

**PERANCANGAN *FRONT-END* DASHBOARD MONITORING LAYANAN
PERPUSTAKAAN BERBASIS WEBSITE MENGGUNAKAN METODE
*AGILE KANBAN***

(STUDI KASUS: UPA PERPUSTAKAAN UNIVERSITAS LAMPUNG)

(Skripsi)

Oleh

MELINDA SARI SUMADYO PUTRI

2115061016



JURUSAN TEKNIK ELEKTRO

FAKULTAS TEKNIK

UNIVERSITAS LAMPUNG

2025

**PERANCANGAN FRONT-END DASHBOARD MONITORING LAYANAN
PERPUSTAKAAN BERBASIS WEBSITE MENGGUNAKAN METODE
AGILE KANBAN
(STUDI KASUS: UPA PERPUSTAKAAN UNIVERSITAS LAMPUNG)**

Oleh

MELINDA SARI SUMADYO PUTRI

Skripsi

**Sebagai Salah Satu Syarat 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

PERANCANGAN FRONT-END DASHBOARD MONITORING LAYANAN PERPUSTAKAAN BERBASIS WEBSITE MENGGUNAKAN METODE *AGILE KANBAN* (STUDI KASUS: UPA PERPUSTAKAAN UNIVERSITAS LAMPUNG)

Oleh

MELINDA SARI SUMADYO PUTRI

UPA Perpustakaan Universitas Lampung membutuhkan sistem penyajian data pengunjung dan peminjaman buku yang cepat, akurat, dan real-time untuk mendukung pelaporan serta pengambilan keputusan. Proses rekapitulasi manual yang digunakan saat ini memperlambat pekerjaan dan berisiko terjadi kesalahan. Penelitian ini bertujuan memberikan solusi atas permasalahan tersebut melalui perancangan dan pengembangan *front-end* dashboard monitoring layanan perpustakaan berbasis web. Dashboard ini menyajikan 11 layanan: 2 layanan utama (monitoring data pengunjung dan peminjaman buku), 1 layanan monitoring jumlah koleksi, serta 8 layanan informasi pada *landing page* (OPAC, E-Journal, E-Book, Verifikasi Karya Ilmiah, Pendaftaran Anggota, Tanya Pustakawan, SKBP, dan DIGILIB). Pengembangan menggunakan metode Agile Kanban dengan *framework* Laravel (Blade) dan perancangan antarmuka melalui Figma. Pengujian *black-box* menunjukkan 13 fitur utama dengan 19 *test case* berhasil 94,74% pada iterasi pertama dan meningkat menjadi 100% pada iterasi kedua. Pengujian *System Usability Scale* (SUS) terhadap 30 responden menghasilkan skor 77,42 (kategori Acceptable, grade B–Excellent), menunjukkan antarmuka mudah digunakan serta meningkatkan efisiensi pemantauan layanan dan pelaporan data secara real-time.

Kata kunci: Dashboard monitoring, Layanan perpustakaan, Figma, Laravel, Agile Kanban, *Black-box testing*, System Usability Scale.

ABSTRAK

PERANCANGAN FRONT-END DASHBOARD MONITORING LAYANAN PERPUSTAKAAN BERBASIS WEBSITE MENGGUNAKAN METODE AGILE KANBAN (STUDI KASUS: UPA PERPUSTAKAAN UNIVERSITAS LAMPUNG)

Oleh

MELINDA SARI SUMADYO PUTRI

The Library Technical Implementation Unit (UPA) of Universitas Lampung requires a system for presenting visitor and book borrowing data that is fast, accurate, and real-time to support reporting and decision-making. The current manual recapitulation process slows down work and is prone to errors. This study aims to provide a solution to these problems through the design and development of a web-based front-end dashboard for monitoring library services. The dashboard presents 11 services: 2 main services (monitoring visitor data and book borrowing), 1 service for monitoring the number of collections, and 8 informational services on the landing page (OPAC, E-Journal, E-Book, Scientific Work Verification, Member Registration, Ask a Librarian, SKBP, and DIGILIB). The development applied the Agile Kanban method using the Laravel (Blade) framework and interface design with Figma. Black-box testing results on 13 main features with 19 test cases achieved a 94.74% success rate in the first iteration and 100% in the second iteration. The System Usability Scale (SUS) testing involving 30 respondents produced a score of 77.42 (Acceptable category, grade B–Excellent), indicating that the interface is easy to use and improves the efficiency of service monitoring and real-time data reporting.

Keywords: *Dashboard monitoring, Library services, Figma, Laravel, Agile Kanban, Black-box testing, System Usability Scale.*

Judul Skripsi : PERANCANGAN FRONT-END DASHBOARD
MONITORING LAYANAN PERPUSTAKAAN
BERBASIS WEBSITE MENGGUNAKAN
METODE
AGILE KANBAN
(STUDI KASUS: UPA PERPUSTAKAAN
UNIVERSITAS LAMPUNG)

Nama Mahasiswa : Melinda Sari Sumadyo Putri

Nomor Pokok Mahasiswa : 2115061016

Program Studi : Teknik Informatika

Jurusan : Teknik Elektro

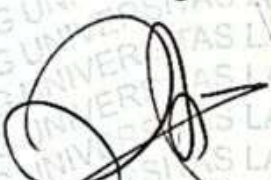
Fakultas : Teknik

MENYETUJUI

1. Komisi Pembimbing

Pembimbing Utama

Pembimbing Pendamping


Puput Budi Wintoro, S.Kom., M.T.I.

NIP. 198410312019031004


Deny Budiyanto, S.Kom, M.T.

NIP 199112082019031011

2. Mengetahui

Ketua Jurusan

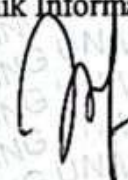
Teknik Elektro


Herlinawati, S.T., M.T.

NIP 197103141999032001

Ketua Program Studi

Teknik Informatika


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

NIP 197312262000122001

MENGESAHKAN

1. Tim Penguji

Ketua : Puput Budi Wintoro, S.Kom., M.T.I.



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: 11 Agustus 2025

SURAT PERNYATAAN

Dengan ini saya menyanggah bahwa skripsi berjudul “Perancangan Front-End Dashboard Monitoring Layanan Perpustakaan Berbasis Webiste Menggunakan Metode Agile Kanban” sepenuhnya merupakan hasil karya saya sendiri. Apabila di kemudian hari terbukti pernyataan ini tidak benar, saya siap menerima sanksi sesuai dengan ketentuan hukum yang berlaku.

Bandar Lampung, 11 Agustus 2025

Penulis,



Melinda Sari Sumadyo Putri

NPM. 2115061016

RIWAYAT HIDUP



Penulis dilahirkan di Lampung pada tanggal 22 Oktober 2002 sebagai anak sulung dari dua bersaudara, dari pasangan Bapak Sumadyo dan Ibu Riati. Penulis menyelesaikan jenjang pendidikan formal di SDN 1 Sumberhadi Lampung Timur pada tahun 2015, kemudian melanjutkan ke SMPN 1 Melinting dan lulus pada tahun 2018, serta menamatkan pendidikan menengah atas di SMAN 1 Bandar Sribhawono pada tahun 2021. Pada tahun 2021, penulis diterima sebagai mahasiswa Program Studi Teknik Informatika Universitas Lampung melalui jalur seleksi SNPMTN. Selama masa perkuliahan, penulis aktif berpartisipasi dalam berbagai kegiatan, antara lain:

1. Melakukan Praktik Kerja Lapangan (PKL) di PT Telkom Witel Lampung bidang marketing pada tahun 2024.
2. Mengikuti program Studi Independen Mitra RevoU Tech Academy bidang Data & Software Engineering Batch 5 pada tahun 2023.
3. Berperan aktif sebagai anggota Himpunan Mahasiswa Teknik Elektro Universitas Lampung pada masa kepengurusan 2022–2023.
4. Berperan sebagai Asisten Laboratorium Teknik Komputer tahun akademik 2024/2025.

MOTTO

Sebenarnya tidak ada yang perlu dikhawatirkan, Allah memang tidak menjanjikan hidupmu selalu mudah, namun dua kali Allah menegaskan dalam firman-Nya:

“Maka sesungguhnya bersama kesulitan ada kemudahan”

“Sesungguhnya beserta kesulitan ada kemudahan”

(QS. Al-Insyirah: 5–6)

“Barangkali sesuatu ditunda karena hendak disempurnakan, dibatalkan karena hendak diganti yang utama, ditolak karena dinanti yang lebih baik”

(Salim A fillah)

Semua jatuh bangunmu hal yang biasa, angan dan pertanyaan waktu yang menjawabnya, berikan tenggat waktu bersedihlah secukupnya, rayakan perasaanmu sebagai manusia

(Baskara Putra-Hindia)

“Tbuku menaruhkan nyawanya demi kelahiranku, maka tidak mungkin jika kehadiranku tidak ada artinya”

(Someone)

PERSEMBAHAN

Segala puji syukur penulis panjatkan ke hadirat Allah SWT atas limpahan rahmat, taufik, dan hidayah-Nya, sehingga penelitian sekaligus penyusunan skripsi ini dapat diselesaikan dengan baik.

