

# Pengujian Aplikasi Menggunakan Metode Load Testing dengan Apache Jmeter pada Sistem Informasi Pertanian

Desy Intan Permatasari<sup>#1</sup>, Misbahul Ardani<sup>#2</sup>, Arvianti Yulia Ma'ulfa<sup>#3</sup>, Nindy Ilhami<sup>#4</sup>, Sandy Gio Pratama<sup>#5</sup>,  
Seria Reni Dwi Astuti<sup>#6</sup>, Nadia Widad Naufalita<sup>#7</sup>

<sup>#</sup>*Departemen Teknik Informatika dan Komputer Politeknik Elektronika Negeri Surabaya  
Jl. Raya ITS, Keputih, Kec. Sukolilo, Surabaya, Indonesia*

<sup>1</sup>desy@pens.ac.id

<sup>5</sup>sandygio@it.student.pens.ac.id

**Abstrak**— *Load Testing* adalah teknik *performance testing* yang mana respon sistem diukur dalam berbagai *load condition*. Penelitian ini membantu menentukan bagaimana software berperilaku ketika beberapa *user* mengakses software secara bersamaan. Pengujian menggunakan metoda *Black Box Testing* dan *Gorilla Testing* sebagai *Test Case Sampling*. *Black Box Testing* adalah Pengujian fungsional software tanpa mengetahui struktur internal program (*no knowledge*). *Gorilla Testing* adalah Teknik pengujian dimana modul program berulang kali diuji untuk memastikan bahwa modul tersebut berfungsi dengan benar dan tidak ada bug. Modul dapat diuji lebih dari seratus kali, dan dengan cara yang sama. Jadi, *Gorilla Testing* juga dikenal sebagai "*Frustrating Testing*". Tujuannya adalah untuk menguji *robustness*. Pada penelitian ini target pengujian aplikasi adalah *Loading time* tidak >3 second, *process memory* tidak >400MB, dan data dapat masuk kedalam fungsi. Hasil pengujian *load testing* menunjukkan dengan menggunakan Apache JMeter bahwa target pengujian *loading time* dan *process memory* sudah terpenuhi.

**Kata kunci**— Software Testing, Load Testing, Performance Testing, Apache JMeter

## I. PENDAHULUAN

Aplikasi MeTANI adalah aplikasi yang memiliki fungsi inventarisasi dan pemetaan produksi hasil dan infrastruktur pertanian. Aplikasi ini mendata seluruh produk hasil pertanian di daerah dan mencatat data infrastruktur pertanian di daerah. Diantaranya adalah hasil pertanian baik sawah, perkebunan, maupun peternakan.

Juga terdapat infrastruktur yang terdiri dari kondisi lahan dan luas lahan pertanian.

Aplikasi MeTANI berguna untuk inventarisasi dan mengetahui rasio produksi pertanian di daerah tiap tahunnya.

Pada aplikasi ini terdapat beberapa modul, antara lain modul pengelolaan jenis tani, pengelolaan jenis tanah, modul hasil tani, modul tanah, modul lokasi, dan lain-lain. Setelah aplikasi MeTANI selesai dibuat, maka perlu dilakukan pengujian untuk menguji modul sudah berjalan sesuai dengan requirement.

Pengujian adalah suatu proses pelaksanaan suatu program dengan tujuan menemukan suatu kesalahan. Suatu kasus test yang baik adalah apabila test tersebut mempunyai kemungkinan menemukan sebuah kesalahan yang tidak terungkap. Suatu test yang sukses adalah bila test tersebut membongkar suatu kesalahan yang awalnya tidak ditemukan. Tujuan utama dari pengujian adalah untuk mendesain test yang secara sistematis membongkar jenis kesalahan dengan usaha dan waktu minimum.

Load Testing adalah Teknik *performance testing* yang mana respon sistem diukur dalam berbagai *load condition*.

Paper ini membantu menentukan bagaimana software berperilaku ketika beberapa *user* mengakses software secara bersamaan. Pengujian menggunakan method *Black Box Testing* dan *Gorilla Testing* sebagai *Test Case Sampling*.

