

# Pengujian Otomatis GUI dengan Katalon Studio pada Situs Lowongan Kerja Jobstreet dan Glints

Fauza Wayah Azhari<sup>1</sup>, Dwi Fatrianto Suyatno<sup>2</sup>

<sup>1,3</sup> Sistem Informasi, Fakultas Teknik, Unoversitas Negeri Surabaya

[1fauza.20061@mhs.unesa.ac.id](mailto:1fauza.20061@mhs.unesa.ac.id)

[3dwifatrianto@unesa.ac.id](mailto:3dwifatrianto@unesa.ac.id)

**Abstrak**— Jobstreet dan Glints merupakan situs rekrutmen kerja *online* yang menyediakan informasi lowongan kerja dan kesempatan karier untuk para pencari kerja. Situs lowongan kerja Jobstreet dan Glints memiliki kompleksitas GUI yang berbeda-beda dan adanya peningkatan jumlah pengguna maka diperlukan pengujian GUI dan pengujian performa. Penelitian ini bertujuan untuk mengetahui hasil pengujian GUI dan pengujian performa pada kedua situs, apakah fungsionalitas dari kedua situs tersebut dapat berjalan dengan baik dan seberapa jauh performa kedua situs saat berjalan di bawah beban kerja. Pengujian GUI ini dilakukan secara otomatis dengan Katalon Studio dan menggunakan metode *Software Testing Life Cycle* (STLC). Sedangkan, pengujian performa dilakukan dengan teknik *load testing*, alat yang digunakan yaitu JMeter dan menggunakan metode *Performance Testing Life Cycle* (PTLC). Hasil dari pengujian yang telah dilakukan menunjukkan bahwa fungsionalitas situs Jobstreet lebih unggul daripada situs Glints dikarenakan semua kasus uji Jobstreet memiliki status keberhasilan *passed* pada semua iterasi pengujian. Sedangkan, pada situs Glints terdapat satu kasus uji yang *failed* yaitu pada TC14 Pengalaman Kerja saat pengujian kedua dikarenakan terdapat *field* yang tidak dapat berinteraksi. Namun, masalah tersebut dapat teratasi. Dari segi *response time* kedua situs memiliki kelebihan dan kekurangan masing-masing. Untuk hasil pengujian performa menunjukkan situs Jobstreet memiliki performa yang baik, sedangkan situs Glints menunjukkan server menolak semua *request* yang dikirim.

**Kata Kunci**— Pengujian GUI, Pengujian Performa, *Load Testing*, Situs Lowongan Kerja, Jobstreet, Glints, Katalon Studio, Jmeter.

## I. PENDAHULUAN

Perkembangan teknologi telah mempengaruhi berbagai bidang termasuk pada bidang rekrutmen. Peningkatan penggunaan internet dan perangkat digital dalam kegiatan sehari-hari semakin memudahkan masyarakat untuk mendapatkan informasi mengenai lowongan pekerjaan dari segala penjuru daerah. Standar perusahaan yang semakin tinggi dan persaingan yang semakin ketat seperti saat ini membuat sebagian besar pencari kerja memilih menggunakan situs lowongan kerja yang ada di internet untuk menelusuri berbagai jenis lowongan kerja yang sesuai dengan kriteria mereka. Menurut hasil survei yang dilakukan oleh Populix pada Mei 2023 dengan jumlah responden 1.014 orang menunjukkan bahwa sebanyak 66% responden menyatakan mendapatkan informasi lowongan kerja melalui situs pencari kerja [1]. Jobstreet dan Glints merupakan contoh dari situs lowongan kerja populer yang banyak digunakan masyarakat Indonesia dan negara lain. Situs lowongan kerja ini berbasis

*website* dengan dukungan antarmuka grafis (GUI) yang baik untuk memudahkan pengguna dalam berinteraksi dengan sistem.

GUI adalah antarmuka program computer yang bekerja sebagai sarana penghubung antara *user* dengan perangkat lunak [2]. GUI memanfaatkan kemampuan motoric *user* untuk memenuhi interaksi dasar antara program komputer dengan manusia seperti eksplorasi, pemanggilan perintah dan pencarian informasi. GUI memiliki beberapa jenis seperti tata letak *widget*, formulir, *hypertext*, *toolbar* dan menu system. GUI dapat dibedakan menjadi 2 yaitu GUI tradisional dan GUI modern. GUI tradisional dikenal dengan istilah *WIMP* (*Windows, Icons, Munes, Pointing devices*). Sedangkan, GUI modern menawarkan berbagai *widget* seperti tombol, kolom entri, dan *chooser* atau pemilih [3]. Setiap sistem memiliki rancangan antarmuka grafis (GUI) dengan kompleksitas yang berbeda-beda. Kompleksitas dari rancangan GUI mampu menimbulkan masalah *response time* dimana rancangan GUI dengan kompleksitas yang semakin tinggi maka durasi waktu respons akan menjadi lebih lama. Pengujian GUI penting dilakukan karena kemampuan setiap *widget* akan mempengaruhi kualitas semua elemen sistem. Pengujian GUI pada intinya akan mengevaluasi waktu respons dari setiap fungsi yang diuji. Kecepatan sistem dalam menangani permintaan pengguna bisa disebabkan oleh kesederhanaan elemen GUI atau antarmuka GUI telah dirancang dengan algoritma yang lebih canggih [4].

Selain kualitas GUI, kinerja situs web juga merupakan faktor krusial yang mempengaruhi pengalaman pengguna sehingga penting dilakukan pengujian performa untuk mengetahui bagaimana situs web dalam menangani beban kerja apakah sistem tetap responsif dan cepat diakses.

