

**PENERAPAN STANDAR ISO/IEC 25010:2023 UNTUK EVALUASI
KUALITAS PERANGKAT LUNAK PADA SISTEM INFORMASI
MONITORING PEMBESARAN KAMBING (SI MBEK)**

(Skripsi)

Oleh

**RIZKI PANGESTU
NPM 2115061082**



**FAKULTAS TEKNIK
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2025**

**PENERAPAN STANDAR ISO/IEC 25010:2023 UNTUK EVALUASI
KUALITAS PERANGKAT LUNAK PADA SISTEM INFORMASI
MONITORING PEMBESARAN KAMBING (SI MBEK)**

Oleh

RIZKI PANGESTU

Skripsi

Sebagai Salah Satu untuk Mencapai Gelar

SARJANA TEKNIK

Pada

Program Studi Teknik Informatika

Jurusan Teknik Elektro

Fakultas Teknik Universitas Lampung



**FAKULTAS TEKNIK
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2025**

ABSTRAK

PENERAPAN STANDAR ISO/IEC 25010:2023 UNTUK EVALUASI KUALITAS PERANGKAT LUNAK PADA SISTEM INFORMASI MONITORING PEMBESARAN KAMBING (SI MBEK)

Oleh

RIZKI PANGESTU

Sistem Informasi Monitoring Pembesaran Kambing (SI MBEK) dikembangkan untuk membantu peternak memantau pertumbuhan kambing secara digital. Namun, sistem ini belum melalui evaluasi kualitas berbasis standar sehingga diperlukan pengujian untuk memastikan keandalan dan kelayakannya. Penelitian ini bertujuan mengevaluasi SI MBEK menggunakan standar ISO/IEC 25010:2023 dengan pendekatan *Software Testing Life Cycle* (STLC). Pengujian dilakukan pada tujuh karakteristik, yaitu *functional suitability*, *performance efficiency*, *compatibility*, *interaction capability*, *reliability*, *security*, dan *maintainability* menggunakan alat *Katalon Studio*, *GTmetrix*, *VPS*, *PowerMapper*, *Apache JMeter*, *BlazeMeter*, *WAPT*, *UEQ*, *UAT*, *OWASP ZAP*, dan *SonarQube Cloud*. Hasil penelitian menunjukkan SI MBEK memenuhi sebagian besar aspek kualitas. Sistem berfungsi sesuai kebutuhan, memiliki kinerja efisien, kompatibel lintas perangkat, serta mendapat penilaian *excellent* pada UEQ dan skor 89,92% pada UAT. Sistem reliabel hingga 400 pengguna simultan, walaupun masih terdapat kerentanan keamanan tingkat rendah dan beberapa bagian kode yang masih perlu perbaikan. Dengan demikian, penerapan ISO/IEC 25010:2023 terbukti efektif dalam mengevaluasi kualitas SI MBEK, sekaligus memberikan masukan untuk peningkatan keamanan dan pemeliharaan sistem ke depan.

Kata Kunci: ISO/IEC 25010, STLC, SI MBEK, pengujian perangkat lunak.

ABSTRACT

IMPLEMENTATION OF ISO/IEC 25010:2023 STANDARD FOR SOFTWARE QUALITY EVALUATION ON THE SISTEM INFORMASI MONITORING PEMBESARAN KAMBING (SI MBEK)

By

RIZKI PANGESTU

The goat husbandry monitoring information system, known as SI MBEK, was developed to assist farmers in digitally monitoring goat growth. However, the system has not yet undergone a standardized quality evaluation, thus requiring testing to ensure its reliability and feasibility. This study aims to evaluate SI MBEK using the ISO/IEC 25010:2023 standard with the Software Testing Life Cycle (STLC) approach. The evaluation was conducted on seven quality characteristics, namely functional suitability, performance efficiency, compatibility, interaction capability, reliability, security, and maintainability, using tools such as Katalon Studio, GTmetrix, PowerMapper, Apache JMeter, VPS, BlazeMeter, WAPT, UEQ, UAT, OWASP ZAP, and SonarQube Cloud. The results show that SI MBEK meets most of the quality aspects. The system functions as required, demonstrates efficient performance, is compatible across devices, and achieved an excellent rating in the UEQ along with a score of 89.92% in User Acceptance Testing (UAT). The system is reliable for up to 400 simultaneous users, although minor security vulnerabilities and some code issues remain to be improved. In conclusion, the application of ISO/IEC 25010:2023 has proven effective in evaluating the quality of SI MBEK while also providing insights for future improvements in security and system maintainability.

Keywords: ISO/IEC 25010, STLC, SI MBEK, software quality evaluation.

Judul Skripsi

: PENERAPAN STANDAR ISO/IEC 25010:2023
UNTUK EVALUASI KUALITAS PERANGKAT
LUNAK PADA SISTEM INFORMASI
MONITORING PEMBESARAN KAMBING (SI
MBEK)

Nama Mahasiswa

: Rizki Pangestu

Nomor Pokok Mahasiswa

: 2115061082

Program Studi

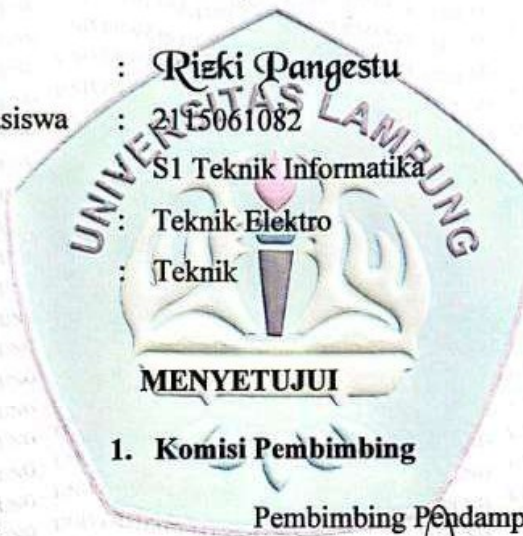
: S1 Teknik Informatika

Jurusan

: Teknik Elektro

Fakultas

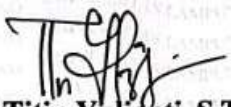
: Teknik



1. Komisi Pembimbing

Pembimbing Utama

Pembimbing Pendamping


Ir. Titin Yulianti, S.T., M.Eng.

NIP. 198807092019032015


Deny Budiyanto, S.Kom., M.T.

NIP. 199112082019031011

2. Mengetahui

Ketua Jurusan

Ketua Program Studi

Teknik Elektro

Teknik Informatika


Herlinawati, S.T., M.T.

NIP. 197103141999032001


Yessi Mulyani, S.T., M.T.

NIP. 197312262000122001

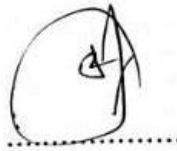
MENGESAHKAN

1. Tim Penguji

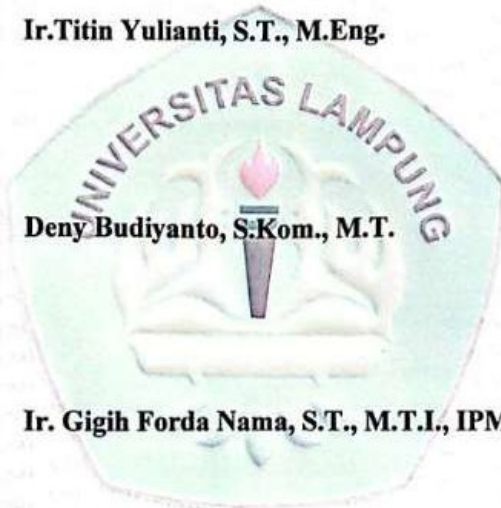
Ketua : Ir. Titin Yulianti, S.T., M.Eng.



Sekretaris : Deny Budiyanto, S.Kom., M.T.



Penguji : Ir. Gigih Forda Nama, S.T., M.T.I., IPM.



2. Dekan Fakultas Teknik



Dr. Eng. Ir. Helmy Fitriawan, S.T., M.Sc.

NIP. 197509282001121002

Tanggal Lulus Ujian Skripsi : 13 Oktober 2025

SURAT PERNYATAAN

Saya yang bertanda tangan dibawah ini, menyatakan dengan sebenarnya bahwa skripsi saya yang berjudul “Penerapan Standar ISO/IEC 25010:2023 untuk Evaluasi Kualitas Perangkat Lunak pada Sistem Informasi Monitoring Pembesaran Kambing (SI MBEK)” merupakan hasil karya saya sendiri. Semua hasil yang tertuang dalam skripsi ini telah mengikuti kaidah penulisan karya ilmiah Universitas Lampung. Apabila di kemudian hari terbukti bahwa skripsi saya ini merupakan salinan atau dibuat oleh orang lain, maka saya bersedia menerima sanksi sesuai dengan ketentuan hukum atau akademik yang berlaku.

Bandar Lampung, 13 Oktober 2025

Pembuat Pernyataan



Rizki Pangestu

2115061082

RIWAYAT HIDUP



Rizki Pangestu dilahirkan di Sidorejo pada tanggal 22 April 2002 sebagai anak keempat dari pasangan (Alm.) Bapak Bahrhun dan Ibu Yuryanah. Riwayat pendidikan dimulai dari Taman Kanak-Kanak (TK) Permata Ibu pada tahun 2008, kemudian dilanjutkan ke Sekolah Dasar Negeri 3 Sidorejo dan diselesaikan pada tahun 2015. Pendidikan menengah pertama ditempuh di Sekolah Menengah Pertama (SMP) Negeri 1 Kalirejo hingga lulus pada tahun 2018. Selanjutnya, pendidikan menengah atas dijalani di Sekolah Menengah Atas (SMA) Negeri 1 Kalirejo dan berhasil diselesaikan pada tahun 2021. Pada tahun yang sama, pendidikan tinggi dilanjutkan di Universitas Lampung, Fakultas Teknik, Jurusan Teknik Elektro, Program Studi S1 Teknik Informatika melalui jalur SBMPTN (Seleksi Bersama Masuk Perguruan Tinggi Negeri). Selama masa perkuliahan, Rizki Pangestu aktif mengikuti berbagai kegiatan akademik maupun non-akademik, di antaranya:

1. Menjadi anggota Himpunan Mahasiswa Teknik Elektro (HIMATRO) pada periode 2021-2022.
2. Mengikuti program Merdeka Belajar Kampus Merdeka (MBKM) Batch 5 pada tahun 2023 melalui program *Studi Independen* sebagai UI/UX Designer di PT GreatEdu Global Mahardika.
3. Melaksanakan Kuliah Kerja Nyata (KKN) di Desa Lembasung, Kecamatan Blambangan Umpu, Kabupaten Way Kanan, Provinsi Lampung pada tahun 2024.

4. Melaksanakan Kerja Praktik (KP) di SMP Al Kautsar Bandar Lampung pada bulan Juni hingga Agustus 2024.
5. Berperan sebagai Asisten Praktikum di Laboratorium Teknik Komputer Universitas Lampung pada tahun 2023-2025.
6. Mengikuti program MBKM Batch 7 pada tahun 2024 melalui program Magang sebagai *System Analyst & Quality Assurance* di Diskominfotiksan Kabupaten Pesawaran, Provinsi Lampung.
7. Berpartisipasi dalam kegiatan Pengabdian kepada Masyarakat tahun 2024 dengan judul “Pembuatan dan Pendampingan SI MBEK: Sistem Informasi Monitoring Pembesaran Kambing di Desa Rukti Endah-Seputih Raman, Kabupaten Lampung Tengah, Provinsi Lampung” dengan penempatan pada bidang UI/UX.
8. Berpartisipasi dalam kegiatan Pengabdian kepada Masyarakat tahun 2025 dengan judul “Pemberdayaan UMKM Peternakan Melalui Pengembangan dan Penerapan SI MBEK Sebagai Solusi Digital Terpadu Menuju Technopreneurship di Desa Rukti Endah” dengan penempatan pada bidang *Quality Assurance*.

Dengan pengalaman akademik, organisasi, pengabdian masyarakat, serta program magang yang telah ditempuh, harapan ke depan adalah agar kemampuan di bidang Teknik Informatika terus dapat dikembangkan, khususnya dalam pengembangan sistem informasi, analisis kualitas perangkat lunak, serta penerapan teknologi digital untuk mendukung inovasi di masyarakat.

MOTTO

“Maka, ingatlah kepada-Ku, Aku pun akan ingat kepadamu. Bersyukurlah kepada-Ku dan janganlah kamu ingkar kepada-Ku.”

(QS. Al-Baqarah: 152)

“Wahai orang-orang yang beriman, mohonlah pertolongan (kepada Allah) dengan sabar dan salat. Sesungguhnya Allah beserta orang-orang yang sabar.”

(QS. Al-Baqarah: 153)

“... Boleh jadi kamu membenci sesuatu, padahal ia amat baik bagimu, dan boleh jadi (pula) kamu menyukai sesuatu, padahal ia amat buruk bagimu; Allah mengetahui, sedang kamu tidak mengetahui.”

(QS. Al-Baqarah: 216)

“Semua masalah kegelasahan dan kekhawatiran adalah bentuk ketidakmampuan kita bersyukur.”

(Fulan)

"Susah tapi Bismillah, semua orang punya waktu terbaiknya masing-masing. Takdir itu milik Allah, doa dan usaha itu milik kita. Biasanya kalo jalannya berat akhirnya sangat indah. Langit tak selamanya mendung, tak selamanya gelap, adakalanya langit indah, cerah dan mempesona datang menyapa."

(Fulan)

PERSEMBAHAN

Alhamdulillahirabbil‘alamin.

Segala puji hanya bagi Allah SWT yang telah melimpahkan rahmat, karunia, serta kekuatan sehingga skripsi ini dapat diselesaikan dengan baik. Shalawat dan salam senantiasa tercurah kepada junjungan Nabi Muhammad SAW, teladan sepanjang masa, yang telah membawa umat manusia dari kegelapan menuju cahaya kebenaran.

**Dengan Penuh Rasa Syukur dan Kerendahan Hati, Karya Sederhana Ini
Kupersembahkan Kepada:**

Kedua orang tuaku tercinta, (Alm.) Bapak Bahrin dan Ibu Yuryanah, yang selalu melantikan doa-doa terbaik, kasih sayang, dan pengorbanan tiada henti. Terima kasih atas cinta tanpa batas, nasihat penuh makna, serta keteladanan hidup yang telah menjadi cahaya dalam setiap langkahku. Semoga Allah SWT senantiasa melimpahkan rahmat, ampunan, dan perlindungan-Nya. Aamiin.

Kakakku tercinta, Fatoni Latif, Bayu Astuti, dan Panggah Santoso, yang selalu hadir sebagai penyemangat, penguat, dan pemberi doa di setiap perjalanan pencapaianku. Terima kasih atas dukungan yang tulus dan kebersamaan yang penuh makna.

Seluruh dosen, civitas akademika, serta rekan-rekan seperjuangan Teknik Informatika Universitas Lampung, atas ilmu, bimbingan, motivasi, serta kebersamaan yang menjadikan perjalanan akademik ini penuh warna dan berharga.

SANWACANA

Alhamdulillah Allah SWT atas limpahan rahmat dan hidayah-Nya, sehingga skripsi berjudul “Penerapan Standar ISO/IEC 25010:2023 untuk Evaluasi Kualitas Perangkat Lunak pada Sistem Informasi Monitoring Pembesaran Kambing (SI MBK)” dapat diselesaikan sebagai salah satu syarat memperoleh gelar Sarjana Teknik pada Program Studi Teknik Informatika, Fakultas Teknik, Universitas Lampung. Berkat doa, dukungan, dan bantuan berbagai pihak, penyusunan skripsi ini dapat terselesaikan dengan baik. Ucapan terima kasih disampaikan kepada semua pihak yang telah memberikan dukungan, baik secara langsung maupun tidak langsung. kepada:

1. Bapak Dr. Eng. Ir. Helmy Fitriawan, S.T., M.Sc., selaku Dekan Fakultas Teknik Universitas Lampung.
2. Ibu Herlinawati, S.T., M.T., selaku Ketua Jurusan Teknik Elektro Universitas Lampung.
3. Ibu Yessi Mulyani, S.T., M.T., selaku Ketua Program Studi Teknik Informatika Universitas Lampung.
4. Ibu Ir. Titin Yulianti, S.T., M.Eng., selaku Pembimbing Utama, atas bimbingan, arahan, serta dukungan yang sangat berarti selama proses penelitian hingga dapat terselesaikan dengan baik, sekaligus selaku Kepala Laboratorium yang telah memberikan fasilitas serta ilmu berharga.
5. Bapak Deny Budiyanto, S.Kom., M.T., selaku Pembimbing Pendamping, yang dengan sabar memberikan waktu, ilmu, dan dukungan hingga penelitian ini dapat terselesaikan.
6. Bapak Rio Ariestia Pradipta, S.Kom., M.T.I., selaku Pembimbing Akademik, atas nasihat, arahan, dan motivasi yang sangat membantu selama proses perkuliahan dan penyusunan skripsi. Terutama dan paling utama kepada kedua orang tua dan keluarga tercinta, (Alm.) Bapak Bahrin dan Ibu

Yuryanah, serta kakak-kakakku, atas doa, dukungan, semangat, dan pengorbanan yang tidak ternilai harganya.

7. Seluruh Asisten Laboratorium Teknik Komputer angkatan 2021-2023 atas pengalaman, kerja sama, dan kebersamaan yang terjalin selama ini.
8. Tim SI MBEK, yaitu Erika Dwi Utami dan Teguh Karya Rizki, atas kerja sama, dukungan, serta perjuangan bersama dalam penyelesaian penelitian ini.
9. Sahabat Livinginbrg yaitu Dimas Ardi Kusuma, Alfi Julian Ashari, Chelly Sabrina, Helma Agustina, dan Agata Sekar Viranti Mukti atas semangat, dukungan, serta kebersamaan dalam setiap proses. Terima kasih juga kepada rekan-rekan Uduk-Uduk Aja atas kebersamaan selama perkuliahan, serta Teman Cerita Random yaitu Budi Cahyono, Teguh Karya Rizki, Yos Marison Sianipar, dan Gibran Alfarabi atas kebersamaan dalam suka maupun duka.
10. Seluruh mahasiswa Teknik Informatika Universitas Lampung atas persahabatan dan kebersamaan selama masa studi, serta seluruh civitas akademika Jurusan Teknik Elektro Universitas Lampung atas dukungan dan bantuan selama proses penyusunan skripsi. Terima kasih juga kepada semua pihak yang tidak dapat disebutkan satu per satu atas segala bentuk dukungan, baik langsung maupun tidak langsung.

Skripsi ini masih memiliki berbagai keterbatasan karena kurangnya pengalaman dan pengetahuan. Oleh karena itu, kritik dan saran yang membangun sangat diharapkan demi penyempurnaan di masa mendatang. Diharapkan karya ini dapat memberikan manfaat, khususnya dalam bidang *Quality Assurance* dan evaluasi perangkat lunak, serta menjadi amal jariyah bagi semua pihak yang terlibat. Aamiin.