*Black Box Testing* adalah Pengujian fungsional software tanpa mengetahui struktur internal program (*no knowledge*).

*Gorilla Testing* adalah Teknik pengujian dimana modul program berulang kali diuji untuk memastikan bahwa modul tersebut berfungsi dengan benar dan tidak ada bug.

Modul dapat diuji lebih dari seratus kali, dan dengan cara yang sama. Jadi, Gorilla Testing juga dikenal sebagai "Frustrating Testing". Tujuannya adalah untuk menguji robustness.

## II. KAJIAN LITELATUR

### A. Pengujian Software

Pengujian *software* sangat diperlukan untuk memastikan *software*/aplikasi yang sudah/sedang dibuat dapat berjalan sesuai dengan fungsionalitas yang diharapkan. Pengembang atau penguji *software* harus menyiapkan sesi khusus untuk menguji program yang sudah dibuat agar kesalahan ataupun kekurangan dapat dideteksi sejak awal dan dikoreksi secepatnya. Pengujian atau testing sendiri merupakan elemen kritis dari jaminan kualitas perangkat lunak dan merupakan bagian yang tidak terpisahkan dari siklus hidup pengembangan *software* seperti halnya analisis, desain, dan pengkodean. (Shi, 2010)

Pengujian *software* haruslah dilakukan dalam proses rekayasa perangkat lunak atau *software engineering*. Sejumlah strategi pengujian *software* telah diusulkan dalam literatur. Semuanya menyediakan template untuk pengujian bagi pembuat *software*. Dalam hal ini, semuanya harus memiliki karakteristik umum berupa (Bhat and Quadri, 2015) :

### B. Performance Testing

Performance testing adalah jenis pengujian untuk memastikan perangkat lunak akan bekerja dengan baik di bawah beban kerja yang diharapkan. Tujuan utamanya bukan untuk mencari bug, tapi untuk mengeliminasi performance bottleneck.

Fokus dari Performance Testing, yaitu :

- Speed - menentukan apakah aplikasi merespon dengan cepat
- Scalability - menentukan apakah jumlah maksimum user load dapat ditangani
- Stability - menentukan apakah aplikasi stabil dengan berbagai beban

### C. Load Testing

Load testing adalah teknik performance testing yang mana respon sistem diukur dalam berbagai load condition.

Pengujian ini membantu menentukan bagaimana *software* berperilaku ketika beberapa user mengakses *software* secara bersamaan.

Load testing diperlukan untuk membuat simulasi akses aplikasi web / website secara simultan. Cara ini lebih baik dibandingkan dengan harus mengundang sekian belas, atau puluh orang sekaligus untuk mengakses sebuah website.

Langkah-langkah pengujian *software* ada 4 yaitu:

1. *Unit testing*-testing per unit yaitu mencoba alur yang spesifik pada struktur modul kontrol untuk memastikan pelengkapan secara penuh dan pendeteksian error secara maksimum
2. *Integration testing* – testing per penggabungan unit yaitu pengalamanan dari isu-isu yang diasosiasikan dengan masalah ganda pada verifikasi dan konstruksi program
3. *High-order test* yaitu terjadi ketika *software* telah selesai diintegrasikan atau dibangun menjadi satu –tidak terpisah-pisah
4. *Validation test* yaitu menyediakan jaminan akhir bahwa *software* memenuhi semua kebutuhan fungsional, kepribadian dan performa.

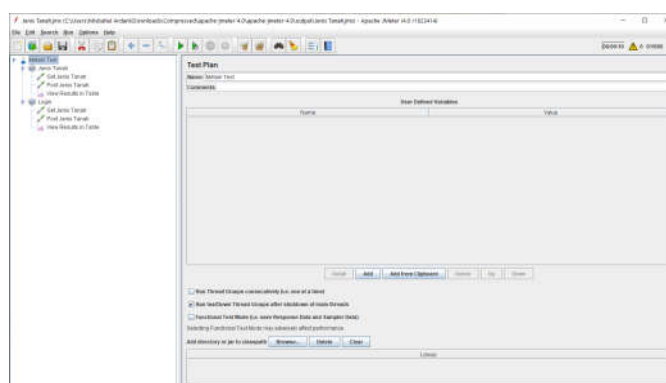
### D. Apache JMeter

Aplikasi Apache JMeter adalah perangkat lunak open source, 100% aplikasi Java murni dirancang untuk memuat tes perilaku fungsional dan mengukur kinerja. Ini pada awalnya dirancang untuk pengujian Aplikasi Web tetapi sejak diperluas untuk fungsi tes lainnya. [7]

