

PENGUJIAN WEBSITE INVITEES MENGGUNAKAN METODE LOAD TESTING DENGAN APACHE JMETER

Oleh:

Wanta Tejaya¹, Syaiful Rahman^{2*}, Abdul Munir³

^{1,2,3}Teknik Informatika, STMIK KHARISMA Makassar

e-mail: ¹wantatejaya_19@kharisma.ac.id, ²syaifulrahman@kharisma.ac.id,

³abdulmunir@kharisma.ac.id

Abstrak: Tujuan penelitian ini adalah untuk melakukan uji performa menggunakan metode load testing dengan tools Apache Jmeter pada website Invitees dan seberapa optimal waktu muat pada fungsi utama website invitees. Teknik pengumpulan data yang digunakan yaitu studi dokumen yang berisi statistik jumlah pengunjung dan user Invitees periode bulan Januari sampai Juni 2022. Teknik analisis data yang digunakan yaitu analisis deskriptif. Waktu muat yang diperoleh dari hasil pengujian kemudian dibandingkan dengan target pengujian untuk mengetahui seberapa optimal waktu muat dari website Invitees. Dari pengujian, diperoleh hasil waktu muat rata-rata website Invitees pada saat layanan digunakan yang diwakili melalui skenario pertama yakni 2.7 detik dan kedua yakni 1.7 detik telah memenuhi target pengujian yaitu di bawah 3 detik. Adapun hasil berdasarkan skenario ketiga yang mewakili kondisi layanan digunakan dengan beban user dua kali lebih banyak, didapatkan waktu muat rata-rata sebesar 4.2 detik yang tidak memenuhi target pengujian. Sehingga dapat disimpulkan bahwa website Invitees telah bekerja secara optimal ketika layanannya digunakan tetapi masih perlu dilakukannya perbaikan jika layanan digunakan oleh user yang berjumlah lebih banyak dari 10 user secara bersamaan.

Kata kunci: Pengujian Beban, Situs Web, Apache Jmeter, Waktu Muat, Aplikasi Situs.

Abstract: The purpose of this research is to conduct a performance test using the load testing method with Apache Jmeter tools on the Invitees website and how optimal the load time is on the main function of the invitees website. The data collection technique used is document study containing statistics on the number of visitors and users of Invitees for the period January to June 2022. The data analysis technique used is descriptive analysis. The load time obtained from the test results is then compared with the test target to find out how optimal the load time of the Invitees website is. From the test, it was obtained that the average load time of the Invitees website when the service was used, represented by the first scenario, namely 2.7 seconds and the second, namely 1.7 seconds, had met the test target, which was under 3 seconds. As for the results based on the third scenario which represents the condition of the service used with twice as much user load, the average load time is 4.2 seconds which does not meet the test target. So it can be concluded that the Invitees website has worked optimally when the service is used but still needs to be improved if the service is used by more than 10 users simultaneously.

Keywords: Load Testing, Website, Apache Jmeter, Load Time, Website Application.

1. PENDAHULUAN

Teknologi yang berkembang pesat di era digital ini, membuat teknologi mulai diterapkan ke berbagai aspek kehidupan sehari-hari contohnya yaitu website yang tentunya dapat

* Corresponding author : Syaiful Rahman (syaifulrahman@kharisma.ac.id)

membantu mengembangkan usaha. Tentu dengan semakin banyaknya *website* yang ada, membuat persaingan market sangat sengit. Salah satu faktor penentu dari keberhasilan suatu *website* untuk dapat bersaing dengan banyaknya kompetitor lainnya yaitu kecepatan muat dari *website* tersebut [1]. Salah satu cara untuk mengukur kecepatan muat dari website adalah dengan melakukan pengujian performa.

Pengujian performa adalah pengujian yang dilakukan untuk mengetahui kondisi dari perangkat lunak ketika dalam kondisi beban kerja tertentu. Pengujian performa terbagi 3 yaitu *Stress test*, *Load test*, dan *Strength test* [2]. *Stress test* adalah pengujian yang dilakukan dengan mensimulasikan penggunaan website dengan jumlah user yang terus bertambah sehingga menyebabkan beban yang berat pada sistem sehingga sistem tidak dapat menyanggupi beban tersebut. Sedangkan *Load test* merupakan pengujian yang dilakukan dengan menambahkan beban secara bertahap untuk melihat perubahan yang terjadi pada sistem. Adapun *Strength test* adalah *load test* atau *stress test* yang dilakukan lebih lama. Tidak seperti *load test* atau *stress test* yang berinterval 10 detik, *strength test* dilakukan dengan interval perjam atau bahkan perhari.

