

**RANCANG BANGUN SOFTWARE ENDURANCE TESTING  
MENGUNAKAN METODE EXTREME PROGRAMMING**

**TUGAS AKHIR**



**Disusun oleh:**

**Rahma Bintang Pratama**

**NIM. 21106050006**

**STATE ISLAMIC UNIVERSITY  
SUNAN KALIJAGA  
YOGYAKARTA**

**PROGRAM STUDI INFORMATIKA**

**FAKULTAS SAINS DAN TEKNOLOGI**

**UNIVERSITAS ISLAM NEGERI SUNAN KALIJAGA**

**YOGYAKARTA**

**2025**

## HALAMAN PENGESAHAN



KEMENTERIAN AGAMA  
UNIVERSITAS ISLAM NEGERI SUNAN KALIJAGA  
FAKULTAS SAINS DAN TEKNOLOGI  
Jl. Marsda Adisucipto Telp. (0274) 540971 Fax. (0274) 519739 Yogyakarta 55281

### PENGESAHAN TUGAS AKHIR

Nomor : B-788/Un.02/DST/PP.00.9/05/2025

Tugas Akhir dengan judul : RANCANG BANGUN SOFTWARE ENDURANCE TESTING MENGGUNAKAN  
METODE EXTREME PROGRAMMING

yang dipersiapkan dan disusun oleh:

Nama : RAHMA BINTANG PRATAMA  
Nomor Induk Mahasiswa : 21106050006  
Telah diujikan pada : Senin, 05 Mei 2025  
Nilai ujian Tugas Akhir : A

dinyatakan telah diterima oleh Fakultas Sains dan Teknologi UIN Sunan Kalijaga Yogyakarta

### TIM UJIAN TUGAS AKHIR



Ketua Sidang

Dr. Agung Fatwanto, S.Si., M.Kom.  
SIGNED

Valid ID: 682b220601d44



Penguji I

Dr. Ir. Sumarsono, S.T., M.Kom.  
SIGNED

Valid ID: 682acae488705



Penguji II

Ir. Maria Ulfah Siregar, S.Kom., MIT., Ph.D.  
SIGNED

Valid ID: 681fc7850ef46



Yogyakarta, 05 Mei 2025  
UIN Sunan Kalijaga

Dekan Fakultas Sains dan Teknologi

Prof. Dr. Dra. Hj. Khurul Wardati, M.Si.  
SIGNED

Valid ID: 682c2095c39b

## HALAMAN PERNYATAAN

### SURAT PERNYATAAN KEASLIAN

Yang bertanda tangan di bawah ini:

Nama : Rahma Bintang Pratama  
NIM : 21106050006  
Program Studi : Informatika  
Fakultas : Sains dan Teknologi  
Dosen Pembimbing : Dr. Agung Fatwanto, S.Si., M.Kom.

Menyatakan dengan sesungguhnya bahwa Tugas Akhir saya yang berjudul **“Rancang Bangun Software Endurance Testing Menggunakan Metode Extreme Programming”** merupakan hasil karya saya sendiri, tidak terdapat pada karya yang pernah diajukan untuk memperoleh gelar akademik di perguruan tinggi manapun, dan sepanjang pengetahuan saya, tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan orang lain, kecuali secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka.

Berdasarkan pernyataan tersebut, Tugas Akhir ini saya ajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana bidang Informatika pada Fakultas Sains dan Teknologi, Universitas Islam Negeri Sunan Kalijaga Yogyakarta.

Yogyakarta, 23 April 2025

Yang menyatakan,



Rahma Bintang Pratama  
NIM. 21106050006

STATE ISLAMIC UNIVERSITY  
SUNAN KALIJAGA  
YOGYAKARTA

## HALAMAN PERSETUJUAN



Universitas Islam Negeri Sunan Kalijaga



FM-UINSK-BM-05-03/R0

### SURAT PERSETUJUAN TUGAS AKHIR

Hal : Surat Persetujuan Tugas Akhir

Lamp : -

Kepada

Yth. Dekan Fakultas Sains dan Teknologi

UIN Sunan Kalijaga Yogyakarta

Di Yogyakarta

*Assalamu 'alaikum Wr. Wb.*

Setelah membaca, meneliti, memberikan petunjuk, mengoreksi, serta mengadakan perbaikan seperlunya, maka kami selaku pembimbing berpendapat bahwa Tugas Akhir saudara:

Nama : Rahma Bintang Pratama

NIM : 21106050006

Judul Tugas Akhir : Rancang Bangun *Software Endurance Testing*  
Menggunakan Metode *Extreme Programming*

sudah dapat diajukan kembali kepada Program Studi Informatika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Sunan Kalijaga Yogyakarta, sebagai salah satu syarat untuk memperoleh gelar Sarjana Strata Satu (S1) dalam Program Studi Informatika.

Dengan ini kami berharap agar Tugas Akhir tersebut dapat segera di-munaqasyahkan. Atas perhatiannya, kami ucapkan terima kasih.

*Wassalamualaikum Wr. Wb.*

Yogyakarta, 23 April 2025

Pembimbing,

Dr. Agung Fatwanto, S.Si., M.Kom.