APACHE JMETER adalah aplikasi open source berbasis Java yang dapat dipergunakan untuk performance test. Bagi seorang QA Engineer jMeter bisa digunakan untuk melakukan load/stress testing Web Application, FTP Application dan Database server test.

Apache JMeter dapat digunakan untuk menguji kinerja baik pada sumber daya statis dan dinamis (web services (SOAP / REST), Web bahasa dinamis - PHP, Java, ASP.NET, File, dll -, Jawa Objects, Pangkalan Data dan Pertanyaan, FTP Server dll). Hal ini dapat digunakan untuk mensimulasikan beban berat pada server, sekelompok server, jaringan atau objek untuk menguji kekuatan atau untuk menganalisa kinerja secara keseluruhan di bawah jenis beban yang berbeda.

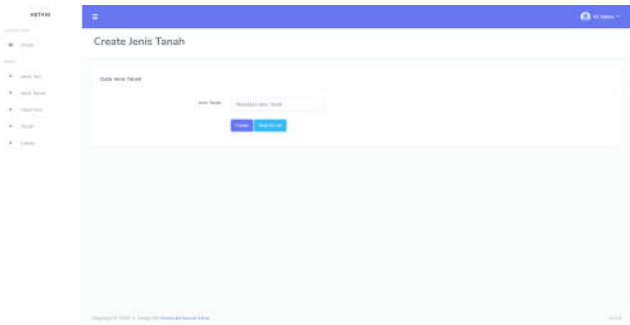
Tampilan awal aplikasi Jmeter sebagai berikut:



Gambar. 1. Tampilan awal aplikasi Jmeter

### III. HASIL DAN PEMBAHASAN

#### A. Pengujian Aplikasi dengan Load Testing



Gambar. 2. Pengujian dengan Load Testinger

User dapat melakukan input data dari aplikasi MeTANI berdasarkan fitur yang disediakan pada aplikasi pengujian.

##### 1. Menentukan test plan dan target pengujian

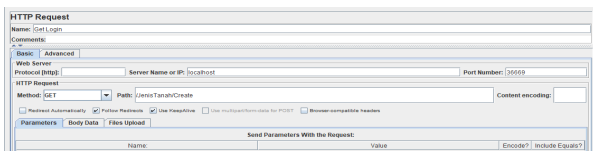
Langkah pertama adalah membuat test plan dan selanjutnya dilakukan testing terhadap aplikasi. Target pengujian aplikasi adalah:

- Loading time tidak >3 second
- Process memory tidak >400MB
- Data dapat masuk kedalam fungsi

##### 2. Thread Login

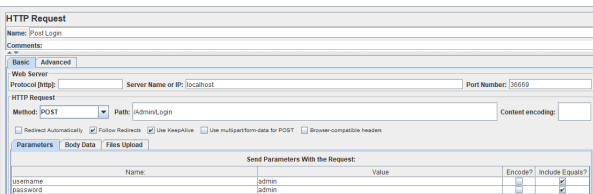
Thread login dibagi menjadi 2 yaitu konfigurasi GET Login dan konfigurasi POST Login.

a. Mengkonfigurasi GET LOGIN dalam aplikasi Apache JMeter dengan memasukkan detail web server dan detail http request.



