi dan efektivitas pengujian manual dan otomatis sehingga dapat digunakan untuk menentukan metode pengujian yang lebih tepat sesuai kebutuhan. Rekomendasi untuk kegiatan selanjutnya adalah melakukan pengujian regresi terhadap fitur yang saling berhubungan, mengembangkan skenario pengujian untuk respons aplikasi terhadap perilaku pengguna yang tidak normal, serta menganalisis waktu terbaik untuk menggunakan pengujian manual dan otomatis.

5. REFERENSI

- [1] Sa'adah, U., Hasim, J. A. N., Yunanto, A. A., Permatasari, D. I., Hardiansyah, F. F., Wulandari, I., & Thooriqoh, H. A. (2021). Framework testing otomatis berbasis Serenity dan Jenkins. *JUTI: Jurnal Ilmiah Teknologi Informasi*, 19(2), 102.
- [2] Dobles, I., Martínez, A., & López, C. Q. (2019). Comparing the effort and effectiveness of automated and manual tests. In *14th Iberian Conference on Information Systems and Technologies (CISTI)* (pp. 1–6).
- [3] B. Prakoso and A. Sujarwo, "Perancangan Automated Testing Pada Studi Kasus Website Indicar," *J. Ilm. Teknol. Inf. dan ...*, vol. 4, pp. 29–32, 2022.
- [4] Dewandra Sapto Prasetyo and W. Silfianti, "Analisis Perbandingan Pengujian Manual Dan Automation Testing Pada Website E-Commerce," *J. Ilm. Tek.*, vol. 2, no. 2, pp. 127–131, 2023.
- [5] C. Garbett, "Investing in information technology," *J. Prop. Financ.*, vol. 4, no. 1, pp. 118–125, 1993.
- [6] Ruliansyah, Tukino, Huda, B., & Hananto, A. L. (2023). Penerapan software testing life cycle pada pengujian otomatisasi platform Dzikra. *CSRID Journal*, 15(1), 1–11.
- [7] I. R. Dhaifullah, M. Muttanifudin H, A. Ananda Salsabila, and M. Ainul Yaqin, "Survei Teknik Pengujian Software," *J. Autom. Comput. Inf. Syst.*, vol. 2, no. 1, pp. 31–38, 2022.
- [8] Anisa Amalia, "Analisis Pemanfaatan Playwright Untuk Pengujian Aplikasi Berbasis Web (Studi Kasus: Sistem Manajemen Jarigan)," 2022.
- [9] Scott, A. M., Forbes, C., Clark, J., Carter, M., Glasziou, P., & Munn, Z. (2021). Systematic review automation tools improve efficiency but lack of knowledge impedes their adoption: A survey. *Journal of Clinical Epidemiology*, 138, 80–94.
- [10] S. Nidhra, "Black Box and White Box Testing Techniques - A Literature Review," *Int. J. Embed. Syst. Appl.*, vol. 2, no. 2, pp. 29–50, 2012.
- [11] A. N. Hasibuan and T. Dirgahayu, "Pengujian dengan Unit Testing dan Test case pada Proyek Pengembangan Modul Manajemen Pengguna," *J. Inform. Univ. Islam Indones.*, vol. 2, no. 1, pp. 103–109, 2020.