Kupersembahkan skripsi ini kepada:

Kedua Orang Tua

“Kepada ayah dan ibu tercinta, yang dengan doa dan kasihnya senantiasa menjadi penopang, serta dengan segala dukungan dan pengorbanannya mengiringi perjalanan penulis menyelesaikan skripsi ini”

SANWACANA

Dengan penuh rasa syukur, penulis menghaturkan terima kasih kepada Allah SWT atas limpahan rahmat dan karunia-Nya sehingga skripsi berjudul *“Perancangan Frontend Dashboard Monitoring Layanan Perpustakaan Berbasis Website Menggunakan Metode Agile Kanban”* dapat diselesaikan dengan baik. Dalam penyusunannya, penulis memperoleh dukungan, arahan, serta bantuan baik moril maupun materil dari berbagai pihak. Atas hal tersebut, penulis dengan tulus menyampaikan penghargaan dan terima kasih yang sebesar-besarnya kepada:

1. Teristimewa ayahanda Sumadyo tersayang, pahlawan tanpa tanda jasa yang diam-diam menguatkan dengan kerja keras dan pengorbananmu. Setiap keringatmu adalah doa dalam wujud nyata, setiap nasihatmu adalah bekal berharga dalam perjalanan hidup ini. Terima kasih, Ayah, telah menjadi panutan yang meneguhkan langkah penulis hingga dapat menyelesaikan perjuangan ini;
2. Gerbang surga ku ibu Riati tersayang, sosok yang selalu menjadi pelabuhan hati dan tempat pulang paling indah. Dari doa-doa yang kau panjatkan di setiap malam, penulis menemukan kekuatan. Dari pelukanmu yang hangat, penulis belajar arti kasih sayang tanpa syarat. Terima kasih, Ibu, telah menuntun setiap langkah dengan doa yang tak pernah terputus, hingga penulis mampu meraih gelar sarjana;
3. Saudara kandung laki-laki Rasyid, yang senantiasa memberikan kasih sayang, perhatian, serta menghadirkan keceriaan dan tawa yang menjadi motivasi bagi penulis untuk terus maju. Terima kasih telah menjadi adik yang penuh kasih, semoga keberhasilan ini dapat menjadi kebanggaan sekaligus inspirasi untukmu;

4. Bapak Dr. Eng. Helmy Fitriawan, S.T., M.Sc., selaku Dekan Fakultas Teknik Universitas Lampung;
5. Ibu Herlinawati, S.T., M.T., selaku Ketua Jurusan Teknik Elektro Universitas Lampung;
6. Ibu Yessi Mulyani, S.T., M.T. selaku Ketua Program Studi Teknik Informatika Universitas Lampung yang telah membantu proses kelancaran pengerjaan penelitian;
7. Bapak Puput Budi Wintoro, S.Kom., M.T.I. selaku Pembimbing Utama, yang dengan penuh kesabaran telah memberikan arahan, bimbingan, serta masukan berharga kepada penulis dalam proses penelitian dan penyusunan skripsi ini;
8. Bapak Deny Budiyanto, S.Kom, M.T. selaku Pembimbing Pendamping atas segala bimbingan, saran, dan dukungan yang sangat membantu penulis dalam menyelesaikan penelitian maupun penulisan skripsi;
9. Bapak Ir. Gigih Forda Nama, S.T., M.T.I., IPM selaku Penguji, atas segala arahan dan masukan yang penuh makna, sehingga penelitian ini dapat terselesaikan dengan lebih baik;
10. Bapak Bayzoni, S.T., M.T. selaku Kepala UPT Perpustakaan Universitas Lampung, yang telah memberikan izin penelitian serta arahan dan saran berharga sehingga penelitian ini dapat terlaksana dengan baik.
11. Ibu Angelino Vinanti Sonjaya, S.I.Pust. beserta jajaran pustakawan UPA Perpustakaan Universitas Lampung yang telah memberikan kesempatan kepada penulis untuk melakukan penelitian, memberikan arahan dan saran dengan penuh kesabaran, serta bersedia meluangkan waktu untuk menjadi penguji sistem;
12. Seluruh Dosen Program Studi Teknik Informatika Universitas Lampung yang telah berperan besar dalam memberikan ilmu dan wawasan kepada penulis;
13. Rekan penelitian sekaligus sahabat penulis, Marselinus Heris Adyatma, yang selalu menunjukkan kerja sama dengan penuh kesabaran serta saling menguatkan sepanjang proses penyelesaian proyek ini, hingga meraih gelar bersama.

14. Sumber ceria penulis, sahabat Pulu-Pulu, Tami, Panca, Marsel, yang selalu kebersamai penulis dengan tawa, dan canda. Terimakasih telah menjadikan perjalanan ini lebih ringan, bermakna, dan penuh warna;
15. Sumber semangat penulis, Panca Rulian Effendi, yang sangat berarti kehadirannya, terimakasih senantiasa mendampingi dengan memberikan dukungan, ketenangan, dan bantuan dalam berbagai hal dan keadaan, sekaligus menjadi penguat di kala terpuruk serta menemani hari-hari di perantauan;
16. Keluarga besar kakung karji, om Muslim, tante Rusmiatun, kakung, mbah putri, Oos, Davis, dan Gogo, terimakasih selalu memberikan bantuan dan motivasi.
17. Asisten Laboratorium Komputer 21 yang selalu berbagi cerita, ilmu, dan wejangan, serta tingkah kocaknya yang tak pernah gagal mencairkan suasana. Terima kasih atas tawa dan kebersamaan yang membuat proses belajar ini semakin berkesan.
18. Teman-teman baik yang menjadi kawan sekaligus rekan bermain selama masa perkuliahan diperantauan, terimakasih atas kebersamaan yang telah diberikan;
19. Terakhir Terima kasih untuk diri sendiri yang telah sabar menapaki jalan panjang penuh tantangan, tetap tegar di bawah tekanan, dengan segala keterbatasan tetap mampu berproses, belajar, dan memilih bertahan hingga titik akhir meraih pencapaian, berbahagialah, semoga Allah selalu melindungi mu, dan memberikan keberkahan dalam hidup.

Harapan penulis, hasil penelitian ini dapat dijadikan bahan pertimbangan dalam memperkaya kajian Teknik Informatika serta memberi manfaat bagi pembaca.

Bandar Lampung, 11 Agustus 2025

Penulis,

Melinda Sari Sumadyo Putri

NPM . 2115061016

DAFTAR ISI

	Halaman
ABSTRAK	i
PERSEMBAHAN.....	viii
SANWACANA	ix
DAFTAR ISI.....	xii
DAFTAR GAMBAR.....	xv
DAFTAR TABEL	xviii
I. PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Perumusan Masalah.....	6
1.3 Tujuan Penelitian.....	6
1.4 Manfaat Penelitian.....	7
1.5 Batasan Masalah.....	7
1.6 Sistematika Penulisan Skripsi / Tugas Akhir	8
II. TINJAUAN PUSTAKA.....	10
2.1 Perpustakaan.....	10
2.2 Struktur Perpustakaan Universitas Lampung	11
2.3 Dashboard.....	16
2.4 Monitoring Data	17
2.5 <i>Front-end</i>	18
2.6 Metode Agile Development	19
2.7 Kanban.....	20

2.8	Framework Laravel	24
2.9	Figma.....	25
2.10	Plugin Anima	26
2.11	Bootstrap.....	27
2.12	Trello.....	27
2.13	Electronic Library (eLib).....	28
2.14	Integrated Library System Lite (INLISLite).....	28
2.15	PHP	29
2.16	MySQL	29
2.17	Laragon	30
2.18	Gitlab	30
2.19	Desain Antarmuka Pengguna	30
2.20	Kebutuhan Sistem.....	31
2.21	Diagram UML (Unified Modeling Language)	32
2.22	<i>Black-box testing</i>	35
2.23	<i>System Usability Scale (SUS)</i>	35
2.24	Penelitian Terdahulu	36
III. METODOLOGI PENELITIAN		49
3.1	Waktu dan Tempat	49
3.2	Jadwal Penelitian.....	49
3.3	Alat dan Bahan Penelitian	50
3.4	Tahapan Penelitian	51
3.4.1	<i>Requirement Gathering</i>	52
3.4.2	Pembuatan <i>User story</i>	53
3.4.3	Penyusunan <i>Backlog</i>	53
3.4.4	Desain dan Perancangan Sistem.....	54

3.4.5 <i>Development</i>	55
3.4.6 Pengujian Sistem.....	56
3.4.7 <i>Deployment</i>	60
IV PEMBAHASAN.....	62
4.1 Requirement Gathering	62
4.1.1 Kebutuhan Sistem	66
4.2 Pembuatan User Story	67
4.3 Penyusunan Backlog	69
4.4 Perancangan Sistem.....	70
4.4.1 Use Case Diagram.....	70
4.4.2 Activity Diagram.....	71
4.4.3 Sequence Diagram	80
4.4.4 Desain Antarmuka.....	88
4.5 Development.....	93
4.5.1 Tampilan Halaman Antarmuka	93
4.5.2 Implementasi Antarmuka ke dalam Sistem Laravel	108
4.5.3 Papan Kanban.....	125
4.6 Pengujian Sistem	131
4.6.1 Pengujian Blackbox Testing	131
4.6.2 Pengujian Usability dengan SUS (<i>System Usability Scale</i>)	144
V KESIMPULAN DAN SARAN	157
5.1 Kesimpulan.....	157
5.2 Saran.....	158
LAMPIRAN.....	159
DAFTAR PUSTAKA.....	160

DAFTAR GAMBAR

	Halaman
Gambar 2.1 Struktur Organisasi UPA Perpustakaan Universitas Lampung.....	12
Gambar 2.2 Kanban Board.....	21
Gambar 2.3 Skor System Usability Scale (SUS)[65].....	36
Gambar 3.1 Tahapan Penelitian.....	51
Gambar 4.1 Use Case Diagram Sistem.....	70
Gambar 4.2 Activity Diagram Login	71
Gambar 4.3 Activity Diagram Lupa Password	73
Gambar 4.4 Activity Diagram Dashboard Monitoring	74
Gambar 4.5 Activity Diagram Pengunjung.....	75
Gambar 4.6 Activity Diagram Peminjaman Buku	76
Gambar 4.7 Activity Diagram Manajemen Data Akun..	77
Gambar 4.8 Activity Diagram Jumlah Koleksi.....	78
Gambar 4. 9 Activity Diagram Kelola Gambar Carousel Landing Page	79
Gambar 4.10 Sequence Diagram Login.....	80
Gambar 4.11 Sequence Diagram Lupa Password.....	81
Gambar 4.12 Gambar Sequence Monitoring Dashboard.	82
Gambar 4.13 Sequence Diagram Monitoring Data Pengunjung.....	83
Gambar 4.14 Sequence Diagram Monitoring Data Peminjaman Buku.	84
Gambar 4.15 Sequence Diagram Manajemen Data Akun.	85
Gambar 4.16 Sequence Diagram Manajemen Jumlah Koleksi.....	86
Gambar 4.17 Diagram Kelola Gambar Carousel Landing Page	87
Gambar 4.18 Tampilan Desain Login.....	88
Gambar 4.19 Tampilan Desain Lupa Password.....	88
Gambar 4.20 Tampilan Reset Password	89