Bandar Lampung, 13 Oktober 2025
Penyusun,

Rizki Pangestu
NPM. 2115061082

DAFTAR ISI

	Halaman
DAFTAR ISI.....	i
DAFTAR TABEL	v
DAFTAR GAMBAR.....	vii
I. PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Tujuan Penelitian	3
1.4 Manfaat Penelitian	3
1.5 Batasan Masalah.....	4
1.6 Sistematika Penulisan	4
II. TINJAUAN PUSTAKA.....	6
2.1 Sistem Informasi	6
2.2 Sistem Informasi Monitoring Pembesaran Kambing (SI MBEK)	6
2.3 Kualitas Sistem Informasi.....	7
2.4 Pengujian Sistem.....	8
2.5 Standar ISO/IEC 25010:2023	10
2.6 Tahapan <i>Software Testing Life Cycle</i> (STLC)	15
2.7 <i>Katalon Studio</i>	18
2.8 <i>Apache JMeter</i>	19
2.9 <i>BlazeMeter</i>	20
2.10 WAPT	20
2.11 <i>GTmetrix</i>	21

2.12 <i>PowerMapper</i>	22
2.13 <i>User Experience Questionnaire (UEQ)</i>	24
2.14 <i>User Acceptance Testing (UAT)</i>	24
2.15 <i>Zed Attack Proxy (ZAP)</i>	25
2.16 <i>SonarQube</i>	26
2.17 <i>Virtual Private Server (VPS)</i>	27
2.18 Penelitian Terkait	28
III. METODOLOGI PENELITIAN	37
3.1 Waktu dan Tempat	37
3.2 Alat dan Bahan Penelitian.....	38
3.2.1 Alat.....	38
3.2.2 Bahan	39
3.3 Tim Penelitian	39
3.4 Tahapan Penelitian.....	41
3.4.1 <i>Requirement Analysis</i>	43
3.4.2 <i>Test Planning</i>	44
3.4.3 <i>Test Case Development</i>	45
3.4.4 <i>Test Environment Setup</i>	46
3.4.5 <i>Test Execution</i>	46
3.4.6 <i>Test Closure</i>	48
IV. HASIL DAN PEMBAHASAN	55
4.1 <i>Requirement Analysis</i>	55
4.1.1 Analisis Kebutuhan Fungsional	55
4.1.2 Analisis Kebutuhan Non-Fungsional	56
4.2 <i>Test Planning</i>	56
4.2.1 Kebutuhan <i>Test Planning</i>	57

4.2.2 <i>Tools yang Digunakan dalam Karakteristik</i>	57
4.3 <i>Test Case Development</i>	58
4.3.1 <i>Test Case Development Karakteristik Functional Suitability</i>	58
4.3.2 <i>Test Case Development Karakteristik Performance Efficiency</i>	90
4.3.3 <i>Test Case Development Karakteristik Compatibility</i>	91
4.3.4 <i>Test Case Development Karakteristik Interaction Capability</i>	92
4.3.5 <i>Test Case Development Karakteristik Reliability</i>	95
4.3.6 <i>Test Case Development Karakteristik Security</i>	96
4.3.7 <i>Test Case Development Karakteristik Maintainability</i>	98
4.4 <i>Environment Setup</i>	99
4.4.1 <i>Environment Setup Karakteristik Functional Suitability</i>	99
4.4.2 <i>Environment Setup Karakteristik Performance Efficiency</i>	103
4.4.3 <i>Environment Setup Karakteristik Compatibility</i>	104
4.4.4 <i>Environment Setup Karakteristik Interaction Capability</i>	105
4.4.5 <i>Environment Setup Karakteristik Reliability</i>	107
4.4.6 <i>Environment Setup Karakteristik Security</i>	113
4.4.7 <i>Environment Setup Karakteristik Maintainability</i>	114
4.5 <i>Test Execution</i>	115
4.5.1 <i>Test Execution Karakteristik Functional Suitability</i>	115
4.5.2 <i>Test Execution Karakteristik Performance Efficiency</i>	125
4.5.3 <i>Test Execution Karakteristik Compatibility</i>	145
4.5.4 <i>Test Execution Karakteristik Interaction Capability</i>	150
4.5.5 <i>Test Execution Karakteristik Reliability</i>	160
4.5.6 <i>Test Execution Karakteristik Security</i>	168
4.5.7 <i>Test Execution Karakteristik Maintainability</i>	182
4.6 <i>Test Cycle Closure</i>	188

4.6.1 <i>Test Cycle Closure</i> Karakteristik <i>Functional Suitability</i>	188
4.6.2 <i>Test Cycle Closure</i> Karakteristik <i>Performance Efficiency</i>	189
4.6.3 <i>Test Cycle Closure</i> Karakteristik <i>Compatibility</i>	191
4.6.4 <i>Test Cycle Closure</i> Karakteristik <i>Interaction Capability</i>	192
4.6.5 <i>Test Cycle Closure</i> Karakteristik <i>Reliability</i>	194
4.6.6 <i>Test Cycle Closure</i> Karakteristik <i>Security</i>	195
4.6.7 <i>Test Cycle Closure</i> Karakteristik <i>Maintainability</i>	196
V. KESIMPULAN DAN SARAN	197
5.1 Kesimpulan	197
5.2 Saran.....	198
DAFTAR PUSTAKA	199

DAFTAR TABEL

	Halaman
Tabel 1. Jenis-Jenis Testing	9
Tabel 2. Karakteristik ISO/IEC 25010:2023 [26]	11
Tabel 3. Status Kerentanan [32]	17
Tabel 4. Penelitian Terkait	31
Tabel 5. Jadwal Pengerjaan Skripsi	37
Tabel 6. Alat Penelitian	38
Tabel 7. Tim Penelitian	40
Tabel 8. Persyaratan Fungsional	43
Tabel 9. Persyaratan Non-Fungsional	44
Tabel 10. Alat Pengujian	45
Tabel 11. Skenario Testing	46
Tabel 12. Status Test Case	47
Tabel 13. Parameter GTmetrix	50
Tabel 14. Aspek UEQ	52
Tabel 15. Kerriteria Penilaian UAT	52
Tabel 16. Analisis Kebutuhan Fungsional	55
Tabel 17. Analisis Kebutuhan Non-Fungsional	56
Tabel 18. Kebutuhan Test Planning	57
Tabel 19. Tools yang Digunakan dalam Karakteristik	58
Tabel 20. Test Case Development Karakteristik Functional Suitability (Pengguna)	59
Tabel 21. Test Case Development Karakteristik Functional Suitability (Super Admin)	74
Tabel 22. Test Case Development Karakteristik Performance Efficiency	90
Tabel 23. Test Case Development Karakteristik Compatibility	91

Tabel 24. Indikator Penilaian UEQ.....	93
Tabel 25. Daftar Pertanyaan UAT	94
Tabel 26. Test Case Development Karakteristik Reliability.....	95
Tabel 27. Test Case Development Karakteristik Security	97
Tabel 28. Test Case Development Karakteristik Maintainability.....	98
Tabel 29. Perangkat Lunak Functional Suitability.....	100
Tabel 30. Lingkungan Sistem Karakteristik Performance Efficiency	104
Tabel 31. Lingkungan Sistem Pengujian Karakteristik Compatibility	105
Tabel 32. Browser yang Digunakan Karakteristik Compatibility.....	105
Tabel 33. Lingkungan Pengujian UEQ	106
Tabel 34. Lingkungan Pengujian UAT	107
Tabel 35. Konfigurasi Uji Karakteristik Security	113
Tabel 36. Hasil Pengujian Karakteristik Functional Suitability Pengguna.....	115
Tabel 37. Hasil Pengujian Karakteristik Functional Suitability Admin	116
Tabel 38. Dokumentasi Kesalahan Sistem.....	117
Tabel 39. Hasil Pengujian Ulang dari Kesalahan Sistem.....	121
Tabel 40. Indikator Penilaian GTmetrix	136
Tabel 41. Hasil Pengujian Performance Efficiency	138
Tabel 42. Hasil Pengujian Compatibility.....	148
Tabel 43. Persona Admin.....	151
Tabel 44. Persona Peternak	151
Tabel 45. Persona Berkecimpung di Dunia IT.....	151
Tabel 46. Persona Pengunjung Umum.....	152
Tabel 47. Data Mentah UEQ.....	152
Tabel 48. Konversi ke Skala UEQ	153
Tabel 49. Rata-Rata UEQ Per Responden	155
Tabel 50. Hasil UAT	156
Tabel 51. Hasil Pengujian Karakteristik Reliability	166
Tabel 52. Hasil Pengujian Karakteristik Security	178
Tabel 53. Hasil Pengujian Karakteristik Maintainability.....	186

DAFTAR GAMBAR

	Halaman
Gambar 1. Jenis-Jenis Testing.....	9
Gambar 2. Tahapan STLC	15
Gambar 3. Tim Penelitian	40
Gambar 4. Tahapan Penelitian STLC	42
Gambar 5. Pengaturan Web Driver Katalon Studio.....	100
Gambar 6. Pembuatan Project di Katalon Studio.....	101
Gambar 7. Penggunaan Spy Web	101
Gambar 8. Penyimpanan Object Repository	102
Gambar 9. Pembuatan Test Case Baru.....	102
Gambar 10. Langkah-Langkah Test Case.....	103
Gambar 11. Eksekusi Test Case.....	103
Gambar 12. Website GTmetrix	104
Gambar 13. UEQ Menggunakan Google Form	106
Gambar 14. UAT Menggunakan Google Form	107
Gambar 15. Filter Pattern pada Blazermeter.....	108
Gambar 16. Start Recording pada Blazermeter.....	108
Gambar 17. Add Listener pada JMeter	109
Gambar 18. Setting Thread Group pada JMeter	110
Gambar 19. Mengisi Jumlah Pengguna pada WAPT	111
Gambar 20. Menentukan Browser pada WAPT	112
Gambar 21. Verify Test pada WAPT.....	112
Gambar 22. Versi ZAP pada Pengujian Security.....	113
Gambar 23. Persiapan Lingkungan Pengujian pada SonarQube Cloud.....	114
Gambar 24. Test Performance Beranda	125
Gambar 25. Detail Halaman Beranda	125

Gambar 26. CrUX Halaman Beranda	126
Gambar 27. Grafik CPU Halaman Beranda.....	126
Gambar 28. Test Performance Produk.....	127
Gambar 29. Detail Halaman Produk.....	127
Gambar 30. CrUX Halaman Produk.....	128
Gambar 31. Grafik CPU Halaman Produk.....	128
Gambar 32. Test Performance Tentang Kami.....	129
Gambar 33. Detail Halaman Tentang Kami.....	129
Gambar 34. CrUX Halaman Tentang Kami.....	130
Gambar 35. Grafik CPU Halaman Tentang Kami	130
Gambar 36. Test Performance Kontak.....	131
Gambar 37. Detail Halaman Kontak.....	131
Gambar 38. CrUX Halaman Kontak.....	131
Gambar 39. Grafik CPU Halaman Kontak	132
Gambar 40. Test Performance Masuk.....	133
Gambar 41. Detail Halaman Masuk.....	133
Gambar 42. CrUX Halaman Masuk.....	133
Gambar 43. Grafik CPU Halaman Masuk	134
Gambar 44. Test Performance Daftar	134
Gambar 45. Detail Halaman Daftar	135
Gambar 46. CrUX Halaman Daftar	135
Gambar 47. Grafik CPU Halaman Daftar	136
Gambar 48. Disk Usage pada VPS	141
Gambar 49. Diakses 500 User dalam Waktu 1 Detik.....	142
Gambar 50. Diakses 500 User dalam 1 Menit	143
Gambar 51. Diakses 500 User dalam Waktu 5 Menit.....	144
Gambar 52. Test Compatibility Halaman Beranda.....	145
Gambar 53. Test Compatibility Halaman Produk.....	146
Gambar 54. Test Compatibility Halaman Tentang Kami	146
Gambar 55. Test Compatibility Halaman Kontak.....	147
Gambar 56. Test Compatibility Halaman Masuk	147
Gambar 57. Test Compatibility Halaman Daftar	148

Gambar 58. Benchmark Hasil UEQ.....	156
Gambar 59. Hasil Pengujian JMeter dan BlazeMeter.....	161
Gambar 60. Pengujian 460 Pengguna	162
Gambar 61. Pengujian 470 Pengguna	163
Gambar 62. Hasil Pengujian WAPT	164
Gambar 63. Simulasi Pengujian Recoverability	165
Gambar 64. Hasil Temuan Pengujian Security (a).....	168
Gambar 65. Hasil Temuan Pengujian Security (b)	169
Gambar 66. Temuan Literal String	182
Gambar 67. Temuan Kompleksitas Kognitif Tinggi	183
Gambar 68. Kurangnya Kurung Kurawal pada Statement Bersarang	184

I. PENDAHULUAN

1.1 Latar Belakang

Dalam era digitalisasi, sistem informasi berperan penting dalam berbagai sektor, termasuk peternakan. Sistem informasi merupakan bagian penting dalam suatu organisasi yang berfungsi untuk mengelola, menyimpan, dan menyajikan data. Dengan sistem ini, pengguna dapat menemukan serta memanfaatkan informasi dari server atau *database* sesuai kebutuhannya [1]. Perkembangan teknologi telah mendorong adopsi sistem informasi guna meningkatkan kelancaran operasional. Kualitas sistem informasi menjadi faktor utama agar dapat berfungsi secara optimal. Pengujian diperlukan untuk memastikan sistem sesuai dengan kebutuhan pengguna, bebas dari kesalahan, serta memiliki performa yang optimal [2]. Tanpa pengujian yang baik, sistem dapat mengalami berbagai permasalahan, seperti *bug*, kompatibilitas terbatas, atau performa yang tidak stabil. Oleh karena itu, pengujian menjadi tahap penting dalam pengembangan perangkat lunak.

Pengujian perangkat lunak umumnya dilakukan dengan teknik seperti *black box testing* dan *white box testing* [3]. Namun, pendekatan ini sering menghadapi tantangan dalam menentukan standar kualitas yang objektif. Tanpa standar yang jelas, sulit memastikan sistem benar-benar memenuhi ekspektasi pengguna dan kebutuhan operasional. Perbedaan persepsi antara pengembang, pemilik sistem dan peternak sering menyebabkan inkonsistensi dalam evaluasi. Dalam bidang peternakan, sistem informasi membantu pengelolaan data terkait hewan ternak, pakan, kesehatan, hingga produktivitas. Beberapa contoh sistem informasi peternakan meliputi:

1. Sistem Informasi Monitoring Perkembangan Sapi Di Lokasi Uji *Performance* (Studi Kasus : Dinas Peternakan Dan Kesehatan Hewan Provinsi Lampung) [4].

2. Pengembangan Sistem Informasi Monitoring Peternakan Ayam berbasis Mobile menggunakan *React Native* dan *Restfull Web Service* (Studi Kasus: Peternakan Alfa Sentosa) [5].

Sistem Informasi Monitoring Pembesaran Kambing (SI MBEK) dikembangkan bersama tim untuk membantu peternak dalam memantau pertumbuhan kambing secara digital, mencatat data pertumbuhan, pola makan, kesehatan, dan produktivitas. Dengan adanya sistem ini, peternak dapat lebih mudah melakukan pemantauan dan analisis untuk mendukung pengambilan keputusan yang lebih baik.

Saat ini, SI MBEK belum memiliki standar kualitas yang jelas karena belum dilakukan pengujian secara menyeluruh. Tanpa pengujian berbasis standar, seperti ISO/IEC 25010:2023, sistem ini berpotensi mengalami kendala seperti kurangnya keandalan dan performa yang tidak optimal. Oleh karena itu, evaluasi kualitas diperlukan untuk memastikan bahwa SI MBEK dapat berfungsi sesuai harapan pengguna serta memenuhi standar perangkat lunak yang diakui secara internasional.

Oleh karena itu, standarisasi seperti ISO/IEC 25010:2023 menjadi penting untuk digunakan dalam evaluasi kualitas perangkat lunak. menyediakan kerangka kerja sistematis dengan tujuh karakteristik utama untuk pengujian yang terstruktur dan dapat diandalkan. Dengan mengadopsinya, evaluasi SI MBEK dapat dilakukan secara objektif, memastikan sistem berfungsi sesuai spesifikasi, optimal dalam performa, kompatibel, serta mudah digunakan dan dipelihara. Standar ini menyediakan kerangka kerja untuk menilai kualitas perangkat lunak berdasarkan berbagai karakteristik seperti, *functional suitability*, *performance efficiency*, *compatibility*, *interaction capability*, *reliability*, *security*, *maintainability*, *flexibility* dan *safety* [6]. Karakteristik ini dipilih karena relevansinya dalam menjamin sistem informasi yang andal, responsif, kompatibel dengan berbagai *browser*, mudah digunakan oleh pengguna, dan mudah dipelihara oleh pengembang.

1.2 Rumusan Masalah

Berdasarkan latar belakang di atas, rumusan masalah dalam penelitian ini adalah:

1. Karakteristik kualitas perangkat lunak apa saja yang menjadi fokus dalam pengujian SI MBEK?
2. Bagaimana penerapan standar ISO/IEC 25010:2023 untuk mengevaluasi kualitas perangkat lunak pada *website* SI MBEK?
3. Bagaimana hasil evaluasi kualitas perangkat lunak SI MBEK berdasarkan standar ISO/IEC 25010:2023?

1.3 Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah sebagai berikut:

1. Mengidentifikasi aspek-aspek kualitas yang perlu diperhatikan dalam pengembangan SI MBEK.
2. Menerapkan standar ISO/IEC 25010:2023 untuk mengevaluasi kualitas perangkat lunak pada SI MBEK.
3. Menyajikan hasil evaluasi kualitas perangkat lunak SI MBEK berdasarkan standar ISO/IEC 25010:2023.

1.4 Manfaat Penelitian

Manfaat yang diharapkan dari penelitian ini adalah:

1. Memberikan kemudahan akses dan penggunaan SI MBEK bagi pengguna melalui berbagai perangkat dan *browser*, serta kemudahan pemeliharaan kode bagi pengembang agar sistem dapat dikelola dan diperbarui dengan baik.
2. Memberikan kemudahan dalam monitoring dan manajemen. Pengguna dapat melihat bagaimana sistem ini membantu pengguna dalam memantau pertumbuhan, kesehatan, dan penjualan kambing dengan baik, sehingga mendukung peningkatan produktivitas.
3. Memberikan jaminan kepada pengguna bahwa sistem SI MBEK telah melalui evaluasi kualitas yang komprehensif berdasarkan standar ISO/IEC 25010:2023, sehingga pengguna dapat merasa yakin terhadap kepuasan dalam penggunaan sistem.

1.5 Batasan Masalah

Dalam penelitian ini, terdapat beberapa batasan yang perlu ditetapkan untuk menjaga fokus penelitian, yaitu:

1. Aspek yang akan dievaluasi mencakup *functional suitability*, *performance efficiency*, *compatibility*, *interaction capability*, *reliability*, *security* dan *maintainability*. pengujian ini tidak mencakup *flexibility* dan *safety*.
2. Penelitian ini tidak mencakup aspek teknis pengembangan perangkat lunak, melainkan lebih pada evaluasi kualitas setelah sistem dikembangkan.

1.6 Sistematika Penulisan

Laporan ini dibagi menjadi beberapa bab untuk memudahkan dalam penguraian, antara lain:

BAB I. PENDAHULUAN

Bab ini memberikan latar belakang mengenai pentingnya evaluasi kualitas perangkat lunak pada SI MBK menggunakan standar ISO/IEC 25010:2023. Rumusan masalah disusun untuk menjawab pertanyaan terkait penerapan standar ini, karakteristik yang diuji, dan hasil evaluasinya. Tujuan penelitian mencakup penerapan standar, identifikasi aspek kualitas, dan penyajian hasil evaluasi. Batasan masalah fokus pada enam karakteristik utama, *functional suitability*, *performance efficiency*, *compatibility*, *interaction capability*, *reliability*, *security* dan *maintainability*. bab ini juga mencakup manfaat penelitian dan sistematika penulisan untuk memberikan gambaran isi setiap bab.

BAB II. TINJAUAN PUSTAKA

Bab ini menguraikan teori dan konsep yang mendukung penelitian. Penjelasan mencakup ISO/IEC 25010:2023 sebagai standar evaluasi kualitas perangkat lunak, termasuk definisi dan detail karakteristik yang diuji. Tahapan pengujian STLC dijelaskan mulai dari tahap *requirement analysis* hingga *test closure*. Selain itu, penelitian sebelumnya yang relevan diulas untuk memberikan konteks dan perbandingan. Kajian ini menjadi landasan teoritis dalam merancang dan melaksanakan penelitian.

BAB III. METODOLOGI PENELITIAN

Bab ini menjelaskan langkah-langkah penelitian yang dilakukan, mulai dari waktu dan tempat penelitian, alat dan bahan yang digunakan, hingga tahapan penelitian. Teknik pengujian melibatkan *Katalon Studio* untuk menguji *functional suitability*, *GTmetrix* dan data di VPS untuk mengukur *performance efficiency*, *PowerMapper* untuk mengukur *compatibility*, *Apache JMeter*, *Blaze Meter*, *WAPT Pro* untuk menguji *reliability*, *User Experience Questionnaire* (UEQ) dan *User Acceptance Testing* (UAT) untuk mengukur *interaction capability*, *Zed Attack Proxy* (ZAP) untuk menguji *security* dan *SonarQube Cloud* untuk menguji *maintainability*. Data dianalisis berdasarkan hasil pengujian dengan membandingkannya terhadap standar ISO/IEC 25010:2023. Penjelasan setiap tahap, dari perencanaan hingga penyusunan laporan, dirinci secara sistematis.

BAB IV. HASIL DAN PEMBAHASAN

Bab ini menyajikan hasil evaluasi kualitas perangkat lunak SI MBEK berdasarkan pengujian lima karakteristik ISO/IEC 25010:2023. Data hasil pengujian disajikan untuk setiap alat yang digunakan, seperti *Katalon Studio*, *GTmetrix*, VPS, *PowerMapper*, *Apache JMeter*, *Blaze Meter*, *WAPT Pro*, *ZAP* dan *SonarQube Cloud*. Analisis membandingkan hasil dengan standar untuk menilai tingkat kualitas perangkat lunak. Pembahasan mendalam dilakukan untuk menginterpretasikan hasil dan mengidentifikasi kekuatan maupun kelemahan sistem.