NIP. 19770103 200501 1 003

# **RANCANG BANGUN SOFTWARE ENDURANCE TESTING MENGUNAKAN METODE EXTREME PROGRAMMING**

**Rahma Bintang Pratama**  
**21106050006**

## **INTISARI**

Seiring dengan pertumbuhan pengguna internet, pengujian kinerja pada aplikasi web menjadi aspek krusial untuk memastikan kestabilan sistem dalam jangka waktu panjang. *Endurance testing* merupakan salah satu metode pengujian kinerja yang digunakan untuk menguji ketahanan aplikasi ketika dijalankan secara terus-menerus di bawah beban tertentu. Namun, beberapa *software* pengujian yang tersedia saat ini memiliki keterbatasan, seperti hanya berbasis *command-line* tanpa antarmuka grafis, penggunaan bahasa pemrograman tertentu yang tidak ramah pengguna, hasil keluaran dalam format yang terbatas, serta memiliki lisensi yang berbayar. Proyek ini bertujuan untuk merancang dan membangun *software endurance testing* untuk menguji kinerja aplikasi web. Rancang bangun dilakukan menggunakan metode *Extreme Programming* dengan pendekatan iteratif yang berfokus pada umpan balik pengguna. Hasil evaluasi menunjukkan skor rata-rata *System Usability Scale* (SUS) sebesar 78,72222222 dengan *margin of error*  $\pm 14,9071195\%$ , menghasilkan rentang kepercayaan dari 66,98700647 hingga 90,45743797. Interpretasi terhadap rentang ini menunjukkan bahwa *software* berada dalam kategori *Grade C* hingga *Grade A+*, dengan predikat *OK* hingga *Best Imaginable* menurut *Adjective Ratings*, masuk dalam kategori *Marginal* hingga *Acceptable* pada *Acceptability Ranges*, serta tergolong *Passive* hingga *Promoter* dalam skala *Net Promoter Score* (NPS).

**Kata kunci:** Aplikasi Web, *Endurance Testing*, *Extreme Programming*, Pengujian Kinerja, *Software*

STATE ISLAMIC UNIVERSITY  
SUNAN KALIJAGA  
YOGYAKARTA

# **DESIGN AND DEVELOPMENT OF ENDURANCE TESTING SOFTWARE USING THE EXTREME PROGRAMMING METHOD**

**Rahma Bintang Pratama**  
**21106050006**

## **ABSTRACT**

*Along with the growth of internet users, performance testing of web applications has become a crucial aspect for ensuring long-term system stability. Endurance testing is one of the performance testing methods used to assess an application's resilience when operated continuously under a specific load. However, several existing testing tools present limitations, such as being command-line based without a graphical interface, requiring the use of specific programming languages that are not user-friendly, producing output in limited formats, and having paid licenses. This project aims to design and develop endurance testing software for evaluating the performance of web applications. The design and development were conducted using the Extreme Programming method through an iterative approach focused on user feedback. The evaluation results show an average System Usability Scale (SUS) score of 78,72222222 with a margin of error of  $\pm 14,9071195\%$ , resulting in a confidence interval ranging from 66,98700647 to 90,45743797. The interpretation of this interval indicates that the software falls within Grade C to Grade A+ category, with ratings from OK to Best Imaginable according to the Adjective Ratings, falls into the Marginal to Acceptable category on the Acceptability Ranges, and is classified from Passive to Promoter on the Net Promoter Score (NPS) scale.*

**Keywords:** Web Applications, Endurance Testing, Extreme Programming, Performance Testing, Software

STATE ISLAMIC UNIVERSITY  
SUNAN KALIJAGA  
YOGYAKARTA

## MOTTO

*“Belajarlah mengucap syukur dari hal-hal baik di hidupmu. Belajarlah menjadi kuat dari hal-hal buruk di hidupmu.”*

~ **Bacharuddin Jusuf Habibie** ~

*“It always seems impossible until it's done.”*

~ **Nelson Rolihlahla Mandela** ~

*“Begitu kamu menyerah, semuanya selesai. Jadi kamu harus melakukan apa yang bisa kamu lakukan. Jangan terlalu berharap.”*

~ **Rimuru Tempest** ~

STATE ISLAMIC UNIVERSITY  
**SUNAN KALIJAGA**  
YOGYAKARTA

## HALAMAN PERSEMBAHAN

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Dengan memanjatkan puji syukur ke hadirat Allah SWT, atas limpahan rahmat, taufik, dan hidayah-Nya, sehingga tugas akhir ini dapat terselesaikan. Sebagai wujud bakti, rasa hormat, dan ungkapan terima kasih yang tulus, karya ini kupersembahkan kepada:

1. Ibu tercinta, Surati, yang menjadi sumber kasih sayang abadi dan untaian doa yang tiada henti. Kelembutan, kesabaran, dan dukungan tulus Ibu telah menjadi sumber kekuatan dan inspirasi terbesar bagi diriku dalam menghadapi segala tantangan. Terima kasih atas segala pengorbanan yang tak terhingga nilainya.
2. Bapak tercinta, Marsono, yang menjadi sosok teladan dalam kegigihan, tanggung jawab, dan kerja keras. Dukungan yang engkau berikan telah menjadi fondasi kokoh bagi diriku dalam menempuh pendidikan dan meraih cita-cita. Terima kasih atas setiap nasihat bijak dan kepercayaan yang tak pernah luntur.

Untuk Ibu dan Bapak, terima kasih atas curahan cinta tak terhingga, doa yang senantiasa mengiringi setiap langkah, dukungan tanpa syarat, serta pengorbanan yang tak ternilai harganya dalam membimbing, mendidik, dan membesarkan diriku hingga mencapai titik ini. Kalian adalah inspirasi terbesar dan sumber semangatku.