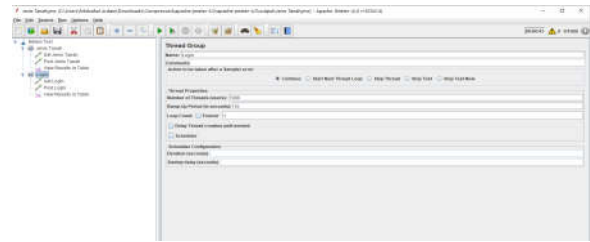
Gambar. 3. Konfigurasi Get Login

b. Mengkonfigurasi POST METHOD dalam aplikasi Apache Jmeter dengan detail yang dibutuhkan dan hampir sama dengan cara sebelumnya.



Gambar. 4. Konfigurasi Post Login

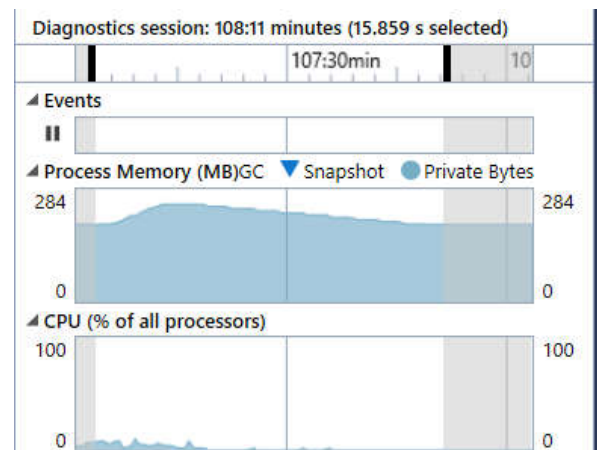
#### 3. Pengujian tread group testing



Gambar. 5. Konfigurasi tread group testing

Berdasarkan data yang sudah kita inputkan maka dapat kita lakukan testing dengan Load Testing menggunakan metode Black Box dan Gorilla Testing. Pertama, testing yang dapat kita lakukan adalah 100 user dengan 1 request baru tiap 0,1 second.

Hasil testing pertama menghasilkan data sebagai berikut:



Gambar. 5. Diagnostic session pengujian pertama

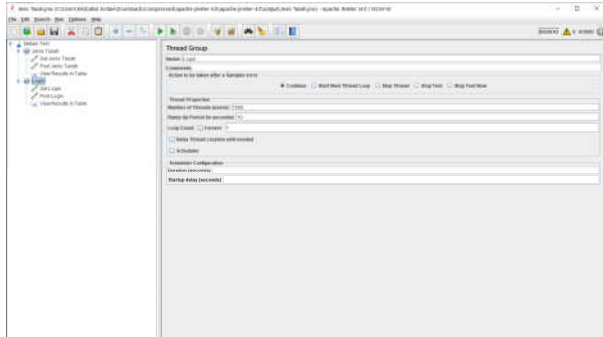
Data yang ditampilkan diatas dapat dianalisa bahwa dengan 100 user dengan 1 request baru tiap 0,1 second. minimum adalah 195 MB dan maximum adalah 245 MB.