Metode *Load test* lebih cocok digunakan dalam pengujian ini dibandingkan metode lainnya, karena metode load test akan memperlihatkan bagaimana sistem dapat bekerja dalam kondisi beban kerja yang beragam dan bertahap. Pada penelitian ini kondisi beban kerja yang beragam dan bertahap yang digunakan yaitu jumlah user yang mengakses website. Apache Jmeter merupakan perangkat lunak *desktop open source* yang berbasis java yang digunakan dalam melakukan pengujian perilaku fungsional dan kinerja dari aplikasi klien/server seperti aplikasi web, aplikasi FTP dan *database server* [3]. Aplikasi ini menghitung waktu respon, beban CPU penggunaan memori serta berbagai sumber daya lainnya. Alasan digunakannya Apache Jmeter pada penelitian ini selain merupakan aplikasi *open source*, Apache Jmeter mempunyai Tampilan *User Interface* yang mudah digunakan, dapat menguji beban dan performa pada berbagai tipe aplikasi/server/protokol, memungkinkan pengambilan sampel secara bersamaan dari berbagai fungsi pengujian yang dilakukan, didesain secara spesifik untuk melakukan pengujian pada *web applications* yang cocok untuk website invitees yang merupakan *web application* serta bisa dijalankan pada *platform* apapun [3][4][5].

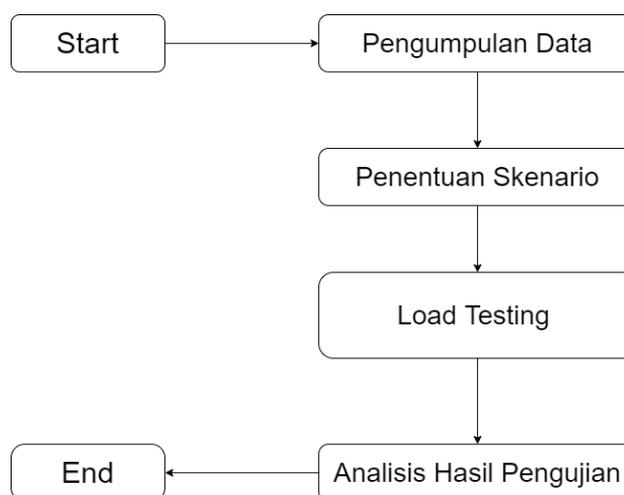
Invitees (<https://invitees.site/>) merupakan *website* yang menyediakan pelayanan jasa pembuatan undangan berbasis web yang dimana client cukup menshare *link* dari web yang berisi undangan *online* yang juga merangkap proses registrasi pendataan tamu undangan. Tentu dalam pelayanan jasa, khususnya berbasis *web* atau *online* maka waktu muat (*Load Time*) dari *website* merupakan hal yang penting. Waktu muat atau *load time* adalah waktu yang dibutuhkan untuk mengunduh dan menampilkan semua konten halaman web pada browser. Pengujian ini dilakukan karena pada undangan *website* Invitees terdapat banyak gambar dari client yang akan dimuat terutama barcode untuk proses registrasi acara yang dimana hal tersebut merupakan salah satu faktor berupa konten yang mempengaruhi performa khususnya

waktu muat atau *load time* dari *website* Invitees [6][7]. Maka untuk mengetahui pengaruh faktor tersebut perlu dilakukan pengujian.

Adapun penelitian Load Testing yang sudah dilakukan oleh beberapa peneliti sebelumnya yaitu, Observasi Traffic dan load testing menggunakan Similarweb dan Loadimpact [8], Pengujian Performa dan Tingkat Stress pada Website Legalisir Ijasah Online Universitas Pendidikan Ganesha [9] yang menggunakan tools yang berbeda, kemudian Pengujian Aplikasi Greenwallet Dengan Metode Load Testing Dan Apache Jmeter [10] yang membandingkan hasil pengujian dari web hosting dan localhost, Pengukuran Throughput Load Testing Menggunakan Test Case Sampling Gorilla Testing [11], dan Pengujian Aplikasi Menggunakan Metode *Load Testing* dengan Apache Jmeter pada Sistem Informasi Pertanian [12] yang juga mengukur *process time* pada pengujian. Tujuan dilakukannya penelitian ini yaitu untuk mengetahui bagaimana melakukan pengujian beban atau load testing pada *website* Invitees dan seberapa optimal load time dari *website* invitees.

2. METODE PENELITIAN

Dalam pelaksanaan penelitian ini adapun tahapan atau alur dari penelitiannya dapat dilihat pada Gambar 1 di bawah ini.



Gambar 1. Alur Tahapan penelitian

Penelitian dimulai dari tahapan pengumpulan data jumlah user atau visitor dari *website* invitees yang terdapat pada statistik data pengunjung user pada *server hosting website*, lalu dilanjutkan dengan penentuan skenario pengujian dengan parameter berdasarkan dari data jumlah *user* atau *visitor website*. Skenario dibagi menjadi tiga yakni skenario pertama yang mewakili kondisi *baseline* (*website* diakses dengan layanan belum terlalu digunakan), kemudian skenario kedua yang mewakili kondisi *peak* (*website* diakses pada saat layanan digunakan), dan skenario ketiga yang mewakili kondisi *stress* (*website* diakses pada saat layanan digunakan dengan jumlah user dua kali lebih banyak) [13]. Selanjutnya dilakukan pengujian dengan metode *load testing* berdasarkan skenario yang telah dibuat dengan

menggunakan aplikasi Apache Jmeter yang kemudian hasilnya dianalisis oleh penulis berdasarkan dari sumber-sumber literatur.