Metode pengujian yang dapat dilakukan untuk pengujian GUI adalah *Blackbox Testing*. *Blackbox testing* memastikan bagian-bagian sistem dapat menampilkan pesan kesalahan yang benar ketika terjadi kesalahan input. *Blackbox testing* merupakan pengujian yang mempertimbangkan nilai inputan pengguna dan mengabaikan mekanisme internal sistem [5]. Metode *blackbox testing* dapat dilakukan secara manual dan otomatis. Pengujian otomatis merupakan pengujian pada sistem secara otomatis dengan alat bantu untuk mengeksekusi skenario pengujian untuk memudahkan dan mempercepat pengujian [6]. Salah satu alat bantu yang dapat digunakan yaitu katalon studio. Katalon studio adalah perangkat lunak *open source* milik Katalon LCC yang membantu pengembang dan tester dalam melakukan organisasi, membuat, menjalankan dan mengatur pengujian otomatis secara efisien

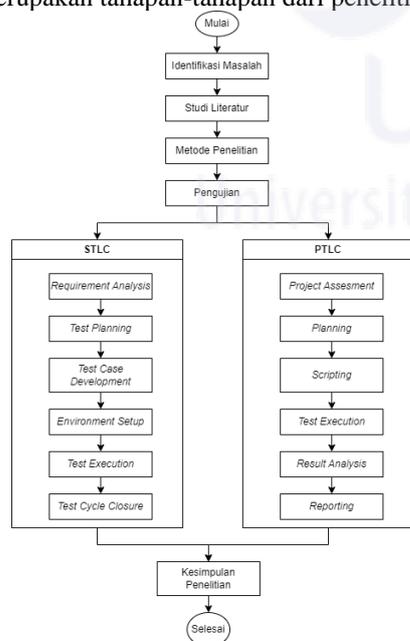
[7]. Dalam menjalankan pengujian perangkat lunak terdapat siklus yang sistematis dan terencana yang dapat diterapkan yaitu *Software Testing Life Cycle* (STLC). *Software testing life cycle* merupakan serangkaian langkah yang dilakukan untuk meningkatkan kualitas perangkat lunak. Siklus ini memiliki enam tahapan meliputi *requirement analysis*, *test planning*, *test case development*, *environment setup*, *test execution* dan *test cycle closure* [8].

Sedangkan, pengujian performa dapat dilakukan dengan teknik *load testing*. *Load testing* merupakan pengujian dimana respons perangkat lunak diukur dengan berbagai beban [9]. JMeter merupakan alat pengujian berbasis Java untuk pengujian beban [10]. Dalam menjalankan pengujian performa terdapat siklus terstruktur dan sistematis yang dapat digunakan yaitu *Performance Testing Life Cycle* (PTLC) dengan 6 fase yaitu *project assessment*, *planning*, *scripting*, *test execution*, *result analysis* dan *reporting* [11].

Jobstreet dan Glints merupakan situs web ketenagakerjaan *online* terpercaya di Asia Tenggara yang melayani jutaan pencari kerja dan ribuan perusahaan berbagai bidang industri. Dengan semakin populernya Jobstreet dan Glints dikalangan pengguna, kedua situs ini dihadapkan dengan tantangan untuk mempertahankan kualitas layanan. Oleh karena itu, diperlukan pengujian GUI dan pengujian performa untuk mengidentifikasi dan memperbaiki bug secara cepat, serta memastikan performa dan fungsionalitas situs tetap optimal di tengah peningkatan jumlah pengguna.

## II. METODOLOGI PENELITIAN

Penelitian ini menggunakan situs lowongan kerja Jobstreet dan Glints sebagai objek penelitian. Alat bantu pengujian otomatis yang digunakan yaitu Katalon Studio dengan jenis pengujian *blackbox testing*. Proses pengujian dilakukan dengan metodologi *software testing life cycle*. Gambar 1 Alur penelitian merupakan tahapan-tahapan dari penelitian ini.



Gbr.1 Alur Penelitian

### A. Identifikasi Masalah

Identifikasi masalah pada penelitian ini dilakukan peneliti dengan melakukan observasi pada objek penelitian yaitu situs lowongan kerja Jobstreet dan Glints untuk membantu merumuskan masalah dan menentukan tujuan penelitian.

### B. Studi Literatur

Studi literatur merupakan proses pengumpulan data dengan melakukan tinjauan pada literatur-literatur yang telah diterbitkan sebelumnya terkait topik pengujian otomatis GUI. Literatur yang ditinjau merupakan karya non fiksi seperti artikel ilmiah, buku, jurnal penelitian, skripsi dan situs di internet yang berkaitan dengan topik penelitian ini.

### C. Menentukan Metode Pengujian

Penelitian ini dilakukan dengan metode *Black Box Testing* dikarenakan peneliti tidak memiliki akses *source code* dan konfigurasi pada situs lowongan kerja Jobstreet dan Glints. Oleh karena itu, metode *Black Box Testing* dirasa metode pengujian yang cocok untuk penelitian ini

### D. Proses Pengujian

Proses pengujian fungsionalitas GUI pada penelitian ini menggunakan siklus pengujian *Software Testing Life Cycle* (STLC). STLC memiliki enam tahapan seperti berikut:

#### 1) Requirement Analysis

*Requirement analysis* merupakan tahapan menganalisis kebutuhan. Terdapat tiga kegiatan pada tahap ini yaitu *gathering data* atau pengumpulan data, analisis kebutuhan dan melakukan pemetaan *requirement traceability matrix*. Pada penelitian ini peneliti melakukan survei pengalaman pengguna untuk mengidentifikasi persyaratan apa saja yang dapat diuji berdasarkan sudut pandang pengguna. Survei ini berperan sebagai langkah awal proses pembentukan butir uji. Berdasarkan survei yang telah dilakukan diketahui bahwa fitur yang sering digunakan yaitu autentikasi (pendaftaran akun dan login), pengaturan profil, cari pekerjaan, dan lamar cepat.