STATE ISLAMIC UNIVERSITY  
SUNAN KALIJAGA  
YOGYAKARTA

## KATA PENGANTAR

Puji syukur ke hadirat Allah SWT, atas limpahan rahmat, karunia, serta hidayah-Nya, sehingga penulis diberikan kemudahan dan kekuatan untuk dapat menyelesaikan tugas akhir yang berjudul “Rancang Bangun *Software Endurance Testing* Menggunakan Metode *Extreme Programming*”. Shalawat serta salam semoga senantiasa tercurah kepada Nabi Muhammad SAW, beserta keluarga, sahabat, dan para pengikutnya hingga akhir zaman, yang telah menjadi suri tauladan terbaik bagi seluruh umat manusia.

Penulis menyadari sepenuhnya bahwa penyelesaian tugas akhir ini tidak terlepas dari berbagai tantangan dan memerlukan usaha yang tekun. Namun, berkat dukungan, bimbingan, serta doa dari berbagai pihak, baik secara langsung maupun tidak langsung, akhirnya tugas akhir ini dapat terselesaikan. Untuk itu, dengan segala kerendahan hati, penulis ingin menyampaikan ucapan terima kasih yang sebesar-besarnya kepada:

1. Kedua Orang Tua Tercinta, Bapak Marsono dan Ibu Surati, yang telah merawat dan membesarkan penulis hingga dapat menempuh pendidikan di Perguruan Tinggi. Dengan segala doa, kasih sayang, dukungan, serta motivasi tiada henti yang telah diberikan sepanjang perjalanan hidup penulis.
2. Bapak Prof. Noorhaidi, M.A., M.Phil., Ph.D. selaku Rektor UIN Sunan Kalijaga Yogyakarta.
3. Ibu Prof. Dr. Dra. Hj. Khurul Wardati, M.Si. selaku Dekan Fakultas Sains dan Teknologi UIN Sunan Kalijaga Yogyakarta.
4. Bapak Dr. Muhammad Mustakim, S.T. M.T. selaku Ketua Program Studi Informatika Fakultas Sains dan Teknologi UIN Sunan Kalijaga Yogyakarta.
5. Bapak Ir. Muhammad Didik Rohmad Wahyudi, S.T., MT. selaku Dosen Penasihat Akademik penulis selama berkuliah di UIN Sunan Kalijaga Yogyakarta.
6. Bapak Dr. Agung Fatwanto, S.Si., M.Kom. selaku Dosen Pembimbing Tugas Akhir penulis yang membimbing serta memberikan masukan kepada penulis dalam penulisan dan penyusunan tugas akhir.

7. Seluruh Angkatan Informatika UIN Sunan Kalijaga, khususnya Angkatan 2021, atas kebersamaan, semangat, dukungan, dan kerja sama yang terjalin selama masa perkuliahan.
8. Semua pihak yang telah memberikan bantuan, dukungan, dan doa, baik secara langsung maupun tidak langsung, yang tidak dapat penulis sebutkan satu per satu, namun turut memperlancar penyusunan tugas akhir ini.

Penulis menyadari sepenuhnya bahwa tugas akhir ini masih jauh dari kesempurnaan dan memiliki banyak kekurangan. Oleh karena itu, penulis menerima segala bentuk kritik dan saran yang membangun dari para pembaca demi perbaikan di masa mendatang. Besar harapan penulis agar tugas akhir ini dapat memberikan kontribusi bagi pengembangan ilmu pengetahuan, khususnya dalam bidang pengujian perangkat lunak.

Yogyakarta, 21 April 2025

Penulis

Rahma Bintang Pratama

NIM. 21106050006

STATE ISLAMIC UNIVERSITY  
SUNAN KALIJAGA  
YOGYAKARTA

## DAFTAR ISI

|                                                         |             |
|---------------------------------------------------------|-------------|
| <b>HALAMAN JUDUL .....</b>                              | <b>i</b>    |
| <b>HALAMAN PENGESAHAN .....</b>                         | <b>ii</b>   |
| <b>HALAMAN PERNYATAAN.....</b>                          | <b>iii</b>  |
| <b>HALAMAN PERSETUJUAN.....</b>                         | <b>iv</b>   |
| <b>INTISARI.....</b>                                    | <b>v</b>    |
| <b>ABSTRACT .....</b>                                   | <b>vi</b>   |
| <b>MOTTO .....</b>                                      | <b>vii</b>  |
| <b>HALAMAN PERSEMBAHAN.....</b>                         | <b>viii</b> |
| <b>KATA PENGANTAR.....</b>                              | <b>ix</b>   |
| <b>DAFTAR ISI.....</b>                                  | <b>xi</b>   |
| <b>DAFTAR TABEL.....</b>                                | <b>xiv</b>  |
| <b>DAFTAR GAMBAR.....</b>                               | <b>xv</b>   |
| <b>BAB I PENDAHULUAN .....</b>                          | <b>1</b>    |
| 1.1. Latar Belakang .....                               | 1           |
| 1.2. Rumusan Masalah .....                              | 3           |
| 1.3. Tujuan .....                                       | 3           |
| 1.4. Manfaat .....                                      | 3           |
| <b>BAB II KAJIAN PUSTAKA .....</b>                      | <b>5</b>    |
| <b>BAB III METODE PENGEMBANGAN SISTEM .....</b>         | <b>9</b>    |
| 3.1. Alat dan Bahan .....                               | 9           |
| 3.1.1. Alat.....                                        | 9           |
| 3.1.2. Bahan.....                                       | 10          |
| 3.2. Waktu dan Tempat .....                             | 11          |
| 3.3. Metode Pengembangan .....                          | 11          |
| 3.4. Metode Pengujian.....                              | 12          |
| 3.5. Tahapan Iterasi .....                              | 13          |
| <b>BAB IV PERANCANGAN DAN IMPLEMENTASI SISTEM .....</b> | <b>17</b>   |
| 4.1. Proses Iterasi Pertama .....                       | 17          |
| 4.1.1. <i>Planning</i> Iterasi Pertama .....            | 17          |