Jenis data pada penelitian ini adalah jenis data kuantitatif. Data yang digunakan pada penelitian ini adalah data hasil pengujian *load time* (waktu muat) dari *website invitees* sehingga data hasil pengujian *load time website invitees* termasuk dalam jenis data kuantitatif. Data hasil pengujian *load time website invitees* dikumpulkan langsung oleh penulis dari pengujian yang dilakukan sehingga sumber data dari penelitian ini merupakan sumber data primer. Setelah diperoleh data hasil pengujian, data tersebut dianalisis dengan cara membandingkannya dengan target pengujian. Jika sesuai dengan target pengujian maka skenario pengujian dinyatakan berhasil, jika tidak memenuhi syarat target pengujian maka skenario pengujian dinyatakan gagal.

3. HASIL DAN PEMBAHASAN

3.1. Deskripsi Data dan Penentuan Skenario Pengujian

Berdasarkan hasil riset dari WebsiteBuilderExpert, *load time* yang ideal untuk website non-ecommerce adalah 3 detik [1] dan juga berdasarkan dari Cami Bird, *load time* ideal sebuah *website* yang juga mempengaruhi *search engine* adalah kurang dari 3 detik [14] sehingga dalam penelitian ini target *load time* dari *website invitees* adalah kurang dari 3 detik.

Maka test plan dan target pengujian sebagai berikut

- *Load time* tidak lebih dari 3 detik.
- Tidak terdapat respon gagal (*error*) dari *request* yang dikirimkan selama pengujian dilakukan.
- Data pengujian berhasil masuk ke dalam fungsi.

Pengujian yang dilakukan secara bertahap dan beragam, akan menggunakan teknik *step load* yang meningkatkan jumlah user secara berkala. Skala yang akan digunakan adalah 2 kali per skenario yang dimana akan lebih efektif untuk melihat performa dari sistem dalam kondisi user yang bervariasi dan bertahap [15].

Berdasarkan dari Logicify, Penentuan jumlah thread yang akan digunakan pada pengujian *load testing* sebaiknya diambil dari data asli pengunjung ataupun *user* pada *website* [15]. Maka dari itu diperoleh data pengunjung dari *website invitees* seperti yang ditunjukkan pada Gambar 2.

Berdasarkan Gambar 2, terdapat 829 total *visitor* atau pengunjung. Dapat dilihat juga, terdapat juga jumlah pengunjung yang banyak pada bulan Januari karena digunakannya layanan *invitees* oleh *client*. Berdasarkan Gambar 2, data periode bulan januari sampai juni dan data bulan januari per harinya akan digunakan sebagai referensi dalam penentuan jumlah thread atau user dalam pengujian ini.

Month	Unique visitors	Number of visits	Pages	Hits	Bandwidth
Jan 2022	284	340	3,550	4,890	534.56 MB
Feb 2022	126	131	818	2,354	525.21 MB
Mar 2022	62	68	360	1,062	225.65 MB
Apr 2022	81	85	404	908	145.70 MB
May 2022	66	71	922	1,990	360.77 MB
Jun 2022	112	134	1,740	3,848	789.98 MB
Jul 2022	0	0	0	0	0
Aug 2022	0	0	0	0	0
Sep 2022	0	0	0	0	0
Oct 2022	0	0	0	0	0
Nov 2022	0	0	0	0	0
Dec 2022	0	0	0	0	0
Total	731	829	7,794	15,052	2.52 GB

Gambar 2. Data Jumlah Pengunjung Website Invitees Periode Januari – Juni 2022

Berdasarkan dari Gambar 2 yang mewakilkan jumlah visitor pada umumnya, di peroleh data user perharinya sebagai berikut :

$$Jumlah\ User\ perhari = \frac{Jumlah\ user\ periode\ Januari - Juni}{Jumlah\ bulan * Jumlah\ hari} \tag{1}$$

$$Jumlah\ User\ perhari = \frac{829}{6 * 30} = 4.6055 \dots \tag{2}$$

Dari persamaan 2 didapatkan bahwa jumlah user perharinya adalah 4.6055.... = ~ 5 , sehingga untuk skenario pertama yang mewakilkan penggunaan layanan invitees pada saat kondisi biasa saja didapatkan jumlah user atau number of thread adalah 5 user.