Sample #	Start Time	Thread Name	Label	Sample Time(s)	Status	Bytes	Send Bytes	Latency	Checked Times(s)
4584	15-12-20 17:15:11	Login-1-84	Get Jenis Tanah	11	Success	3394	130	11	0
4585	15-12-20 17:15:11	Login-1-84	Post Jenis Tanah	36	Success	4889	483	2	0
4586	15-12-20 17:15:11	Login-1-85	Get Jenis Tanah	4	Success	3390	130	4	0
4587	15-12-20 17:15:11	Login-1-85	Post Jenis Tanah	25	Success	4889	483	2	0
4588	15-12-20 17:15:11	Login-1-86	Get Jenis Tanah	11	Success	3390	130	11	0
4589	15-12-20 17:15:11	Login-1-86	Post Jenis Tanah	35	Success	4889	483	2	0
4590	15-12-20 17:15:11	Login-1-87	Get Jenis Tanah	11	Success	3394	130	11	0
4591	15-12-20 17:15:11	Login-1-87	Post Jenis Tanah	36	Success	4889	483	2	0
4592	15-12-20 17:15:11	Login-1-88	Get Jenis Tanah	11	Success	3394	130	11	0
4593	15-12-20 17:15:11	Login-1-88	Post Jenis Tanah	36	Success	4889	483	2	0
4594	15-12-20 17:15:11	Login-1-89	Get Jenis Tanah	11	Success	3390	130	11	0
4595	15-12-20 17:15:11	Login-1-89	Post Jenis Tanah	36	Success	4889	483	2	0
4596	15-12-20 17:15:11	Login-1-90	Get Jenis Tanah	11	Success	3394	130	11	0
4597	15-12-20 17:15:11	Login-1-90	Post Jenis Tanah	37	Success	4889	483	2	0
4598	15-12-20 17:15:11	Login-1-91	Get Jenis Tanah	11	Success	3390	130	11	0
4599	15-12-20 17:15:11	Login-1-91	Post Jenis Tanah	35	Success	4889	483	2	0
4600	15-12-20 17:15:11	Login-1-92	Get Jenis Tanah	11	Success	3394	130	11	0
4601	15-12-20 17:15:11	Login-1-92	Post Jenis Tanah	44	Success	4889	483	4	0
4602	15-12-20 17:15:11	Login-1-93	Get Jenis Tanah	11	Success	3394	130	11	0
4603	15-12-20 17:15:11	Login-1-93	Post Jenis Tanah	35	Success	4889	483	2	0
4604	15-12-20 17:15:11	Login-1-94	Get Jenis Tanah	4	Success	3390	130	4	0
4605	15-12-20 17:15:11	Login-1-94	Post Jenis Tanah	34	Success	4889	483	2	0
4606	15-12-20 17:15:11	Login-1-95	Get Jenis Tanah	11	Success	3390	130	11	0
4607	15-12-20 17:15:11	Login-1-95	Post Jenis Tanah	31	Success	4889	483	2	0
4608	15-12-20 17:15:11	Login-1-96	Get Jenis Tanah	11	Success	3394	130	11	0
4609	15-12-20 17:15:11	Login-1-96	Post Jenis Tanah	35	Success	4889	483	2	0
4610	15-12-20 17:15:11	Login-1-97	Get Jenis Tanah	11	Success	3394	130	11	0
4611	15-12-20 17:15:11	Login-1-97	Post Jenis Tanah	41	Success	4889	483	4	0
4612	15-12-20 17:15:11	Login-1-98	Get Jenis Tanah	11	Success	3394	130	11	0
4613	15-12-20 17:15:11	Login-1-98	Post Jenis Tanah	35	Success	4889	483	2	0
4614	15-12-20 17:15:11	Login-1-99	Get Jenis Tanah	4	Success	3390	130	4	0
4615	15-12-20 17:15:11	Login-1-99	Post Jenis Tanah	34	Success	4889	483	2	0
4616	15-12-20 17:15:11	Login-1-100	Get Jenis Tanah	11	Success	3390	130	11	0
4617	15-12-20 17:15:11	Login-1-100	Post Jenis Tanah	33	Success	4889	483	2	0
4618	15-12-20 17:15:11	Login-1-101	Get Jenis Tanah	11	Success	3394	130	11	0
4619	15-12-20 17:15:11	Login-1-101	Post Jenis Tanah	35	Success	4889	483	2	0
4620	15-12-20 17:15:11	Login-1-102	Get Jenis Tanah	11	Success	3394	130	11	0
4621	15-12-20 17:15:11	Login-1-102	Post Jenis Tanah	35	Success	4889	483	2	0
4622	15-12-20 17:15:11	Login-1-103	Get Jenis Tanah	11	Success	3390	130	11	0
4623	15-12-20 17:15:11	Login-1-103	Post Jenis Tanah	35	Success	4889	483	2	0
4624	15-12-20 17:15:11	Login-1-104	Get Jenis Tanah	11	Success	3394	130	11	0
4625	15-12-20 17:15:11	Login-1-104	Post Jenis Tanah	35	Success	4889	483	2	0
4626	15-12-20 17:15:11	Login-1-105	Get Jenis Tanah	11	Success	3390	130	11	0
4627	15-12-20 17:15:11	Login-1-105	Post Jenis Tanah	36	Success	4889	483	2	0

Gambar. 6. Data hasil pengujian pertama

Dari tabel diatas, dapat diambil analisa bahwa dalam 100 user dengan 1 request tiap 0,1 second di dapat sample time maximum sebesar 38ms.