|            |                                            |    |
|------------|--------------------------------------------|----|
| 4.1.1.1.   | Analisis Kebutuhan Fungsional.....         | 17 |
| 4.1.1.2.   | Analisis Kebutuhan Non-Fungsional.....     | 18 |
| 4.1.2.     | <i>Design</i> Iterasi Pertama .....        | 18 |
| 4.1.2.1.   | Arsitektur Sistem .....                    | 18 |
| 4.1.2.1.1. | Dekomposisi Sistem .....                   | 18 |
| 4.1.2.1.2. | <i>Mapping</i> .....                       | 19 |
| 4.1.2.1.3. | <i>Platform Unit</i> .....                 | 19 |
| 4.1.2.1.4. | Gaya Arsitektur .....                      | 19 |
| 4.1.2.2.   | <i>Use Case Diagram</i> .....              | 21 |
| 4.1.2.3.   | <i>Activity Diagram</i> .....              | 22 |
| 4.1.2.4.   | Desain <i>User Interface</i> .....         | 25 |
| 4.1.3.     | <i>Coding</i> Iterasi Pertama .....        | 26 |
| 4.1.4.     | <i>Testing</i> Iterasi Pertama.....        | 27 |
| 4.1.5.     | <i>User Feedback</i> Iterasi Pertama ..... | 41 |
| 4.2.       | Proses Iterasi Kedua.....                  | 41 |
| 4.2.1.     | <i>Planning</i> Iterasi Kedua.....         | 42 |
| 4.2.1.1.   | Analisis Kebutuhan Fungsional .....        | 42 |
| 4.2.1.2.   | Analisis Kebutuhan Non-Fungsional.....     | 42 |
| 4.2.2.     | <i>Design</i> Iterasi Kedua.....           | 43 |
| 4.2.2.1.   | <i>Activity Diagram</i> .....              | 43 |
| 4.2.2.2.   | Desain <i>User Interface</i> .....         | 44 |
| 4.2.3.     | <i>Coding</i> Iterasi Kedua.....           | 45 |
| 4.2.4.     | <i>Testing</i> Iterasi Kedua.....          | 46 |
| 4.2.5.     | <i>User Feedback</i> Iterasi Kedua .....   | 52 |
| 4.3.       | Proses Iterasi Ketiga.....                 | 53 |
| 4.3.1.     | <i>Planning</i> Iterasi Ketiga .....       | 54 |
| 4.3.1.1.   | Analisis Kebutuhan Fungsional.....         | 54 |
| 4.3.1.2.   | Analisis Kebutuhan Non-Fungsional.....     | 55 |
| 4.3.2.     | <i>Design</i> Iterasi Ketiga.....          | 55 |
| 4.3.2.1.   | <i>Use Case Diagram</i> .....              | 55 |
| 4.3.2.2.   | <i>Activity Diagram</i> .....              | 57 |

|                                                      |    |
|------------------------------------------------------|----|
| 4.3.2.3. Desain <i>User Interface</i> .....          | 59 |
| 4.3.3. <i>Coding</i> Iterasi Ketiga .....            | 60 |
| 4.3.4. <i>Testing</i> Iterasi Ketiga .....           | 61 |
| 4.3.5. <i>User Feedback</i> Iterasi Ketiga.....      | 68 |
| 4.4. Panduan Pengguna .....                          | 68 |
| 4.4.1. Sistem.....                                   | 69 |
| 4.4.2. Gemini API .....                              | 71 |
| 4.4.3. Discord Webhook URL.....                      | 72 |
| 4.5. Evaluasi <i>Software</i> .....                  | 76 |
| 4.5.1. Analisis Hasil Evaluasi <i>Software</i> ..... | 78 |
| <b>BAB V KESIMPULAN DAN SARAN</b> .....              | 83 |
| 5.1. Kesimpulan .....                                | 83 |
| 5.2. Saran.....                                      | 83 |
| <b>DAFTAR PUSTAKA</b> .....                          | 84 |
| <b>LAMPIRAN</b> .....                                | 88 |

## DAFTAR TABEL

|                                                 |    |
|-------------------------------------------------|----|
| Tabel 1. 1 Perbandingan Software .....          | 2  |
| Tabel 2. 1 Studi Performance Testing.....       | 6  |
| Tabel 3. 1 Spesifikasi Laptop.....              | 9  |
| Tabel 3. 2 Spesifikasi IDE .....                | 9  |
| Tabel 3. 3 Spesifikasi Framework.....           | 10 |
| Tabel 3. 4 Spesifikasi VCS .....                | 10 |
| Tabel 3. 5 Perbandingan LLM .....               | 15 |
| Tabel 4. 1 Test Case Iterasi Pertama.....       | 28 |
| Tabel 4. 2 Test Case Iterasi Kedua.....         | 46 |
| Tabel 4. 3 Test Case Iterasi Ketiga .....       | 62 |
| Tabel 4. 4 Skala Likert.....                    | 76 |
| Tabel 4. 5 Pernyataan Kuesioner .....           | 77 |
| Tabel 4. 6 Skor Hasil Perhitungan SUS .....     | 79 |
| Tabel 4. 7 Penjabaran Kuantitatif Skor SUS..... | 81 |