BAB V. KESIMPULAN DAN SARAN

Bab ini menyimpulkan penelitian berdasarkan jawaban atas rumusan masalah. Kesimpulan mencakup penerapan standar ISO/IEC 25010:2023, hasil pengujian kualitas perangkat lunak, dan implikasi dari hasil evaluasi. Saran diberikan untuk pengembangan lebih lanjut SI MBEK, seperti peningkatan aspek *maintainability* atau pengujian lebih mendalam pada karakteristik lain di luar cakupan penelitian ini.

II. TINJAUAN PUSTAKA

2.1 Sistem Informasi

Sistem informasi merupakan sebuah kumpulan dari beberapa komponen yang mengelola data agar dapat diolah menjadi informasi yang bermakna serta membantu mencapai tujuan organisasi [7]. Fungsi utama dalam sebuah sistem informasi adalah mengumpulkan, memproses, menyimpan, menganalisis, dan mendistribusikan informasi untuk tujuan tertentu. Seiring dengan pesatnya perkembangan teknologi, sistem informasi kini telah diterapkan di berbagai bidang, seperti pendidikan, pemerintahan, kesehatan, dan bisnis. Peningkatan pemanfaatan sistem informasi ini didorong oleh kebutuhan manusia akan teknologi yang dapat mendukung aktivitas mereka agar lebih efisien dan efektif [8].

2.2 Sistem Informasi Monitoring Pembesaran Kambing (SI MBEK)

Monitoring atau pemantauan bertujuan memastikan tercapainya target organisasi dan manajemen. Ini mencakup penilaian kesesuaian kegiatan dengan rencana, identifikasi masalah, evaluasi pola kerja, serta analisis hubungan antara kegiatan dan tujuan. Monitoring juga mendeteksi permasalahan sejak dini untuk tindakan korektif yang cepat [9]. Dalam sistem informasi peternakan seperti SI MBEK, monitoring mencakup pengelolaan data pertumbuhan, kesehatan, dan produktivitas ternak, membantu peternak dalam pengambilan keputusan berbasis data.

Kambing sebagai salah satu ternak ruminansia kecil, berperan penting dalam memenuhi kebutuhan daging dan susu masyarakat. Kambing memiliki karakteristik unik terbentuk melalui adaptasi selama beberapa generasi, menjadikannya sesuai dengan kondisi lingkungan setempat [10]. Pembesaran kambing bertujuan mendorong pertumbuhan optimal untuk meningkatkan manfaat ekonomi bagi peternak [11]. Proses ini melibatkan perhatian terhadap beberapa faktor kunci,

seperti pemberian pakan berkualitas, manajemen kesehatan, dan pemantauan lingkungan.

Dalam informasi berbasis web telah menjadi solusi untuk membantu peternak mencatat, memantau, dan menganalisis data pembesaran kambing secara lebih sistematis. Sistem Informasi Monitoring Pembesaran Kambing (SI MBEK) adalah sistem yang dirancang khusus untuk tujuan tersebut. SI MBEK menyediakan fitur monitoring pertumbuhan, kesehatan, serta penjualan kambing secara *online*, yang semuanya dapat diakses dan dikelola dengan mudah oleh peternak dan pelanggan. Sistem ini dirancang untuk mempermudah proses pencatatan data seperti berat badan, jenis pakan yang diberikan, dan jadwal pemberian vaksin, sehingga pengambilan keputusan dapat dilakukan dengan lebih cepat, akurat, dan berbasis data. Dengan adanya SI MBEK, para peternak dan pelanggan dapat dengan mudah mengakses informasi terkait kambing secara daring, sehingga meningkatkan kepercayaan pelanggan serta efektivitas operasional peternakan.

Dalam penerapannya, sistem informasi seperti SI MBEK harus melalui proses pengujian yang menyeluruh guna memastikan bahwa sistem berfungsi sesuai dengan kebutuhan pengguna dan mampu memberikan informasi yang akurat serta andal. Pengujian menjadi tahap krusial dalam siklus pengembangan perangkat lunak karena dapat mengidentifikasi kekurangan atau kesalahan sistem. Dengan menerapkan standar ISO/IEC 25010:2023, evaluasi terhadap aspek-aspek kualitas seperti *functional suitability*, *performance efficiency*, *compatibility*, *interaction capability*, *reliability*, *security* dan *maintainability* dapat dilakukan secara sistematis. Hasil pengujian ini akan memastikan bahwa SI MBEK tidak hanya mampu mendukung kegiatan monitoring dan pengelolaan ternak, tetapi juga memiliki keandalan yang tinggi dalam membantu pengambilan keputusan berbasis data bagi peternak.

2.3 Kualitas Sistem Informasi

Kualitas sistem informasi merujuk pada ukuran efektivitas suatu sistem dalam mendukung interaksi antara pengguna dan sistem itu sendiri. Konsep ini mencakup

berbagai aspek yang menentukan sejauh mana sistem dapat memenuhi harapan, kebutuhan, serta kepuasan pengguna. Kualitas tersebut dapat diidentifikasi melalui karakteristik dan fitur sistem yang berkontribusi terhadap pencapaian manfaat yang diinginkan, seperti kemudahan akses, keandalan, kinerja optimal, serta kemampuan sistem dalam menyesuaikan diri dengan kebutuhan pengguna yang terus berkembang [12].

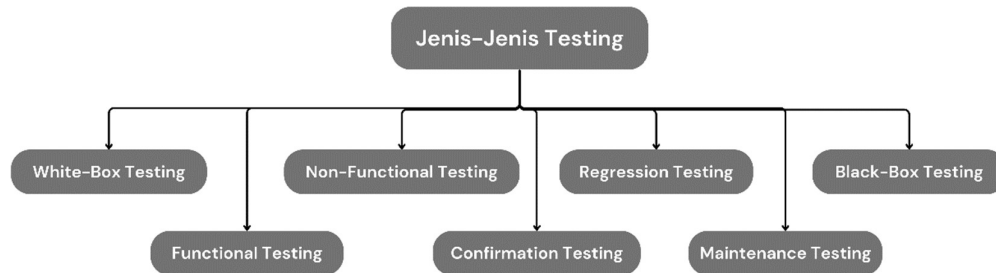
Kualitas suatu sistem sangat penting untuk memastikan bahwa seluruh kebutuhan dan spesifikasi pengguna dapat terpenuhi dengan baik. Saat ini, terdapat berbagai standar pengujian sistem yang diakui secara internasional, seperti Boehm, McCall, FURPS, Dromey, ISO 9126, dan ISO 25010. Di antara berbagai standar tersebut, ISO 25010 menjadi salah satu acuan utama dalam mengukur kualitas sistem informasi secara komprehensif, karena mencakup berbagai aspek penting dalam evaluasi perangkat lunak [15].

Kualitas sistem informasi harus dikembangkan sesuai dengan kebutuhan dan spesifikasi pengguna agar dapat berfungsi secara optimal. Sistem informasi yang berkualitas ditandai oleh tiga aspek utama, yaitu efektivitas dalam menjalankan tugasnya, kemudahan penggunaan, serta kemampuannya dalam memberikan nilai tambah bagi produsen dan pengguna. Untuk memastikan sistem memenuhi standar yang diharapkan, evaluasi menjadi langkah penting dalam menilai kepuasan pengguna terhadap kualitas sistem informasi. Evaluasi ini bertujuan untuk mengukur tingkat kegunaan dan fungsionalitas sistem serta mengidentifikasi berbagai masalah yang mungkin muncul, sehingga perbaikan dapat dilakukan secara tepat guna meningkatkan kualitas dan kinerja sistem secara keseluruhan [13].

2.4 Pengujian Sistem

Pengujian sistem merupakan tahap penting dalam siklus pengembangan perangkat lunak untuk memastikan bahwa sistem yang dikembangkan berfungsi sesuai dengan kebutuhan pengguna dan spesifikasi yang telah ditetapkan. Pengujian ini bertujuan untuk menemukan kesalahan dalam sistem sebelum diterapkan secara luas [14].

Berdasarkan standar ISTQB (*International Software Testing Qualifications Board*), jenis-jenis pengujian perangkat lunak dapat dikategorikan menjadi *white-box testing*, *functional testing*, *non-functional testing*, *confirmation testing*, *regression testing*, *maintenance testing* dan *Black-box Testing* [16]. Dapat dilihat dalam gambar diagram berikut.



Gambar 1. Jenis-Jenis *Testing*

Tabel 1. Jenis-Jenis *Testing*

No	Jenis <i>Testing</i>	Deskripsi
1.	<i>White-Box Testing</i>	Pengujian yang melakukan analisis mendalam terhadap logika dan struktur internal dari kode program. Dalam teknik ini, penguji harus memiliki pemahaman penuh terhadap kode sumber untuk dapat mengidentifikasi dan menguji setiap bagian dari sistem secara menyeluruh [17].
2.	<i>Functional Testing</i>	Pengujian perangkat lunak yang berfokus pada validasi layanan yang disediakan oleh sistem. Pengujian ini mengevaluasi bagaimana sistem memproses masukan data serta memastikan bahwa perilakunya sesuai dengan yang diharapkan dalam berbagai kondisi atau situasi tertentu [18].
3.	<i>Non-Functional Testing</i>	Proses pengujian perangkat lunak yang berfokus pada validasi aspek non-fungsional dari suatu aplikasi. Banyak jenis pengujian non-fungsional memiliki istilah yang sering digunakan secara bergantian karena cakupannya yang saling tumpang tindih. Misalnya, kinerja perangkat lunak merupakan konsep luas yang mencakup berbagai aspek

No	Jenis Testing	Deskripsi
		spesifik seperti keandalan, skalabilitas dan sebagainya [19].
4.	<i>Confirmation Testing</i>	Pengujian ulang yang dilakukan untuk memastikan bahwa <i>bug</i> yang sebelumnya dilaporkan telah diperbaiki dengan benar. Setelah tim pengembang merilis versi terbaru perangkat lunak dengan perbaikan, tim penguji melakukan <i>re-Testing</i> pada kasus uji yang sebelumnya gagal untuk memverifikasi bahwa <i>bug</i> tidak lagi muncul dan sistem berfungsi dengan baik [20].
5.	<i>Regression Testing</i>	Proses pengujian ulang untuk mendeteksi cacat baru atau <i>bug</i> yang muncul kembali setelah perubahan kode dilakukan. Pengujian ini melibatkan pengulangan <i>test case</i> sebelumnya guna memastikan bahwa modifikasi pada kode tidak mempengaruhi kualitas dan fungsionalitas perangkat lunak secara keseluruhan [21].
6.	<i>Maintenance Testing</i>	Pengujian ini dilakukan setelah perangkat lunak dirilis untuk memastikan bahwa sistem tetap berfungsi dengan baik dan tidak mengalami masalah yang mungkin muncul di kemudian hari [22].
7.	<i>Black-box Testing</i>	Pengujian yang dilakukan tanpa mengetahui cara kerja internal dari aplikasi. Teknik ini hanya berfokus pada fungsi utama system. Pengujian ini bertujuan untuk mengidentifikasi berbagai jenis kesalahan, seperti fungsi yang salah atau tidak lengkap, kesalahan antarmuka, kesalahan dalam struktur data atau akses ke database eksternal (jika ada), kesalahan kinerja, serta kesalahan dalam proses inisialisasi dan terminasi system [23].

2.5 Standar ISO/IEC 25010:2023

ISO/IEC 25010:2023 adalah bagian dari *Systems and Software Engineering – Systems and Software Quality Requirements and Evaluation (SQuaRE)* yang berfungsi sebagai model kualitas sistem dan perangkat lunak. Model ini dapat digunakan untuk spesifikasi kebutuhan dan evaluasi kualitas produk target

sepanjang siklus hidupnya oleh berbagai pemangku kepentingan, termasuk pengembang, pihak pengakuisisi, staf jaminan dan pengendalian kualitas [24].

ISO/IEC 25010:2023 adalah model terbaru dari seri ISO/IEC 250n yang dikenal sebagai standar untuk persyaratan dan evaluasi kualitas perangkat lunak dalam kerangka kerja SQuaRE. Model ini digunakan untuk menilai kualitas perangkat lunak berdasarkan karakteristik yang telah ditetapkan dalam masing-masing model, sehingga memungkinkan evaluasi yang lebih komprehensif terhadap aspek kegunaan dan kualitas produk perangkat lunak [24].

ISO/IEC 25010:2023 digunakan sebagai tolak ukur analisis kualitas perangkat lunak, baik oleh perusahaan, instansi, maupun organisasi. Melalui standar ini, evaluasi kualitas sistem perangkat lunak dapat dilakukan secara spesifik yang terdiri atas sembilan karakteristik [6]. Berikut adalah tabel dari sembilan karakteristik ISO/IEC 25010:2023.

Tabel 2. Karakteristik ISO/IEC 25010:2023 [26]

No	Karakteristik	Deskripsi
1.	<i>Functional Suitability</i>	Kemampuan dari sistem untuk menyediakan fungsi-fungsi yang memenuhi kebutuhan dan harapan pengguna yang telah ditetapkan. Produk harus mampu melakukan fungsi yang dibutuhkan sesuai dengan spesifikasi dan kebutuhan pengguna.
	<i>a. Functional Completeness</i>	Kemampuan sebuah produk untuk menyediakan fungsi-fungsi yang memenuhi kebutuhan dan tujuan pengguna yang telah ditentukan dan diharapkan.
	<i>b. Functional Correctness</i>	Kemampuan produk untuk menyediakan seluruh fungsi yang diperlukan untuk menyelesaikan semua tugas dan memenuhi tujuan pengguna yang telah ditetapkan.
	<i>c. Functional Appropriateness</i>	Kemampuan produk untuk memberikan hasil yang akurat dan benar saat digunakan.
2.	<i>Performance Efficiency</i>	Kemampuan dari sebuah produk untuk menjalankan fungsi-fungsinya dalam waktu yang telah ditentukan, serta dapat menggunakan sumber

No	Karakteristik	Deskripsi
		daya secara efisien dalam kondisi yang telah ditetapkan.
	a. <i>Time Behaviour</i>	Waktu <i>respons</i> sistem dalam menyelesaikan permintaan pengguna.
	b. <i>Resource Utilization</i>	Kemampuan sistem untuk menggunakan sumber daya yang tidak melebihi jumlah yang ditetapkan untuk menjalankan fungsi tertentu dalam kondisi tertentu.
	c. <i>Capacity</i>	Kemampuan suatu produk untuk memenuhi persyaratan mengenai batas maksimum suatu parameter produk.
3.	<i>Compatibility</i>	Kemampuan sebuah produk untuk berbagi informasi dengan produk lain, dan melaksanakan fungsi yang dibutuhkan sambil berbagi lingkungan dan sumber daya yang sama.
	a. <i>Co-existence</i>	Sistem dapat berjalan berdampingan dengan aplikasi lain tanpa gangguan.
	b. <i>InterOperability</i>	Sistem dapat bekerja dengan berbagai <i>platform</i> , <i>browser</i> , dan perangkat yang berbeda tanpa mengalami <i>error</i> .
4.	<i>Interaction Capability</i>	Kemampuan suatu produk untuk berinteraksi dengan pengguna tertentu untuk bertukar informasi antar pengguna dan sistem melalui antarmuka pengguna untuk menyelesaikan tugas yang dimaksudkan.
	a. <i>Appropriateness recognizability</i>	Kemampuan pengguna untuk mengenali apakah suatu fitur relevan dan sesuai untuk mencapai tujuannya.
	b. <i>Learnability</i>	Kemudahan pengguna dalam mempelajari cara menggunakan sistem.
	c. <i>Operability</i>	Kemampuan sistem untuk memudahkan pengguna dalam menjalankan dan mengontrol interaksi.
	d. <i>User error protection</i>	Kemampuan sistem dalam mencegah atau mengurangi kemungkinan terjadinya kesalahan pengguna.
	e. <i>User engagement</i>	Kemampuan sistem untuk menarik perhatian pengguna dan menciptakan pengalaman interaktif yang menyenangkan.

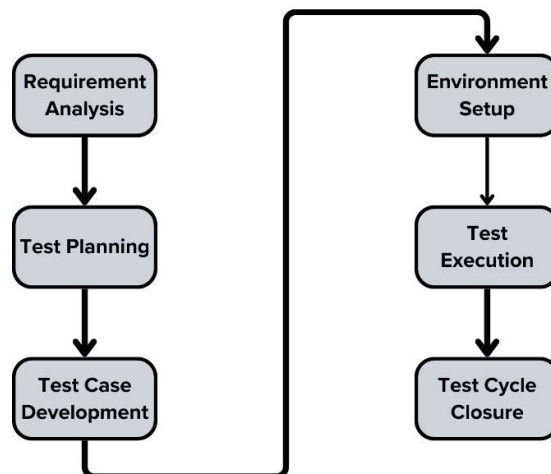
No	Karakteristik	Deskripsi
	<i>f. Inclusivity</i>	Kemampuan dari sebuah produk untuk dapat digunakan oleh orang-orang dari berbagai latar belakang.
	<i>g. User assistance</i>	Ketersediaan bantuan dan panduan yang dapat membantu pengguna memahami atau menyelesaikan tugas.
	<i>h. Self-descriptiveness</i>	Seberapa jelas dan intuitif tampilan antarmuka sehingga pengguna memahami fungsinya tanpa penjelasan tambahan.
5.	<i>Reliability</i>	Kemampuan suatu produk untuk melakukan fungsi tertentu dalam kondisi tertentu selama periode waktu tertentu tanpa gangguan dan kegagalan.
	<i>a. Faultlessness</i>	Kemampuan sistem untuk berfungsi tanpa kesalahan selama pengoperasian normal.
	<i>b. Fault tolerance</i>	Kemampuan sistem untuk terus beroperasi meskipun terjadi gangguan.
	<i>c. Availability</i>	Sejauh mana sistem tersedia dan dapat digunakan saat dibutuhkan.
	<i>d. Recoverability</i>	Kemampuan sistem untuk dipulihkan setelah mengalami kegagalan
6.	<i>Security</i>	Kemampuan suatu produk untuk melindungi informasi dan data sehingga orang atau produk lain memilikinya tingkat akses data yang sesuai dengan jenis dan tingkat otorisasinya, dan untuk mempertahankan diri dari serangan pola oleh aktor jahat
	<i>a. Confidentiality</i>	Memastikan data hanya dapat diakses oleh pihak yang berwenang.
	<i>b. Integrity</i>	Menjaga agar data tidak dimodifikasi secara tidak sah.
	<i>c. Non-Repudiation</i>	Menjamin bahwa tindakan pengguna dapat dilacak.
	<i>d. Accountability</i>	Setiap aksi yang dilakukan dapat ditelusuri ke pengguna tertentu.
	<i>e. Authenticity</i>	Sistem dapat memverifikasi identitas pengguna dengan benar.
	<i>f. Resistance</i>	Kemampuan sistem menahan berbagai bentuk serangan umum.

No	Karakteristik	Deskripsi
7.	<i>Maintainability</i>	Mengacu pada sejauh mana suatu produk atau sistem dapat dimodifikasi dengan efektif dan efisien untuk perbaikan, peningkatan, atau penyesuaian terhadap perubahan lingkungan.
	<i>a. Reusability</i>	Kemampuan komponen kode untuk digunakan kembali pada konteks yang berbeda.
	<i>b. Modifiability</i>	Kemudahan dalam memodifikasi sistem tanpa menyebabkan kesalahan tambahan.
	<i>c. Testability</i>	Kemampuan sistem untuk diuji ulang setelah dimodifikasi.
	<i>d. Modularity</i>	Sejauh mana sistem dibagi menjadi komponen-komponen terpisah yang dapat dimodifikasi.
	<i>e. Analysability</i>	Sejauh mana sistem dapat dipahami dan dianalisis untuk diagnosis kesalahan.
8.	<i>Flexibility</i>	Kemampuan produk untuk diadaptasi terhadap berbagai perubahan pada kebutuhan, konteks penggunaan, atau lingkungan sistem. Seperti kemudahan instalasi, mengukur tingkat kesulitan dalam memasang perangkat lunak pada lingkungan target.
	<i>a. Scalability</i>	Kemampuan produk untuk menangani beban kerja yang tumbuh atau menyusut, serta mengubah kapasitasnya sesuai kebutuhan.
	<i>b. Installability</i>	Kemampuan produk untuk dipasang dan/atau dicopot secara efektif dan efisien di lingkungan tertentu.
	<i>c. Replaceability</i>	Kemampuan produk untuk mengganti produk lain dengan tujuan yang sama dalam lingkungan yang sama, termasuk dalam konteks pembaruan versi untuk mengurangi risiko ketergantungan dan memfasilitasi penggunaan file standar.
9.	<i>Safety</i>	Kemampuan dari suatu produk, dalam kondisi tertentu yang telah didefinisikan, untuk menghindari kondisi yang dapat menyebabkan bahaya terhadap kehidupan manusia, kesehatan, properti, atau lingkungan.
	<i>a. Operational Constrain</i>	Kemampuan produk untuk mengendalikan operasi agar tetap berada dalam parameter atau kondisi aman saat menghadapi bahaya operasional.
	<i>b. Risk Identification</i>	Kemampuan produk untuk mengidentifikasi jalur kejadian atau operasi yang berpotensi

No	Karakteristik	Deskripsi
		menyebabkan risiko yang tidak dapat diterima terhadap kehidupan, properti, atau lingkungan.
	<i>c. Fail Safe</i>	Kemampuan produk untuk otomatis pindah ke mode operasi aman atau kembali ke kondisi aman saat terjadi kegagalan.
	<i>d. Hazard Warning</i>	Kemampuan produk untuk memberikan peringatan terhadap risiko tidak dapat diterima agar pengguna atau sistem dapat bereaksi cukup cepat dalam menjaga keselamatan.
	<i>e. Safe Integration</i>	Kemampuan produk untuk mempertahankan keselamatan selama dan setelah integrasi dengan satu atau lebih komponen lain. Ini menekankan bahwa keselamatan harus tetap terjaga saat produk digabungkan ke dalam sistem yang lebih besar atau bersama komponen lain.