4. Test plan 1000 user untuk 1 request tiap 0.1 second



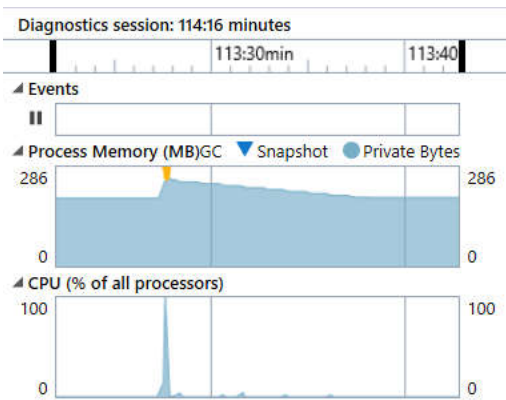
Gambar. 7. Konfigurasi pengujian 1000 user

Setelah diinputkan detail thread group termasuk thread properties untuk 1000 user dengan 1 request tiap 0,1 second maka dapat di lihat hasil testing sebagai berikut:

Sampein #	Start Time	Thread Name	Last	Sampein (msec)	Status	Batas	Start Batas	Label	Content Type
85646	11:20:20.888	Post Java Tanah	20000	4688	OK	4395	4395	0	
85645	11:20:20.852	Post Java Tanah	22777	4688	OK	4392	4392	0	
85640	11:20:20.815	Post Java Tanah	25974	4688	OK	4426	4426	0	
85637	11:20:20.813	Post Java Tanah	26947	4688	OK	4424	4424	0	
85630	11:20:20.803	Post Java Tanah	28352	4688	OK	4452	4452	0	
85628	11:20:20.871	Post Java Tanah	23325	4688	OK	4398	4398	0	
85623	11:20:20.841	Post Java Tanah	22852	4688	OK	4442	4442	0	
85621	11:20:20.821	Post Java Tanah	25974	4688	OK	4452	4452	0	
85620	11:20:20.843	Post Java Tanah	25989	4688	OK	4427	4427	1	
85619	11:20:20.864	Post Java Tanah	25990	4688	OK	4426	4426	0	
85614	11:20:20.894	Post Java Tanah	23011	4688	OK	4419	4419	1	
85605	11:20:20.884	Post Java Tanah	23003	4688	OK	4420	4420	0	
85600	11:20:20.861	Post Java Tanah	21985	4688	OK	4443	4443	0	
85600	11:20:20.820	Post Java Tanah	23252	4688	OK	4420	4420	0	
85598	11:20:20.871	Post Java Tanah	23252	4688	OK	4419	4419	0	
85598	11:20:20.814	Post Java Tanah	23210	4688	OK	4415	4415	0	
85594	11:20:20.852	Post Java Tanah	23247	4688	OK	4449	4449	0	
85593	11:20:20.834	Post Java Tanah	23115	4688	OK	4449	4449	0	
85592	11:20:20.813	Post Java Tanah	23207	4688	OK	4419	4419	0	
85591	11:20:20.873	Post Java Tanah	23205	4688	OK	4439	4439	0	
85584	11:20:20.877	Post Java Tanah	23097	4688	OK	4434	4434	1	
85580	11:20:20.813	Post Java Tanah	23207	4688	OK	4445	4445	0	
85580	11:20:20.810	Post Java Tanah	23465	4688	OK	4451	4451	1	
85577	11:20:20.823	Post Java Tanah	23209	4688	OK	4445	4445	1	
85576	11:20:20.867	Post Java Tanah	23495	4688	OK	4457	4457	1	
85576	11:20:20.878	Post Java Tanah	23497	4688	OK	4454	4454	0	
85570	11:20:20.891	Post Java Tanah	23463	4688	OK	4452	4452	0	
85571	11:20:20.817	Post Java Tanah	23486	4688	OK	4476	4476	1	
85562	11:20:20.813	Post Java Tanah	23445	4688	OK	4498	4498	0	
85562	11:20:20.877	Post Java Tanah	23204	4688	OK	4450	4450	0	
85564	11:20:20.893	Post Java Tanah	23487	4688	OK	4523	4523	0	
85565	11:20:20.830	Post Java Tanah	23452	4688	OK	4474	4474	1	
85565	11:20:20.890	Post Java Tanah	23500	4688	OK	4512	4512	0	
85571	11:20:20.893	Post Java Tanah	23426	4688	OK	4462	4462	1	