STATE ISLAMIC UNIVERSITY  
SUNAN KALIJAGA  
YOGYAKARTA

## DAFTAR GAMBAR

|                                                                         |    |
|-------------------------------------------------------------------------|----|
| Gambar 4. 1 Gaya Arsitektur .....                                       | 20 |
| Gambar 4. 2 Use Case Diagram Iterasi Pertama.....                       | 21 |
| Gambar 4. 3 Activity Diagram Help .....                                 | 22 |
| Gambar 4. 4 Activity Diagram Info .....                                 | 23 |
| Gambar 4. 5 Activity Diagram Proses Pengujian Iterasi Pertama .....     | 24 |
| Gambar 4. 6 Desain User Interface Iterasi Pertama.....                  | 25 |
| Gambar 4. 7 Proses Coding Iterasi Pertama.....                          | 26 |
| Gambar 4. 8 Pengujian Input URL .....                                   | 31 |
| Gambar 4. 9 Pengujian Input Request.....                                | 31 |
| Gambar 4. 10 Pengujian Input Timeout.....                               | 32 |
| Gambar 4. 11 Pengujian Input Durasi Pengujian.....                      | 33 |
| Gambar 4. 12 Pengujian Pilih Satuan Durasi Pengujian.....               | 33 |
| Gambar 4. 13 Pengujian Mengirim HTTP Request .....                      | 34 |
| Gambar 4. 14 Pengujian Menghentikan Pengujian Manual.....               | 35 |
| Gambar 4. 15 Pengujian Menghentikan Pengujian Otomatis .....            | 35 |
| Gambar 4. 16 Pengujian Menampilkan Halaman User Guide .....             | 36 |
| Gambar 4. 17 Pengujian Menampilkan Halaman Info.....                    | 37 |
| Gambar 4. 18 Pengujian Menghapus Parameter dan Hasil Pengujian.....     | 37 |
| Gambar 4. 19 Pengujian Mengekspor Hasil Pengujian Ke Format Excel ..... | 38 |
| Gambar 4. 20 Pengujian Mengekspor Hasil Pengujian Ke Format CSV .....   | 39 |
| Gambar 4. 21 Pengujian Mengekspor Hasil Pengujian Ke Format JSON.....   | 39 |
| Gambar 4. 22 Pengujian Mengekspor Hasil Pengujian Ke Format HTML .....  | 40 |
| Gambar 4. 23 Activity Diagram Proses Pengujian Iterasi Kedua .....      | 43 |
| Gambar 4. 24 Desain User Interface Iterasi Kedua.....                   | 44 |
| Gambar 4. 25 Proses Coding Iterasi Kedua .....                          | 45 |
| Gambar 4. 26 Logika Penghitungan Jumlah Request .....                   | 45 |
| Gambar 4. 27 Pengujian Memilih Mode Stable .....                        | 48 |
| Gambar 4. 28 Pengujian Memilih Mode Progressive .....                   | 49 |
| Gambar 4. 29 Pengujian Memilih Mode Fluctuative.....                    | 50 |

|                                                                                        |    |
|----------------------------------------------------------------------------------------|----|
| Gambar 4. 30 Pengujian Input Max Request .....                                         | 50 |
| Gambar 4. 31 Pengujian Input Min dan Max Request.....                                  | 51 |
| Gambar 4. 32 Use Case Diagram Iterasi Ketiga .....                                     | 55 |
| Gambar 4. 33 Activity Diagram Analisis Test Summary .....                              | 57 |
| Gambar 4. 34 Activity Diagram Notifikasi Pada Server Discord.....                      | 58 |
| Gambar 4. 35 Desain User Interface Iterasi Ketiga .....                                | 59 |
| Gambar 4. 36 Proses Coding Iterasi Ketiga .....                                        | 60 |
| Gambar 4. 37 Pengujian Analisis AI.....                                                | 63 |
| Gambar 4. 38 Pengujian Notifikasi Ke Server Discord (Tampilan Software).....           | 64 |
| Gambar 4. 39 Pengujian Notifikasi Ke Server Discord (Tampilan Server Discord)<br>..... | 64 |
| Gambar 4. 40 Pengujian Tampilan Splash Screen .....                                    | 65 |
| Gambar 4. 41 Pengujian Input Request Maksimal .....                                    | 66 |
| Gambar 4. 42 Pengujian Input Timeout Maksimal.....                                     | 66 |
| Gambar 4. 43 Pengujian Input Durasi Pengujian Maksimal.....                            | 67 |
| Gambar 4. 44 Halaman Utama Software .....                                              | 69 |
| Gambar 4. 45 Persetujuan Pengguna Gemini .....                                         | 72 |
| Gambar 4. 46 Pembuatan API Key .....                                                   | 72 |
| Gambar 4. 47 Penyiapan API Key .....                                                   | 72 |
| Gambar 4. 48 Pembuatan Server Discord.....                                             | 73 |
| Gambar 4. 49 Pilihan Server Discord .....                                              | 73 |
| Gambar 4. 50 Penyesuaian Server Discord.....                                           | 74 |
| Gambar 4. 51 Penamaan Server Discord .....                                             | 74 |
| Gambar 4. 52 Ikon Konfigurasi Channel Discord .....                                    | 74 |
| Gambar 4. 53 Menu Konfigurasi Channel Discord .....                                    | 75 |
| Gambar 4. 54 Pembuatan Webhook.....                                                    | 75 |
| Gambar 4. 55 Penyiapan Webhook.....                                                    | 75 |
| Gambar 4. 56 Klasifikasi Skor SUS.....                                                 | 81 |