2.6 Tahapan *Software Testing Life Cycle* (STLC)

Software Testing Life Cycle (STLC) adalah proses yang terstruktur dan sistematis dalam pengujian perangkat lunak. STLC sering dianggap sebagai bagian integral dari *Software Development Life Cycle* (SDLC). Proses dalam STLC terdiri dari serangkaian tahapan, di mana setiap tahapan memiliki tujuan. Pendekatan ini memastikan pengujian dilakukan secara menyeluruh dan terorganisir, sehingga kualitas perangkat lunak dapat terjamin [27].



Gambar 2. Tahapan STLC

1. *Requirement Analysis*

Tahap pertama dalam *Software Testing Life Cycle* (STLC) adalah analisis kebutuhan, di mana *tester* mengkaji dan mengidentifikasi persyaratan yang dapat diuji berdasarkan perspektif pengujian. Proses ini melibatkan interaksi dengan *stakeholder*, untuk memastikan bahwa setiap kebutuhan dipahami dengan jelas sebelum pengujian dilakukan. Persyaratan tersebut terbagi menjadi dua kategori utama, yaitu persyaratan fungsional yang mencakup fitur dan fungsi utama perangkat lunak, serta persyaratan non-fungsional yang berfokus pada aspek kinerja, kemampuan sistem, dan keandalan sistem [28].

2. *Tahap Planning*

Test planning adalah tahap perencanaan strategi pengujian yang mencakup pemilihan jenis atau teknik pengujian, serta sumber daya yang dibutuhkan. Pada tahap ini, *tester* menentukan pendekatan yang paling efektif untuk menguji perangkat lunak, baik dengan menggunakan *automation testing tools* maupun pengujian manual, berdasarkan kompleksitas sistem dan kebutuhan proyek. [29].

3. *Test Case Development*

Test case development merupakan tahap pengembangan dalam proses pengujian perangkat lunak yang berfokus pada penyusunan skenario pengujian sebagai acuan utama dalam pengujian. Pada tahap ini, *tester* bertanggung jawab untuk membuat *test case*, menyusun *test data*, serta mengembangkan *automation test script* berdasarkan *test case* yang telah dirancang. *Test case* sendiri merupakan serangkaian skenario yang dirancang secara sistematis untuk memastikan bahwa sistem yang diuji dapat berfungsi sesuai dengan persyaratan yang ditetapkan serta memenuhi standar kualitas yang diharapkan [30].

4. *Environment Setup*

Environment setup merupakan tahap penting dalam proses pengujian perangkat lunak yang bertujuan untuk memastikan bahwa lingkungan pengujian, baik

dari segi perangkat keras (*hardware*) maupun perangkat lunak (*software*), telah dikonfigurasi dan berjalan sesuai dengan kebutuhan pengujian. Pada tahap ini, dilakukan pengaturan sistem, instalasi perangkat lunak pendukung, serta konfigurasi alat uji untuk memastikan kesesuaian dengan spesifikasi yang telah direncanakan. Proses ini dapat dilakukan secara paralel dengan tahap *test case development* guna menghemat waktu dan memastikan kesiapan pengujian. Selain itu, *environment setup* bersifat independen, artinya tidak terikat secara langsung dengan tahap lainnya, namun tetap memiliki peran krusial dalam menjamin kelancaran dan akurasi proses pengujian [30].

5. *Test Execution*

Tahap *test execution* adalah proses pelaksanaan pengujian berdasarkan skenario dan jadwal yang telah disusun sebelumnya. Pada tahap ini, pengujian dilakukan dengan menjalankan perangkat lunak sesuai dengan kasus pengujian (*test case*), sambil mencatat hasil aktual yang diperoleh serta membandingkannya dengan hasil yang diharapkan. Jika ditemukan ketidaksesuaian antara hasil aktual dan hasil yang diharapkan, penguji akan mencatat temuan tersebut sebagai *bug* dan melaporkannya kepada tim pengembang untuk ditangani [31]. Selain itu, selama tahap pelaksanaan pengujian juga dapat ditemukan berbagai kerentanan sistem yang memengaruhi kualitas perangkat lunak. Temuan-temuan tersebut kemudian dikelompokkan dan dianalisis lebih lanjut untuk menilai tingkat keparahan serta dampaknya terhadap sistem. Rincian jenis kerentanan yang ditemukan selama pengujian dapat dilihat pada Tabel 3 berikut.

Tabel 3. Status Kerentanan [32]

Status	Keterangan
<i>High</i>	Masalah kritis yang dapat dieksploitasi dengan mudah dan berpotensi menyebabkan dampak serius.
<i>Medium</i>	Masalah tingkat sedang, cukup signifikan tetapi memerlukan kondisi tertentu atau upaya tambahan untuk dieksploitasi.
<i>Low</i>	Masalah minor yang berdampak kecil, sulit dieksploitasi, atau membutuhkan kondisi khusus.

Status	Keterangan
<i>Informational</i>	Bukan masalah langsung, tetapi informasi yang mungkin berguna bagi penyerang untuk melakukan eksploitasi lebih lanjut.

6. *Test Cycle Closure*

Test cycle closure merupakan tahap akhir dalam *Software Testing Life Cycle* (STLC) yang mencakup pelaporan hasil pengujian, penyelesaian tes, serta analisis efektivitas pengujian. Pada tahap ini, *tester* bersama tim melakukan diskusi dan evaluasi untuk meninjau seluruh proses pengujian, mengidentifikasi kendala yang dihadapi, serta merumuskan perbaikan untuk pengujian di masa mendatang. Tahap ini bertujuan untuk memastikan bahwa seluruh aktivitas pengujian telah diselesaikan dengan baik dan memberikan wawasan berharga bagi pengembangan perangkat lunak selanjutnya [30].

2.7 *Katalon Studio*

Katalon Studio adalah aplikasi open source yang dikembangkan oleh *Katalon LLC*, yang memungkinkan pelaksanaan pengujian otomatis pada berbagai sistem Operasi seperti *Windows*, *macOS*, dan *Linux*. Alat ini menawarkan berbagai fitur lengkap dengan antarmuka pengguna yang intuitif, sehingga cocok digunakan untuk pengujian aplikasi *mobile*, *web browser*, dan *API*. Pengguna tidak memerlukan kemampuan pemrograman khusus, sehingga menjadikannya sesuai untuk pengguna dari berbagai tingkat keahlian, baik pemula maupun yang sudah berpengalaman [33].

Katalon Studio merupakan alat pengujian otomatis yang mendukung proses pengujian secara otomatis pada tiga *platform* utama, yaitu *Webite Testing*, *API Testing*, dan *Mobile Testing*. Selain itu, *Katalon* telah terintegrasi dengan berbagai teknologi eksternal seperti *qTest*, *JIRA*, *Kobiton*, *GitHub*, dan lainnya, sehingga mempermudah kolaborasi serta pengelolaan pengujian perangkat lunak [34].

Katalon Studio memiliki versi gratis dan versi berbayar, versi gratis *Katalon Studio* sudah menyediakan fitur inti seperti perekaman otomatis (*record & playback*),

scripting dengan bahasa *Groovy/Java*, manajemen *test case*, laporan eksekusi dasar, serta integrasi dengan *Git* dan *Jenkins*. Namun, versi gratis memiliki keterbatasan dalam hal fitur *enterprise* seperti *advanced analytics*, *test reporting* yang lebih komprehensif, integrasi penuh dengan *Katalon TestOps*, serta dukungan teknis resmi. Untuk kebutuhan profesional, *Katalon* menawarkan versi berbayar (*Katalon Studio Enterprise*) dengan biaya mulai dari sekitar US\$ 759 per user per tahun, sedangkan paket lain seperti *Katalon Runtime Engine* dikenakan biaya tambahan sekitar US\$ 599 per mesin per tahun. Dengan adanya pilihan versi gratis dan berbayar, *Katalon Studio* dapat digunakan baik oleh individu maupun organisasi, menyesuaikan dengan skala proyek dan tingkat kebutuhan pengujian perangkat lunak yang dijalankan [35].

2.8 Apache JMeter

Apache JMeter adalah sebuah alat pengujian perangkat lunak gratis berbasis *Java* yang dikembangkan oleh *Apache Software Foundation*. *Apache JMeter* adalah alat *open-source* yang digunakan untuk menguji performa dan beban (*Load testing*) pada aplikasi web, API, dan layanan lainnya. *JMeter* dirancang untuk mensimulasikan beban pengguna yang tinggi pada sistem, sehingga memungkinkan pengembang dan penguji untuk mengevaluasi kinerja dan keandalan aplikasi. [36].

JMeter menyediakan berbagai fitur yang mendukung pengujian performa dan keandalan. Beberapa fitur utamanya meliputi *load testing* untuk mensimulasikan beban pengguna yang tinggi, *stress testing* untuk menguji batas maksimal kemampuan sistem, dan *performance testing* untuk mengukur waktu *respons*, *throughput*, serta utilisasi sumber daya sistem. Selain itu, *JMeter* juga mendukung *distributed testing*, yang memungkinkan pengujian dijalankan dari beberapa mesin secara bersamaan untuk mensimulasikan beban yang lebih besar [37].

Dalam konteks *performance efficiency*, *JMeter* dapat digunakan untuk mengukur aspek-aspek seperti waktu respon (*response time*), *throughput*, dan utilisasi sumber daya. Waktu *respons* mengukur seberapa cepat sistem merespon permintaan pengguna, sementara *throughput* mengukur jumlah permintaan yang dapat diproses

oleh sistem dalam satuan waktu tertentu. Utilisasi sumber daya memantau penggunaan CPU, memori, dan jaringan selama pengujian [37].

2.9 BlazeMeter

BlazeMeter merupakan sebuah *platform performance testing* yang digunakan untuk mengukur, menganalisis, dan mengoptimalkan kinerja aplikasi web, situs web, layanan web, serta infrastruktur terkait. Alat ini memungkinkan pengguna melakukan *load testing* dan *stress testing* secara realistis terhadap aplikasi web yang diuji. *BlazeMeter* memiliki fitur unggulan seperti kemampuan untuk merekam dan memutar ulang skenario pengujian (*record and replay*), melakukan scaling secara *fleksibel* untuk mensimulasikan berbagai beban lalu lintas (*traffic load*), serta menyediakan pemantauan kinerja secara real-time [38].

Kemampuannya dalam melakukan simulasi lalu lintas pengguna yang besar dan beragam, *BlazeMeter* sering digunakan untuk mengevaluasi *performance efficiency* sebuah sistem. Hasil pengujian dari *BlazeMeter* dapat membantu pengembang memprediksi performa sistem pada kondisi beban tertentu, serta melakukan optimalisasi sebelum sistem digunakan secara penuh di lingkungan produksi.

BlazeMeter juga menyediakan *extension* untuk *browser* (misalnya *Chrome Extension*) yang berfungsi sebagai perekam (*recorder*) aktivitas pengguna. *Extension* ini merupakan alat gratis yang memungkinkan perekaman interaksi pengguna di *website* secara langsung, kemudian hasilnya dapat diekspor dalam format JMX untuk dijalankan di *JMeter* [39].

2.10 WAPT

WAPT (*Web Application Performance Testing*) merupakan alat pengujian yang digunakan untuk melakukan *load testing*, *stress testing*, dan *performance testing* terhadap situs web maupun aplikasi web yang memiliki antarmuka berbasis web. WAPT terdiri dari komponen *workplace* dan beberapa *load agents* yang dapat diinstal di berbagai lokasi serta dikelola secara jarak jauh.

Alat ini menyediakan informasi detail mengenai aktivitas *virtual users* selama pengujian berlangsung, termasuk hasil keluaran (output) secara langsung kepada pengguna selama proses *load testing*. WAPT dikenal sebagai salah satu alat pengujian yang efektif dari segi biaya (*cost-effective*) untuk menganalisis performa layanan web (*web services*) [40]. WAPT sangat berguna untuk mengevaluasi ketahanan dan kinerja sistem pada kondisi beban tinggi sebelum sistem digunakan di lingkungan produksi.

WAPT merupakan perangkat lunak berbayar, meskipun tersedia dalam versi trial dengan waktu penggunaan terbatas yaitu 30 hari. Versi berbayar WAPT menawarkan beberapa edisi, antara lain WAPT Pro yang mendukung pengujian skala besar dengan distribusi beban ke beberapa mesin, *scripting* kustom, serta integrasi dengan CI/CD. Dari sisi biaya, lisensi WAPT yaitu US\$ 500 untuk versi standar, sedangkan WAPT Pro dapat mencapai US\$ 1000, terdapat biaya tambahan pemeliharaan sebesar US\$ 500. Dengan dukungan teknis resmi dan kemudahan penggunaan, WAPT banyak dimanfaatkan oleh perusahaan atau tim QA yang membutuhkan alat uji performa profesional dengan tingkat akurasi dan fleksibilitas tinggi [41].

2.11 GTmetrix

GTmetrix adalah alat pengujian yang digunakan untuk menguji dan memantau kinerja suatu halaman web. Alat ini dirancang untuk memastikan pengalaman pengguna yang cepat dan konsisten baik pada perangkat desktop maupun mobile, serta pada berbagai jenis perangkat keras, dari yang rendah hingga tinggi spesifikasinya, dan dari berbagai lokasi geografis. Sejak didirikan pada tahun 2009, *GTmetrix* bertujuan membantu pengguna dengan berbagai tingkat keahlian dalam menghadirkan pengalaman web yang cepat dan lancar. Awalnya, alat ini dikembangkan untuk membantu pelanggan membedakan apakah masalah kinerja *website* berasal dari sisi server atau sisi *front-end*. Respon positif yang besar dari pengguna mendorong pengembangan alat ini agar dapat diakses oleh publik umum dan menjadikannya sebagai alat utama untuk pengujian performa web [42].

GTmetrix digunakan untuk analisis kinerja web yang khusus digunakan untuk mengevaluasi dan memberikan wawasan terkait kecepatan serta optimasi sebuah situs web. Alat ini mengukur berbagai aspek, termasuk waktu muat halaman, dan menyajikan rekomendasi untuk peningkatan performa web. Dalam penelitian ini, fokus kami bukanlah untuk menguji performa keseluruhan dari situs web utama, melainkan untuk mengevaluasi sistem yang digunakan dalam pelaksanaan survei spasial [43].

GTmetrix merupakan layanan analisis performa *website* yang tersedia dalam dua versi, yaitu gratis (*basic*) dan berbayar (*pro*). Versi gratis menyediakan fitur dasar seperti pengujian kecepatan dengan tujuh lokasi global, grafik tren performa, notifikasi penurunan kinerja, serta penyimpanan laporan terbatas hanya selama satu minggu dengan kuota sekitar lima tes per bulan. Sementara itu, versi berbayar atau *GTmetrix Pro* ditujukan bagi pengguna profesional dengan kebutuhan analisis lebih mendalam, meliputi akses ke lokasi pengujian premium tambahan, perangkat mobile nyata, histori laporan lebih panjang, kapasitas tes *on-demand* dan API credit lebih besar, hingga prioritas antrean pengujian serta pelaporan lebih lengkap seperti PDF dan video *capture*. Dari sisi biaya, *GTmetrix Pro* menawarkan beberapa paket tahunan, antara lain *Micro* sekitar US\$ 4,25 per bulan, *Solo* US\$ 12,33 per bulan, *Starter* US\$ 23,75 per bulan, dan *Growth* US\$ 47,33 per bulan, di mana setiap tingkatan memberikan kapasitas monitoring serta fitur yang semakin lengkap sesuai kebutuhan pengguna [42].

2.12 PowerMapper

PowerMapper merupakan alat yang mudah digunakan untuk pengujian, dan analisis situs web. Produk-produk *PowerMapper* telah digunakan di lebih dari 50 negara oleh sejumlah organisasi besar dunia. *PowerMapper* menawarkan solusi yang sederhana, dinamis, dan efektif dalam pembuatan peta situs (*siteMaps*). Semua alat yang mereka kembangkan dapat dijalankan dengan satu klik, sebagai bagian dari komitmen mereka untuk menyediakan solusi yang mudah diakses oleh semua kalangan pengguna. Selain itu, *PowerMapper* secara otomatis menghasilkan peta situs dalam berbagai gaya visual yang menarik guna mendukung perjalanan

pengalaman pengguna (UX). Dengan kemampuan membangun peta situs berkualitas secara instan, alat ini menghilangkan kebutuhan akan biaya dan tenaga kerja yang biasanya diperlukan dalam pemetaan situs secara manual. Kepercayaan terhadap solusi pemetaan situs dari *PowerMapper* tercermin dari fakta bahwa lebih dari 30% perusahaan Fortune 100 mengandalkan produk ini [44].

PowerMapper digunakan untuk menguji kompatibilitas sistem di berbagai lingkungan. Alat ini memastikan bahwa sistem dapat berjalan tanpa masalah pada berbagai jenis *browser* web [45]. *PowerMapper* juga tersedia secara *online* dan digunakan untuk mengukur tingkat aksesibilitas sebuah situs web. Dalam pengujian aksesibilitas, *PowerMapper* menganalisis sampai dengan sepuluh halaman situs dan memberikan rincian terkait berbagai masalah kesalahan yang ditemukan. Komponen utama yang diukur oleh *PowerMapper* dalam evaluasi aksesibilitas meliputi kualitas keseluruhan (*Overall Quality*), kesalahan (*Errors*), aksesibilitas (*Accessibility*), kompatibilitas (*Compatibility*), pencarian (*Search*), kepatuhan terhadap standar (*Standards*), kegunaan (*Usability*), serta isu-isu umum (*Overall Issues*). Semua komponen ini kemudian digabungkan untuk menghasilkan evaluasi menyeluruh atas performa dan aksesibilitas situs web yang diuji [46].

PowerMapper merupakan alat bantu otomatis untuk pengujian kualitas web yang tersedia dalam beberapa edisi dengan model lisensi berbeda. Tersedia versi gratis yang dapat diakses langsung melalui situs resminya, meskipun bersifat sekali pakai dan hasil datanya tidak tersimpan. Pada edisi *Desktop Standard*, lisensi bersifat *perpetual* (sekali beli) dengan harga sekitar US\$149 per pengguna dan fitur dasar seperti pembuatan sitemap visual hingga 22.000 halaman, sementara edisi *Desktop Professional* dibanderol sekitar US\$349 per pengguna dengan tambahan fitur ekspor *Excel*, catatan, serta *scheduler*. Tersedia pula *Desktop Suite* yang menggabungkan *PowerMapper* dan *SortSite* dengan harga sekitar US\$499 per pengguna, serta opsi biaya upgrade tahunan opsional untuk pembaruan versi dan dukungan teknis. Untuk kebutuhan berbasis *cloud*, *PowerMapper* menawarkan *OnDemand Suite* dengan model berlangganan, mulai dari paket kecil seharga US\$49 per bulan (1 pengguna), paket menengah US\$149 per bulan (3 pengguna),

hingga paket besar US\$299 per bulan (10 pengguna). Dengan fleksibilitas lisensi *perpetual* maupun berlangganan *cloud*, *PowerMapper* dapat digunakan baik oleh individu maupun organisasi dalam melakukan pengujian aksesibilitas, validasi standar web, kompatibilitas lintas *browser*, serta pemetaan struktur situs [44].