Gambar 4.21 Desain Tampilan Landing Page	89
Gambar 4.22 Desain Tampilan Dashboard Monitoring.....	90
Gambar 4.23 Desain Tampilan Data Pengunjung.....	90
Gambar 4.24 Desain Tampilan Data Peminjaman Buku	91
Gambar 4.25 Desain Tampilan Data Akun	91
Gambar 4.26 Desain Tampilan Jumlah Koleksi	92
Gambar 4.27 Desain Tampilan Form Edit Jumlah Koleksi	92
Gambar 4.28 Tampilan Halaman Landing Page	93
Gambar 4.29 Tampilan Pop- Up E-Book.....	93
Gambar 4.30 Tampilan Halaman Login.....	94
Gambar 4.31 Tampilan Halaman Lupa Password.....	95
Gambar 4.32 Tampilan Halaman Reset Password	95
Gambar 4.33 Tampilan Halaman Sidebar	96
Gambar 4.34 Tampilan Halaman Dashboard Monitoring.....	97
Gambar 4.35 Tampilan Halaman Data Pengunjung	98
Gambar 4.36 Tampilan Fitur Filter dan Pencarian pada Data Pengunjung	99
Gambar 4.37 Tampilan Fitur Download PDF Data Pengunjung	99
Gambar 4.38 Tampilan Halaman Data Peminjaman Buku.....	100
Gambar 4. 39 Tampilan Fitur dan Filter dan Pencarian pada Data Peminjaman Buku	101
Gambar 4.40 Tampilan Fitur Download PDF Data Peminjaman Buku.....	102
Gambar 4.41 Tampilan Halaman Data Akun.....	103
Gambar 4.42 Tampilan Halaman Tambah Akun	104
Gambar 4.43 Tampilan Halaman Jumlah Koleksi	105
Gambar 4.44 Tampilan Halaman Form Edit Jumlah Kolesi.....	105
Gambar 4.45 Tampilan Halaman Form Edit Jumlah Kolesi.....	106
Gambar 4.46 Tampilan Fitur Pagination.....	107
Gambar 4.47 Tampilan Fitur Breadcrumb	107
Gambar 4.48 Plugin Anima pada Figma.....	108
Gambar 4.49 Implementasi Halaman Login	109
Gambar 4.50 Kode Blade POST	110
Gambar 4.51 Kode Bootstrap Toast.....	111

Gambar 4.52 Implementasi Halaman Dashboard Utama.....	112
Gambar 4.53 Kode Blade Fitur Filter Dashboard Utama.	113
Gambar 4.54 Kode Blade Tampilan Card Jumlah Data pada Dshboard Utama.	113
Gambar 4.55 Kode Blade Grafik Chart.js.....	114
Gambar 4.56 Kode Blade Menampilkan Grafik	114
Gambar 4.57 Implementasi Halaman Data Pengunjung.....	115
Gambar 4.58 Kode Program Blade Layout Utama Data Pengunjung	116
Gambar 4.59 Kode Blade fitur Filter Data Pengunjung Request ke Backend	116
Gambar 4.60 Kode Program Menampilkan Halaman Data Pengunjung.	117
Gambar 4.61 Kode Blade Filter Modal Bootsrap 5 Data Pengunjung.....	117
Gambar 4.62 Kode Blade Bootsrap Total Jumlah Pengunjung.	118
Gambar 4.63 Kode Blade Bootsrap Unduh Data Pengunjung.....	118
Gambar 4.64 Kode Blade Unduh PDF Data Pengunjung.....	118
Gambar 4.65 Kode Blade Unduh Excel Data Pengunjung	119
Gambar 4.66 Implementasi Halaman Data Pengunjung.....	120
Gambar 4.67 Kode Program Blade Layout Utama Data Peminjaman Buku.....	120
Gambar 4.68 Kode Blade Fitur Filter Data Peminjaman Request ke Backend. .	120
Gambar 4.69 Kode Blade Menampilkan Halaman Data Peminjaman Buku.	121
Gambar 4.70 Kode Blade Unduh PDF Data Peminjaman Buku.	121
Gambar 4.71 Kode Blade Unduh Excel Data Peminjaman Buku.....	122
Gambar 4.72 View Blade pada Laravel	122
Gambar 4.73 Papan Kanban.....	125
Gambar 4.74 Grafik Durasi Pengerjaan Frontend Webiste Perpustakaan	130
Gambar 4.75 Representasi Responden <i>System Usability Scale</i>	147
Gambar 1 Wawancara Bersama Pustakawan Perpustakaan Universitas Lampung.....	159

DAFTAR TABEL

	Halaman
Tabel 2.1 Penelitian Terdahulu	43
Tabel 3.1 Jadwal Penelitian.....	49
Tabel 3. 2 Alat dan Bahan pengembangan sistem	50
Tabel 3.3 Pertanyaan <i>System Usability Scale</i> (SUS)	57
Tabel 3.4 Jawaban dan Skor Metode SUS	57
Tabel 3. 5 Acceptable Ranges	59
Tabel 3. 6 Skor SUS.....	60
Tabel 4.1 Hasil Wawancara.....	62
Tabel 4.2 Kebutuhan Fungsional	66
Tabel 4.3 Kebutuhan Non-Fungsional	67
Tabel 4.4 User Story	67
Tabel 4.5 Backlog	69
Tabel 4.6 View Blade Pada Laravel.....	123
Tabel 4.7 Durasi Fronend Task Card Kanban.....	127
Tabel 4.8 Pengujian Login ke Halaman Dashboard.....	131
Tabel 4.9 Pengujian Tidak Berhasil Login ke Halaman Dashboard.....	132
Tabel 4.10 Pengujian Lupa Password iterasi 1	132
Tabel 4.11 Pengujian Logout Dashboard.....	133
Tabel 4.12 Persentase Keberhasilan Pengujian Fitur Login iterasi 1.....	133
Tabel 4.13 Pengujian Lupa Password iterasi 2	133
Tabel 4.14 Persentase Keberhasilan Pengujian Fitur Login iterasi 2.....	134
Tabel 4.15 Pengujian Fitur Dashboard Monitoring	134
Tabel 4.16 Persentase Keberhasilan Pengujian Fitur Dashboard Monitoring	135
Tabel 4.17 Pengujian Fitur Filter Data.....	135

Tabel 4.18 Persentase Keberhasilan Pengujian Fitur Filter Data.....	135
Tabel 4.19 Pengujian Fitur Search Data Pengunjung	136
Tabel 4.20 Pengujian Fitur Search Data Pengunjung Tidak Sesuai Kata Kunci	136
Tabel 4.21 Pengujian Fitur Search Data Peminjaman Buku.....	137
Tabel 4.22 Pengujian Fitur Search Data Peminjaman Buku Tidak Sesuai Kata Kunci	137
Tabel 4.23 Persentase Keberhasilan Pengujian Fitur Search Data Pengunjung & Peminjaman Buku	137

I. PENDAHULUAN

1.1 Latar Belakang

Perpustakaan memiliki posisi strategis dalam menunjang penyelenggaraan pendidikan tinggi, termasuk di Universitas Lampung. Sebagai pusat informasi akademik, perpustakaan menyediakan berbagai sumber literatur seperti buku referensi, jurnal ilmiah, dan karya tulis mahasiswa yang berperan penting dalam mendukung kegiatan pembelajaran dan penelitian. Mahasiswa, dosen, dan peneliti mengandalkan perpustakaan sebagai tempat untuk mengakses berbagai sumber informasi yang relevan dengan kebutuhan akademik[1]. Selain itu, perpustakaan juga menjadi tempat belajar yang kondusif, di mana pengunjung dapat berdiskusi, membaca, atau melakukan penelitian. Peran perpustakaan tidak hanya terbatas pada penyimpanan buku, tetapi juga sebagai pendukung inovasi dan pengembangan ilmu pengetahuan di lingkungan kampus.

Perpustakaan Universitas Lampung telah memiliki sistem otomatis untuk mencatat data pengunjung dan peminjaman buku yang dikelola oleh Divisi Penelitian dan Pengembangan (Litbang) dan Kerja Sama. Divisi ini bertanggung jawab dalam pengembangan sistem informasi perpustakaan, seperti website, aplikasi, dan layanan digital lainnya. Awalnya, data peminjaman buku disimpan dalam sistem Elib, yaitu perpustakaan digital berbasis teknologi yang menyediakan dan mengelola seluruh sumber daya informasi dalam format digital [2]. Sementara itu, data pengunjung tersimpan dalam database INLISLite, sebuah perangkat lunak open source yang dikembangkan oleh Perpustakaan Nasional Republik Indonesia untuk mendukung pengelolaan dan layanan perpustakaan secara digital [3]. Seiring

berjalannya waktu, dilakukan proses migrasi agar seluruh data terpusat dalam satu sistem, yaitu database INLISLite. Migrasi ini bertujuan menyederhanakan manajemen data dan memanfaatkan keunggulan INLISLite sebagai perangkat lunak open source yang lebih fleksibel dibandingkan Elib.