# BAB I

## PENDAHULUAN

### 1.1. Latar Belakang

Di era digital saat ini, aplikasi web telah menjadi bagian integral dari berbagai aspek kehidupan modern. Di Indonesia, pertumbuhan pengguna internet menunjukkan peningkatan yang konsisten dan signifikan, dengan tingkat penetrasi mencapai 79,5% pada tahun 2024, meningkat 1,4% dari tahun sebelumnya [1]. Baik untuk kebutuhan komunikasi, perdagangan, hiburan, maupun pendidikan, aplikasi web memainkan peran penting dalam menyediakan layanan dan informasi secara luas dan mudah diakses. Seiring dengan pertumbuhan aplikasi web, kebutuhan untuk memastikan kualitas dan kinerja aplikasi tersebut juga semakin meningkat [2]. Pengujian kinerja (*performance testing*) menjadi aspek krusial dalam rekayasa perangkat lunak untuk memastikan aplikasi web dapat beroperasi dengan stabil, responsif, dan efisien, terutama dalam menghadapi beban kerja yang berkelanjutan.

Salah satu jenis pengujian kinerja yang penting adalah *endurance testing*, yang juga dikenal sebagai *soak testing*. *Endurance testing* berfokus pada evaluasi kemampuan aplikasi untuk mempertahankan kinerja yang stabil dalam jangka waktu yang lama di bawah beban kerja yang berkelanjutan. Pengujian ini sangat penting untuk mengungkap masalah yang mungkin tidak muncul dalam pengujian jangka pendek, seperti *memory leak*, *resource depletion*, atau penurunan kinerja secara bertahap [3]. Dengan melakukan *endurance testing*, pengembang dapat mengidentifikasi dan mengatasi potensi masalah kinerja sebelum aplikasi digunakan secara luas oleh pengguna.

Meskipun penting, implementasi *endurance testing* sering kali menghadapi berbagai tantangan yang perlu diperhatikan. Salah satu tantangan utama adalah kebutuhan untuk menggunakan *software* atau alat bantu yang tepat agar pengujian dapat dilakukan secara efektif. Terdapat beberapa *software* yang telah mendukung *endurance testing*, seperti Siege, LoadRunner, Gatling, dan Loadster. Tabel 1.1 berikut merangkum kelebihan dan kekurangan masing-masing *software* tersebut [4], [5], [6]:

Tabel 1. 1 Perbandingan Software

| Nama       | Kelebihan                                                                              | Kekurangan                                                                                                    |
|------------|----------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------|
| Siege      | Memiliki kecepatan <i>setup</i> yang cepat serta memberikan hasil dengan cepat         | Hanya berbasis <i>command-line</i> tanpa antarmuka GUI                                                        |
| LoadRunner | Optimal dalam menguji beban berat dan mampu menangani banyak pengguna secara bersamaan | Lisensi yang mahal                                                                                            |
| Gatling    | Skrip yang ringan, mudah dibaca, dan berbasis bahasa pemrograman objek                 | Menggunakan bahasa Scala yang mungkin menjadi hambatan bagi pengguna yang tidak memiliki keahlian pemrograman |
| Loadster   | Memiliki editor grafis yang memudahkan pengguna dalam membuat skenario pengujian       | Hanya menghasilkan laporan dalam format HTML                                                                  |

Sebagai alternatif dari *software* yang telah ada, proyek ini berfokus untuk merancang dan membangun *software endurance testing* berbasis GUI yang lebih mudah digunakan tanpa memerlukan kemampuan pemrograman. Selain itu, *software* ini bersifat *open-source* dan gratis, sehingga lebih mudah diakses oleh berbagai kalangan pengembang dan peneliti. *Software* ini juga menyediakan fleksibilitas dalam ekspor hasil pengujian ke berbagai format, termasuk Excel, CSV, JSON, dan HTML, yang memungkinkan analisis data lebih lanjut sesuai kebutuhan pengguna.

Dalam konteks ini, penting untuk merancang dan membangun *software endurance testing* yang dapat digunakan untuk menguji kinerja aplikasi web. Dengan meningkatnya kompleksitas aplikasi dan jumlah pengguna yang terus bertambah, kebutuhan akan pengujian yang komprehensif menjadi semakin mendesak. Proyek ini berfokus pada rancang bangun *software endurance testing* menggunakan metode *Extreme Programming* sebagai kerangka kerja pengembangan *software* yang berfokus pada fleksibilitas dan respons cepat terhadap perubahan. Metode ini mendorong proses rancang bangun *software* secara iteratif dengan siklus rilis yang lebih pendek, memastikan bahwa *software* diuji dan disempurnakan secara berkelanjutan. Dengan keterlibatan aktif antara pengembang dan pengguna, metode ini memungkinkan adaptasi yang lebih baik terhadap kebutuhan yang berkembang. Pendekatan ini juga berfokus pada implementasi yang adaptif, memungkinkan pengembang untuk menyesuaikan *software* dengan kebutuhan yang berkembang selama proses rancang bangun.

## **1.2. Rumusan Masalah**

Berdasarkan latar belakang yang telah dipaparkan, rumusan masalah dalam proyek ini adalah: “Bagaimana merancang dan membangun *software endurance testing* untuk menguji kinerja aplikasi web menggunakan metode *Extreme Programming*?”.