#### 2) Test Planning

*Test planning* merupakan tahapan merumuskan perencanaan pengujian. Pada penelitian ini pengujian dilakukan secara otomatis dengan alat bantu Katalon Studio dengan iterasi pengujian sebanyak tiga kali untuk memastikan konsistensi dari fitur yang diuji. Parameter pengujian ini meliputi fungsionalitas (validasi form dan navigasi), performa (*response time*) dan keamanan (autentikasi dan validasi data)

#### 3) Test Case Development

*Test case development* merupakan tahapan pembuatan kasus uji atau *test case* sebagai bahan acuan untuk pengujian yang akan dilakukan. Kasus uji ini merupakan sekumpulan scenario pengujian untuk memastikan sistem yang diuji telah

memenuhi persyaratan dan kebutuhan pengguna. Kasus uji pada pengujian GUI situs lowongan kerja Jobstreet dan Glints ini terdiri dari 2 jenis kasus uji yaitu kasus uji valid dan invalid. Berikut ini beberapa daftar kasus uji pengujian GUI situs Jobstreet pada penelitian ini.

TABEL I  
 TEST CASE JOBSTREET

Test Case ID	Type	Test Scenario	Expected Result
TC_06	Valid	Melakukan login dengan data yang valid	Login berhasil dan menampilkan halaman profil.
TC_17	Invalid	Menambahkan pendidikan dengan mengosongkan field kursus atau kualifikasi	Menampilkan peringatan "Silakan tambahkan Kursus atau Kualifikasi"
TC_24	Invalid	Melakukan pencarian lowongan kerja dengan kata kunci yang tidak valid	Menampilkan pemberitahuan "Tidak ada hasil pencarian yang sesuai"

Sedangkan, beberapa daftar kasus uji pengujian GUI situs Glints dapat dilihat pada daftar kasus uji berikut ini.

TABEL II  
 TEST CASE GLINTS

Test Case ID	Type	Test Scenario	Expected Result
TC_01	Valid	Melakukan pendaftaran akun dengan data yang valid	Pendaftaran akun berhasil dan menampilkan halaman beranda.
TC_14	Invalid	Menambahkan pengalaman kerja tetapi mengosongkan posisi pekerjaan	Button Simpan tidak dapat ditekan.
TC_25	Valid	Melakukan lamaran cepat dengan data yang valid	Berhasil mengirim lamaran

#### 4) Environment Setup

Environment Setup merupakan tahapan persiapan lingkungan pengujian. Pada tahap ini terdapat tiga kegiatan utama yaitu environment setup, konfigurasi perangkat lunak

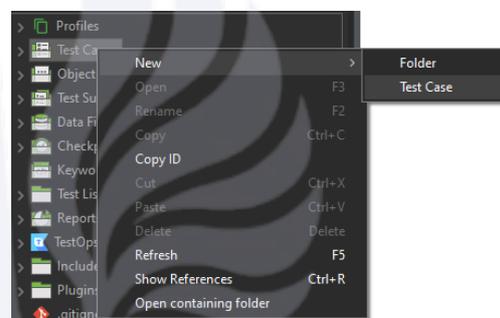
dan perangkat keras dan melakukan pengujian *smoke* atau pengujian pengujian yang dilakukan untuk memastikan kesiapan dari fitur system yang akan diuji sebelum dijalankan pengujian dengan *script* atau pengujian otomatis. Lingkungan pengujian yang disiapkan pada penelitian pengujian otomatis GUI ini meliputi katalon studio, JDK versi 8 dan *web browser* Google Chrome.

#### 5) Test Execution

Test Execution merupakan tahapan pelaksanaan pengujian perangkat lunak berdasarkan *test plan* dan *test case* yang telah dirancang sebelumnya. Berikut ini langkah-langkah pengujian otomatis GUI dengan menggunakan alat pengujian otomatis Katalon Studio.

##### a) Membuat Test Case di Katalon Studio

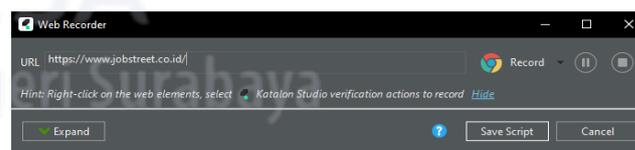
Langkah pertama yaitu membuat projek baru di katalon studio terlebih dahulu, kemudian buat *test case* dengan cara tekan kanan pada folder Test Cases lalu pilih New dan pilih Test Case. Selanjutnya beri nama dan deskripsi *test case*.



Gbr.2 Buat Test Case

##### b) Melakukan Record Web

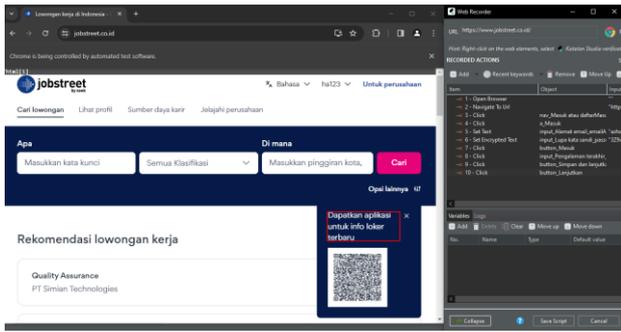
Langkah selanjutnya yaitu merekam *object* atau elemen dari setiap langkah pengujian dengan cara tekan *icon record web* kemudian input URL dari situs lowongan kerja yang akan diuji lalu tekan *button* Record.