Meskipun proses pencatatan data sudah terotomatisasi dan terintegrasi ke dalam satu basis data, pustakawan masih menghadapi kendala dalam menyajikan informasi secara visual dan real-time. Hal ini disebabkan oleh banyaknya data yang tersimpan di dalam *Inlislite*, serta tidak adanya sistem pendukung yang dapat menampilkan data tersebut dalam bentuk ringkasan dan visualisasi yang cepat serta mudah dipahami. Ketika asesor melakukan audit dan saat manajemen membutuhkan laporan seperti jumlah pengunjung atau peminjaman harian, bulanan, tahunan, hingga informasi pengunjung dan peminjam terbanyak berdasarkan fakultas dan program studi, pustakawan harus mengolah data secara manual. Proses tersebut tidak hanya membutuhkan waktu yang lama, tetapi juga memiliki potensi tinggi terhadap terjadinya kesalahan.

Keterbatasan ini menjadi hambatan signifikan dalam penyediaan informasi yang aktual, terstruktur, dan akurat. Proses manual seperti ini tidak efisien, terutama dalam menghadapi kebutuhan informasi yang mendesak. Selain itu, analisis mendalam seperti tren kunjungan atau pola peminjaman berdasarkan prodi memerlukan upaya tambahan yang kompleks dan tidak praktis jika dilakukan tanpa dukungan sistem visualisasi data.

Pada era digital saat ini, tuntutan terhadap penyajian data yang cepat, interaktif, dan mudah dipahami menjadi semakin penting. Penyajian informasi melalui visualisasi seperti grafik atau tabel interaktif membantu pengguna dalam menangkap pola serta tren data dengan lebih intuitif. Sementara itu, kemampuan sistem untuk menampilkan data secara real-time memungkinkan pengambilan keputusan yang lebih responsif dan tepat sasaran.

Berdasarkan permasalahan tersebut, diperlukan pengembangan dashboard monitoring berbasis website yang mampu menyajikan data pengunjung dan peminjaman buku secara otomatis, visual, dan real-time. Dashboard ini tidak hanya menjadi alat bantu pelaporan, tetapi juga menjadi media strategis untuk mendukung

pengambilan keputusan manajerial dan akademik. Dengan tampilan data yang informatif dan mudah dipahami, sistem ini akan sangat membantu pustakawan dalam menyusun laporan harian, bulanan, maupun tahunan, baik untuk kepentingan audit, evaluasi manajemen, maupun akreditasi institusi dan program studi.

Selain dashboard monitoring, sistem ini juga dilengkapi dengan *landing page* yang berfungsi sebagai halaman utama yang menampilkan layanan-layanan umum yang disediakan oleh perpustakaan. *Landing page* ini dirancang untuk memberikan informasi awal kepada pengguna terkait layanan yang tersedia, terdapat 9 layanan yaitu OPAC , E-Journal, E-Book, Verivikasi Karya Ilmiah, Pendaftaran Anggota, Tanya Pustakawan, layanan SKBP (Surat Keterangan Bebas Pustaka), DIGILIB, dan informasi jumlah koleksi yang ada pas perpustakaan Universitas Lampung, seperti jumlah buku, jurnal, KKI, anggota aktif, e-book dan e-jurnal, serta jumlah total koleksi. Dengan adanya *landing page*, pengguna dapat mengakses berbagai informasi dan layanan yang tersedia di perpustakaan secara mudah tanpa harus terlebih dahulu masuk ke dalam sistem utama. Pada sistem ini terdapat total 11 layanan, yang terdiri dari 2 layanan utama, yaitu penyajian data pengunjung dan penyajian data peminjaman buku, serta 9 layanan yang ditampilkan pada halaman *landing page*. Dari keseluruhan layanan tersebut, hanya 3 layanan yang termonitoring, yaitu penyajian data pengunjung, penyajian data peminjaman buku, dan informasi jumlah koleksi. Sementara itu, layanan lainnya tidak termonitoring, melainkan hanya menampilkan tautan eksternal yang mengarah ke situs resmi masing-masing layanan.

Metode *Agile development* dipilih sebagai pendekatan dalam pengembangan dashboard ini. *Agile development* adalah metode pengembangan perangkat lunak yang dilakukan secara bertahap dan berulang, dengan menekankan kerja sama tim yang intensif dalam setiap prosesnya [4]. Dengan *Agile*, setiap komponen dalam dashboard dapat diperbaiki dan disesuaikan berdasarkan umpan balik dari pengguna selama proses pengembangan. Hal ini memungkinkan dashboard untuk lebih responsif terhadap kebutuhan pengguna, sekaligus meningkatkan kualitas desain dan fungsionalitas antarmuka. Dalam implementasinya, metode *Agile Kanban* digunakan sebagai kerangka kerja dalam proses pengembangan. *Kanban* membantu tim dalam memvisualisasikan alur kerja dan tugas proyek yang

direncanakan, sedang berlangsung, serta telah diselesaikan. Metode ini menggunakan papan fisik atau digital, yang memungkinkan tim untuk melihat dan mengelola tugas serta alur kerja [5]. *Kanban* berfokus pada visualisasi alur kerja dengan membagi setiap tugas ke dalam beberapa tahap, seperti *To do*, *In-progress*, *Testing*, dan *Done*. Dengan pendekatan ini, setiap fitur, termasuk *dashboard* monitoring dan *landing page*, dikembangkan secara bertahap dan fleksibel sesuai dengan prioritas *backlog* yang telah ditentukan. Pemilihan *Kanban* dalam proyek ini juga didukung oleh struktur tim yang terdiri dari pengembang *front-end* dan *back-end*, di mana setiap anggota tim dapat mengambil tugas sesuai keahliannya tanpa harus terikat dalam siklus sprint tetap, dengan *Kanban* memungkinkan semua anggota tim untuk lebih memahami status proyek serta meningkatkan koordinasi dan kolaborasi, dengan menyediakan artefak informasi yang dapat diakses bersama, metode ini mendukung memori kolektif, peningkatan kualitas koordinasi serta efektivitas dalam komunikasi [6]. Selain itu *Kanban* fokus mengoptimalkan alur kerja dengan menghilangkan aktivitas yang tidak perlu, mempercepat umpan balik melalui *visualisasi* alur kerja, pembatasan *WIP (Work In Process)* , pengelolaan proses, penetapan kebijakan yang jelas, serta peningkatan produktivitas dengan kolaborasi [7].

Hal ini memungkinkan proses pengembangan menjadi lebih efisien, responsif terhadap perubahan, dan memastikan setiap fitur yang dikembangkan telah diuji sebelum diimplementasikan dalam sistem utama. Dengan menerapkan metode *Agile Kanban*, pengembangan *dashboard* monitoring dan *landing page* diharapkan dapat berjalan dengan lebih sistematis, fleksibel, serta dapat terus berkembang berdasarkan kebutuhan pengguna dan evaluasi berkelanjutan. Selain itu, metode ini juga mendukung dokumentasi progres proyek secara transparan, sehingga memudahkan pemantauan kinerja tim serta peningkatan kualitas sistem secara bertahap.

Penelitian pengembangan sistem ini dirancang menggunakan framework Laravel untuk menjamin kemampuan sistem dalam beradaptasi dan berkembang seiring meningkatnya kebutuhan. Laravel digunakan karena memiliki kemampuan dalam menangani data yang kompleks serta mendukung pembangunan aplikasi web yang stabil dan andal [8]. Selain itu, *Laravel* dipilih agar lebih mudah ketika

berkolaborasi antara *front-end* dan *back-end*, karena memiliki arsitektur yang terstruktur, mendukung integrasi *API* dengan baik, serta menyediakan fitur seperti *Blade Templating*, *Eloquent ORM*, dan *RESTful routing*, yang mempermudah komunikasi antara tim *front-end* dan *back-end* dalam proses pengembangan. Fokus utama dari penelitian ini adalah pada pengembangan *front-end dashboard* yang mampu menampilkan data pengunjung dan peminjaman buku secara interaktif dan mudah dipahami. Desain antarmuka pengguna (*Front-end*) dirancang menggunakan Figma untuk menghasilkan tampilan yang interaktif dan ramah pengguna [9]. Pengujian sistem dilakukan dengan pendekatan black-box testing, yang menitikberatkan pada pemeriksaan output sistem sesuai dengan kebutuhan pengguna tanpa melihat struktur internal dari kode program. Melalui pendekatan tersebut, tim mampu memverifikasi bahwa setiap fitur inti dalam sistem berjalan sesuai dengan harapan dan kebutuhan pengguna [10]. Selain itu pengujian juga dilakukan menggunakan SUS (*System Usability Scale*) untuk mengukur nilai pengalaman pengguna.

Implementasi sistem dashboard monitoring ini diharapkan mampu mendorong peningkatan efisiensi serta mutu layanan digital di lingkungan Perpustakaan Universitas Lampung secara menyeluruh, khususnya dalam penyajian data yang real-time dan mudah dipahami. Dashboard ini menjadi salah satu elemen pendukung utama bagi Divisi Litbang dalam menghadirkan sistem layanan perpustakaan berbasis website yang lebih informatif, terintegrasi, dan responsif terhadap kebutuhan pengguna. Pengembangan sistem ini tidak hanya menjadi langkah awal dalam modernisasi layanan perpustakaan Universitas Lampung, tetapi juga berpotensi menjadi model pengembangan layanan digital di lingkungan perguruan tinggi lainnya. Penelitian ini diharapkan memberikan manfaat jangka panjang bagi pengelola perpustakaan maupun seluruh sivitas akademika Universitas Lampung.

1.2 Perumusan Masalah

Adapun perumusan masalah, berdasarkan latar belakang yang mendasari penelitian ini, adalah sebagai berikut:

1. Bagaimana merancang *front-end dashboard* monitoring data pengunjung dan peminjaman buku yang efektif dan informatif untuk digunakan di perpustakaan Universitas Lampung ?
2. Bagaimana memastikan bahwa fungsi utama pada *front-end dashboard*, seperti tampilan data pengunjung, peminjaman buku, tabel dan grafik *visual*, berjalan sesuai kebutuhan pengguna melalui pengujian *fungsiional* menggunakan *black-box testing*?
3. Bagaimana mengukur tingkat kemudahan penggunaan dan kepuasan pengguna terhadap *front-end dashboard* yang dirancang ?