## **1.3. Tujuan**

Dari rumusan masalah yang telah disebutkan, tujuan yang ingin dicapai dari proyek ini adalah untuk merancang dan membangun *software endurance testing* untuk menguji kinerja aplikasi web menggunakan metode *Extreme Programming*.

## **1.4. Manfaat**

Hasil dari rancang bangun ini diharapkan dapat memberikan sejumlah manfaat sebagai berikut:

a. Bagi Pengembang Web

Menyediakan alat yang dapat membantu pengembang dalam menguji kinerja aplikasi web, sehingga dapat meningkatkan keandalan dan kepuasan pengguna.

b. Bagi Industri Berbasis Teknologi

Menyediakan solusi yang dapat digunakan oleh perusahaan dalam memastikan aplikasi web mereka mampu beroperasi dengan baik di bawah beban kerja yang berkelanjutan serta mengurangi risiko *downtime* dan masalah kinerja.

c. Bagi Pengembang *Software*

Menjadi referensi bagi pengembangan *software* lebih lanjut dalam bidang pengujian kinerja aplikasi web khususnya *endurance testing* untuk mengembangkan alat pengujian yang lebih canggih.

d. Bagi Pendidikan

Menjadi sumber pembelajaran bagi mahasiswa dan praktisi di bidang rekayasa perangkat lunak mengenai pentingnya pengujian kinerja dan metode yang dapat digunakan untuk melakukannya.

## BAB V

### KESIMPULAN DAN SARAN

#### 5.1. Kesimpulan

Rancang bangun *software endurance testing* untuk menguji kinerja aplikasi web telah berhasil dilakukan menggunakan metode *Extreme Programming*. Metode ini memungkinkan proses rancang bangun dilakukan secara iteratif dengan siklus rilis yang lebih cepat, sehingga *software* dapat diuji dan disempurnakan secara berkelanjutan sesuai kebutuhan pengguna. *Software* yang dikembangkan memiliki antarmuka grafis (GUI) yang memudahkan pengguna tanpa memerlukan keahlian pemrograman. Selain itu, *software* ini bersifat *open-source* dan menyediakan fitur ekspor hasil pengujian dalam berbagai format seperti Excel, CSV, JSON, dan HTML, sehingga memungkinkan fleksibilitas dalam analisis data. Dengan adanya *software* ini, pengujian dapat dilakukan dengan lebih mudah dan terstruktur. Oleh karena itu, tugas akhir ini telah mencapai tujuan utama, yaitu merancang dan membangun *software endurance testing* menggunakan metode *Extreme Programming*.

#### 5.2. Saran

Rancang bangun *software endurance testing* telah memenuhi tujuan yang ditetapkan, namun masih terdapat beberapa aspek yang dapat ditingkatkan di masa mendatang. Oleh karena itu, beberapa saran untuk pengembangan lebih lanjut adalah sebagai berikut:

1. Otomatisasi analisis hasil pengujian dengan menambahkan fitur deteksi anomali berbasis *machine learning*.
2. Menambahkan dukungan untuk sistem operasi lain seperti Linux dan macOS.

Dengan adanya pengembangan lebih lanjut, diharapkan *software* ini dapat memberikan manfaat yang lebih luas bagi pengembang dan peneliti dalam melakukan pengujian kinerja aplikasi web.

## DAFTAR PUSTAKA

- [1] S. M. Prasetyo, R. Gustiawan, Farhat and F. R. Albani, "Analisis Pertumbuhan Pengguna Internet Di Indonesia," *BIIKMA : Buletin Ilmiah Ilmu Komputer dan Multimedia*, vol. 2, no. 1, pp. 65-71, June 2024.
- [2] D. M. Dave and A. Bhanushali, "Performance Testing: Methodology for Determining Scalability of Web Systems," *International Journal of Science and Research (IJSR)*, vol. 13, no. 1, pp. 1254-1261, January 2024.
- [3] S. Pargaonkar, "A Comprehensive Review of Performance Testing Methodologies and Best Practices: Software Quality Engineering," *International Journal of Science and Research (IJSR)*, vol. 12, no. 8, pp. 2008-2014, August 2023.
- [4] R. Abbas, Z. Sultan and S. N. Bhatti, "Comparative Analysis of Automated Load Testing Tools: Apache JMeter, Microsoft Visual Studio (TFS), LoadRunner, Siege," in *2017 International Conference on Communication Technologies (ComTech)*, Rawalpindi, 2017.
- [5] A. Kołtun and B. Pańczyk, "Comparative analysis of web application performance testing tools," *Journal Computer Sciences Institute*, vol. 17, pp. 351-357, December 2020.
- [6] M. Hayek and P. Farhat, "Web 2.0 Testing Tools: A Compendium," in <https://doi.org/10.1109/3ICT.2019.8910274>, Sakhier, 2019.
- [7] Indrianto, "Performance Testing on Web Information System Using Apache JMeter and BlazeMeter," *Jurnal Ilmiah Ilmu Terapan Universitas Jambi*, vol. 7, no. 2, pp. 138-149, December 2023.
- [8] N. M, K. K. S, A. Saha and D. Chahar, "Comparative Study on Performance Testing with JMeter," *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 5, no. 2, pp. 70-76, October 2016.