Gbr.3 Melakukan Record Web

##### c) Simpan Elemen pada Object Repository

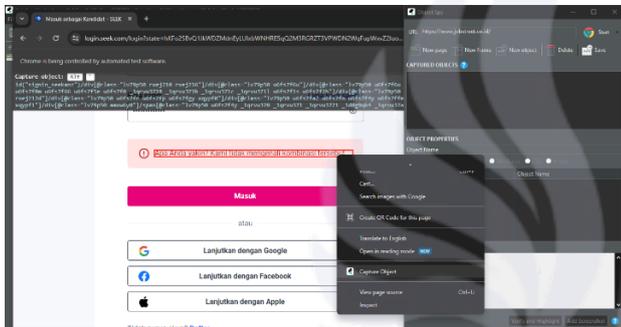
Setelah semua *object* / elemen dari setiap langkah kasus uji terekam kemudian tekan tombol *Save Script* untuk menyimpan semua *object* / elemen pada folder *Object Repository*.



Gbr.4 Simpan Elemen di Object Repository

d) Melakukan *Spy Web*

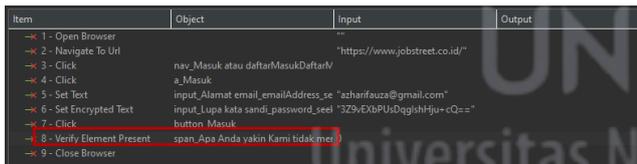
Mengingat terdapat kasus uji dengan memasukkan invalid data. Maka, untuk memvalidasi apakah situs lowongan kerja tersebut menampilkan pesan kesalahan maka dilakukan dengan *spy web* untuk merekam *object* / elemen dari pesan kesalahan tersebut.



Gbr.5 *Spy Web*

e) Melakukan Verifikasi Data

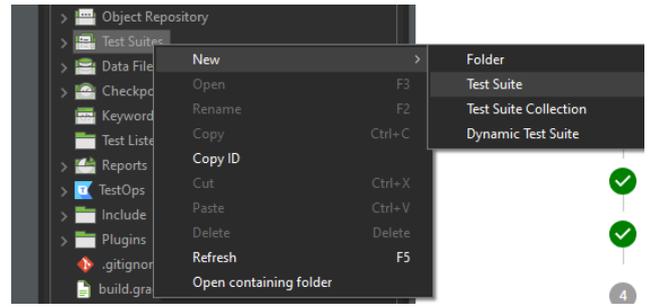
Selanjutnya verifikasi data pada *test case* dengan menambahkan item *verify element present* / *verify element text* pada *object* / elemen dari pesan kesalahan.



Gbr.6 Verifikasi Data

f) Membuat *Test Suite*

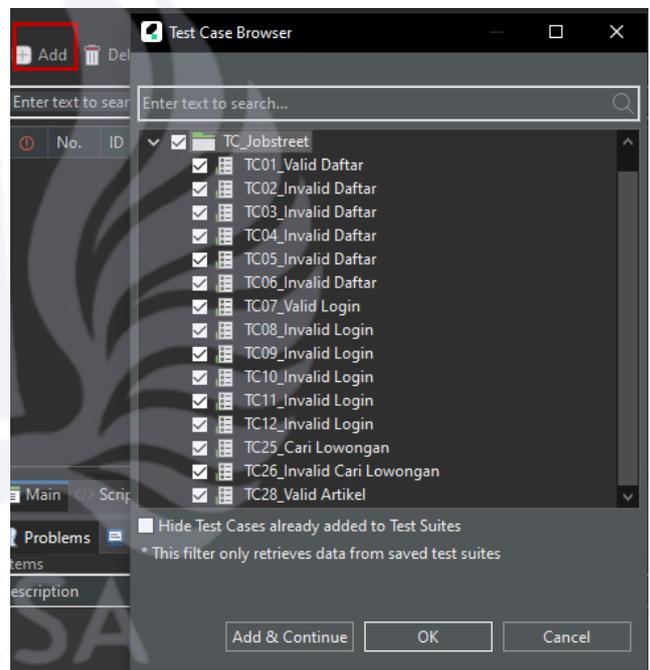
Langkah pertama untuk pembuatan *test suite* yaitu dengan klik kanan pada *Test Suites* kemudian pilih *Test Suite*. Selanjutnya beri nama dan deskripsi mengenai *test suite*.



Gbr.7 Buat *Test Suite*

g) Menambahkan *Test Case* pada *Test Suite*

Untuk menambahkan *test case* pada *test suite* dapat dilakukan dengan cara klik *add* kemudian pilih *test case* yang ingin ditambahkan kemudian klik *button OK*. Selanjutnya untuk menjalankan eksekusi pengujian dapat menekan *button Run*.



Gbr.8 Menambahkan *Tes Case* ke *Test Suite*

6) *Test Cycle Closure*

*Test Cycle Closure* merupakan tahapan mengidentifikasi dan menganalisis dari hasil pengujian yang dihasilkan oleh katalon studio untuk mengevaluasi setiap tahapan STLC yang telah dilaksanakan.

Sedangkan, pengujian performa pada penelitian ini menggunakan siklus pengujian *Performance Testing Life Cycle* dengan tahapan seperti berikut.

1) *Project Assessment*

*Project Assessment* merupakan tahapan mengumpulkan informasi mengenai situs lowongan Jobstreet dan Glints.

Informasi yang didapatkan akan digunakan sebagai bahan untuk melakukan pengujian kinerja.

2) *Planning*

*Planning* merupakan tahapan merencanakan pengujian performa yang akan dilakukan. Jenis pengujian performa pada penelitian ini yaitu *load testing*. Perangkat yang digunakan yaitu Apache JMeter. Fitur yang akan diuji yaitu halaman utama, halaman daftar, halaman login, lihat profil dan cari lowongan. Pengujian *load testing* pada situs Jobstreet akan disimulasikan pada kondisi normal dan kondisi puncak dengan tiga *ramp-up period* yaitu 600s untuk memastikan sistem mampu menangani terhadap peningkatan beban yang relatif cepat, 900s untuk memastikan stabilitas sistem terhadap peningkatan yang relatif stabil dan 1200s untuk memastikan kinerja sistem terhadap peningkatan beban yang sangat lambat. Untuk skenario pengujian *load testing* pada situs Jobstreet dapat dilihat pada table berikut.