1.3 Tujuan Penelitian

Adapun tujuan penelitian ini sebagai berikut:

1. Merancang dan mengembangkan *front-end dashboard* monitoring data pengunjung dan peminjaman buku yang efektif dengan tampilan informatif, sesuai kebutuhan pengguna di perpustakaan Universitas Lampung, untuk mempermudah pemantauan data secara *real-time* menggunakan metode *Agile development* dalam kerangka waktu yang ditentukan.
2. Memastikan bahwa fungsi utama *front-end dashboard*, seperti tampilan data pengunjung, peminjaman buku, tabel dan grafik *visual*, berjalan dengan baik sesuai dengan spesifikasi dan kebutuhan pengguna melalui pengujian *fungsiional* menggunakan metode *black-box testing*.
3. Mengukur tingkat kemudahan penggunaan dan kepuasan pengguna terhadap *front-end dashboard* , serta menganalisis hasilnya untuk memberikan wawasan tentang pengalaman pengguna.

1.4 Manfaat Penelitian

Adapun manfaat penelitian ini sebagai berikut:

1. Penelitian ini membantu perpustakaan dalam meningkatkan efisiensi pengelolaan data pengunjung dan peminjaman buku. *Dashboard* yang dirancang mempermudah staf perpustakaan untuk memantau data secara *real-time*, sehingga mempercepat proses penyajian laporan dan mendukung pengambilan keputusan strategis.
2. *Dashboard* yang dirancang secara sederhana dan informatif pada dashboard ini memberikan kemudahan akses dan interaksi, sehingga meningkatkan efisiensi kerja staf perpustakaan dan auditor. Sistem ini memungkinkan akses cepat ke informasi, seperti jumlah pengunjung dan jumlah peminjam buku, serta data program studi dengan aktivitas tertinggi.
3. Penelitian ini dapat dijadikan acuan dalam pengembangan sistem monitoring sejenis di masa mendatang, terutama dalam penerapan metode *Agile Kanban*. Hasil pengujian dan implementasi diharapkan dapat menjadi acuan untuk pengembangan perangkat lunak berbasis data *real-time* yang lebih baik di berbagai institusi.

1.5 Batasan Masalah

Adapun batasan masalah dalam penelitian ini mencakup hal-hal berikut:

1. Penelitian ini difokuskan pada pengembangan antarmuka (*front-end*) *dashboard* untuk menampilkan data pengunjung dan peminjaman buku secara *real-time* di perpustakaan Universitas Lampung. Pengelolaan data *back-end* dan integrasi dengan sistem lain berada di luar cakupan penelitian ini.
2. Penelitian ini dilakukan secara kolaboratif dengan pengembangannya menggunakan *framework laravel* untuk mendukung pengelolaan data di sisi *front-end*, dengan fokus pada pengembangan antarmuka pengguna tanpa mencakup pengembangan fitur *back-end* secara terperinci, di mana data yang ditampilkan pada *dashboard* monitoring sepenuhnya berasal dari sistem *back-end* yang telah ada dan ditampilkan secara *real-time*.

3. Proses pengujian sistem menggunakan metode black-box testing guna memverifikasi bahwa seluruh fitur utama pada dashboard telah berjalan sesuai fungsinya, seperti visualisasi data *real-time* dan grafik informatif, berfungsi sesuai spesifikasi dan pengujian SUS (*System Usability Scale*) menguji tingkat kegunaan (usability) sistem berdasarkan pengalaman pengguna melalui kuisisioner standar, tanpa mengevaluasi logika internal dari sistem *back-end*.

1.6 Sistematika Penulisan Skripsi / Tugas Akhir

Penulisan skripsi/tugas akhir ini disusun ke dalam lima bab dengan rincian sebagai berikut:

BAB I : PENDAHULUAN

Bab ini membahas latar belakang penelitian terkait kebutuhan akan sistem monitoring yang efisien untuk data pengunjung dan peminjaman buku di perpustakaan Universitas Lampung. Bab ini juga memuat rumusan masalah, tujuan yang hendak dicapai melalui penelitian ini, serta manfaat yang diharapkan dari hasil penelitian. Selain itu, bab ini menjelaskan batasan masalah yang difokuskan pada pengembangan front-end dashboard, dan disertai dengan sistematika penulisan sebagai panduan untuk memahami alur keseluruhan laporan.

BAB II : DASAR TEORI

Bab ini membahas kajian literatur yang mendukung penelitian, meliputi konsep dasar pentingnya Perpustakaan, Struktur Perpustakaan Universitas Lampung, *Dashboard*, Monitoring Data, *Front-end*, Metode *Agile development*, *Kanban*, *Framework Laravel*, Figma, Bootstrap, Trello, eLib, INLISLite, PHP, MySQL, Laragon, Gitlab, Desain Antarmuka Pengguna, Kebutuhan, Diagram, *Black-Box Testing*, *System Usability Scale* dan Penelitian Terdahulu.

BAB III : METODOLOGI PENELITIAN

Bab ini menguraikan metode yang digunakan dalam pelaksanaan penelitian, meliputi rincian waktu dan tempat penelitian, perangkat yang digunakan, serta tahapan-tahapan yang mengikuti pendekatan Agile dengan kerangka kerja *Kanban*.

BAB IV : PEMBAHASAN

Bab ini memaparkan hasil dari proses perancangan dan implementasi dashboard monitoring pada sisi *front-end*, yang bertujuan untuk menampilkan data pengunjung dan peminjaman buku secara *real-time*. Sistem ini dirancang dengan mempertimbangkan aspek fungsionalitas serta kemudahan penggunaan antarmuka. Pengujian dilakukan menggunakan metode *Black-box Testing*, yang difokuskan untuk mengevaluasi apakah fitur-fitur utama berjalan sesuai kebutuhan pengguna tanpa memeriksa struktur internal kode. Selain itu, bab ini juga mencakup evaluasi antarmuka pengguna menggunakan metode *System Usability Scale* (SUS) guna memastikan bahwa desain yang diterapkan mampu mendukung pengalaman pengguna yang efektif dan efisien dalam mengakses informasi melalui sistem dashboard.

BAB V : KESIMPULAN DAN SARAN

Bab ini menyajikan ringkasan dari hasil penelitian yang telah dilakukan, mencakup pencapaian terhadap tujuan yang telah ditetapkan, serta memberikan rekomendasi untuk pengembangan di masa mendatang, seperti penggabungan dengan sistem back-end atau penambahan fitur baru pada dashboard.

II. TINJAUAN PUSTAKA

2.1 Perpustakaan

Perpustakaan adalah lembaga yang memegang peranan penting dalam memberikan akses informasi dan sumber pengetahuan guna menunjang proses pendidikan, riset, serta pengembangan keilmuan. Perpustakaan berperan sebagai pusat informasi yang menjadi sumber pengetahuan, tempat mendukung kegiatan riset, media hiburan, wadah pelestarian budaya bangsa, serta menyediakan beragam layanan lainnya [11]. Fungsi ini menjadikan perpustakaan sebagai salah satu sarana penting dalam meningkatkan literasi informasi masyarakat.

Perpustakaan menyediakan beragam jenis pustaka, seperti buku, jurnal, dan koleksi digital, untuk mendukung kegiatan akademik dan penelitian. Bahan pustaka adalah koleksi yang ada pada perpustakaan yaitu karya cetak (seperti buku teks dan referensi), terbitan berkala (seperti majalah dan surat kabar), media audiovisual, serta sumber elektronik [12]. Dalam pengelolaannya, perpustakaan dibantu oleh pustakawan sebagai tenaga profesional yang dibekali dengan kemampuan serta pemahaman mendalam dibidang kepustakawanan seperti pengelolaan informasi , memberikan layanan perpustakaan dan bertugas mengelola koleksi pustaka, serta memastikan proses sirkulasi dan akses informasi berjalan dengan baik [13]. Kehadiran pustakawan dan tersedianya pustaka menjadi dua komponen utama yang saling melengkapi dalam menunjang fungsi perpustakaan sebagai ruang belajar dan pusat pengetahuan yang terstruktur.

Dalam perkembangannya, perpustakaan tidak hanya menyediakan koleksi fisik,

seperti buku, majalah, dan surat kabar, tetapi juga sumber daya digital, seperti e-book, jurnal elektronik, dan database penelitian. Perpustakaan modern juga memiliki peran dalam pengembangan keterampilan literasi informasi di kalangan penggunanya [14]. Literasi informasi merujuk pada kemampuan mengakses, mengevaluasi, dan menerapkan informasi secara efektif sesuai kebutuhan.

Selain itu, perpustakaan juga berperan dalam meningkatkan keterampilan dalam melakukan penelitian. Hal ini dilakukan melalui berbagai program, seperti pelatihan untuk menggunakan basis data akademik, lokakarya tentang cara menulis karya ilmiah, dan seminar literasi digital. Program-program ini dirancang untuk membantu pengguna dalam mengoptimalkan pemanfaatan seluruh sumber daya yang tersedia. Dengan adanya program ini, mahasiswa, dosen, atau peneliti dapat lebih mudah mencari informasi yang relevan, memahami cara penggunaannya, dan menerapkannya dalam pekerjaan akademik. Oleh karena itu, perpustakaan berperan tidak hanya sebagai sarana akses informasi, melainkan juga sebagai mitra strategis dalam menunjang pencapaian akademik.