Berdasarkan Gambar 2 yang mewakilkan jumlah user pada saat layanan digunakan, diperoleh data user perharinya sebagai berikut :

$$Jumlah\ User\ perhari = \frac{Jumlah\ user\ periode\ Januari}{Jumlah\ hari} \tag{3}$$

$$Jumlah\ User\ perhari = \frac{340}{30} = 11.333 \dots \tag{4}$$

Dari persamaan 4 didapatkan bahwa jumlah user perharinya adalah 11.333.... pada saat layanan digunakan. Dikarenakan skala yang digunakan yaitu 2 kali dari skenario sebelumnya, maka dari data tersebut akan di bulatkan ke bawah dan data yang akan digunakan untuk skenario kedua yaitu sebanyak 10 user yang akan mewakilkan jumlah user pada saat kondisi layanan digunakan oleh client. Sedangkan untuk skenario ketiga, juga akan menggunakan skala 2 kali dari skenario sebelumnya sehingga didapatkan jumlah user untuk skenario ketiga yakni 20 orang. Sehingga dihasilkan skenario sesuai pada Tabel 1.

Tabel 1. Skenario Pengujian

Skenario 1	Number of Threads (users)	5
	Ramp-up period (seconds)	1
	Loop count	1
Skenario 2	Number of Threads (users)	10
	Ramp-up period (seconds)	1
	Loop count	1
Skenario 3	Number of Threads (users)	20
	Ramp-up period (seconds)	1
	Loop count	1

Dari skenario pada Tabel 1, skenario 1 mensimulasikan bagaimana *load time* ketika website sedang dalam kondisi service-level agreement (Baseline Testing) dengan berdasarkan data user perhari dari bulan Januari sampai Juni, skenario 2 mensimulasikan bagaimana *load time* ketika website sedang dalam kondisi layanan digunakan (Peak Testing) berdasarkan data user per hari dari bulan Januari, dan skenario 3 mensimulasikan bagaimana *load time* dalam kondisi layanan digunakan secara maksimum (Stress Testing) dengan jumlah user dua kali lebih banyak [13].

3.2. Pembahasan

Adapun langkah-langkah pengujian dan hasil dari pengujian yang dilakukan sebagai berikut :

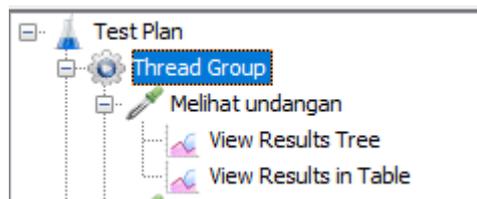
1. Pembuatan dan Konfigurasi *Thread Group*

Gambar 3. Konfigurasi parameter pengujian skenario 1

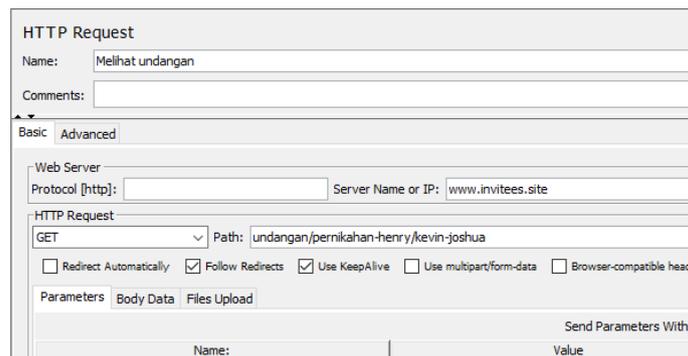
Membuat thread group dan mengisi masing – masing parameter berdasarkan dari skenario yang telah dibuat seperti pada Gambar 3.

2. Pembuatan dan Konfigurasi *HTTP Request*

Selanjutnya, dibuat *Sampler HTTP Request* untuk fungsi melihat undangan dan *Listener* untuk memperlihatkan hasil pengujian. Konfigurasi dari *sampler HTTP Request* yang diisi adalah *method request GET*, *Server name* atau *IP* yang berisikan *link* dari *website* invitees yaitu www.invitees.site dan *path* yang berisikan salah satu contoh undangan yang dibuat untuk pengujian ini. Adapun detailnya dapat dilihat pada Gambar 4 dan 5.