TABEL III  
 SKENARIO LOAD TESTING JOBSTREET

Skenario	Number of Threads	Ramp – up period (s)
Kondisi Normal	528	600
		900
		1200
Kondisi Puncak	1056	600
		900
		1200

Sedangkan, Pengujian *load testing* pada situs Glints akan disimulasikan pada kondisi normal dan kondisi puncak dengan tiga *ramp-up period* yaitu 300s untuk memastikan sistem mampu menangani terhadap peningkatan beban yang relatif cepat, 600s untuk memastikan stabilitas sistem terhadap peningkatan yang relatif stabil dan 900s untuk memastikan kinerja sistem terhadap peningkatan beban yang sangat lambat.

TABEL IV  
 SKENARIO LOAD TESTING GLINTS

Skenario	Number of Threads	Ramp – up period (s)
Kondisi Normal	440	300
		600
		900
Kondisi Puncak	880	300
		600
		900

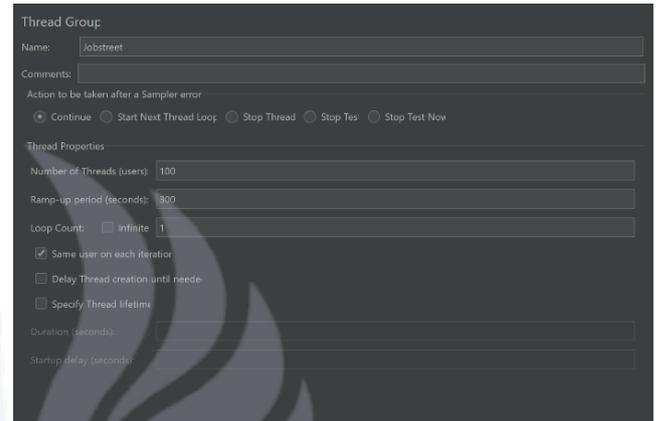
3) *Scripting*

*Scripting* merupakan tahapan penulisan menuliskan langkah pengujian. *Scripting* dapat dilakukan secara manual dengan menuliskan HTTP request secara langsung pada

JMeter atau dapat secara otomatis dengan menggunakan Blazemeter untuk merekam HTTP request.

4) *Test Execution*

*Test Execution* merupakan tahapan menjalankan skenario pengujian performa. Sebelum menjalankan pengujian, penguji dapat mengatur parameter request berupa *number of thread (user)*, *ramp-up period (s)*, dan jumlah perulangan pada *Thread Group* seperti gambar berikut.



Gbr.9 Thread Group

5) *Result Analysis*

*Result analysis* merupakan tahapan menganalisis hasil pengujian performa untuk menentukan nilai performansi dari kedua situs yang diuji. Proses analisis ini dilakukan dengan menganalisis pada *summary report* yang dihasilkan JMeter.

6) *Reporting*

*Reporting* merupakan tahapan akhir dari pengujian performa setelah proses analisis hasil pengujian dan dibuat sebuah laporan hasil pengujian performa.

E. *Kesimpulan*

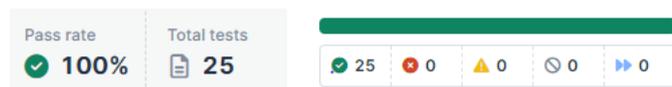
Kesimpulan merupakan tahapan akhir dari penelitian ini yang berisi ringkasan hasil yang dicapai dari penelitian ini dan saran baik untuk perbaikan system maupun untuk penelitian selanjutnya.

III. HASIL DAN PEMBAHASAN

A. *Hasil Pengujian GUI*

Hasil pengujian yang akan dibahas ini berdasarkan hasil eksekusi yang dilakukan sebanyak tiga kali *run test*. Berikut ini merupakan hasil dari ketiga *run test* pengujian GUI situs lowongan kerja Jobstreet dan Glints.

1) Hasil Pengujian Pertama



Gbr.10 Hasil Pengujian Pertama Jobstreet

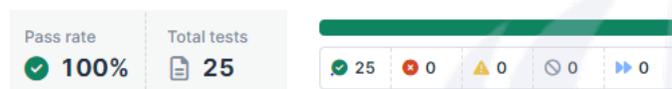
Hasil pengujian pertama situs Jobstreet menunjukkan status keberhasilan *Passed* dan persentase 100% dengan jumlah *test case* 25.



Gbr.11 Hasil Pengujian Pertama Glints

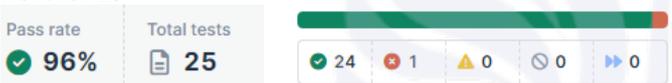
Hasil pengujian pertama situs Glints menunjukkan status keberhasilan *Passed* dan persentase 100% dengan jumlah *test case* 25.

## 2) Hasil Pengujian Kedua



Gbr.12 Hasil Pengujian Kedua Jobstreet

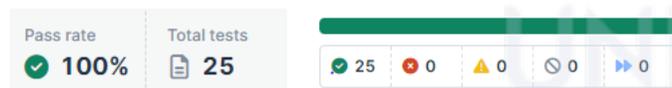
Hasil pengujian kedua situs Jobstreet menunjukkan status keberhasilan *Passed* dan persentase 100% dengan jumlah *test case* 25.