2.2 Struktur Perpustakaan Universitas Lampung

Pengembangan sistem informasi dan dokumentasi berupa perpustakaan di Universitas Lampung (Unila) dimulai sejak tahun 1980 berlandaskan Peraturan Pemerintah Nomor 5 Tahun 1980 dan kemudian diperkuat dengan Keputusan Presiden Nomor 43 Tahun 1982 yang menetapkan perpustakaan sebagai unit terpusat. Pembentukan perpustakaan terpusat ini dilatar belakangi oleh kurang optimalnya pengelolaan koleksi dan sumber daya perpustakaan pada masing-masing fakultas di Universitas Lampung, sehingga perlu adanya pelayanan informasi yang terintegrasi untuk seluruh sivitas akademika Unila. Awalnya, perpustakaan terpusat berlokasi di satu gedung dan dikelola oleh staf dari berbagai perpustakaan fakultas. Selanjutnya, pembentukan Unit Pelaksana Teknis (UPT) Perpustakaan Universitas Lampung didasarkan pada Keputusan Menteri Pendidikan dan Kebudayaan Republik Indonesia No. 091/0/1993 serta Keputusan Rektor Universitas Lampung No. 09/KTPS/1994. UPT Perpustakaan ini berada di bawah tanggung jawab langsung Rektor dan dibina oleh Wakil Rektor Bidang Akademik (sebelumnya Pembantu Rektor I).

Struktur organisasi UPT Perpustakaan Universitas Lampung kemudian mengalami pembaruan, di mana kepala perpustakaan memegang tanggung jawab penuh kepada Rektor dan berkoordinasi dengan Wakil Rektor Bidang Akademik. Dalam pelaksanaan tugasnya, kepala perpustakaan dibantu oleh Kepala Subbagian Tata Usaha serta kelompok jabatan fungsional pustakawan. Pengaturan struktur ini merujuk pada Peraturan Menteri Pendidikan dan Kebudayaan Republik Indonesia No. 72 Tahun 2014, khususnya Pasal 99 hingga Pasal 103.

Selanjutnya, Susunan Organisasi dan Tata Kerja (STOK) Perpustakaan Unila kembali disempurnakan dari Unit Pelaksana Teknis (UPT) resmi diubah menjadi Unit Penunjang Akademik (UPA) berdasarkan Peraturan Menteri Pendidikan, Kebudayaan, Riset, dan Teknologi RI No. 49 Tahun 2024.

Adapun berikut merupakan struktur organisasi UPA Perpustakaan Universitas Lampung yang terdapat pada gambar 2.1:

Struktur organisasi merupakan pengaturan yang menunjukkan keterkaitan antara posisi, tugas, wewenang, dan tanggung jawab di dalam sebuah organisasi. Struktur ini berfungsi untuk mengatur pembagian peran, kekuasaan, serta aliran informasi agar terkoordinasi dengan baik di antara seluruh bagian organisasi.



Gambar 2.1 Struktur Organisasi UPA Perpustakaan Universitas Lampung

Sumber : <https://library.unila.ac.id/upt-perpustakaan-universitas->

Berdasarkan hasil wawancara dengan salah satu pustakawan UPA Perpustakaan Universitas Lampung yang bertugas pada divisi Litbang yaitu Ibu Angelino Vinanti Sonjaya, S.I.Pust. Diperoleh penjelasan mengenai struktur organisasi UPA Perpustakaan Universitas Lampung, sebagai berikut:

Struktur organisasi UPA Perpustakaan Universitas Lampung terdiri dari beberapa tingkatan yang terkoordinasi dengan jelas. Pada tingkat tertinggi, terdapat Rektor yang menaungi dan bertanggung jawab langsung atas UPA Perpustakaan Universitas Lampung dengan dibantu oleh Wakil Rektor. Lalu dibawahnya terdapat Kepala Perpustakaan yang bertanggung jawab langsung dalam operasional perpustakaan yang bekerjasama dengan Kelompok Pustakawan dalam pelaksanaan tugas-tugas teknis kepastakaan.

Kelompok Pustakawan terbagi menjadi 3 divisi utama, yaitu:

1. Divisi Layanan Pengguna

Divisi yang bertugas dalam memberikan pelayanan langsung kepada menggunakan perpustakaan khususnya mahasiswa. Divisi ini terbagi menjadi beberapa sub-divisi yaitu:

- Sub Divisi Layanan Sirkulasi
Pada Sub Divisi Layanan Sirkulasi, mahasiswa dilayani dalam proses peminjaman dan pengembalian buku, pembuatan Surat Keterangan Bebas Pustaka, serta pembayaran tunggakan denda peminjaman atau penggantian buku yang hilang.
- Sub Divisi Layanan Cadangan
Pada Sub Divisi Layanan Cadangan, disediakan layanan tandon yang berfungsi menyediakan koleksi cadangan atau salinan terakhir dari buku-buku yang ada di perpustakaan. Setiap judul buku hanya tersedia satu eksemplar. Buku-buku yang ada di layanan cadangan juga tersedia di layanan sirkulasi, namun buku di layanan cadangan hanya dapat dibaca di tempat dan tidak diperkenankan untuk dipinjam.
- Sub Divisi Layanan Referensi dan Jurnal
Pada Sub Divisi Layanan Referensi dan Jurnal, disediakan berbagai koleksi buku seperti kamus, ensiklopedia, buku-buku dengan subjek

ilmu yang lebih spesifik, jurnal tercetak, serta akses ke jurnal online. Layanan ini hanya memperbolehkan pengguna untuk membaca koleksi di tempat dan tidak diperkenankan untuk meminjam dan membawa pulang.

- Sub Divisi Layanan Koleksi Karya Ilmiah (KKI)

Pada Sub Divisi Layanan Koleksi Karya Ilmiah (KKI), disediakan berbagai macam koleksi karya ilmiah, baik dalam bentuk cetak maupun digital yang telah diunggah ke dalam Digilib (Digital Library). Karya ilmiah yang tersedia hanya dapat dibaca di tempat dan tidak diperbolehkan untuk dipinjam. Untuk karya ilmiah cetak, koleksi yang tersedia adalah terbitan tahun 2015, sedangkan karya ilmiah terbaru sepenuhnya telah tersedia dalam format digital melalui Digilib.

- Sub Divisi Layanan Teknologi Informasi

Pada sub divisi Layanan Teknologi Informasi, yang juga dikenal sebagai layanan verifikasi karya ilmiah, disediakan pelayanan untuk mahasiswa tingkat akhir yang ingin membuat akun Digilib guna mengunggah skripsi. Setelah mahasiswa mengunggah skripsi, petugas akan melakukan verifikasi untuk memastikan bahwa dokumen tersebut telah memenuhi seluruh persyaratan yang ditetapkan. Jika skripsi dinyatakan sesuai, petugas akan menerbitkan tanda terima dalam bentuk file PDF yang dikirimkan melalui email mahasiswa untuk dicetak secara mandiri. Selain itu, terdapat beberapa fakultas yang mensyaratkan cap dan tanda tangan pengesahan, yang dapat diperoleh melalui layanan verifikasi ini.

2. Divisi Layanan Teknik

Divisi Layanan Teknik merupakan bagian yang berperan dalam mendukung operasional perpustakaan secara teknis, namun tidak berinteraksi secara langsung dengan pengguna. Pada divisi Layanan Teknik terdapat beberapa sub-divisi yaitu :

- Sub Divisi Pengadaan Bahan Pustaka

Pada Sub-divisi Pengadaan Bahan Pustaka, setiap koleksi buku baru, baik berupa buku tercetak maupun referensi lainnya, akan diproses terlebih dahulu di bagian ini. Proses pengadaan dilakukan melalui survei kebutuhan buku yang diperlukan oleh dosen dan mahasiswa, sebelum koleksi tersebut tersedia untuk layanan perpustakaan.

- Sub Divisi Pengolahan Bahan Pustaka

Di sub-divisi Pengolahan Bahan Pustaka, setelah melewati tahapan survei dan pendataan di Divisi Pengadaan. Buku-buku yang telah dibeli dan terdata kemudian diproses di sini. Pada tahap ini, pustakawan bekerja dengan penuh ketelitian: memberi tag, melabeli setiap buku, serta menginput data menggunakan sistem ELIP. Setelah seluruh proses pengolahan selesai, buku-buku tersebut didistribusikan ke masing-masing divisi layanan pengguna. Buku cetak akan diteruskan ke Divisi Sirkulasi, sementara salinan tambahan disimpan di Layanan Cadangan. Buku yang diperuntukkan sebagai bahan referensi dikirimkan ke Divisi Layanan Referensi. Semua proses ini memastikan setiap buku siap menemui pembacanya di tempat yang tepat.

- Sub Divisi Perawatan Bahan Pustaka

Subdivisi ini menyediakan layanan dalam menangani mahasiswa yang tanpa sengaja merusak buku yang dipinjam. Di sini, setiap kasus diselesaikan dengan penggantian buku atau pembayaran denda, sesuai ketentuan yang berlaku. Selain itu, di sub-divisi ini, pustakawan di subdivisi ini juga teliti memeriksa dan memperbaiki buku-buku yang mengalami kerusakan agar tetap layak digunakan.

3. Divisi Litbang & Kerjasama

Divisi ini bertanggung jawab mengelola berbagai aplikasi dan website perpustakaan, seperti e-book digital Universitas Lampung. Selain itu, divisi ini juga menangani kerja sama dengan pihak luar, mengelola akun WhatsApp admin, serta mengatur sosial media perpustakaan untuk memastikan informasi

dan layanan selalu terhubung dengan pengguna. Pada divisi ini terdapat sub-divisi sebagai berikut:

- Sub Divisi Pengembangan Sistem Informasi Publikasi

Sub-divisi Pengembangan Sistem Informasi dan Publikasi bertugas mengelola dan mengembangkan sistem informasi perpustakaan, seperti website, dan layanan digital lainnya. Selain itu, subdivisi ini juga bertanggung jawab mempublikasikan berbagai informasi tentang layanan dan koleksi perpustakaan melalui media sosial, atau platform digital, serta menyusun strategi promosi untuk meningkatkan pemanfaatan layanan oleh pengguna. Di samping itu, subdivisi ini juga mengelola data statistik penggunaan layanan dan menyajikannya dalam bentuk laporan atau infografis yang menarik.