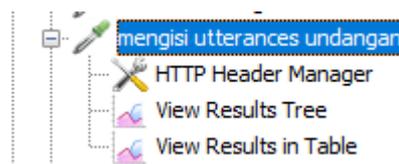


Gambar 4. *Sampler HTTP Request* untuk fungsi melihat undangan

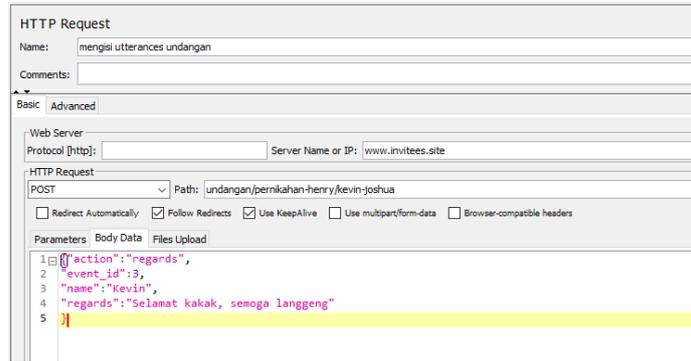


Gambar 5. Konfigurasi *Sampler HTTP Request* melihat undangan

Setelah konfigurasi tersebut dilanjutkan dengan konfigurasi fungsi mengisi ucapan undangan dengan membuat *HTTP Request* baru pada *Thread Group* dengan konfigurasi *server name* dan *path* yang sama dengan *HTTP Request* sebelumnya tetapi *Request Method*nya menjadi *POST* dan *Body Data* seperti pada Gambar 6 dan 7.

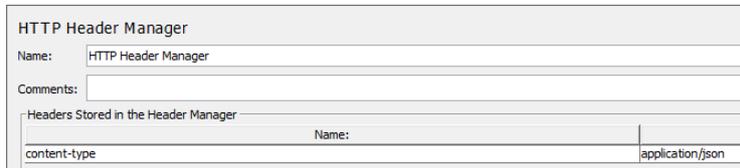


Gambar 6. *Sampler HTTP Request* untuk fungsi mengisi ucapan undangan



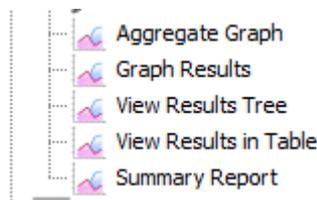
Gambar 7. Konfigurasi *Sampler HTTP Request* Mengisi *utterances* undangan

Agar *Body Data* dapat diinput ke website kita perlu menambahkan *HTTP Header Manager* untuk mendeklarasikan *content-type* yang kita *input* berupa JSON dan juga ditambahkan *Listener* untuk fungsi ini. Adapun Konfigurasinya dapat dilihat pada Gambar 8.



Gambar 8. Konfigurasi HTTP Header Manager

3. Pembuatan *Event Listener*



Gambar 9. Pembuatan Event Listener

Setelah semua fungsi telah dikonfigurasi untuk pengujian, ditambahkan beberapa *listener* tambahan pada *thread Group* seperti pada Gambar 9 yakni *Aggregate Graph* yang berfungsi menunjukkan bagan *load time* dari hasil pengujian, *Graph result* untuk menunjukkan grafik data hasil pengujian, *View Result Tree* untuk menunjukkan hasil pengujian detail, *View Result in Table* untuk menunjukkan hasil pengujian dalam bentuk tabel dan *Summary Report* untuk menunjukkan rangkuman hasil pengujian.

Berdasarkan pengujian yang telah dilakukan , Maka didapatkan hasil dari pengujian *website Invitees* skenario pertama yang ditunjukkan pada Tabel 2 yang rinciannya dapat dilihat pada lampiran.

Tabel 2. Tabel Data Hasil Pengujian Skenario 1

No	Fungsi	ThreadName	Load Time	Response Message	Success State	Failure Message
1	Melihat undangan	Thread Group 1-1	2117	OK	TRUE	-
2	Melihat undangan	Thread Group 1-2	2414	OK	TRUE	-
3	mengisi utterances undangan	Thread Group 1-1	767	OK	TRUE	-
...
10	mengisi utterances undangan	Thread Group 1-3	1115	OK	TRUE	-

Berdasarkan dari Tabel 2 dapat disimpulkan bahwa, fungsi melihat undangan dan fungsi mengisi undangan pada skenario 1 berjalan dengan lancar, serta tidak didapatkannya kegagalan pada setiap fungsi yang dijalankan (tidak ada *error*).

Invitees			
47	Steven	Selamat kakak, semoga langgeng	2022-06-22 11:53:12
48	Steven	Selamat kakak, semoga langgeng	2022-06-22 11:53:12
49	kevin	Selamat kakak, semoga langgeng	2022-06-22 12:01:05
50	kevin	Selamat kakak, semoga langgeng	2022-06-22 12:01:05
51	kevin	Selamat kakak, semoga langgeng	2022-06-22 12:01:05
52	kevin	Selamat kakak, semoga langgeng	2022-06-22 12:01:05
53	kevin	Selamat kakak, semoga langgeng	2022-06-22 12:01:06
54	Jason	Selamat kakak, semoga langgeng	2022-06-22 12:01:10

Gambar 10. Data Skenario 1 Masuk ke Sistem

Dari Gambar 10, dapat dilihat bahwa data pengujian dari fungsi mengisi undangan skenario 1 dapat masuk ke dalam sistem.