Gbr.13 Hasil Pengujian Kedua Glints

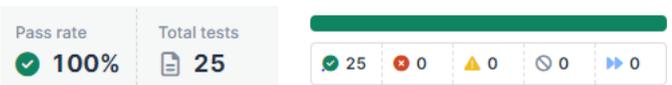
Hasil pengujian kedua situs Glints menunjukkan status keberhasilan *Failed* dan persentase 96% dengan jumlah *test case* 25. Pada pengujian ini terjadi *error* pada kasus uji TC14\_Pengalaman Karier karena terdapat elemen yang tidak dapat berinteraksi saat pengujian.

## 3) Hasil Pengujian Ketiga



Gbr.14 Hasil Pengujian Ketiga Jobstreet

Hasil pengujian ketiga situs Jobstreet menunjukkan status keberhasilan *Passed* dan persentase 100% dengan jumlah *test case* 25.



Gbr.15 Hasil Pengujian Ketiga Glints

Hasil pengujian ketiga situs Glints menunjukkan status keberhasilan *Passed* dan persentase 100% dengan jumlah *test case* 25. Pada pengujian ketiga ini masalah yang ditemukan pada pengujian kedua telah teratasi.

## B. Analisis Hasil Pengujian GUI

Pengujian GUI untuk situs lowongan kerja Jobstreet dan Glints menggunakan Katalon Studio yang telah dilaksanakan memiliki fokus pada tiga parameter utama yaitu fungsionalitas, *performance* dan keamanan. Berikut ini merupakan analisis hasil pengujian berdasarkan parameter:

### a. Fungsionalitas

Pada situs lowongan kerja Jobstreet semua *button*, menu, dan formulir berfungsi dengan baik tanpa error dalam validasi dan semua link dan *button* navigasi mengarahkan ke halaman yang semestinya. Tidak ada *test case* yang gagal menunjukkan bahwa konsistensi dan kualitas dari fungsionalitas GUI sudah berfungsi dengan baik dan memiliki navigasi yang jelas.

Untuk fungsionalitas dari situs lowongan kerja Glints semua form baik pendaftaran akun dan login maupun form pada profil berfungsi dengan baik dalam validasi. Namun, terdapat *button* yang tidak dapat berinteraksi saat dilakukan pengujian pada *test case* 14. Kegagalan tersebut dapat mempengaruhi navigasi antar halaman. Namun, kegagalan tersebut dapat diatasi dengan menambahkan langkah pengujian dengan metode "*waitForElementVisible*" untuk menunggu hingga elemen terlihat pada layar.

### b. Performance

Pada situs lowongan kerja Jobstreet memiliki rata-rata *response time* 19,2 detik untuk halaman pendaftaran akun, 19,5 detik untuk halaman login, 40,3 detik untuk halaman profil, 14,7 detik untuk halaman cari lowongan kerja dan 43 detik untuk halaman lamar cepat. Performa pada situs lowongan kerja Jobstreet menunjukkan stabilitas dan kecepatan yang baik.

Sedangkan, untuk *performance* situs lowongan kerja Glints memiliki rata-rata *response time* 34,7 detik untuk halaman pendaftaran akun, 18,4 detik untuk halaman login, 50,3 detik untuk halaman profil, 27,2 detik untuk halaman cari lowongan kerja dan 34,7 untuk halaman lamar cepat. Performa dari situs lowongan kerja Glints memiliki *response time* yang sedikit lebih tinggi dibandingkan Jobstreet dikarenakan memiliki fungsionalitas yang sedikit lebih kompleks.

### c. Keamanan

Pada situs lowongan kerja Jobstreet dan Glints *test case* login dan pendaftaran akun lulus tanpa masalah karena mampu membaca alamat email dan password yang dimasukkan. Hal ini menunjukkan bahwa mekanisme autentikasi dan otorisasi dasar bekerja dengan baik dan responsif

## C. Hasil Requirement Traceability Matrix

Setelah tahap pengujian telah selesai dilakukan maka salah satu hasil dari pengujian yaitu *requirement traceability matrix*.

Berdasarkan hasil uji sebelumnya, ditemukan hasil dari Jobstreet seperti berikut:

- a. Aspek *Security* : 100% dari 3 kali pengujian dengan 10 *test case* yang berbeda.
- b. Aspek *Data Integrity* : 100% *Pass* dari 3 kali pengujian dengan 11 *test case* yang berbeda
- c. Aspek *Functional* : 100% *Pass* dari 3 kali pengujian dengan 3 *test case* yang berbeda.
- d. Aspek *Recovery* : 100% *Pass* dari 3 kali pengujian dengan 1 *test case*.

- a. Aspek *Security* : 100% dari 3 kali pengujian dengan 10 *test case* yang berbeda.
- b. Aspek *Data Integrity* : 96% *Pass* dari 3 kali pengujian dengan 11 *test case* yang berbeda
- c. Aspek *Functional* : 100% *Pass* dari 3 kali pengujian dengan 3 *test case* yang berbeda.
- d. Aspek *Recovery* : 100% *Pass* dari 3 kali pengujian dengan 1 *test case*.