Gambar. 8. Data hasil pengujian 1000 user

Proses yang terjadi mengalami delay sebesar 23435ms dalam 1 request tiap 0,1 second untuk jumlah user sebanyak 1000.



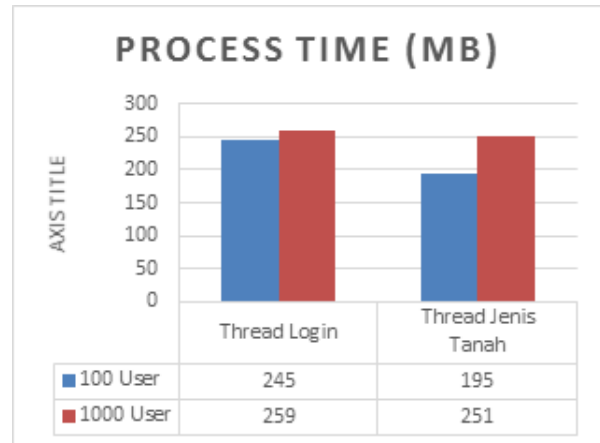
Gambar. 9. Diagnostic session pengujian 1000 user

Berdasarkan diagnostics session, dihasilkan process memory minimal 195MB dan maximum sebesar 259MB.

B. Hasil Analisa Pengujian

1. Process Time

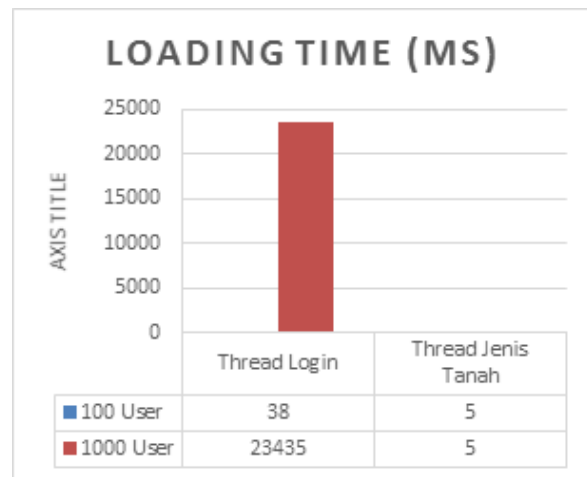
Dari hasil pengujian, dapat dianalisa bahwa process memory tidak lebih dari 400MB. Gambar grafiknya sebagai berikut:



Gambar. 10. Analisis process time

2. Loading Time

Dari hasil pengujian, dapat dianalisa bahwa loading time tidak lebih dari 3 second. Gambar grafiknya sebagai berikut:



Gambar. 11. Analisis loading time

## IV. KESIMPULAN

Load Testing merupakan jenis performance testing yang dapat mengukur respon sistem dalam berbagai load condition.

Hasil pengujian ini membantu menentukan bagaimana software berperilaku ketika beberapa user mengakses software secara bersamaan.

Teknik pengujian Gorilla Testing diterapkan pada pengujian di bagian 3. Teknik ini memastikan bahwa modul tersebut berfungsi dengan benar dan tidak ada bug. Modul dapat diuji lebih dari seratus kali, dan dengan cara yang sama. Gorilla Testing sangat bermanfaat untuk menguji robustness dari aplikasi.