Label	# Samples	Average	Min	Max	Std. Dev.	Error %
Melihat undangan	5	2754	2117	3532	593.82	0.00%
mengisi utterances u...	5	744	528	1115	200.24	0.00%
TOTAL	10	1749	528	3532	1098.45	0.00%

Gambar 11. Summary Report Pengujian Skenario 1

Berdasarkan Gambar 11, dari data yang ditampilkan dapat dianalisis bahwa dengan 5 user dengan 1 request baru tiap 1 detiknya menghasilkan *load time* rata-rata sebesar 1.7 detik dengan fungsi melihat undangan sebesar 2.7 detik, fungsi mengisi ucapan undangan sebesar 744 milidetik, nilai *error* sebesar 0%, data dapat masuk ke dalam sistem, serta nilai *Throughput* 126.103/menit. Data ini membuktikan bahwa website Invitees dapat bekerja secara optimal pada kondisi *traffic* yang rendah.

Dilanjutkan ke skenario kedua, dengan hasil yang didapatkan pada Tabel 3 yang rinciannya dapat dilihat pada lampiran.

Tabel 3. Tabel Data Hasil Pengujian Skenario 2

No	Fungsi	Thread Group	Load Time	Response Message	Success State	Failure Message
1	Melihat undangan	Thread Group 1-1	546	OK	TRUE	-
2	Melihat undangan	Thread Group 1-3	506	OK	TRUE	-
3	Melihat undangan	Thread Group 1-4	677	OK	TRUE	-
...
20	mengisi utterances undangan	Thread Group 1-8	602	OK	TRUE	-

Berdasarkan dari Tabel 3 dapat disimpulkan bahwa, fungsi melihat undangan dan fungsi mengisi undangan pada skenario 2 berjalan dengan lancar, serta tidak didapatkannya kegagalan pada setiap fungsi yang dijalankan (tidak ada *error*).

Invitees			
53	kevin	Selamat kakak, semoga langgeng	2022-06-22 12:01:06
54	Jason	Selamat kakak, semoga langgeng	2022-06-22 12:01:10
55	Jason	Selamat kakak, semoga langgeng	2022-06-22 12:01:11
56	Jason	Selamat kakak, semoga langgeng	2022-06-22 12:01:12
57	Jason	Selamat kakak, semoga langgeng	2022-06-22 12:01:13
58	Jason	Selamat kakak, semoga langgeng	2022-06-22 12:01:13
59	Jason	Selamat kakak, semoga langgeng	2022-06-22 12:01:13
60	Jason	Selamat kakak, semoga langgeng	2022-06-22 12:01:14
61	Jason	Selamat kakak, semoga langgeng	2022-06-22 12:01:14
62	Jason	Selamat kakak, semoga langgeng	2022-06-22 12:01:14
63	Jason	Selamat kakak, semoga langgeng	2022-06-22 12:01:14
64	lan	Selamat kakak, semoga langgeng	2022-06-23 03:00:52

Gambar 12. Data Skenario 2 Masuk ke Sistem

Dari Gambar 12 dapat dilihat bahwa data pengujian dari fungsi mengisi undangan skenario 2 dapat masuk ke dalam sistem.

Label	# Samples	Average	Min	Max	Std. Dev.	Error %
Melihat undangan	10	2042	506	5369	1442.65	0.00%
mengisi utterances u...	10	1553	602	2093	449.54	0.00%
TOTAL	20	1798	506	5369	1096.10	0.00%

Gambar 13. Summary Report Pengujian Skenario 2

Berdasarkan Gambar 13, dari data yang ditampilkan dapat dianalisis bahwa dengan 10 user dengan 1 request baru tiap 1 detiknya menghasilkan *load time* rata-rata sebesar 1.8 detik dimana fungsi melihat undangan sebesar 2 detik dan fungsi mengisi ucapan undangan sebesar 1.5 detik, nilai *error* sebesar 0%, data dapat masuk ke dalam sistem, serta nilai *Throughput* 179.883/menit. Data ini membuktikan bahwa website Invitees dapat bekerja secara optimal pada kondisi *traffic* yang cukup sibuk .

Dilanjutkan ke skenario ketiga, dengan hasil yang didapatkan pada Tabel 4 yang rinciannya dapat dilihat pada lampiran.

Tabel 4. Tabel Data Hasil Pengujian Skenario 3

No	Fungsi	Thread Group	Load Time	Response Message	Success State	Failure Message
1	Melihat undangan	Thread Group 1-2	2094	OK	TRUE	-
2	Melihat undangan	Thread Group 1-16	1665	OK	TRUE	-
3	Melihat undangan	Thread Group 1-10	2657	OK	TRUE	-
...
40	mengisi utterances undangan	Thread Group 1-8	7604	OK	TRUE	-

Berdasarkan dari Tabel 4 dapat disimpulkan bahwa, fungsi melihat undangan dan fungsi mengisi undangan pada skenario 3 berjalan dengan lancar, serta tidak didapatkannya kegagalan pada setiap fungsi yang dijalankan (tidak ada *error*).