TABEL V  
 REQUIREMENT TRACEABILITY MATRIX JOBSTREET

Require ment Traceability	Business Requirement			
	BR1	BR2	BR3	BR4
	Autentikasi	Pengaturan Profil	Cari Lowongan	Lamar Pekerjaan
TC01	✓			
TC02	✓			
TC03	✓			
TC04	✓			
TC05	✓			
TC06	✓			
TC07	✓			
TC08	✓			
TC09	✓			
TC10	✓			
TC11		✓		
TC12		✓		
TC13		✓		
TC14		✓		
TC15		✓		
TC16		✓		
TC17		✓		
TC18		✓		
TC19		✓		
TC20		✓		
TC21		✓		
TC22			✓	
TC23			✓	
TC24				✓
TC25		✓		

- SSTD – Jobstreet Tabel RTM *Security*
- SSTD – Jobstreet Tabel RTM *Data Integrity*
- SSTD – Jobstreet Tabel RTM *Functional*
- SSTD – Jobstreet Tabel RTM *Recovery*

Sedangkan, untuk hasil *requirement traceability matrix* dari hasil pengujian Glints yang telah dilakukan sebelumnya dapat dilihat di bawah ini:

TABEL VI  
 Requirement Traceability Matrix Glints

Require ment Traceability	Business Requirement			
	BR1	BR2	BR3	BR4
	Autentikasi	Pengaturan Profil	Cari Lowongan	Lamar Pekerjaan
TC01	✓			
TC02	✓			
TC03	✓			
TC04	✓			
TC05	✓			
TC06	✓			
TC07	✓			
TC08	✓			
TC09	✓			
TC10	✓			
TC11		✓		
TC12		✓		
TC13		✓		
TC14		✓		
TC15		✓		
TC16		✓		
TC17		✓		
TC18		✓		
TC19		✓		
TC20		✓		
TC21		✓		
TC22			✓	
TC23			✓	
TC24				✓
TC25		✓		

- SSTD – Glints Tabel RTM *Security*
- SSTD – Glints Tabel RTM *Data Integrity*
- SSTD – Glints Tabel RTM *Functional*
- SSTD – Glints Tabel RTM *Recovery*

D. Hasil Load Testing

Berikut ini hasil dari pengujian *load testing* yang telah dilakukan dengan tiga kali *run test* dengan dua scenario pengujian yaitu saat kondisi normal dan puncak.

TABEL VII  
 HASIL LOAD TESTING

Skenario	Jobstreet			Glints		
	Response time (s)	Throughput (req/min)	Error Rate (%)	Response Time (s)	Throughput (req/min)	Error Rate (%)
Halaman Utama (Kondisi Normal)	599	0.88	0	231	1.5	100
	1116	0.59	0	124	44.1	100
	435	0.44	0	180	29.4	100
Halaman Utama (Kondisi Puncak)	428	1.8	0	220	2.9	100
	419	1.2	0	145	1.5	100
	425	0.88	0	129	58.7	100
Cari Lowongan (Kondisi Normal)	784	0.88	0	223	1.5	100
	1361	0.59	0	116	44.1	100
	643	0.44	0	174	29.4	100
Cari Lowongan (Kondisi Puncak)	649	1.8	0	214	2.9	100
	631	1.2	0	139	1.5	100
	642	0.88	0	122	58.7	100
Detail Pekerjaan (Kondisi Normal)	751	0.88	0	223	1.5	100
	1235	0.59	0	115	44.1	100
	608	0.44	0	174	29.4	100
Detail Pekerjaan (Kondisi Puncak)	619	1.8	0	214	2.9	100
	598	1.2	0	139	1.5	100
	617	0.88	0	121	58.7	100
Profil (Kondisi Normal)	442	0.88	0	222	1.5	100
	578	0.59	0	115	44.1	100
	279	0.44	0	173	29.4	100
Profil (Kondisi Puncak)	394	1.8	0	214	2.9	100
	410	1.2	0	138	1.5	100
	273	0.88	0	121	58.8	100
Halaman Daftar (Kondisi Normal)	723	0.88	0	223	1.5	100
	996	0.59	0	114	44.1	100
	538	0.44	0	173	29.4	100
Halaman	606	1.8	0	214	2.9	100

Skenario	Jobstreet			Glints		
	Response time (s)	Throughput (req/min)	Error Rate (%)	Response Time (s)	Throughput (req/min)	Error Rate (%)
Daftar (Kondisi Puncak)	620	1.2	0	139	1.5	100
	536	0.88	0	121	58.8	100
Halaman Login (Kondisi Normal)	593	0.88	0	222	1.5	100
	777	0.59	0	115	44.1	100
Halaman Login (Kondisi Puncak)	461	0.44	0	174	29.4	100
	524	1.8	0	215	2.9	100
	529	1.2	0	138	1.5	100
	462	0.88	0	121	58.8	100

E. Analisis Hasil Load Testing

Berdasarkan hasil pengujian *Load Testing* setiap halaman situs lowongan kerja Jobstreet dan Glints yang dilakukan dengan JMeter, berikut ini merupakan analisis dari setiap indikator pengujian:

a. Response Time

Menurut (Suwarsono et al., 2022) *response time* dikategorikan cukup baik apabila berada diantara 1 sampai 30 detik. Dari hasil *load testing* yang telah dilakukan nilai *average response time* situs lowongan kerja Jobstreet berada di bawah 30 detik sehingga dapat dikategorikan cukup baik. Namun, nilai *average response time* mengalami penurunan seiring dengan bertambahnya *number of thread (user)*. Hal tersebut membuktikan bahwa situs Jobstreet memiliki optimisasi yang baik.

Sedangkan, nilai *response time* dari situs lowongan kerja Glints yang ditampilkan tidak berarti dikarenakan tidak ada respons yang diterima untuk dihitung dan nilai *response time* yang rendah menunjukkan bahwa *request* yang dikirim gagal sebelum server dapat menghitung *response time*.

b. Throughput

Nilai *throughput* situs lowongan kerja Jobstreet yang ditunjukkan pada hasil pengujian cenderung mengalami peningkatan seiring dengan bertambahnya *user* dan berbanding terbalik dengan nilai *response time* membuktikan server situs Jobstreet bekerja dengan sangat baik dan dapat menangani lebih banyak permintaan tanpa harus meningkatkan *response time*.

Sedangkan, pada situs lowongan kerja Glints menunjukkan tidak ada *throughput* yang dapat dihitung karena *request* yang dikirim tidak ada yang berhasil sehingga *throughput* yang ditampilkan tidak memiliki arti.

c. Error Rate

Nilai *error rate* dari situs lowongan kerja Jobstreet menunjukkan 0.00% yang berarti *endpoint* dari setiap halaman fitur mampu mengatasi *request* dengan baik tanpa kesalahan.