2.13 User Experience Questionnaire (UEQ)

User Experience Questionnaire (UEQ) adalah kuesioner yang cepat dan andal untuk mengukur pengalaman pengguna terhadap produk interaktif. UEQ tersedia secara gratis dalam lebih dari 30 bahasa dan mudah digunakan berkat materi pendukung yang lengkap. Kuesioner ini mengukur berbagai aspek pengalaman pengguna, termasuk aspek *usability* klasik seperti efisiensi, kemudahan pemahaman, dan keandalan, serta aspek pengalaman pengguna seperti orisinalitas dan stimulasi [47].

Metode *User Experience Questionnaire* (UEQ) merupakan salah satu metode pengukuran dalam uji kegunaan (*usability testing*) yang digunakan untuk memperoleh gambaran komprehensif dan praktis terkait aspek *usability* dan pengalaman pengguna melalui survei penilaian kualitas subjektif. Hasil evaluasi menggunakan UEQ memungkinkan identifikasi area yang perlu ditingkatkan, sehingga dapat memprediksi perbaikan yang akan memberikan dampak signifikan terhadap pengalaman pengguna [48].

2.14 User Acceptance Testing (UAT)

User Acceptance Testing (UAT) merupakan tahap pengujian akhir dalam proses pengembangan perangkat lunak, yang biasanya dilakukan oleh klien atau pengguna akhir. Tujuan utama dari UAT adalah untuk memastikan bahwa sistem yang dikembangkan benar-benar sesuai dengan kebutuhan pengguna dan dapat memberikan manfaat nyata sebelum diimplementasikan dalam lingkungan operasional yang sesungguhnya [49].

UAT memberikan kesempatan kepada pengguna untuk berinteraksi langsung dengan perangkat lunak sebelum resmi diluncurkan. Melalui proses ini, pengguna

dapat memverifikasi apakah fungsi-fungsi yang ada berjalan sebagaimana mestinya sesuai dengan kebutuhan dan harapan mereka. Dengan demikian, UAT menjadi bukti bahwa sistem yang dikembangkan telah diterima oleh pengguna karena memenuhi kebutuhan yang telah ditetapkan. Selain itu, UAT juga dapat dilakukan oleh staf atau karyawan perusahaan yang berperan sebagai pengguna akhir. Mereka berfungsi sebagai penguji untuk memastikan bahwa sistem dapat digunakan dengan baik dalam konteks pekerjaan nyata. Hasil dari pengujian ini kemudian didokumentasikan sebagai bukti penerimaan, sekaligus menjadi indikator bahwa sistem layak untuk diimplementasikan [50].

Dalam beberapa penelitian, UAT juga digunakan untuk menguji aspek kegunaan (*usability*) baik secara internal maupun eksternal. Hal ini bertujuan agar perangkat lunak, termasuk aplikasi maupun sistem pembelajaran digital, benar-benar sesuai dengan kebutuhan pengguna serta memenuhi kriteria fungsional dan edukatif yang telah ditentukan [50].

2.15 Zed Attack Proxy (ZAP)

Zed Attack Proxy (ZAP) adalah perangkat lunak keamanan *open source* yang dikembangkan sebagai proyek unggulan oleh *Open Web Application Security Project (OWASP)*. ZAP dirancang untuk membantu dalam *penetration testing* aplikasi web dengan mendeteksi dan mengevaluasi potensi kerentanan keamanan. *penetration test* adalah proses yang dilakukan untuk mengungkap dan menemukan kerentanan pada suatu sistem [51].

OWASP ZAP merupakan sebuah alat yang tersedia secara gratis dan *open source* yang dirancang untuk mendeteksi kerentanan pada aplikasi web. ZAP merupakan alat integrasi pengujian penetrasi yang cocok digunakan oleh pengembang maupun penguji fungsional dengan berbagai tingkat pengalaman keamanan. Alat ini menyediakan pemindai otomatis maupun fungsi deteksi kerentanan secara manual. Metodologi OWASP mencakup pemahaman terhadap kerangka kerja keamanan, pengenalan risiko-risiko kritis yang dihadapi oleh aplikasi web, serta penentuan ruang lingkup dan tujuan penerapan langkah-langkah keamanan. Berdasarkan

laporan, informasi kerentanan yang diperoleh dari pemindaian menggunakan ZAP dikategorikan dalam tingkat risiko tinggi, sedang, rendah, serta informasional [52].

2.16 *SonarQube*

SonarQube adalah *platform open-source* yang digunakan untuk melakukan *static code analysis* guna mengevaluasi kualitas kode sumber (*source code*) dari suatu perangkat lunak. Alat ini membantu pengembang dan tim penguji untuk mengidentifikasi masalah kualitas kode, seperti *bug*, kerentanan keamanan, duplikasi kode, dan ketidakpatuhan terhadap standar pengkodean. Dalam konteks standar ISO/IEC 25010:2023, *SonarQube* sangat berguna untuk mengukur *maintainability* (kemampuan pemeliharaan), yang mencakup aspek-aspek seperti modularitas, kemudahan analisis, dan stabilitas kode. Dengan menggunakan *SonarQube*, tim pengembang dapat memastikan bahwa kode yang dihasilkan mudah dipelihara dan ditingkatkan seiring waktu [53].

Aspek utama yang diukur oleh *SonarQube* adalah *maintainability*. *Maintainability* dalam ISO/IEC 25010:2023 mengacu pada kemampuan sistem untuk dimodifikasi dengan mudah, baik untuk memperbaiki *bug*, meningkatkan performa, atau menambahkan fitur baru. *SonarQube* mengukur *maintainability* melalui beberapa indikator, seperti *technical debt*, *code smells*, dan *complexity*. *Technical debt* mengukur estimasi waktu yang dibutuhkan untuk memperbaiki masalah dalam kode, sementara *code smells* mengidentifikasi pola kode yang berpotensi menyebabkan masalah di masa depan. *complexity* mengukur tingkat kerumitan kode, yang dapat memengaruhi kemudahan pemeliharaan. Alat ini bekerja dengan cara memindai kode sumber (*source code*) dan menganalisisnya berdasarkan sejumlah metrik kualitas, seperti kompleksitas kode, duplikasi kode, kepatuhan terhadap standar pengkodean [54].

SonarQube dapat diakses melalui *browser* yaitu dengan menggunakan *SonarCloud* yang mendukung berbagai bahasa pemrograman yang digunakan dalam mata kuliah STW, seperti HTML, CSS, XML, *Java*, dan *JavaScript* (SonarSource, 2023b). Alat ini memverifikasi kepatuhan kode sumber dari suatu aplikasi perangkat lunak

terhadap sekumpulan aturan yang telah ditentukan. Jika kode sumber yang dianalisis melanggar aturan tersebut, *SonarCloud* akan mendeteksi adanya masalah. Ketika *SonarCloud* menemukan sebuah masalah, alat ini memberikan informasi kepada pengembang mengenai jenis masalah tersebut, yang dapat berupa kerentanan (*vulnerability*), titik rawan keamanan (*security hotspot*), *bug*, atau *code smell*. Selain itu, *SonarCloud* menyertakan deskripsi masalah serta saran atau langkah untuk mengatasinya. Setiap masalah juga diberikan tingkat keparahan (*severity level*) yang berkisar dari yang paling rendah hingga yang paling tinggi [55].

SonarCloud dapat digunakan secara gratis untuk proyek *open source* memiliki batas 50 ribu baris kode pribadi dan maksimal 5 anggota, sedangkan untuk *upgrade* tersedia model berlangganan mulai dari US\$32 per bulan untuk 100.000 baris kode, dengan biaya tambahan yang disesuaikan berdasarkan jumlah baris kode yang dianalisis. Dengan kemudahan integrasi, dukungan otomatisasi *Continuous Integration/Continuous Deployment* (CI/CD), serta kemampuan kolaborasi tim secara *real-time*, *SonarCloud* menjadi salah satu solusi populer dalam menjaga kualitas dan keamanan kode sumber pada proyek perangkat lunak modern [56].

2.17 Virtual Private Server (VPS)

Virtual Private Server (VPS) merupakan sebuah mesin virtual yang berfungsi untuk mengelola perangkat lunak serta data yang dibutuhkan dalam menjalankan aplikasi maupun situs web. Disebut virtual karena VPS hanya memanfaatkan sebagian dari sumber daya fisik yang dimiliki server utama yang dikelola oleh penyedia layanan pihak ketiga. Meskipun demikian, pengguna tetap memperoleh alokasi sumber daya khusus pada perangkat keras tersebut sehingga dapat mengoperasikan aplikasi atau situs web secara lebih stabil dan terkontrol [57].

Sumber daya atau *resources* pada VPS merujuk pada kapasitas server virtual yang digunakan, meliputi RAM, CPU, *Disk Space*, dan *Bandwidth*. Keempat komponen ini memiliki peranan penting karena saling berhubungan dalam menentukan kecepatan, kapasitas, serta keandalan performa server. RAM (*Random Access*

Memory) berfungsi sebagai ruang memori sementara yang memungkinkan server menjalankan berbagai aplikasi dan proses secara bersamaan. CPU (*Central Processing Unit*) bertugas memproses instruksi dan data yang dibutuhkan untuk menjalankan aplikasi maupun situs web. *Disk Space* menyediakan ruang penyimpanan bagi seluruh file, data, dan konten, baik berupa file statis seperti gambar maupun file dinamis seperti basis data. Adapun *bandwidth*, atau sering disebut transfer data, mengacu pada kapasitas total data yang dapat keluar masuk dari VPS dalam periode tertentu, sehingga semakin besar *bandwidth* yang tersedia, semakin optimal pula akses dan kinerja layanan yang dijalankan [58].

2.18 Penelitian Terkait

Dalam penyusunan skripsi ini, terdapat beberapa penelitian terkait yang dijadikan sebagai referensi untuk mendukung dan memperkuat dasar penelitian. Adapun rujukan pertama adalah penelitian oleh Abdillah dkk. (2024) menilai *website XYZ.com* yang menunjukkan hasil layak pada aspek efisiensi, keandalan, dan portabilitas, namun masih memiliki kelemahan pada keamanan terutama ancaman *cross site scripting* [59]. Hakim (2024) melakukan evaluasi aplikasi CBT di SMP Islam Dian Didaktika yang dinyatakan memenuhi standar ISO/IEC 25010 secara keseluruhan dengan *usability* yang sangat baik [45]. Sementara Nocera et al. (2025) mengkaji pengaruh pelatihan penggunaan *SonarCloud* pada mahasiswa pengembang aplikasi web, memperlihatkan peningkatan signifikan pada reliabilitas dan *maintainability* meskipun pelatihan hanya fokus pada keamanan, menandakan motivasi internal dalam peningkatan kualitas perangkat lunak. Temuan ini menegaskan bahwa metode ISO/IEC 25010 yang didukung oleh alat pengujian modern efektif dalam memberikan penilaian kualitas yang objektif dan relevan pada konteks perangkat lunak yang beragam [55].

Erlangga et al. (2023) menerapkan ISO 25010 pada *Learning Management System (LMS) Moodle* dalam pembelajaran fisika dengan menekankan desain emosional melalui pendekatan retorika (*ethos, pathos, logos*) untuk meningkatkan pengalaman pengguna, di mana hasil pengujian *User Experience Questionnaire (UEQ)* terhadap 122 mahasiswa menunjukkan kualitas pragmatis seperti kejelasan dan efisiensi di

atas rata-rata, sedangkan aspek hedonik seperti stimulasi dan kebaruan masih relatif rendah [60]. Dalam konteks berbeda, Agpaoa et al. (2024) mengembangkan aplikasi web e201 file untuk pengelolaan data pegawai di lembaga pendidikan dengan mengacu pada ISO 25010, khususnya aspek fungsionalitas, kegunaan, dan keamanan, menggunakan OWASP ZAP untuk mendeteksi kerentanan, yang setelah perbaikan menghasilkan sistem siap *live deployment* dengan skor penerimaan pengguna rata-rata 4,41 dari skala 5 [52]. Sementara itu, Bilal Naqvi dkk. (2025) mengevaluasi kualitas *Low-Code/No-Code Development Platforms* (LC/NC) berdasarkan ISO 25010:2023 pada sembilan karakteristik kualitas perangkat lunak, dan menemukan bahwa meskipun *platform* LC/NC unggul dalam aspek *usability* dan *accessibility*, tantangan masih terdapat pada *performance efficiency*, *security*, dan *maintainability*, terutama dalam pengembangan aplikasi yang menuntut tingkat penyesuaian tinggi dan keamanan ketat [6].

Selanjutnya dalam aspek pengembangan sistem informasi berbasis web, Agustina Eka Kumala, dkk (2018) mengembangkan Sistem Informasi Monitoring Perkembangan Sapi menggunakan PHP dan MySQL. Pengujian mencakup penginputan data sapi, penilaian tumbuh kembang sapi, dan pembuatan laporan perkembangan sapi, yang menunjukkan bahwa sistem mampu mempermudah proses monitoring dan menghasilkan informasi yang akurat [4]. Sejalan dengan itu, penelitian oleh Zulvita, dkk (2024) merancang sistem informasi berbasis web untuk objek wisata di Kabupaten Pinrang menggunakan metode *Research and Development* (R&D) dengan model SDLC serta pengujian berdasarkan ISO 25010. Pengujian terhadap delapan aspek kualitas menunjukkan bahwa sistem layak digunakan dengan keamanan cukup baik, kompatibilitas luas, serta performa yang efisien [61]. Selain itu, penelitian oleh Mustari S. Lamada, dkk (2020) menilai kualitas Sistem Monitoring Perkuliahan dengan *Black-box Testing*, *USE Questionnaire*, *stress testing (Server Stress Tools)*, *Load testing (GTmetrix)*, dan *Maintainability Index*. Hasil menunjukkan *Functional Suitability* 100%, *Usability* 88,5%, *Reliability* 100%, *Performance Efficiency* 4,2 detik, dan *Maintainability* 100%, sehingga sistem memenuhi standar ISO 25010 [62].

Pengujian aplikasi menggunakan *Software Testing Life Cycle* (STLC) dilakukan oleh Kurnia Gusti Ayu, dkk (2024) menganalisis aplikasi HESTI (*Helpdesk Technology Information*) dengan *black-box testing* menggunakan *Katalon Studio*. Pengujian terhadap *functional suitability*, *compatibility*, dan *performance efficiency* menunjukkan *functional suitability* 100%, *performance efficiency* 90% (*Grade A* dari *GTmetrix*), serta tidak ditemukan permasalahan dalam *compatibility*, sehingga aplikasi dinilai memiliki kualitas tinggi [33]. Selain itu penelitian oleh Mutiara, dkk, (2024) bertujuan untuk menguji kualitas Sistem Informasi Indo Towing dengan metode *automation testing* menggunakan *Katalon Studio*. Pengujian dilakukan berdasarkan *Software Testing Life Cycle* (STLC) dengan teknik *equivalence partitioning*, yang mencakup tahap *requirement analysis*, *test planning*, *test case development*, *environment setup*, *test execution*, dan *test cycle closure*. Karakteristik yang diuji meliputi fungsi sistem informasi, dengan 16 *test case* yang mencakup fitur utama seperti pemesanan layanan *towing* dan verifikasi kode pemesanan. Hasil pengujian menunjukkan bahwa semua *test case* berhasil dijalankan tanpa ditemukan *bug*, sehingga sistem dinyatakan berjalan dengan baik dan sesuai dengan fungsinya [28].

Terkait metode pengujian lain, penelitian oleh Dyah Darma Andayani dan Gufran Darma Dirawan menggunakan pendekatan *waterfall* dalam SDLC dan pengujian dengan ISO 9126. Pengujian *functionality* dilakukan dengan *white-box testing*, *reliability* dengan *webserver stress tool*, *reliability* dengan *browserstack*, dan *usability* dengan kuesioner. Hasil menunjukkan *functionality* 100%, *Reliability* stabil dengan 5 pengguna simultan, *reliability* berjalan di berbagai *browser*, dan *usability* mencapai 87% kepuasan pengguna [63].

Selanjutnya, penelitian oleh Arista Pratama, dkk (2022) bertujuan untuk mengevaluasi pengalaman pengguna Sistem Informasi Pembelajaran Terpadu menggunakan metode *User Experience Questionnaire* (UEQ). Pengujian dilakukan dengan menyebarkan kuesioner *online* kepada 50 mahasiswa, mengukur enam aspek utama dalam UEQ: *Attractiveness*, *perspicuity*, *efficiency*, *dependability*, *stimulation*, dan *novelty*. Hasil penelitian menunjukkan bahwa

attractiveness (2.121), *perspicuity* (2.152), *efficiency* (2.319), dan *stimulation* (1.716) masuk dalam kategori *excellent*, sedangkan *dependability* (1.505) dalam kategori *good*, dan *novelty* (1.020) dalam kategori *above average*. Kesimpulannya, sistem ini memiliki pengalaman pengguna yang baik dan dapat ditingkatkan lebih lanjut pada aspek *novelty* [48].

Selain itu dalam aspek keamanan, penelitian oleh Nopan Pirsas dan Sumijan (2020) meningkatkan keamanan SIPADUKO dengan *Grey-Box Penetration Test* menggunakan *Computer Assisted Audit Techniques* (CAAT). Pengujian dengan OWASP ZAP menemukan 97 kerentanan kategori tinggi, 1 sedang, dan 26 rendah, memberikan rekomendasi bagi Dinas Komunikasi dan Informatika Kota Payakumbuh dalam meningkatkan keamanan sistem [51].

Terakhir, dalam pengujian performa, Cahaya Putri Agustika, dkk (2021) menguji aplikasi *Greenwallet* PT. Terminal Teluk Lamong menggunakan *Load testing* dengan *Apache JMeter*. Pengujian dilakukan pada *server localhost* dan *hosting* untuk mengukur *response time*, dengan target maksimal 0,8 detik (*localhost*) dan 1 detik (*hosting*). Hasil menunjukkan sebagian besar skenario berhasil dengan *response time* di bawah 1 detik, kecuali fitur registrasi yang melebihi batas waktu yang ditentukan [64].

Untuk mempermudah pemahaman terhadap berbagai sumber literatur yang telah disebutkan, disusun sebuah tabel yang merangkum poin-poin penting yang dapat dijadikan referensi dalam penelitian ini. Berikut adalah tabel tersebut.