No	Isi	Waktu
57	Jason Selamat kakak, semoga langgeng	2022-06-22 12:01:13
58	Jason Selamat kakak, semoga langgeng	2022-06-22 12:01:13
59	Jason Selamat kakak, semoga langgeng	2022-06-22 12:01:13
60	Jason Selamat kakak, semoga langgeng	2022-06-22 12:01:14
61	Jason Selamat kakak, semoga langgeng	2022-06-22 12:01:14
62	Jason Selamat kakak, semoga langgeng	2022-06-22 12:01:14
63	Jason Selamat kakak, semoga langgeng	2022-06-22 12:01:14
64	lan Selamat kakak, semoga langgeng	2022-06-23 03:00:52
65	lan Selamat kakak, semoga langgeng	2022-06-23 03:00:52
66	lan Selamat kakak, semoga langgeng	2022-06-23 03:00:53
67	lan Selamat kakak, semoga langgeng	2022-06-23 03:00:53
68	lan Selamat kakak, semoga langgeng	2022-06-23 03:00:54
69	lan Selamat kakak, semoga langgeng	2022-06-23 03:00:54
70	lan Selamat kakak, semoga langgeng	2022-06-23 03:00:54
71	lan Selamat kakak, semoga langgeng	2022-06-23 03:00:54
72	lan Selamat kakak, semoga langgeng	2022-06-23 03:00:57
73	lan Selamat kakak, semoga langgeng	2022-06-23 03:00:57
74	lan Selamat kakak, semoga langgeng	2022-06-23 03:00:58
75	lan Selamat kakak, semoga langgeng	2022-06-23 03:00:58
76	lan Selamat kakak, semoga langgeng	2022-06-23 03:00:58
77	lan Selamat kakak, semoga langgeng	2022-06-23 03:00:58
78	lan Selamat kakak, semoga langgeng	2022-06-23 03:00:59
79	lan Selamat kakak, semoga langgeng	2022-06-23 03:00:59
80	lan Selamat kakak, semoga langgeng	2022-06-23 03:00:59
81	lan Selamat kakak, semoga langgeng	2022-06-23 03:00:59
82	lan Selamat kakak, semoga langgeng	2022-06-23 03:00:59
83	lan Selamat kakak, semoga langgeng	2022-06-23 03:00:59

Gambar 14. Data Skenario 3 Masuk ke Sistem

Dari Gambar 14 dapat dilihat bahwa data pengujian dari fungsi mengisi undangan skenario 3 dapat masuk ke dalam sistem.

Label	# Samples	Average	Min	Max	Std. Dev.	Error %
Melihat undangan	20	5542	1665	8244	2131.66	0.00%
mengisi utterances u...	20	4267	1506	7604	1867.22	0.00%
TOTAL	40	4905	1506	8244	2102.73	0.00%

Gambar 15. Summary Report Pengujian Skenario 3

Berdasarkan Gambar 15, dari data yang ditampilkan dapat dianalisis bahwa dengan 20 user dengan 1 request baru tiap 1 detiknya menghasilkan load time rata-rata sebesar 4.9 detik dimana fungsi melihat undangan sebesar 5.5 detik dan fungsi mengisi undangan sebesar 4.2 detik, nilai error sebesar 0%, data dapat masuk ke dalam sistem, serta nilai Throughput 160.278/menit. Data ini membuktikan bahwa website Invitees kurang optimal pada kondisi traffic yang sangat sibuk.

4. KESIMPULAN

Setelah dilakukannya penelitian load testing terhadap website invitees terutama pada fungsionalitas utamanya, berdasarkan dari skenario 1 yang mewakili kondisi umum (baseline testing) website invitees bekerja dengan optimal dikarenakan skenario pertama telah memenuhi target pengujian dengan hasil rata-rata waktu loadnya sebesar 2.7 detik yang kurang dari 3 detik, tidak didapatkan kegagalan respon, dan data test bisa masuk ke dalam sistem.

Berdasarkan dari skenario 2 yang mewakili kondisi penggunaan layanan pada traffic tinggi (peak testing), *website* invitees telah bekerja dengan optimal dikarenakan skenario kedua telah memenuhi target pengujian dengan hasil rata-rata waktu *load*nya sebesar 1.8 detik yang kurang dari 3 detik, tidak didapatkan kegagalan respon, dan data test bisa masuk ke dalam sistem.

Hasil dari pengujian pada skenario yang ketiga menunjukkan bahwa *website* invitees kurang optimal bekerja pada saat kondisi penggunaan layanan berat (stress testing) yakni dua kali lebih berat dari skenario kedua dikarenakan tidak memenuhi salah satu dari target pengujian dengan hasil rata-rata waktu *load*nya sebesar 4.9 detik yang lebih besar dari target pengujian yaitu 3 detik, tidak didapatkan *error* atau kegagalan respon, dan data test bisa masuk ke dalam sistem.