- [9] A. Z. Musthafawi, A. Mas'adah, Sukmadiningtyas and F. Ramdani, "Performance Testing on the Shopee Website in the Pandemic Period of COVID-19," in *SIET '20: 5th International Conference on Sustainable Information Engineering and Technology*, Malang, 2020.
- [10] M. A. Putri, "Implementing and Analyzing Web Performance Testing for Universitas Terbuka's Website with GTMetrix and Pingdom," *Jurnal Teknologi Sistem Informasi dan Aplikasi*, vol. 7, no. 4, pp. 1598-1602, October 2024.
- [11] J. R. J. Sumakul and E. Mailoa, "Analisa Performa Website Kabupaten Kota di Provinsi Sulawesi Utara Menggunakan Website Performance Testing Tools," *Jurnal Minfo Polgan*, vol. 12, no. 1, pp. 1262-1271, 1 July 2023.
- [12] M. R. Maulana, E. B. Susanto and S. W. Binabar, "Analisis Kinerja Website Pemerintah Kota Pekalongan," *Jurnal Litbang Kota Pekalongan*, vol. 19, no. 1, pp. 48-54, 1 July 2021.
- [13] A. Ismail, A. Y. Ananta, S. N. Arief and E. N. Hamdana, "Performance Testing Sistem Ujian Online Menggunakan JMeter Pada Lingkungan Virtual," *JIP (Jurnal Informatika Polinema)*, vol. 9, no. 2, pp. 159-164, February 2023.
- [14] Q. M. Yas, A. ALazzawi and B. Rahmatullah, "A Comprehensive Review of Software Development Life Cycle methodologies: Pros, Cons, and Future Directions," *Iraqi Journal for Computer Science and Mathematics*, vol. 4, no. 4, pp. 173-190, 2023.
- [15] Mintarsih, "Pengujian Black Box Dengan Teknik Transition Pada Sistem Informasi Perpustakaan Berbasis Web Dengan Metode Waterfall Pada SMC Foundation," *Jurnal Teknologi Dan Sistem Informasi Bisnis*, vol. 5, no. 1, pp. 33-35, January 2023.
- [16] Supriyono, "Software Testing with the approach of Blackbox Testing on the Academic Information System," *International Journal of Information System & Technology*, vol. 3, no. 2, pp. 227-233, 2020.

- [17] D. I. Pirdaus and R. A. Hidayana, "Analysis Testing Black Box and White Box on Application To-Do List Based Web," *International Journal of Mathematics, Statistics, and Computing (IJMSC)*, vol. 2, no. 2, pp. 68-75, 2024.
- [18] A. Ali, H. A. Maghawry and N. Badr, "Automation of Performance Testing: A Review," *International Journal of Intelligent Computing and Information Sciences*, vol. 22, no. 4, pp. 35-50, December 2022.
- [19] F. Neha and D. Bhati, "A Survey of DeepSeek Models," 7 February 2025. [Online]. Available: <https://www.techrxiv.org/doi/full/10.36227/techrxiv.173896582.25938392>. [Accessed 18 March 2025].
- [20] M. A. Valashani and A. M. Abukari, "ERP Systems Architecture For The Modern Age: A Review of The State of The Art Technologies," *Journal of Applied Intelligent Systems & Information Sciences (JAISIS)*, vol. 1, no. 2, pp. 70-90, August 2020.
- [21] Sugiyono, METODE PENELITIAN KUANTITATIF, KUALITATIF, DAN R&D, 25 ed., Bandung: Alfabeta, 2017, p. 93.
- [22] L. S. Vailshery, "Developers population worldwide 2018-2024," Statista, 26 November 2024. [Online]. Available: <https://www.statista.com/statistics/627312/worldwide-developer-population/>. [Accessed 9 April 2025].
- [23] S. B. Nyambaka and C. Mutwiri, "Usability of KOHA OPAC by Undergraduate Users for Information Retrieval with Regard to Usability Testing in Technical University of Kenya Library," *International Journal of Current Aspects*, vol. 7, no. 1, pp. 37-50, 2023.
- [24] J. Brooke, "SUS -- a quick and dirty usability scale," 1 January 1996. [Online]. Available: [https://www.researchgate.net/publication/319394819\\_SUS\\_-\\_a\\_quick\\_and\\_dirty\\_usability\\_scale](https://www.researchgate.net/publication/319394819_SUS_-_a_quick_and_dirty_usability_scale). [Accessed 9 April 2025].

- [25] M. Agustina and S. Aqdi, "Evaluation of The South Sumatra Samsat E-Demopo Application using the System Usability Scale Method," *International Journal of Information System & Technology*, vol. 7, no. 2, pp. 151-161, 2023.
- [26] J. Sauro, "5 Ways to Interpret a SUS Score," 19 September 2018. [Online]. Available: <https://measuringu.com/interpret-sus-score/>. [Accessed 21 April 2025].
- [27] S. I. Putri and K. Liu, "Assessing Ticket.com App Usability Through the System Usability Scale (SUS) Method," *International Journal for Applied Information Management*, vol. 4, no. 1, pp. 30-40, April 2024.
- [28] J. R. Lewis and J. Sauro, "Item Benchmarks for the System Usability Scale," *Journal of Usability Studies*, vol. 13, no. 3, pp. 158-167, May 2018.
- [29] B. Lestari, P. I. Rifiani and A. B. Gati, "The Use of the Usability Scale System as an Evaluation of the Kampung Heritage Kajoetangan Guide Ebook Application," *European Journal of Business and Management Research*, vol. 6, no. 6, pp. 156-161, December 2021.
- [30] O. Suria, "A Statistical Analysis of System Usability Scale (SUS) Evaluations in Online Learning Platform," *Journal of Information Systems and Informatics*, vol. 6, no. 2, pp. 992-1007, June 2024.