2.3 Dashboard

Dashboard adalah media penyajian data dalam bentuk visual yang dirancang untuk memberikan informasi secara ringkas, jelas, dan mudah diinterpretasikan oleh pengguna. Alat ini memungkinkan pengguna untuk melihat data yang relevan secara *real-time* dalam bentuk grafik, tabel, atau indikator *visual* lainnya. Informasi tersebut disusun dan ditampilkan pada satu layar agar mudah diakses, memungkinkan pengguna untuk memahaminya secara sekilas tanpa memerlukan waktu yang lama [15]. Dashboard yang dirancang dengan baik dapat menyajikan informasi yang bersifat actionable, yaitu informasi yang dapat langsung dimanfaatkan sebagai landasan dalam proses pengambilan keputusan [16]. Elemen utama dari dashboard yang baik meliputi penyajian data yang relevan, penggunaan desain yang intuitif, serta kemampuan untuk menampilkan data secara akurat dan cepat.

Dalam konteks perpustakaan, dashboard memiliki peran penting dalam mendukung pengelolaan data operasional. Sebagai contoh, dashboard dapat digunakan untuk memantau data pengunjung perpustakaan secara *real-time*, sehingga pihak pengelola dapat mengetahui tingkat kunjungan harian, mingguan, atau bulanan. Selain itu, dashboard juga dapat digunakan untuk melacak jumlah peminjaman buku berdasarkan kategori tertentu, seperti jenis buku yang paling sering dipinjam,

program studi peminjam terbanyak, atau tren peminjaman selama periode tertentu. Dashboard yang diterapkan dalam sistem perpustakaan memungkinkan pengelola untuk memperoleh wawasan yang lebih baik terkait aktivitas pengguna dan kebutuhan mereka [17]. Hal ini mendukung efisiensi operasional perpustakaan serta memberikan nilai tambah bagi pengguna layanan perpustakaan.

Dengan kemampuan untuk menyajikan data secara *visual* dan informatif, dashboard tidak hanya mempermudah pengambilan keputusan tetapi juga membantu pengelola perpustakaan untuk lebih responsive terhadap perubahan kebutuhan pengguna. Implementasi *dashboard* di perpustakaan modern menjadi salah satu langkah strategis dalam era digitalisasi, di mana data menjadi asset penting untuk mendukung pengelolaan dan pengembangan layanan

2.4 Monitoring Data

Monitoring data merupakan proses penting yang melibatkan pengumpulan, analisis, dan pelaporan informasi untuk memastikan bahwa sistem berjalan sesuai dengan fungsinya serta mencapai tujuan yang telah ditetapkan secara efektif dan efisien [18]. Monitoring data tidak hanya mencakup pemantauan performa sistem, tetapi juga membantu dalam mengidentifikasi kendala, mengevaluasi pencapaian target, serta memberikan wawasan yang relevan untuk pengambilan keputusan strategis. Monitoring dapat menentukan keberhasilan atau kegagalan suatu tujuan organisasi. Dalam hal ini, monitoring bertujuan untuk menyediakan informasi yang relevan untuk mendukung proses pengambilan keputusan, baik dalam hal perbaikan proses, penyusunan strategi baru, maupun evaluasi efektivitas dari implementasi kebijakan yang ada.

Dalam konteks sistem informasi perpustakaan, monitoring data menjadi salah satu elemen kunci untuk memastikan layanan perpustakaan dapat berjalan secara optimal. Monitoring data pada sistem informasi perpustakaan mampu membantu pengelola dalam memahami pola penggunaan layanan perpustakaan, seperti tingkat kunjungan pengguna, frekuensi peminjaman buku, serta kategori sumber daya yang paling sering diakses [19]. Informasi ini dapat digunakan untuk menyusun strategi pengelolaan perpustakaan yang lebih responsif terhadap kebutuhan pengguna.

Implementasi monitoring data yang terintegrasi pada sistem informasi perpustakaan juga memungkinkan pengelola untuk melakukan evaluasi layanan secara berkelanjutan. Hal ini mencakup pemantauan performa sistem digital, seperti kecepatan akses, keandalan fitur, dan tingkat kepuasan pengguna. Dengan demikian, monitoring data tidak sekadar digunakan untuk evaluasi, tetapi juga berperan sebagai sarana dalam mendukung keputusan yang didasarkan pada data, serta mendorong peningkatan mutu layanan perpustakaan secara menyeluruh.

2.5 Front-end

Front-end merupakan komponen dari aplikasi web atau situs yang langsung berhubungan dengan pengguna, mencakup seluruh elemen tampilan seperti teks, gambar, tombol, serta navigasi menu. Pengembangan front-end bertujuan untuk menghadirkan pengalaman pengguna yang optimal dengan memanfaatkan elemen-elemen utama seperti HTML untuk menyusun struktur serta isi halaman, CSS untuk mengatur tampilan visual dan tata letak, serta JavaScript guna memberikan fitur interaktif pada web.

Dalam penelitian ini, fokus pengembangan diarahkan pada *front-end dashboard*, yang berfungsi sebagai alat visualisasi untuk menyajikan data secara ringkas dan mudah dipahami. *Dashboard* yang dirancang dengan baik dapat membantu pengguna dalam mengakses informasi secara efisien dan mendukung pengambilan keputusan yang lebih cepat. Keberhasilan dalam membangun sebuah dashboard tidak semata-mata ditentukan oleh jumlah fitur yang tersedia, melainkan juga oleh sejauh mana sistem tersebut mampu memberikan kenyamanan dan kemudahan bagi pengguna dalam melakukan interaksi [20].

Keberhasilan *front-end dashboard* diuji melalui pengujian *fungsi*ional menggunakan *Black-Box Testing* untuk memastikan setiap fitur bekerja dengan baik, serta pengujian *usability* menggunakan *System Usability Scale (SUS)* untuk menilai tingkat kenyamanan dan kemudahan penggunaan oleh pengguna. Hal ini menekankan pentingnya *front-end* sebagai wajah aplikasi yang mempengaruhi pengalaman pengguna secara langsung.

Dengan adanya *front-end* yang optimal, sistem dapat memberikan nilai tambah

dalam mendukung kebutuhan pengguna secara *real-time*, seperti memantau data peminjaman buku atau statistik pengunjung perpustakaan. Oleh karena itu, peran *front-end development* menjadi krusial dalam memastikan bahwa aplikasi atau situs web mampu memenuhi ekspektasi pengguna sekaligus meningkatkan efisiensi operasional.

2.6 Metode Agile Development

Metode *Agile Development* merupakan pendekatan inovatif untuk mengembangkan perangkat lunak dengan sifat fleksibilitas, kolaborasi, dan kemampuan untuk beradaptasi terhadap perubahan kebutuhan secara cepat. Penerapan metodologi pengembangan *agile* memungkinkan perubahan perangkat lunak diputuskan berdasarkan kondisi pasar sasaran, menjadikan hasil perangkat lunak lebih fleksibel dan efisien, serta memastikan bahwa perangkat lunak terus diperbarui sesuai dengan kondisi selama iterasi[21].

Agile development menempatkan pengguna dan tim pengembang dalam satu ekosistem kerja yang saling terhubung sehingga pengembangan perangkat lunak lebih responsif dan iteratif. *Agile* memiliki karakteristik umum seperti desain yang sederhana, kemampuan untuk melakukan rilis dalam waktu singkat, kolaborasi tim yang luas, praktik pair programming, pengujian yang dilakukan secara terus-menerus selama pengembangan, kemampuan untuk mengantisipasi perubahan kebutuhan, serta penerimaan perubahan sebagai peluang untuk keuntungan[22].

Metode *Agile development* memiliki tahapan utama yaitu melakukan perencanaan dengan mengidentifikasi kebutuhan pengguna, spesifikasi awal, dan perencanaan sprint. Kemudian melakukan desain, dimana desain awal dibuat, terutama untuk fitur yang dikembangkan dalam sprint tertentu. Desain ini dapat mencakup gambar rangka, *prototipe*, atau desain antarmuka pengguna yang mendetail. Lalu melakukan implementasi yakni pengembangan dimulai dengan menulis kode, menjalankan tes awal, dan menyelesaikan fitur prioritas dalam sprint. Selanjutnya melakukan pengujian yaitu hasil sprint diuji untuk memastikan fitur memenuhi persyaratan. Pengujian juga mencakup evaluasi penggunaan pengguna akhir. Kemudian melakukan revisi setelah pengujian, yaitu masukan dikumpulkan dari pengguna atau tim untuk perbaikan pada sprint berikutnya. Lalu tahap terakhir

merilis dan evaluasi, setelah beberapa kali iterasi, produk akhir dirilis dan dievaluasi untuk memastikan memenuhi harapan.

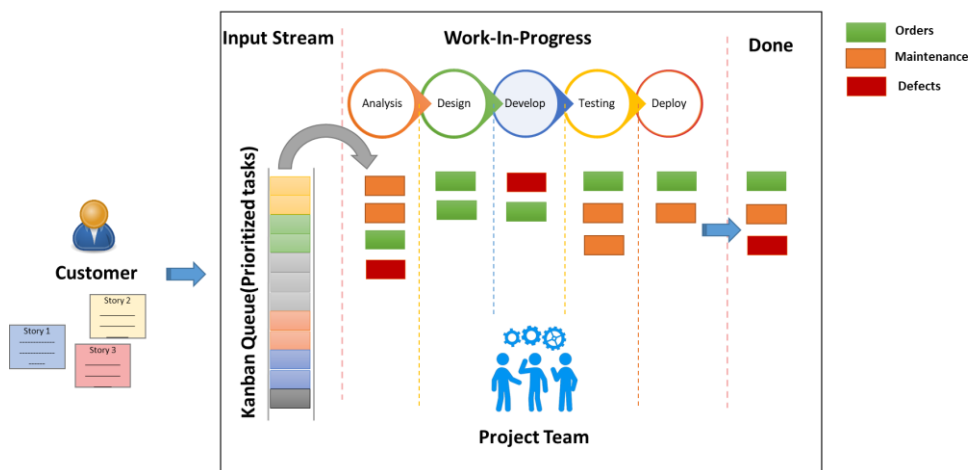
Dalam konteks pengembangan *dashboard*, *Agile* memungkinkan tim untuk merespons kebutuhan pengguna yang dinamis. Proses ini memanfaatkan prinsip kolaborasi yang intensif di antara anggota tim untuk memastikan pengembangan berjalan sesuai tujuan. Metode *Agile* tidak hanya meningkatkan fleksibilitas dalam pengembangan perangkat lunak tetapi juga mendorong pendekatan berorientasi pengguna. Dengan pendekatan iteratif ini, pengembangan dashboard dapat disesuaikan secara lebih tepat dengan kebutuhan pengguna serta menghasilkan solusi yang selaras dengan sasaran organisasi [23].