Maka dapat disimpulkan dari hasil pengujian yang dilakukan bahwa *website* Invitees berjalan dengan kurang optimal dengan rata-rata total waktu muat dari skenario pertama, kedua dan ketiga adalah 3.1 detik berdasarkan skenario pertama dan kedua, tetapi masih diperlukan perbaikan terutama pada fungsi melihat undangan. Dikarenakan pada fungsi melihat undangan, terdapat banyak gambar yang merupakan salah satu faktor yang mempengaruhi waktu muat maka perbaikan yang dapat dilakukan untuk mengurangi waktu *load*nya yaitu kompresi dan optimisasi gambar.

5. SARAN

Pada penelitian ini dilakukan pengujian hanya untuk 1 undangan. Saran untuk penelitian selanjutnya adalah ditambahkan variasi skenario pengujian dengan menggunakan lebih dari satu undangan yang diuji secara bersamaan.

DAFTAR PUSTAKA

- [1] Maura Monaghan, "Website Load Time Statistics: Why Speed Matters in 2022," 2022. [Online]. Available: <https://www.websitebuilderexpert.com/building-websites/website-load-time-statistics/>. [Accessed: 20-Jun-2022].
- [2] K. Zhu, J. Fu, and Y. Li, "Research the performance testing and performance improvement strategy in web application," pp. 328–332, 2010.
- [3] "Apache Jmeter," *Apache Software Foundation*. [Online]. Available: <https://jmeter.apache.org/>. [Accessed: 08-Aug-2022].
- [4] S. Suryadevara, S. Ali, and N. Zealand, "Preperformance Testing of A Website," pp. 33–52, 2020.
- [5] N. Kaur and K. Bahl, "Performance Testing Of Insititute Website Using Jmeter," vol. 3, no. 4, pp. 534–537, 2016.
- [6] Z. Munyaradzi, G. Maxmillan, and M. N. Amanda, "Effects of Web Page Contents on Load Time over the Internet," no. January, 2016.
- [7] J. Manhas, "A Study of Factors Affecting Websites Page Loading Speed for Efficient

- Web A Study of Factors Affecting Websites Page Loading Speed for Efficient Web Performance,” no. December 2013, 2015.
- [8] A. Suprpto and D. Sasongko, “EVALUASI PERFORMA WEBSITE BERDASARKAN PENGUJIAN BEBAN DAN STRESS MENGGUNAKAN LOADIMPACT (STUDI KASUS WEBSITE IAIN SALATIGA),” vol. 6, no. 1, pp. 31–37, 2021.
- [9] I. M. E. Listartha, “Penguujian Performa dan Tingkat Stress pada Website Legalisir Ijasah Online Universitas Pendidikan Ganesha.”
- [10] C. P. Agustika, W. S. J. Saputra, and M. Idhom, “PENGUJIAN APLIKASI GREENWALLET DENGAN METODE LOAD TESTING DAN APACHE JMETER,” vol. 2, no. 2, pp. 190–195, 2021.
- [11] D. I. Permatasari *et al.*, “Pengukuran throughput load testing menggunakan test case sampling gorilla testing,” no. September, pp. 2008–2014, 2019.
- [12] D. I. Permatasari *et al.*, “Penguujian Aplikasi Menggunakan Metode Load Testing dengan Apache Jmeter pada Sistem Informasi Pertanian,” vol. 8, no. 1, pp. 135–139, 2020.
- [13] “5 Load Testing Scenarios for Your API,” *SoapUI*. [Online]. Available: <https://www.soapui.org/learn/load-testing/5-load-testing-scenarios-for-your-api/>.
- [14] C. Bird, “What Is a Good Page Load Time for SEO — How Fast Is Fast Enough?,” 2020. [Online]. Available: https://www.semrush.com/blog/how-fast-is-fast-enough-page-load-time-and-your-bottom-line/?kw=&cmp=AA_SRCH_DSA_Blog_Core_BU_EN&label=dsa_pagefeed&Network=g&Device=c&utm_content=515715541581&kwid=dsa-1057183196155&cmpid=12565136841&agpid=119030055826&BU=Core&extid=23631749011&adpos=&gclid=CjwKCAjw-8qVBhANEiwAfjXLrNrNGI314RKvhgXnDyZvxF-VF7ZO98bqoy0wICLx-YfwVazXpEb0MRoC6soQAvD_BwE. [Accessed: 22-Jun-2022].
- [15] Logicify, “How to Choose the Number of Threads for an Apache JMeter Test Plan.” [Online]. Available: <https://medium.com/@logicify/how-to-choose-the-number-of-threads-for-an-apache-jmeter-test-plan-427cf615d1c9>. [Accessed: 28-Jul-2022].

Lampiran

[Tabel Hasil Pengujian Skenario 1](#)

[Tabel Hasil Pengujian Skenario 2](#)

[Tabel Hasil Pengujian Skenario 3](#)