Tabel 4. Penelitian Terkait

No	Peneliti dan Tahun	Tujuan	Metode & Alat	Karakteristik yang Diuji	Hasil Penelitian
1.	Abdillah dkk. (2024) [59].	Menganalisis kualitas <i>website XYZ.com</i> menggunakan	<i>GTmetrix</i> , <i>WAPT</i> , <i>Apache JMeter</i> , <i>Acunetix</i> , <i>Sucuri</i>	<i>Performance</i> <i>Efficiency</i> , <i>Reliability</i> , <i>Security</i> , <i>Reliability</i>	Website layak pada <i>Performance</i> <i>Efficiency</i> , <i>Reliability</i> , dan

No	Peneliti dan Tahun	Tujuan	Metode & Alat	Karakteristik yang Diuji	Hasil Penelitian
		ISO/IEC 25010			<i>Reliability</i> ; belum layak pada <i>Security</i> .
2.	Hakim (2024) [45].	Evaluasi kualitas aplikasi CBT SMP Islam Dian Didaktika	<i>GTmetrix</i> , <i>WAPT</i> , <i>Sucuri</i> , <i>SiteCheck</i> , <i>Lambdatest</i> , <i>PowerMapper</i>	<i>Functional Suitability</i> , <i>Reliability</i> , <i>Usability</i> , <i>Performance</i> , <i>Efficiency</i> , <i>Security</i> , <i>Maintainability</i> , <i>Reliability</i> , <i>Compatibility</i>	Aplikasi CBT layak dan memenuhi standar ISO 25010 dengan nilai <i>usability</i> sangat baik.
3.	Nocera et al. (2025) [55].	Menilai dampak penggunaan <i>SonarCloud</i> pada kualitas aplikasi web mahasiswa	<i>SonarCloud (Static Analysis Tool)</i>	<i>Security</i> , <i>Reliability</i> , <i>Maintainability</i>	Pelatihan dengan <i>SonarCloud</i> meningkatkan signifikan reliabilitas dan <i>maintainability</i> meskipun tidak fokus pelatihan pada aspek ini.
4.	Erlangga et al. (2023) [60].	Meningkatkan kualitas UX <i>LMS Moodle</i> dengan emotional design	<i>Design-Based Research</i> , UEQ	<i>Attractiveness</i> , <i>Clarity</i> , <i>Efficiency</i> , <i>Dependability</i> , <i>Stimulation</i> , <i>Novelty</i>	Kualitas pragmatis di atas rata-rata; kualitas hedonik di bawah rata-rata; desain retorika meningkatkan UX
5.	Agpaoa et al. (2024) [52].	Pengembangan aplikasi web e201 file dengan pengamanan siber	R&D model, <i>Agile Iterative</i> , OWASP ZAP (ISO 25010)	<i>Functional Suitability</i> , <i>Usability</i> , <i>Security</i>	Sangat diterima pengguna, peningkatan keamanan dengan pengaturan

No	Peneliti dan Tahun	Tujuan	Metode & Alat	Karakteristik yang Diuji	Hasil Penelitian
					header dan token Anti-CSRF
6.	Bilal Naqvi, dkk (2025) [6].	Mengevaluasi kualitas <i>Low-Code/No-Code Development Platforms</i> (LC/NC) berdasarkan standar ISO 25010:2023.	Analisis berdasarkan ISO 25010	<i>Functional Suitability, Performance Efficiency, Compatibility, Usability, Reliability, Security, Maintainability, Flexibility, dan Safety.</i>	Memiliki keunggulan dalam <i>usability</i> dan <i>accessibility</i> , tetapi masih memiliki kelemahan dalam <i>Performance Efficiency, security, dan maintainability</i> , terutama untuk aplikasi dengan kebutuhan kompleks dan keamanan tinggi.
7.	Mustari S, dkk (2020)[62]	Menjamin kualitas Sistem Monitoring Perkuliahan agar tidak terjadi <i>error</i> serta memastikan fitur dan fungsionalitasnya sesuai standar ISO 25010.	Teknik : <i>Black box testing.</i> Alat : <i>Testing Use Questionnaire, Server Stress Tools, GTmetrix</i>	<i>Functional Suitability, Usability, Reliability, Performance, Efficiency, Maintainability</i>	Hasil pengujian menunjukkan <i>Functional Suitability</i> 100% (baik), <i>Usability</i> 88.5% (sangat layak), <i>Reliability</i> 100% (lolos), <i>Performance Efficiency</i> 4.2 detik (diterima), dan <i>Maintainability</i> 100 (sangat mudah dirawat).
8.	Agustina Eka Kumala, dkk (2018) [4]	Mengembangkan Sistem Informasi Monitoring Perkembangan Sapi untuk	Metode : DFD. Alat :	Penginputan data sapi, penilaian tumbuh kembang sapi, dan pembuatan	Sistem mampu mempermudah proses monitoring, serta menghasilkan

No	Peneliti dan Tahun	Tujuan	Metode & Alat	Karakteristik yang Diuji	Hasil Penelitian
		Dinas Peternakan dan Kesehatan Hewan Provinsi Lampung.	PHP, <i>MySQL</i> , dan <i>Adobe Dreamweaver</i> .	laporan perkembangan sapi.	informasi yang akurat dan tepat waktu dalam laporan perkembangan sapi.
9.	Zulvita, Sanatang, dan Abdul Wahid (2024) [61]	Merancang Sistem Informasi Pariwisata berbasis Web untuk objek wisata di Kabupaten Pinrang.	<i>Research and Development (R&D)</i> dengan <i>System Development Life Cycle (SDLC)</i> dan pengujian ISO 25010.	<i>Functional Suitability, usability, Compatibility, Performance Efficiency, Reliability, security, maintainability, Reliability.</i>	Sistem layak digunakan dengan keamanan cukup baik, kompatibilitas luas, dan kinerja efisien.
10.	Kurnia Gusti Ayu, dkk (2024) [33]	Menganalisis kualitas aplikasi HESTI untuk memastikan seluruh fitur berjalan dengan baik dan andal.	Metode : STLC (<i>Software Testing Life Cycle</i>). Alat : <i>Katalon Studio, GTmetrix, PowerMapper</i> dan analisis kualitas ISO/IEC 25010	<i>Functional Suitability, Performance Efficiency, Compatibility</i>	<i>Functional Suitability:</i> 100%; <i>Performance Efficiency:</i> 90%; <i>Compatibility:</i> tidak ditemukan masalah
11.	Mutiara, dkk (2023) [28].	Menguji kualitas Sistem Informasi Indo Towing menggunakan metode <i>Automation Testing</i> .	Metode: <i>Software Testing Life Cycle (STLC)</i> Alat: <i>Katalon Studio, Equivalence Partitioning</i> .	Fungsi sistem, informasi, Pemesanan layanan <i>towing</i> , Verifikasi kode pemesanan.	Semua <i>test case</i> berhasil dijalankan tanpa ditemukan <i>bug</i> , sehingga sistem dinyatakan berjalan dengan baik dan sesuai fungsinya.
12.	Dyah Darma Andayani &	Mengembangkan sistem	Standar ISO 9126	<i>Functionality, Reliability,</i>	Sistem layak digunakan

No	Peneliti dan Tahun	Tujuan	Metode & Alat	Karakteristik yang Diuji	Hasil Penelitian
	Gufran Darma Dirawan (2015) [63]	informasi publikasi ilmiah berbasis web untuk mempermudah publikasi karya ilmiah dosen dan mahasiswa.	Metode: <i>Waterfall Model</i> (SDLC) Teknik : <i>White box testing</i> Alat : <i>WebServer Stress Tool, BrowserStack, Kuesioner Usability.</i>	<i>Reliability, Usability</i>	dengan <i>Functionality</i> 100%, <i>Reliability</i> stabil, <i>Reliability</i> lancar di berbagai browser, dan <i>Usability</i> mencapai 87% kepuasan pengguna.
13.	Arista Pratama, dkk (2022) [48]	Mengevaluasi pengalaman pengguna Sistem Informasi Pembelajaran Terpadu menggunakan UEQ.	Metode: <i>User Experience Questionnaire</i> (UEQ) Alat: Kuesioner, UEQ Data Analysis Tools	<i>Attractiveness, Perspicuity, Efficiency, Dependability, Stimulation, Novelty</i>	Hasil evaluasi menunjukkan <i>Attractiveness</i> (2.121), <i>Perspicuity</i> (2.152), <i>Efficiency</i> (2.319), dan <i>Stimulation</i> (1.716) masuk kategori <i>Excellent</i> , <i>Dependability</i> (1.505) kategori <i>good</i> , dan <i>Novelty</i> (1.020) kategori <i>above average</i> .
14.	Nopan Pirsia & Sumijan (2020) [51]	Meningkatkan keamanan Sistem Informasi Puskesmas Terpadu (SIPADUKO) Kota Payakumbuh.	Metode: <i>Grey-Box Penetration Test</i> Alat: OWASP ZAP, Computer Assisted Audit Techniques (CAAT)	<i>Cross-Site Scripting</i> (XSS) - <i>SQL Injection, Path Traversal, Information Disclosure, Keamanan Cookie</i>	Ditemukan 97 kerentanan tinggi, 1 sedang, dan 26 rendah. Hasil pengujian digunakan untuk meningkatkan keamanan SIPADUKO.

No	Peneliti dan Tahun	Tujuan	Metode & Alat	Karakteristik yang Diuji	Hasil Penelitian
15.	Cahaya Putri Agustika, dkk (2021) [64]	Menguji performa aplikasi web <i>Greenwallet</i> PT. Terminal Teluk Lamong menggunakan.	<i>Load testing</i> alat : <i>Apache JMeter</i>	<i>Response time</i> dalam berbagai skenario penggunaan aplikasi.	Sebagian besar skenario pengujian berhasil dengan <i>response time</i> di bawah 1 detik, kecuali fitur registrasi yang mengalami <i>response time</i> lebih dari 1 detik.

Dari berbagai penelitian ini, dapat disimpulkan bahwa penerapan ISO/IEC 25010 dalam pengujian kualitas perangkat lunak memberikan hasil yang akurat dalam menilai berbagai aspek, seperti *functional suitability*, *interaction capability*, *reliability*, *performance efficiency*, *compatibility*, *security* dan *maintainability*. Selain itu, penggunaan STLC juga terbukti meningkatkan kualitas perangkat lunak. Hal ini menunjukkan bahwa kombinasi pengujian kualitas perangkat lunak dan metodologi pengembangan yang tepat sangat penting dalam menghasilkan sistem informasi yang berkualitas tinggi dan sesuai dengan kebutuhan pengguna. Berdasarkan penelitian-penelitian tersebut, penerapan ISO/IEC 25010 terbukti relevan untuk mengevaluasi kualitas perangkat lunak, termasuk dalam studi kasus SI MBEK.

III. METODOLOGI PENELITIAN

3.1 Waktu dan Tempat

Dalam menyelesaikan penelitian ini, telah dibuat jadwal pengerjaan serta tempat berlangsungnya kegiatan penelitian. Adapun rinciannya seperti :

Tempat : Laboratorium Teknik Komputer Jurusan Teknik Elektro,
Universitas Lampung

Waktu : Maret 2025 – Agustus 2025

Tabel 5. Jadwal Pengerjaan Skripsi

[illegible]

3.2 Alat dan Bahan Penelitian

3.2.1 Alat

Adapun alat dan bahan yang dibutuhkan dalam penelitian ini adalah sebagai berikut:

Tabel 6. Alat Penelitian

No	Perangkat	Spesifikasi	Kegunaan
1.	Laptop <i>DELL Inspiron 15 3000</i>	<i>Intel Core i5, RAM 16 GB, SSD 512 GB, Windows 11</i>	Perangkat utama untuk pengembangan, pengujian sistem dan analisis laporan.
2.	<i>Katalon Studio</i>	<i>Versi 10.10</i>	Digunakan untuk pengujian <i>functional suitability</i>
3.	<i>JMeter</i>	<i>Apache JMeter versi 5.6.3</i>	Untuk melakukan pengujian <i>reliability</i>
4.	OWASP ZAP	Versi 2.26.0	Digunakan untuk pengujian kerentanan system.
5.	<i>SonarQube Cloud</i>	Berbasis Website	Alat yang berfokus pada analisis kualitas kode dan <i>maintainability</i> dalam pengembangan perangkat lunak.
7.	<i>Spreadsheet</i>	<i>Flatform Google</i>	Untuk membuat skenario <i>testing</i> dan analisis data <i>intraction capability</i> .
8.	<i>Google Chrome, Edge</i>	<i>Browser Web</i>	Sebagai <i>browser</i> yang digunakan.
9.	<i>Github</i>	Versi 3.3.18, Sebagai <i>platform hosting</i>	Untuk kontrol versi kode sumber dan kolaborasi tim dalam pengembangan proyek.
10.	<i>GTmetrix</i>	Berbasis Website	Untuk melakukan pengujian <i>performance efficiency</i>
10.	<i>PowerMapper</i>	Berbasis Website	Alat yang digunakan untuk pengujian <i>compatibility</i>
11.	<i>BlazeMeter</i>	Versi 6.6.7	Alat yang digunakan untuk mendukung pengujian <i>reliability</i> bersama <i>JMeter</i>
12.	WAPT	WAPT Pro 5.1	Alat pengujian yang digunakan untuk melakukan <i>load testing</i> , <i>stress testing</i> , dan <i>performance testing</i> terhadap sebuah <i>website</i> .

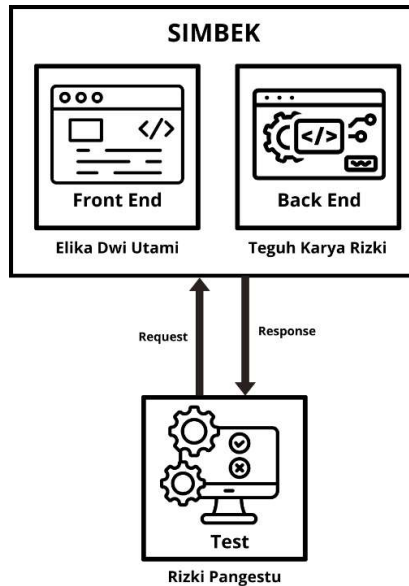
No	Perangkat	Spesifikasi	Kegunaan
13.	<i>User Experience Questionnaire</i> (UEQ)	Bahasa Indonesia	Alat pengujian yang digunakan untuk melakukan pengujian <i>interaction capability</i> .

3.2.2 Bahan

Bahan yang digunakan dalam penelitian ini mencakup data yang diperoleh dari Sistem Informasi Monitoring Pembesaran Kambing (SI MBEK) serta data kuesioner dari pengguna. Data pada sistem SI MBEK digunakan sebagai objek evaluasi untuk menguji karakteristik kualitas perangkat lunak berdasarkan standar ISO/IEC 25010:2023. Selain itu, data kuesioner dikumpulkan dari pengguna untuk mengevaluasi karakteristik *interaction capability*. Data yang diperoleh akan digunakan sebagai dasar analisis dalam pengujian, guna memastikan kualitas perangkat lunak SI MBEK sesuai dengan ISO/IEC 25010:2023.

3.3 Tim Penelitian

Tim proyek ini terdiri dari tiga anggota utama yang masing-masing memiliki peran strategis dalam pengembangan Sistem Informasi Pembesaran Kambing (SI MBEK). Peran dan tanggung jawab dijabarkan dalam tabel berikut untuk memberikan gambaran yang jelas mengenai kontribusi setiap anggota dalam proyek ini.



Gambar 3. Tim Penelitian

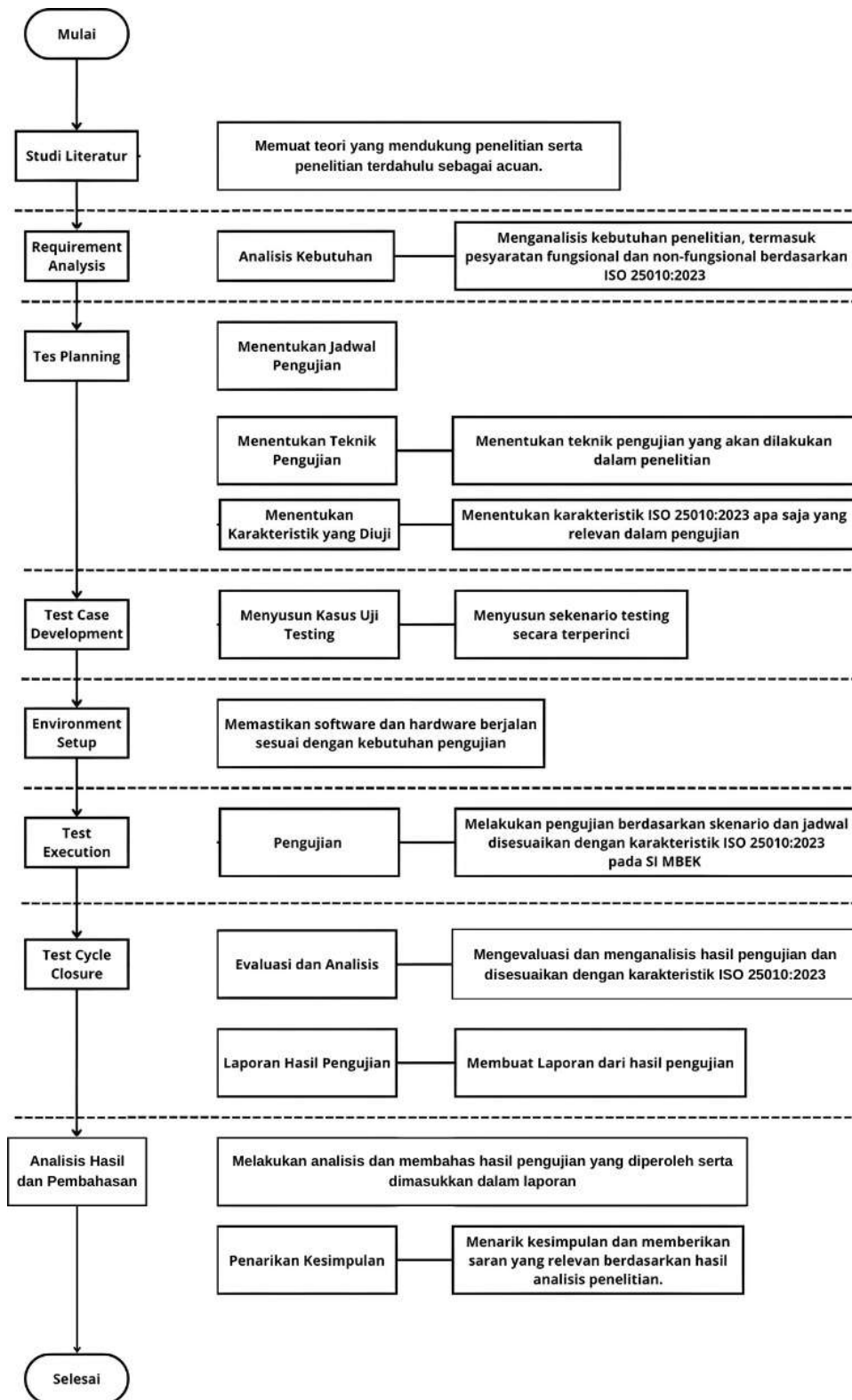
Tabel 7. Tim Penelitian

No	Nama	Peran	Deskripsi Pekerjaan
1.	Rizki Pangestu	<i>Quality Assurance</i>	- Menyusun dan mengelola <i>test case</i> berdasarkan standar ISO/IEC 25010:2023. - Melakukan pengujian perangkat lunak menggunakan <i>Katalon Studio</i>, <i>User Experience Questionnaire (UEQ)</i>, <i>JMeter</i>, <i>BlazeMeter</i>, <i>WAPT</i>, <i>ZAP</i> dan <i>SonarQube Cloud</i>. - Mengevaluasi kualitas perangkat lunak berdasarkan tujuh karakteristik ISO/IEC 25010:2023 yang diuji.
2.	Elika Dwi Utami	<i>Front End Dev</i>	- Mengembangkan <i>frontend</i> sistem untuk tampilan pengguna, termasuk fitur penjualan pada admin, pengguna dan pembeli. - Mengoptimalkan responsivitas agar sistem dapat diakses di berbagai perangkat. - Berkolaborasi dengan <i>backend</i> untuk memastikan visualisasi data sesuai kebutuhan pengguna.

No	Nama	Peran	Deskripsi Pekerjaan
3.	Teguh Karya Rizki	<i>Backend Dev</i>	• Membangun dan mengelola struktur <i>backend</i> menggunakan <i>Laravel</i> untuk mendukung fungsi utama sistem. • Mengembangkan fitur dan logika bisnis dalam <i>backend</i> agar sesuai dengan kebutuhan sistem. • Memastikan kompatibilitas sistem di berbagai perangkat dan <i>browser</i>.

3.4 Tahapan Penelitian

Tahapan penelitian ini terdiri dari beberapa langkah yang dilakukan secara sistematis untuk mencapai tujuan penelitian, yaitu mengevaluasi kualitas perangkat lunak Sistem Informasi Monitoring Pembesaran Kambing (SI MBEK). Penelitian ini dilakukan berdasarkan tahapan STLC, yang terdiri dari enam langkah utama dapat dilihat pada gambar berikut.



Gambar 4. Tahapan Penelitian STLC

Penelitian ini mengikuti tahapan *Software Testing Life Cycle* (STLC), yang terdiri dari enam tahap.

3.4.1 Requirement Analysis

Tahap pertama dalam *Software Testing Life Cycle* (STLC) adalah *requirement analysis*, yang berfokus pada identifikasi persyaratan pengujian yang dapat diuji. Proses ini melibatkan interaksi dengan *stakeholder* untuk memastikan bahwa setiap kebutuhan sistem dipahami dengan jelas sebelum pengujian dilakukan. Persyaratan tersebut dibagi menjadi dua kategori utama: persyaratan fungsional dan persyaratan non-fungsional yang akan di uji.

1. Persyaratan Fungsional

Persyaratan fungsional mencakup fitur dan fungsi utama perangkat lunak yang akan diuji. Berikut adalah tabel yang menunjukkan fitur yang diuji dan yang tidak diuji dalam penelitian ini.

Tabel 8. Persyaratan Fungsional

No	Halaman Pengguna yang Akan di Uji	Halaman Super Admin yang Akan di Uji
1.	Beranda	Masuk
2.	Produk	Dashboard
3.	Tentang Kami	Pengguna
4.	Kontak	Tambah
5.	Masuk	Produk
6.	Daftar	Penjualan
7.	Lupa Kata Sandi	Pengaturan
8.	Dashboard	Profil
9.	Produk	Perbarui Kata Sandi
10.	Transaksi	Keluar
11.	Profil	Lihat Web

2. Persyaratan Non-Fungsional

Persyaratan non-fungsional berfokus pada aspek kinerja, kemampuan sistem, dan keandalan sistem. Dalam penelitian ini, enam karakteristik dari ISO/IEC 25010:2023 yang akan diuji sebagai berikut.