## REFERENSI

- [1] Shi, Mingtao, 2010, Software Functional Testing from the Perspective of Business Practice Computer and Information Science, [www.ccsenet.org/cis](http://www.ccsenet.org/cis).
- [2] Bhat, A, and Quadri, S.M.K, 2015, Equivalence Class Partitioning and Boundary Value Analysis = A review, 2nd International Conference on Computing for Sustainable Global Development (INDIACom)
- [3] Khan, Mohd Ehmer, 2011, Different Approach to Blackbox Testing Technique for Finding Error, International Journal of Software Engineering & Applications (IJSEA), Vol.2, No.4, October 2011
- [4] Nidhra, Srinivas, and Dondeti, Jagruthi, 2012, Blackbox and Whitebox Testing Techniques - A Literature Review, International Journal of Embedded Systems and Applications (IJESA) Vol.2, No.2, June 2012
- [5] Mustaqbal, Sidiq, and Firdaus, Roeri Fajar, 2019, Pengujian aplikasi menggunakan black box testing boundary value analysis, Jurnal Ilmiah Teknologi Informasi Terapan (JITTER)
- [6] <https://jmeter.apache.org/>
- [7] <https://lms.onnocenter.or.id/wiki/index.php/Jmeter>
- [8] S. Sharmila1, Dr. E. Ramadevi2. Analysis of Performance Testing on Web Applications. International Journal of Advanced Research in Computer and Communication Engineering Vol. 3, Issue 3, March 2014
- [9] Pooja Ranout, Neha Thakur. Comparative Analysis of Load testing tool: LOAD Complete and WAPT pro. IJEDR Volume 5, Issue 2 2017.
- [10] Narinder Kaur, Kailash Bahl. Performance Testing Of Insitute Website Using Jmeter. International Journal of Innovative Science, Engineering & Technology, Vol. 3 Issue 4, April 2016.
- [11] Shikha Dhiman Pratibha Sharma et al. Performance Testing: A Comparative Study and Analysis of Web Service Testing Tools. International Journal of Computer Science and Mobile Computing. Vol.5 Issue.6, June- 2016, pg. 507-512
- [12] M. Sidi Mustaqbal, Roeri Fajri Firdaus, Hendra Rahmadi. PENGUJIAN APLIKASI MENGGUNAKAN BLACK BOX TESTING BOUNDARY VALUE ANALYSIS. Jurnal Ilmiah Teknologi Informasi Terapan Volume I, No 3, 10 Agustus 2015
- [13] I. Bornstein, "Practical Software Testing: process oriented approach," Springer Professional Computing, 2003.
- [14] M. E. Khan, "Different Forms of Software Testing Techniques for Finding Errors," International Journal of Software Engineering (IJSE), vol. 7, no. 3,2010.
- [15] IJREAT International Journal of Research in Engineering & Advanced Technology, Volume 3, Issue 3, June-July, 2015
- [16] International Journal of Technical Research and Applications e-ISSN: 2320-8163, [www.ijtra.com](http://www.ijtra.com) Volume 2, Issue 2 (March-April 2014)
- [17] International Journal of Current Engineering and Technology E-ISSN 2277 – 4106, P-ISSN 2347 - 5161 ©2014 INPRESSCO®, All Rights Reserved Available at [32TUhttp://inpressco.com/category/ijcetU32T](http://inpressco.com/category/ijcetU32T)
- [18] International Journal of Advance Research in Computer Science and Management Studies Volume 3, Issue 10, October 2015
- [19] F.Elberzhag, J.Munch and R.Eschbach."Reducing test effort: A Systematic Mapping study on Existing approaches," Information and Technology 54,2012
- [20] [http://www.pcmag.com/encyclopedia\\_term/0,2542,t=WHlTe+box+testing&i=54432,00.asp](http://www.pcmag.com/encyclopedia_term/0,2542,t=WHlTe+box+testing&i=54432,00.asp), February 08, 2009.
- [21] [http://www.testingstandards.co.uk/living\\_glossary.htm#Testing](http://www.testingstandards.co.uk/living_glossary.htm#Testing), February 08, 2009