Sedangkan, pengujian *load testing* pada situs lowongan kerja Glints mengalami *error rate* sebesar 100% pada semua fitur. Setiap *request* yang dikirim menghasilkan *response code* 403 dan *response message Forbidden*. Masalah tersebut menunjukkan bahwa *request* yang dikirim oleh JMeter ke server ditolak. Hal ini dapat terjadi dikarenakan situs web tidak memberikan izin alat pengujian performa untuk mengakses sumber daya yang diminta.

#### IV. KESIMPULAN DAN SARAN

Berdasarkan hasil penelitian dan pembahasan, terdapat beberapa hal yang dapat menjadi kesimpulan seperti berikut:

- 1) penggunaan Katalon Studio sebagai alat pengujian otomatis GUI efektif dalam melakukan validasi data seperti mampu menemukan kesalahan dan menampilkan pesan kesalahan saat data yang dimasukkan tidak sesuai. Hasil pengujian GUI pada kedua situs lowongan kerja menunjukkan performa yang baik pada parameter fungsionalitas, *performance* dan keamanan. Meskipun ditemukan elemen yang tidak berinteraksi pada *test case* 14 pengujian kedua Glints. Namun, masalah tersebut dapat teratasi.
- 2) Hasil *load testing* pada Jobstreet menunjukkan performa yang baik dengan *error rate* yang dihasilkan 0%, *response time* yang baik dan *throughput* yang meningkat seiring dengan bertambahnya *user*. Sedangkan, hasil *load testing* pada Glints menunjukkan server menolak semua *request* yang dikirim. Hal ini dibuktikan dengan *response code* 403 dan *response message Forbidden* sehingga nilai *response time* yang ditampilkan tidak valid dan nilai *throughput* tidak berarti.

Saran yang dapat penulis berikan untuk penelitian selanjutnya yaitu pengujian dapat melakukan pengujian GUI tidak hanya berbasis web tetapi juga dapat menguji berbasis API atau Mobile App. Pengujian juga dapat melakukan pengujian dengan menggunakan *software testing* selain Katalon Studio seperti Selenium, Serenity, Cypress dan lain

sebagainya. Selain itu, pengujian dapat menggunakan web browser selain Google Chrome seperti Mozilla Firefox, Safari, Microsoft Edge dan lain sebagainya saat pengujian. Meskipun hasil *load testing* pada Jobstreet menunjukkan hasil yang baik, perlu dilakukan pemantauan secara berkala untuk memastikan situs dapat menangani peningkatan jumlah pengguna di waktu yang akan datang.

#### REFERENSI

- [1] "Ini 9 Situs Lowongan Kerja yang Paling Banyak Digunakan Masyarakat Indonesia - GoodStats." <https://goodstats.id/article/ini-9-situs-lowongan-kerja-yang-paling-banyak-digunakan-masyarakat-indonesia-tSQyA> (accessed Jul. 20, 2024).
- [2] M. M. Muhtadi, M. D. Friyadi, and A. Rahmani, "Analisis GUI Testing pada Aplikasi E-Commerce menggunakan Katalon," *Pros. Ind. Res. Work. Natl. Semin.*, vol. 10, no. 1, pp. 1387–1393, 2019.
- [3] A. Oulasvirta, N. R. Dayama, M. Shiripour, M. John, and A. Karrenbauer, "Combinatorial Optimization of Graphical User Interface Designs," *Proc. IEEE*, vol. 108, no. 3, pp. 434–464, 2020, doi: 10.1109/JPROC.2020.2969687.
- [4] A. Indrayanti, B. A. Wardijono, D. Nur, and R. Aulia, "Analisis Pengujian Graphical User Interface E-Commerce Dengan Menggunakan Katalon Studio," *Semin. Nas. Teknol. Inf. dan Komun. STI&K*, vol. 5, no. 1, 2021.
- [5] Supriyono, "Software Testing with the approach of Blackbox Testing on the Academic Information System," *Int. J. Inf. Syst. Technol.*, vol. 3, no. 36, pp. 227–233, 2020.
- [6] U. Sa'adah et al., "Framework Testing Otomatis Berbasis Serenity dan Jenkins," *JUTI J. Ilm. Teknol. Inf.*, vol. 19, no. 2, p. 102, 2021, [Online]. Available: <http://juti.if.its.ac.id/index.php/juti/article/view/1017/460>.
- [7] U. Nugraha, S. Atikah Nurduha Robaiah, and D. Rospinoedji, "TESTING THE INFORMATION SYSTEM SOFTWARE USING BEHAVIOR DRIVEN DEVELOPMENT METHOD Ucu Nugraha, Siti Atikah Nurduha Robaiah, Djoko Rospinoedji. Testing The Information System Software Using Behavior Driven Development Method-Palarch's Journal Of Archaeology," vol. 17, no. 10, pp. 3732–3742, 2020.
- [8] 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, doi: 10.47134/jacis.v2i1.42.
- [9] D. I. Permatasari, "Pengujian Aplikasi menggunakan metode Load Testing dengan Apache JMeter pada Sistem Informasi Pertanian," *J. Sist. dan Teknol. Inf.*, vol. 8, no. 1, p. 135, 2020, doi: 10.26418/justin.v8i1.34452.
- [10] Y. Pryayoga, M. R. Prasetya, A. D. Irawan, T. Informatika, U. Pamulang, and T. Selatan, "OTOMATISASI PENGUJIAN PERFORMA APLIKASI E-COMMERCE," vol. 2, no. 2, pp. 1585–1596, 2024.
- [11] J. du Plessis, "White Paper Performance Testing Methodology," 2008.