Tabel 9. Persyaratan Non-Fungsional

No	Karakteristik yang Akan di Uji	Deskripsi
1.	<i>Functional Suitability</i>	Evaluasi kesesuaian fungsional sistem.
2.	<i>Performance Efficiency</i>	Pengujian waktu respons, penggunaan sumber daya, dan kapasitas sistem.
3.	<i>Compatibility</i>	Pengujian kompatibilitas dengan berbagai <i>browser</i> dan perangkat.
4.	<i>Interaction Capability</i>	Evaluasi pengalaman, kemudahan dan pemahaman penggunaan sistem.
5.	<i>Reliability</i>	Pengujian stabilitas sistem dalam berbagai kondisi beban.
6.	<i>Security</i>	Pengujian keamanan sistem.
7.	<i>Maintainability</i>	Analisis kemudahan pemeliharaan kode.

3.4.2 Test Planning

Test Planning adalah tahap perencanaan strategi pengujian yang mencakup pemilihan teknik pengujian serta sumber daya yang dibutuhkan. Pada tahap ini, pendekatan yang paling sesuai untuk menguji perangkat lunak ditentukan berdasarkan kebutuhan proyek.

1. Teknik Pengujian

Teknik pengujian yang digunakan dalam penelitian ini adalah *black box testing*, *load & stress testing*, *user experience questionnaire* (UEQ), *security testing*, *static code analysis*, dan *cross-browser testing*. Pengujian dilakukan secara otomatis untuk memastikan proses berjalan lebih cepat.

2. Alat yang Digunakan

Adapun alat yang digunakan dalam penelitian ini adalah sebagai berikut.

Tabel 10. Alat Pengujian

No	Alat	Gratis/Berbayar	Deskripsi
1.	<i>Katalon Studio</i>	Versi Gratis	Untuk pengujian <i>functional suitability</i>
2.	<i>GTmetrix</i>	Versi <i>trial</i> 7 hari	Untuk mengukur <i>performance efficiency</i>
3.	<i>PowerMapper</i>	Versi Gratis	Untuk mengukur <i>compatibility</i>
4.	<i>JMeter</i>	Gratis	Untuk mengukur <i>reliability</i>
5.	<i>Blaze Meter</i>	Versi Gratis	
6.	WAPT Pro	Versi <i>trial</i> 30 hari	
7.	<i>User Experience Questionnaire (UEQ)</i>	Gratis	Untuk mengevaluasi <i>interaction capability</i>
8.	<i>User Acceptance Testing (UAT)</i>	Gratis	Untuk mengevaluasi <i>interaction capability</i>
9.	<i>Zed Attack Proxy (ZAP)</i>	Gratis	Untuk pengujian <i>security</i>
10.	<i>SonarQube Cloud</i>	Versi Gratis	Untuk menganalisis <i>maintainability</i>

3.4.3 Test Case Development

Test Case Development merupakan tahap pengembangan dalam proses pengujian perangkat lunak yang berfokus pada penyusunan skenario pengujian sebagai acuan utama dalam pelaksanaan uji. Pada tahap ini, pembuatan *test case* dilakukan sebagai tanggung jawab utama. *Test case* sendiri merupakan serangkaian skenario yang dirancang secara sistematis untuk memastikan bahwa sistem yang diuji dapat berfungsi sesuai dengan persyaratan yang ditetapkan serta memenuhi standar kualitas yang diharapkan.

Penyusunan skenario *testing* dilakukan dengan mendokumentasikan setiap langkah pengujian, termasuk input, kondisi awal, langkah-langkah yang harus dilakukan, serta hasil yang diharapkan. Skenario ini disusun dalam bentuk tabel untuk mempermudah pemantauan dan pelaksanaan pengujian.

Tabel 11. Skenario *Testing*

<i>Test Case ID</i>	<i>Fitur</i>	<i>Test Case Description</i>	<i>Test Step</i>	<i>Expected Result</i>	<i>Actual Result</i>
01					
02					
03					
...					

3.4.4 *Test Environment Setup*

Tahap *test environment setup* bertujuan untuk memastikan bahwa lingkungan pengujian, baik dari sisi perangkat keras maupun perangkat lunak, telah disiapkan dengan benar sesuai rencana yang telah ditetapkan pada tahap sebelumnya. Lingkungan pengujian harus mendukung kebutuhan pengujian perangkat lunak secara optimal dan memungkinkan pelaksanaan seluruh skenario pengujian tanpa hambatan. Tahap ini dapat dilakukan bersamaan dengan fase *test case development*, namun tetap merupakan fase independen yang tidak terikat langsung dengan tahapan lainnya.

Pada tahap ini diperlukan pemahaman terhadap arsitektur sistem perangkat lunak untuk memastikan bahwa semua komponen telah dikonfigurasi dengan benar. Persiapan mencakup instalasi perangkat lunak yang diperlukan untuk pengujian, seperti *Katalon Studio*, *Apache JMeter*, *BlazeMeter*, *WAPT*, *ZAP*, dan *SonarQube Cloud*, serta mempersiapkan *User Experience Questionnaire* (UEQ). Selain itu, perangkat keras yang digunakan juga dipastikan memenuhi spesifikasi minimum, seperti laptop dan koneksi internet yang stabil.

3.4.5 *Test Execution*

Tahap *Test Execution* adalah langkah pelaksanaan pengujian berdasarkan *test case* yang telah dirancang pada tahap sebelumnya. Pada tahap ini, setiap *test case* dijalankan secara sistematis untuk memastikan bahwa perangkat lunak Sistem Informasi Monitoring Pembesaran Kambing (SI MBEK) memenuhi kebutuhan dan persyaratan yang telah ditentukan. Hasil dari pengujian ini berupa laporan yang

mencakup status keberhasilan setiap *test case*, serta identifikasi bug atau masalah yang ditemukan selama pengujian.

Pelaksanaan pengujian dilakukan secara terstruktur dengan mengikuti langkah-langkah berikut:

1. Menjalankan *Test Case* sesuai Rencana

Setiap *test case* yang telah dirancang dijalankan dengan menggunakan alat dan metode pengujian yang telah ditetapkan. Pengujian dilakukan berdasarkan skenario yang telah ditentukan, mencakup input yang dimasukkan ke dalam sistem, keluaran yang diharapkan, serta hasil aktual yang diperoleh.

2. Memberikan Status untuk Hasil Pengujian

Tabel 12. Status *Test Case*

Status	Keterangan
<i>PASS</i>	<i>Test Case</i> dinyatakan berhasil jika hasil aktual sesuai dengan hasil yang diharapkan.
<i>FAIL</i>	<i>Test Case</i> dinyatakan gagal jika hasil aktual tidak sesuai dengan hasil yang diharapkan.

3. Identifikasi dan Pelaporan *Bug*

Bug atau masalah yang ditemukan selama pengujian didokumentasikan dalam laporan *bug*. Laporan ini mencakup:

- Deskripsi *bug*: Penjelasan singkat tentang masalah yang ditemukan.
- Dampak: Penilaian tingkat keparahan *bug* terhadap fungsi sistem.
- Tangkapan layar atau *log*: Bukti visual atau teknis dari *bug*.
- Laporan *bug* disampaikan kepada tim pengembang untuk dikoreksi.

4. Melakukan Pengujian Ulang (*Retesting*)

Setelah *bug* diperbaiki oleh tim pengembang, pengujian ulang dilakukan untuk memastikan bahwa masalah telah teratasi dan fitur terkait berfungsi dengan baik. *retesting* ini dilakukan secara spesifik pada *test case* yang sebelumnya dinyatakan gagal, tanpa mengulang *test case* yang telah berhasil.

5. Dokumentasi Hasil Pengujian

Semua hasil pengujian, baik *test case* yang berhasil maupun yang gagal, didokumentasikan secara terperinci. Dokumentasi ini mencakup status setiap *test case*, temuan *bug*, dan hasil *retesting*. Dokumentasi hasil pengujian menjadi bagian penting dari laporan akhir yang akan digunakan untuk analisis dan evaluasi kualitas perangkat lunak.

Hasil dari tahap *test execution* adalah rekapitulasi lengkap mengenai status setiap *test case* yang diuji, laporan *bug* yang ditemukan, dan hasil pengujian ulang terhadap fitur yang telah diperbaiki. *Test case* yang berhasil dieksekusi sesuai dengan hasil yang diharapkan diberi status *PASS*, sedangkan *test case* yang gagal diberi status *FAIL* dengan detail deskripsi masalah, langkah reproduksi, dan tingkat keparahan *bug*. Semua *bug* yang ditemukan selama pengujian didokumentasikan dalam laporan *bug* dan disampaikan kepada tim pengembang untuk dilakukan perbaikan. Setelah perbaikan selesai, pengujian ulang dilakukan secara spesifik pada *test case* yang sebelumnya gagal, untuk memastikan bahwa masalah telah teratasi tanpa menimbulkan masalah baru. Dokumentasi hasil pengujian mencakup status akhir dari semua *test case*, daftar *bug* yang telah diperbaiki, dan konfirmasi keberhasilan *retesting*. Hasil dari tahap ini memberikan gambaran yang jelas mengenai tingkat kesiapan perangkat lunak, serta menjadi dasar untuk analisis lebih lanjut terkait kualitas perangkat lunak berdasarkan standar ISO/IEC 25010:2023.

3.4.6 Test Closure

Tahap *test cycle closure* merupakan langkah terakhir dalam *Software Testing Life Cycle* (STLC) yang berfokus pada evaluasi menyeluruh terhadap proses pengujian yang telah dilakukan. Tahap ini bertujuan untuk memastikan bahwa semua aktivitas pengujian telah diselesaikan, hasil pengujian terdokumentasi dengan baik, dan analisis lengkap terhadap temuan serta pelaksanaan pengujian telah dilakukan. *Test cycle closure* memberikan kesempatan bagi tim pengujian untuk merefleksikan proses yang telah berjalan, mengidentifikasi keberhasilan dan kelemahan, serta memberikan rekomendasi untuk perbaikan di masa mendatang.

Adapun aktivitas dalam *test cycle closure* adalah sebagai berikut :

1. Analisis Data

Analisis data dalam penelitian ini bertujuan untuk mengevaluasi hasil pengujian perangkat lunak Sistem Informasi Monitoring Pembesaran Kambing (SI MBEK) berdasarkan standar ISO/IEC 25010:2023. Data yang dianalisis mencakup hasil dari pengujian pada masing-masing karakteristik kualitas perangkat lunak yang diuji, yaitu *functional suitability*, *performance efficiency*, *compatibility*, *reliability*, *interaction capability* dan *maintainability*.

a. *Functional Suitability*

Pengujian pada karakteristik *functional suitability* dilakukan menggunakan *Katalon Studio* dengan tujuan untuk memastikan bahwa setiap fungsi dan fitur yang terdapat pada Sistem Informasi Monitoring Pembesaran Kambing (SI MBEK) telah berjalan sesuai dengan kebutuhan dan spesifikasi yang telah ditetapkan. Proses pengujian ini difokuskan pada tiga subkarakteristik, yaitu *functional completeness*, untuk menilai kelengkapan fungsi yang tersedia; *functional correctness*, untuk memverifikasi kebenaran hasil keluaran sistem sesuai input dan tujuan pengguna; serta *functional appropriateness*, untuk menilai kesesuaian fungsi dalam membantu pengguna mencapai tujuan penggunaan sistem secara efektif..

b. *Performance Efficiency*

Pengujian pada karakteristik *Performance Efficiency* dilakukan menggunakan *GTmetrix* dengan tujuan untuk mengukur kinerja sistem dalam menangani beban akses pengguna. *GTmetrix* digunakan untuk memperoleh berbagai metrik kinerja penting seperti *grade*, *performance score*, *structure score*, *Largest Contentful Paint (LCP)*, *Total Blocking Time (TBT)*, *Cumulative Layout Shift (CLS)*, serta jumlah *total requests*. Data hasil pengujian tersebut dianalisis berdasarkan tiga subkarakteristik utama, yaitu *time behaviour* yang menilai waktu respons sistem terhadap permintaan pengguna, *resource utilization* yang mengukur efisiensi penggunaan sumber daya seperti CPU dan

penyimpanan, serta *capacity* untuk memenuhi persyaratan mengenai batas maksimum suatu parameter produk.

Tabel 13. Parameter *GTmetrix*

Parameter	Keterangan
<i>GTmetrix Grade</i>	Merupakan evaluasi terhadap kinerja keseluruhan halaman. Penilaian ini mencerminkan seberapa cepat halaman yang dimuat bagi pengguna, serta seberapa baik halaman tersebut dibangun untuk mendukung performa.
<i>Performance</i>	Skor berdasarkan <i>Web Vitals</i> yang berfokus pada pengalaman pengguna akhir
<i>Structure</i>	Penilaian eksklusif dari <i>GTmetrix</i> yang menunjukkan seberapa baik struktur halaman yang dibangun untuk mencapai performa yang optimal.
<i>Web Vitals</i>	Serangkaian metrik inti yang menunjukkan apakah memberikan pengalaman yang cepat dan menyenangkan (menurut standar <i>Google</i>) kepada pengunjung.
LCP (<i>Largest Contentful Paint</i>)	Mengukur berapa lama waktu yang dibutuhkan hingga elemen konten terbesar (seperti gambar utama atau teks judul) di halaman terlihat oleh pengunjung di layar mereka.
TBT (<i>Total Blocking Time</i>)	Menunjukkan berapa banyak waktu yang tertahan oleh skrip selama proses pemuatan halaman. Untuk pengalaman pengguna yang optimal.
CLS (<i>Cumulative Layout Shift</i>)	Menunjukkan seberapa banyak perubahan tata letak yang dialami oleh pengunjung saat halaman Anda dimuat. Untuk memastikan kenyamanan pengguna, pertahankan skor CLS tidak lebih dari 0,1.
FCP (<i>First Contentful Paint</i>)	Waktu yang dibutuhkan sejak halaman mulai dimuat hingga <i>browser</i> menampilkan konten pertama yang “bermakna” (teks, gambar, kanvas, atau SVG) di layar.
INP (<i>Interaction to Next Paint</i>)	Metrik kinerja yang mengukur responsivitas halaman web terhadap interaksi pengguna (klik, ketukan, atau input keyboard).
TTFB (<i>Time To First Byte</i>)	Waktu yang dibutuhkan <i>browser</i> untuk menerima <i>byte</i> pertama dari server setelah permintaan HTTP dikirim.
CrUX (<i>Chrome User Experience Report</i>)	Data yang berisi metrik LCP, INP, CLS, FCP dan TTFB

Parameter	Keterangan
CPU (<i>Central Processing Unit</i>)	CPU memengaruhi kecepatan eksekusi <i>JavaScript</i> , rendering halaman, dan <i>respons</i> interaksi.

c. *Compatibility*

Pengujian pada karakteristik *compatibility* dilakukan menggunakan *PowerMapper* dengan tujuan untuk memastikan bahwa Sistem Informasi Monitoring Pembesaran Kambing (SI MBEK) dapat berfungsi dengan baik pada berbagai lingkungan dan perangkat. Hasil pengujian ini kemudian dianalisis berdasarkan dua subkarakteristik utama, yaitu *co-existence*, yang menilai kemampuan sistem untuk berjalan berdampingan dengan aplikasi lain tanpa menimbulkan gangguan, serta *interoperability*, yang mengevaluasi kemampuan sistem dalam beroperasi dengan berbagai *platform*, peramban (*browser*), dan perangkat yang berbeda tanpa menimbulkan kesalahan (*error*).

d. *Reliability*

Analisis pada karakteristik *reliability* dilakukan untuk mengevaluasi tingkat kestabilan dan ketahanan Sistem Informasi Monitoring Pembesaran Kambing (SI MBEK) terhadap kemungkinan terjadinya kesalahan selama proses operasional. Pengujian ini menggunakan tiga alat utama, yaitu *Apache JMeter*, *BlazeMeter*, dan WAPT, yang masing-masing berfungsi untuk mengukur performa sistem dalam kondisi beban tertentu serta kemampuan sistem dalam menjaga konsistensi layanan. Hasil dari pengujian ini kemudian dianalisis berdasarkan empat subkarakteristik, yaitu *faultlessness*, yang menilai kemampuan sistem untuk berfungsi tanpa kesalahan selama pengoperasian normal; *fault tolerance*, yang mengevaluasi kemampuan sistem untuk tetap beroperasi meskipun terjadi gangguan; *availability*, yang mengukur sejauh mana sistem tersedia dan dapat digunakan ketika dibutuhkan; serta *recoverability*, yang menilai kemampuan sistem untuk dipulihkan kembali setelah mengalami kegagalan.

e. *Interaction Capability*.

Analisis pada karakteristik *interaction capability* dilakukan untuk mengukur tingkat kemudahan dan kenyamanan pengguna dalam berinteraksi dengan Sistem Informasi Monitoring Pembesaran Kambing (SI MBEK). Pengujian ini dilakukan menggunakan *User Experience Questionnaire* (UEQ) dan *User Acceptance Testing* (UAT) untuk memperoleh gambaran mengenai persepsi pengguna terhadap kualitas interaksi sistem. Hasil pengujian kemudian dianalisis berdasarkan beberapa subkarakteristik, yaitu *appropriateness*, *recognizability*, *learnability*, *operability*, *user error protection*, *user engagement*, *inclusivity*, *user assistance*, dan *self-descriptiveness*.

Tabel 14. Aspek UEQ

Aspek	Ketengan
<i>Attractiveness</i>	Kesan terhadap sistem, apakah menyenangkan atau tidak.
<i>Perspicuity</i>	Kejelasan sistem, apakah mudah dipahami dan dipelajari.
<i>Efficiency</i>	Efisiensi dalam penggunaan, apakah cepat dan praktis.
<i>Dependability</i>	Tingkat kendali dan prediktabilitas interaksi, apakah dapat diandalkan.
<i>Stimulation</i>	Tingkat motivasi dan daya tarik sistem, apakah membuat pengguna bersemangat.
<i>Novelty</i>	Kebaruan sistem, apakah inovatif dan kreatif dibandingkan pengalaman sebelumnya.

Tabel 15. Keriteria Penilaian UAT

Presentase	Keterangan
0% - 20%	Sangat Buruk
21% - 40%	Buruk
41% - 60%	Netral
61% - 80%	Baik
81% - 100%	Sangat Baik

f. *Security*

Analisis pada karakteristik *security* dilakukan untuk mengukur kemampuan Sistem Informasi Monitoring Pembesaran Kambing (SI MBEK) dalam melindungi informasi dan data dari berbagai ancaman serta serangan yang dapat membahayakan integritas dan kerahasiaan sistem. Pengujian ini bertujuan untuk memastikan bahwa sistem memiliki mekanisme keamanan yang memadai dalam menjaga keaslian, kerahasiaan, serta keandalan data yang dikelola. Hasil pengujian kemudian dianalisis berdasarkan beberapa subkarakteristik, yaitu *confidentiality*, *integrity*, *non-repudiation*, *accountability*, *authenticity*, dan *resistance*.

g. *Maintainability*

Analisis pada karakteristik *maintainability* dilakukan untuk menilai sejauh mana Sistem Informasi Monitoring Pembesaran Kambing (SI MBEK) memiliki kemudahan dalam proses pemeliharaan dan perbaikan setelah sistem dikembangkan. Pengujian ini dilakukan menggunakan *SonarQube Cloud* untuk menganalisis kualitas kode sumber, mendeteksi potensi kesalahan, serta menilai tingkat kemudahan sistem dalam dimodifikasi atau diuji ulang. Hasil pengujian kemudian dianalisis berdasarkan beberapa subkarakteristik, yaitu *reusability*, *modifiability*, *testability*, *modularity*, dan *analysability*.

2. Dokumentasi Hasil

Semua hasil pengujian, analisis, dan rekomendasi didokumentasikan dengan baik. Dokumentasi ini menjadi referensi penting untuk pengujian perangkat lunak di masa depan, baik untuk pengembangan lebih lanjut dari sistem yang sama maupun untuk proyek lain.

Tahap *test cycle closure* memastikan bahwa seluruh proses pengujian selesai secara menyeluruh, terdokumentasi dengan baik, dan memberikan wawasan yang berharga untuk pengembangan perangkat lunak yang lebih baik. Dengan menganalisis hasil pengujian dan temuan *bug*, tim pengembang dapat memahami kelemahan sistem secara mendalam dan mengambil langkah-langkah perbaikan

yang tepat. Selain itu, laporan yang disusun pada tahap ini menjadi bukti kesesuaian perangkat lunak dengan standar kualitas ISO/IEC 25010:2023, memberikan kepercayaan kepada pengguna bahwa perangkat lunak telah melalui proses pengujian yang ketat dan memenuhi kebutuhan.

V. KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan, maka dapat diambil kesimpulan sebagai berikut:

1. Aspek kualitas perangkat lunak penting diperhatikan dalam pengembangan SI MBEK meliputi: *functional suitability*, *performance efficiency*, *compatibility*, *interaction capability*, *reliability*, *security*, dan *maintainability*. Setiap aspek memiliki peran penting dalam memastikan sistem tidak hanya berfungsi sesuai kebutuhan, tetapi juga efisien, aman, mudah digunakan, dapat diandalkan, dan mudah dipelihara.
2. Standar ISO/IEC 25010:2023 berhasil diterapkan sebagai kerangka evaluasi kualitas perangkat lunak pada SI MBEK. Evaluasi dilakukan dengan mengacu pada sub-karakteristik dari masing-masing aspek. Dengan penerapan standar ini, kualitas sistem dapat diukur secara komprehensif dan terstruktur.
3. Hasil pengujian menunjukkan bahwa SI MBEK telah memenuhi sebagian besar karakteristik kualitas perangkat lunak dengan hasil yang baik. Dari sisi fungsionalitas, hampir semua *test case* berjalan sesuai harapan dengan pengecualian minor pada integrasi pembayaran *Midtrans* yang kemudian diatasi dengan metode pembayaran manual transfer. Pada aspek kinerja, sistem terbukti efisien dengan waktu muat cepat, penggunaan sumber daya optimal, dan tampilan stabil. Pengujian kompatibilitas membuktikan sistem berjalan lancar di berbagai perangkat dan *browser*. Dari sisi pengalaman pengguna, hasil UEQ menunjukkan kategori *excellent* di seluruh skala dan mendapatkan skor UAT sebesar 89,92% (sangat baik). Pada aspek keandalan, sistem stabil hingga 400 pengguna simultan dan mampu pulih dari gangguan. Dari sisi keamanan, ditemukan risiko rendah hingga sedang yang dapat diperbaiki melalui penambahan konfigurasi header dan kebijakan CSP. Analisis

maintainability melalui *SonarQube* memberikan grade A, meskipun masih terdapat isu minor seperti duplikasi kode dan kompleksitas kognitif.

5.2 Saran

Berdasarkan hasil dari penelitian yang telah dilakukan, terdapat beberapa saran sebagai berikut.

1. Perlu dilakukan *refactoring* terhadap kode yang memiliki kompleksitas tinggi, duplikasi *hardcoded values*, serta parameter fungsi yang berlebihan. Hal ini penting untuk meningkatkan *modularity*, *reusability*, dan *testability* pada penelitian berikutnya.
2. Mengingat standar kualitas perangkat lunak bersifat dinamis, perlu dilakukan evaluasi berkala dengan ISO/IEC 25010:2023 untuk memastikan kualitas SI MBEK tetap terjaga seiring dengan perkembangan teknologi, kebutuhan pengguna, serta potensi ancaman keamanan baru.

DAFTAR PUSTAKA

- [1] D. P. Nugroho dan J. Sutopo, “Analisis dan Pengembangan Sistem Pengelolaan Peternakan Kambing Berbasis Web dengan Metode Pieces,” *Ris. dan E-J. Manaj. Inform. Komput.*, vol. 8, no. 1, 2024.
- [2] R. Wahyudi dan E. S. Astuti, “Pengaruh Kualitas Sistem, Informasi dan Pelayanan Siakad Terhadap Kepuasan Mahasiswa,” *J. Adm. Bisnis SI Univ. Brawijaya*, vol. 1, no. 1, hlm. 28–39, Apr 2013.
- [3] Sumihai, “Pengujian Sistem Informasi,” Sumihai Teknologi Indonesia. Diakses: 5 Maret 2025. [Daring]. Tersedia pada: <https://sumihai.co.id/id/pengujian-sistem-informasi/>
- [4] A. E. Kumala, R. I. Borman, dan P. Prasetyawan, “Sistem Informasi Monitoring Perkembangan Sapi di Lokasi Uji Performance (Studi Kasus: Dinas Peternakan dan Kesehatan Hewan Provinsi Lampung),” *J. TEKNOKOMPAK*, vol. 12, no. 1, 2018.
- [5] M. F. Fadlurrahman, I. Arwani, dan W. H. N. Putra, “Pengembangan Sistem Informasi Monitoring Peternakan Ayam Berbasis Mobile menggunakan React Native dan Restfull Web Service (Studi Kasus: Peternakan Alfa Sentosa),” *J-PTIHK*, vol. 5, hlm. 5196–5204, Desember 2021.
- [6] B. Naqvi, D. Kedziora, L. Zhang, dan S. Oyedeki, “Quality of Low-Code/No-Code Development Platforms Through the Lens of ISO 25010:2023,” *Computer*, vol. 58, no. 3, hlm. 30–40, 2025, doi: 10.1109/MC.2024.3493192.
- [7] M. R. Ridho, “Rancang Bangun Sistem Informasi Point of Sale dengan Framework Codeigniter pada CV Powershop,” *J. COMASIE*, vol. 04, no. 02, 2021.
- [8] E. Susanti dan T. E. Tarigan, “Penilaian Kualitas Sistem Informasi menggunakan ISO/IEC 25010 dengan Metode Profile Matching,” *Jutisi J. Ilm.*

- Tek. Inform. dan Sist. Inf.*, vol. 12, no. 1, Apr 2023, doi: 10.35889/jutisi.v12i1.1189.
- [9] A. Herliana dan P. M. Rasyid, “Sistem Informasi Monitoring Pengembangan Software pada Tahap Development berbasis Web,” *J. Inform.*, vol. 3, no. 1, 2016.
- [10] E. A. Ibrahim, R. Rajab, dan B. J. Papilaya, “Karakterisasi Fenotipik Kambing Lokal di Kecamatan Leihitu Kabupaten Maluku Tengah,” *Agrinimal J. Ilmu Ternak dan Tanam.*, vol. 10, no. 2, hlm. 86–95, Nov 2022, doi: 10.30598/ajitt.2022.10.2.86-95.
- [11] D. S. Nugroho, Suyatno, H. Hermawan, dan H. M. Naim, “Penguatan Kapasitas SDM dalam Pengelolaan Wisata Berbasis Edukasi,” *J. Abdimas Pariwisata*, vol. 5, no. 2, hlm. 33–41, Jul 2024, doi: 10.36276/jap.v5i2.624.
- [12] M. D. Mulyawan, I. N. S. Kumara, I. B. A. Swamardika, dan K. O. Saputra, “Kualitas Sistem Informasi Berdasarkan ISO/IEC 25010: Literature Review,” *Maj. Ilm. Teknol. Elektro*, vol. 20, no. 1, hlm. 15, Mar 2021, doi: 10.24843/MITE.2021.v20i01.P02.
- [13] N. W. Hidayatulloh dan P. Dellia, “Evaluasi Sistem Informasi Sekolah Terintegrasi Instagram dan WhatsApp Berdasarkan Pengujian ISO 25010,” *JSI J. Sist. Inf. E-J.*, vol. 15, no. 2, Okt 2023, doi: 10.18495/jsi.v15i2.22233.
- [14] H. S. Suparto dan R. H. Dai, “Evaluasi Kualitas Sistem Informasi Pengukuran Prestasi Kerja Berdasarkan ISO/IEC 25010,” *Jambura J. Inform.*, vol. 3, Oktober 2021, doi: 10.37905/jji.v3i2.11744.
- [15] S. S. A. Larasati, D. S. Rusdianto, dan T. A. Kurniawan, “Pembangunan Sistem Ujian Harian Siswa Berbasis Web dengan Mengacu pada Standar Kualitas ISO 25010,” *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 2, hlm. 4357–4364, 2018.
- [16] ISTQB, “Mind Maps of FL v4.0 Keywords - ISTQB Glossary,” ISTQB Glossary. Diakses: 14 Februari 2025. [Daring]. Tersedia pada: https://glossary.istqb.org/en_US/graph/mind-maps-of-fl-v40-keywords
- [17] M. Ehmer dan F. Khan, “A Comparative Study of White Box, Black Box and Grey Box Testing Techniques,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 3, no. 6, 2012, doi: 10.14569/IJACSA.2012.030603.

- [18] R. A. Sianturi, A. M. Sinaga, Y. Pratama, H. Simatupang, J. Panjaitan, dan S. Sihotang, “Perancangan Pengujian Fungsional dan Non Fungsional Aplikasi Siappara di Kabupaten Humbang Hasundutan,” *J. Komput. dan Inform.*, vol. 9, no. 2, hlm. 133–141, Sep 2021, doi: 10.35508/jicon.v9i2.4706.
- [19] H. S. Samra, “Study on Non Functional Software Testing,” *Int. J. Comput. Technol.*, vol. 4, no. 1, hlm. 151–155, Feb 2013, doi: 10.24297/ijct.v4i1c.3115.
- [20] Rajkumar, “What is Confirmation testing in Software?” Diakses: 16 Februari 2025. [Daring]. Tersedia pada: <https://www.softwaretestingmaterial.com/confirmation-testing/>
- [21] S. I Dewa Gde Satria Pramana Erlangga dan Afina Lina Nurlaili, “Pengujian Aplikasi Bangbeli pada Perusahaan PT. Doa Anak Digital menggunakan User Acceptance Test dan Regression Testing,” *SANTIKA*, vol. 3, hlm. 163–168, 2023.
- [22] Lamdatest, “What is Maintenance Testing in Software Testing?” Diakses: 16 Februari 2025. [Daring]. Tersedia pada: <https://www.lamdatest.com/learning-hub/maintenance-testing>
- [23] G. Y. Saputra dan Y. Hanum, “Comparison of Zoom and Google Meet in Online Worship with ISO 25010 Quality Model (Case Study: Churches in Tigaraksa),” *Int. Res. J. Adv. Eng. Sci.*, vol. 7, no. 2, 2022.
- [24] ISO, “ISO/IEC 25010:2023,” ISO. Diakses: 11 Maret 2025. [Daring]. Tersedia pada: <https://www.iso.org/standard/78176.html>
- [25] M. D. Mulyawan, I. B. A. Swamardika, dan K. O. Saputra, “Analisis Kesesuaian Fungsional dan Usability pada Sistem Informasi Karma Simanis Berdasarkan ISO/IEC 25010,” *J. Teknol. dan Sist. Inf.*, vol. VII, no. 3, hlm. 293–302, 2021, doi: <https://doi.org/10.33330/jurteksi.v7i3.1139>.
- [26] ISO/IEC 25010:2023, “Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Product quality model,” *Int. Organ. Stand. Geneva Switz.*, 2023.
- [27] B. Huda dan A. L. Hananto, “Penerapan Software Testing Life Cycle pada Pengujian Otomatisasi Platform Dzikra,” *CSRID J.*, vol. 15, no. 1, 2023.

- [28] M. Mutiara, N. Qomariyah, dan D. K. Widyawati, “Automated Testing of Indo Towing Information System with Katalon Studio,” *ROUTERS J. Sist. dan Teknol. Inf.*, hlm. 35–41, Okt 2023, doi: 10.25181/rt.v2i1.3277.
- [29] I. R. Dhaifullah, M. Muttanifudin H, A. Ananda Salsabila, dan M. Ainul Yaqin, “Survei Teknik Pengujian Software,” *J. Autom. Comput. Inf. Syst.*, vol. 2, no. 1, hlm. 31–38, Jun 2022, doi: 10.47134/jacis.v2i1.42.
- [30] A. Arfan dan Hendrik, “Penerapan STLC dalam Pengujian Automation Aplikasi Mobile (Studi kasus: LMS Amikom Center PT.GIT Solution),” *J. UII*, vol. 3, no. 2, 2022.
- [31] H. Bhasin, E. Khanna, dan S. Sudha, “Black Box Testing Based on Requirement Analysis and Design Specifications,” *Int. J. Comput. Appl.*, vol. 87, no. 18, hlm. 36–40, Feb 2014, doi: 10.5120/15311-4024.
- [32] F. Kristianto, S. Rahman, dan S. Bahri, “Analisis Kerentanan pada Website Servio Menggunakan Acunetix Web Vulnerability,” *JTRISTE*, vol. 9, no. 1, hlm. 46–55, Mar 2022, doi: 10.55645/jtriste.v9i1.363.
- [33] K. G. Ayu, D. W. Sari, A. Zein, dan R. S. Fadila, “Pengujian Black-Box dan Analisa Kualitas ISO 25010 pada Aplikasi HESTI,” *CESS J. Comput. Eng. Syst. Sci.*, vol. 9, no. 2, hlm. 710, Jul 2024, doi: 10.24114/cess.v9i2.61621.
- [34] F. Ardi dan H. P. Putro, “Pengujian Black Box Aplikasi Mobile Menggunakan Katalon Studio,” *J. UII*, vol. 2, no. 1, 2021.
- [35] Katalon, “About Us | Katalon Software Quality Management Platform,” [katalon.com](https://katalon.com/about-us). Diakses: 12 Maret 2025. [Daring]. Tersedia pada: <https://katalon.com/about-us>
- [36] Apache JMeter, “Apache JMeter™,” About Apache JMeter. Diakses: 31 Januari 2025. [Daring]. Tersedia pada: <https://jmeter.apache.org/>
- [37] E. H. Halili, *Apache JMeter: A Practical Beginner’s Guide to Automated Testing and Performance Measurement for Your Websites*, 1st ed. dalam From Technologies to Solutions Series. Birmingham, U.K.: Packt Publishing, 2008.
- [38] I. Indrianto, “Performance Testing on Web Information System Using Apache Jmeter and Blazemeter,” *J. Ilm. Ilmu Terap. Univ. Jambi*, vol. 7, no. 2, hlm. 138–149, Des 2023, doi: 10.22437/jiituj.v7i2.28440.

- [39] BlazeMeter, “Ekstensi Chrome BlazeMeter: Merekam JMeter, Selenium, atau JMeter dan Selenium yang Tersinkronisasi.” Diakses: 11 September 2025. [Daring]. Tersedia pada: <https://help.blazemeter.com/docs/guide/recorders-blazemeter-chrome-extension.html>
- [40] M. Kaushik dan P. Fageria, “Performance Testing Tools: A Comparative Study,” *IJISET - Int. J. Innov. Sci. Eng. Technol.*, vol. 1, no. 4.
- [41] WAPT, “Harga Lisensi Alat Uji Beban WAPT.” Diakses: 11 September 2025. [Daring]. Tersedia pada: <https://www.loadtestingtool.com/order.shtml>
- [42] GTmetrix, “About Us,” About GTmetrix. Diakses: 1 Agustus 2025. [Daring]. Tersedia pada: <https://gtmetrix.com/about.html>
- [43] Y. Calleo dan F. Pilla, “Optimizing spatial survey administration adopting RT-GSCS: A statistical perspective on performance metrics,” *MethodsX*, vol. 12, hlm. 102578, 2024, doi: <https://doi.org/10.1016/j.mex.2024.102578>.
- [44] PowerMapper, “PowerMapper Visual Site Mapping,” PowerMapper Product. Diakses: 1 Agustus 2025. [Daring]. Tersedia pada: <https://www.powermapper.com/products/mapper/>
- [45] I. Hakim dan H. H, “Quality Analysis of SMP Islam Dian Didaktika CBT Application System Based on ISO 25010 Standard,” *Int. J. Comput. Appl.*, vol. 186, no. 44, hlm. 23–28, Okt 2024, doi: [10.5120/ijca2024924066](https://doi.org/10.5120/ijca2024924066).
- [46] Irum Naz Sodhar dan Hina Bhanbhro, “Evaluation of Web Accessibility of Engineering University,” *IJCSNS Int. J. Comput. Sci. Netw. Secur.*, vol. 19, no. 12, 2019, doi: [10.13140/RG.2.2.12124.54403](https://doi.org/10.13140/RG.2.2.12124.54403).
- [47] UEQ, “User Experience Questionnaire,” User Experience Questionnaire. Diakses: 16 Maret 2025. [Daring]. Tersedia pada: <https://www.ueq-online.org>
- [48] A. Pratama, A. Farqi, dan E. P. Mandyartha, “Evaluation of User Experience in Integrated Learning Information Systems Using User Experience Questionnaire (UEQ),” *J. Inf. Syst. Inform.*, vol. 4, no. 4, hlm. 1019–1029, Nov 2022, doi: [10.51519/journalisi.v4i4.394](https://doi.org/10.51519/journalisi.v4i4.394).
- [49] C. K. N. Che Ku Mohd dan F. Shahbodin, “Personalized Learning Environment: Alpha Testing, Beta Testing & User Acceptance Test,” *Procedia - Soc. Behav. Sci.*, vol. 195, hlm. 837–843, Jul 2015, doi: [10.1016/j.sbspro.2015.06.319](https://doi.org/10.1016/j.sbspro.2015.06.319).

- [50] N. A. Vanesha, R. Rizky, dan A. Purwanto, "Comparison Between Usability and User Acceptance Testing on Educational Game Assessment," *J. Sisfokom Sist. Inf. dan Komput.*, vol. 13, no. 2, hlm. 210–215, Jun 2024, doi: 10.32736/sisfokom.v13i2.2099.
- [51] N. Pirsu dan Sumijan, "Meningkatkan Keamanan Sistem Informasi Puskesmas Terpadu dengan Metode Grey-Box Penetration Test Menggunakan Computer Assisted Audit Techniques," *J. Inf. dan Teknol.*, Sep 2020, doi: 10.37034/jidt.v2i4.79.
- [52] Aristotel Aaron Agpaoa, "Development and Employment of Cyber Security of e201 File Web Application for Data Center College of the Philippines of Laoag City, Inc.," *J. Electr. Syst.*, vol. 20, no. 5s, hlm. 677–683, Apr 2024, doi: 10.52783/jes.2289.
- [53] Team, "Software Maintainability: Software Maintainability Enhanced with Clean Code," Sonar Qube. Diakses: 31 Januari 2025. [Daring]. Tersedia pada: <https://www.sonarsource.com/solutions/maintainability/>
- [54] M. Fadhli, Yuli Fitriisa, dan D. Nurmallasari, "Pengujian Kualitas Coding Pada Aplikasi Bank Sampah DLHK Kota Pekanbaru Menggunakan Code Smell Tools," *J. Komput. Terap.*, vol. 10, no. 1, hlm. 86–97, Jul 2024, doi: 10.35143/jkt.v10i1.6211.
- [55] S. Nocera, S. Romano, R. Francese, dan G. Scanniello, "Software engineering education: Results from a training intervention based on SonarCloud when developing web apps," *J. Syst. Softw.*, vol. 222, hlm. 112308, Apr 2025, doi: 10.1016/j.jss.2024.112308.
- [56] SonarQube, "Billing and upgrade - SonarQube Cloud." Diakses: 11 September 2025. [Daring]. Tersedia pada: <https://sonarcloud.io/organizations/rizkipgt/billing/upgrade>
- [57] AWS, "Apa itu VPS? - Server Privat Virtual - AWS," Amazon Web Services, Inc. Diakses: 18 September 2025. [Daring]. Tersedia pada: <https://aws.amazon.com/id/what-is/vps/>
- [58] A. Gusti, "Resource VPS untuk Kinerja Optimal: RAM, CPU, Disk Space!," IDwebhost. Diakses: 18 September 2025. [Daring]. Tersedia pada: <https://idwebhost.com/blog/resource-vps-untuk-kinerja-optimal/>

- [59] M. K. Abdillah, “Analisis Kualitas Website xyz.com menggunakan Model ISO/IEC 25010 Product Quality,” *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 8, hlm. 41–50, Jan 2024, doi: 10.35889/jutisi.v10i2.661.
- [60] S. Y. Erlangga, P. H. Winingsih, R. Yektyastuti, E. A. Poort, dan A. N. Amrurrahim, “Enhancing Statistical Physics Learning: Rhetoric and Emotion in Moodle for Physics Education,” *J. Pendidik. Fis. Indones.*, vol. 19, no. 2, hlm. 163–179, Des 2023, doi: 10.15294/jpfi.v19i2.46090.
- [61] A. Wahid, “Perancangan Sistem Informasi Berbasis Web untuk Objek Pariwisata di Kabupaten Pinrang,” *PISCES*, vol. 02, no. 1, hlm. 25–34, 2024.
- [62] M. S. Lamada, A. S. Miru, dan R.- Amalia, “Pengujian Aplikasi Sistem Monitoring Perkuliahan Menggunakan Standar ISO 25010,” *J. Mediat.*, vol. 3, no. 3, Sep 2020, doi: 10.26858/jmtik.v3i3.15172.
- [63] D. D. Andayani dan G. D. Dirawan, “Pengembangan sistem informasi publikasi ilmiah menggunakan Model System Development Life Cycle berbasis dynamic web,” *Pros. Semin. Nas. Lemb. Penelit. Univ. Negeri Makassar*, 2015.
- [64] Cahaya Putri Agustika, W. SJ Saputra, dan M. Idhom, “Pengujian Aplikasi Greenwallet dengan Metode Load Testing dan Apache Jmeter,” *J. Inform. dan Sist. Inf.*, vol. 2, no. 2, hlm. 190–195, Jul 2021, doi: 10.33005/jifosi.v2i2.357.