2.7 Kanban

Salah satu kerangka kerja pada metode *Agile development* yaitu *Kanban*. *Kanban*, yang berarti "sinyal *visual*," awalnya digunakan oleh pekerja Toyota untuk mengawasi alur kerja dalam sistem manufaktur mereka, dengan metode *Kanban* tim dapat berkomunikasi lebih efisien, karena menyediakan informasi yang jelas mengenai tugas yang harus diselesaikan dan waktu pengerjaannya [7]. *Kanban* menghindari gangguan pasokan dan kelebihan stok dalam sistem manufaktur, mempermudah identifikasi kemacetan yang menghambat alur kerja, mengukur produktivitas melalui waktu tunggu, serta membuktikan efisiensinya di berbagai industri, menjadikannya salah satu metodologi produksi paling populer di dunia [7].

Kanban berfungsi sebagai alat untuk mengelola alur kerja secara *visual*, memastikan efisiensi dan transparansi dalam proses pengembangan perangkat lunak. *Kanban* digunakan untuk mendukung pengembangan perangkat lunak dengan cara yang fleksibel dan responsif terhadap perubahan kebutuhan pengguna. Tim dapat dengan cepat menyesuaikan prioritas dan alur kerja sesuai kebutuhan tanpa harus mengikuti rencana yang kaku. Kerangka kerja *Kanban* ini dapat meningkatkan pemahaman, visibilitas, dan pengendalian alur kerja[24].

Berikut merupakan gambar *Kanban Board*:



Gambar 2.2 Kanban Board

Kanban menawarkan pendekatan *visual* untuk mengelola alur kerja dengan menggunakan papan *Kanban*. Berdasarkan Gambar 2.2, merupakan gambar Kanban Board [7]. *Kanban Queue (Prioritized Tasks)*, *Work-In-Progress (WIP)*, dan *Done*. Masing-masing kolom merepresentasikan tahapan yang dilalui oleh setiap tugas, dimulai dari pengumpulan permintaan hingga penyelesaian pekerjaan. Pada bagian paling kiri terdapat permintaan *customer* lalu tugas tersebut di masukkan kedalam *Kanban Queue* untuk diprioritaskan dan siap dikerjakan melalui *Input Stream*, merupakan aliran masuk dari permintaan atau kebutuhan (biasanya berupa user story atau task) yang datang dari pihak eksternal, seperti customer. Setelah itu, kebutuhan serta *user story* yang telah diseleksi dan ditinjau akan masuk ke *Kanban Queue (Prioritized Tasks)*. Tahap ini berisi daftar tugas yang telah diprioritaskan dan siap untuk dikerjakan, namun belum ditangani oleh tim pengembang. *Kanban Queue* berfungsi sebagai antrian tugas yang siap ditarik ke tahap pengerjaan, berdasarkan kapasitas dan kebutuhan tim. Tugas-tugas dari *Kanban Queue* kemudian masuk ke tahap *Work-In-Progress (WIP)*, yaitu inti dari proses kerja aktif. Tahapan *WIP* terdiri dari beberapa kolom yang merepresentasikan langkah-langkah pengerjaan tugas, yaitu *Analysis* (tahap *analysis* merupakan langkah awal dalam proses pengembangan di mana tim menganalisis tugas untuk memahami ruang lingkup, kebutuhan teknis, dan potensi hambatan, serta berperan sebagai upaya pengendalian untuk mencegah kesalahan sejak dini, menghindari penumpukan pekerjaan berlebihan, dan memastikan kelancaran serta efisiensi alur kerja secara keseluruhan [7], *Design* (merancang

tampilan antarmuka atau struktur sistem berdasarkan hasil analisis sebelumnya), *Develop* (proses pengembangan atau pengkodean sistem berdasarkan rancangan yang telah ditetapkan sebelumnya), *Testing* (pengujian sistem untuk menguji sistem guna memastikan setiap fitur berfungsi sesuai spesifikasi dan bebas dari error), *Deploy* (tahap implementasi fitur atau sistem yang telah lolos uji ke dalam lingkungan produksi). Setelah melewati seluruh proses dalam *WIP*, tugas akan dipindahkan ke kolom *Done*, yang menandakan bahwa pekerjaan telah selesai sepenuhnya dan fitur siap digunakan oleh pengguna. Di sisi kanan atas gambar terdapat legenda warna pada tugas menunjukkan status dan prioritas tugas tersebut, yakni warna hijau (*Orders*) menandakan bahwa item tersebut merupakan permintaan fitur atau kebutuhan dari pengguna, jadi tugas utama yang diprioritaskan karena harus diselesaikan sesuai kebutuhan sistem atau stakeholder, warna oranye (*Maintenace*) menunjukkan bahwa pekerjaan tersebut adalah bagian dari pemeliharaan sistem, seperti perbaikan kecil, peningkatan performa, atau pembaruan sistem yang bukan fitur baru, dan warna merah (*Defects*) menandakan adanya bug atau error dalam sistem yang perlu segera diperbaiki. Tugas ini biasanya bersifat prioritas tinggi karena menyangkut kesalahan yang berdampak pada fungsionalitas sistem.

Beberapa istilah yang umum digunakan dalam proses pengembangan dengan *Kanban Board* antara lain:

- a. *Requirment Gatterring* adalah Tahap awal dalam siklus agile seperti analisis sistem dan pengumpulan kebutuhan pengguna yang sangat menentukan keberhasilan akhir proyek sistem informasi [25] [26].
- b. *User story* adalah deskripsi singkat dari kebutuhan atau fungsi sistem yang ditulis dari sudut pandang pengguna sistem nantinya, dan sebagai dasar dalam penyusunan product backlog [27].
- c. *Backlog* adalah kumpulan tugas yang harus diselesaikan selama proses pengembangan sistem, yang disusun berdasarkan user story yang telah ditentukan[28].
- d. *Task* adalah unit kerja yang lebih kecil yang berasal dari *backlog* dan harus diselesaikan untuk memenuhi kebutuhan sistem .
- e. Trello adalah aplikasi kolaborasi berbasis cloud yang mendukung manajemen

proyek dengan metode Kanban, memungkinkan tim mengatur dan memantau seluruh aktivitas tugas dalam satu tampilan visual melalui (*board*), daftar (*lists*), dan kartu (*cards*)[29].

Dalam pengembangan *dashboard*, kerangka kerja *Kanban* sangat membantu tim untuk memvisualisasikan alur kerja secara *real-time*. Dengan papan *Kanban*, setiap tugas dapat dipantau secara jelas, sehingga memudahkan tim untuk mengidentifikasi hambatan atau bottleneck yang mengganggu kelancaran proses pengembangan [30]. Selain itu memudahkan identifikasi masalah dan pengambilan keputusan. Misalnya, jika terdapat tugas yang terhenti di kolom “*In-progress*,” tim dapat segera mengevaluasi dan mencari solusi untuk mempercepat penyelesaiannya. Keunggulan lain dari *Kanban* adalah fleksibilitasnya dalam mengelola tugas. Selain itu, *Kanban* mendorong pembagian kerja yang jelas, di mana setiap anggota tim memahami tanggung jawab mereka masing-masing. Hal ini menciptakan lingkungan kerja yang lebih terorganisasi dan kolaboratif, yang sangat penting dalam pengembangan perangkat lunak yang kompleks

Dalam konteks pengembangan *dashboard*, *Kanban* juga memfasilitasi tim untuk memastikan setiap fitur atau elemen desain diselesaikan dengan efisien. Dengan memvisualisasikan alur kerja, tim dapat memprioritaskan tugas yang memiliki dampak besar terhadap pengalaman pengguna. Misalnya, penyelesaian komponen antarmuka utama dapat didahulukan sebelum fitur tambahan lainnya. Hal ini tidak hanya membantu mengoptimalkan alur kerja, tetapi juga memastikan bahwa hasil pengembangan sesuai dengan kebutuhan pengguna.

Secara keseluruhan, penerapan kerangka kerja *Kanban* dalam pengembangan *dashboard* memberikan banyak manfaat, mulai dari peningkatan transparansi alur kerja hingga pengelolaan tugas yang lebih efisien. Dengan pendekatan ini, tim dapat mengelola penelitian secara lebih efektif, mengidentifikasi masalah dengan cepat, dan memastikan bahwa hasil pengembangan memenuhi ekspektasi pengguna.

2.8 Framework Laravel

Laravel merupakan framework PHP yang terkenal dan sering dimanfaatkan dalam membangun aplikasi web masa kini. Framework ini dirancang untuk menyederhanakan proses pengembangan, menawarkan berbagai fitur bawaan yang mendukung efisiensi dan skalabilitas dalam membangun aplikasi web, termasuk yang bersifat kompleks [31]. Kerangka kerja ini menyediakan struktur kode yang terorganisir dengan baik, memungkinkan pengembang untuk mengelola logika bisnis, tugas, dan data. Sintaks *Laravel* yang elegan dan dokumentasi yang komprehensif memudahkan pengembang untuk membuat aplikasi yang skalabel, aman, dan efisien. Laravel didesain untuk menyederhanakan proses pembuatan website atau aplikasi, dengan menyediakan berbagai fitur dan tools yang mendukung pengembang dalam membangun aplikasi yang efisien, aman, serta mudah untuk dipelihara [32].

Salah satu keunggulan utama *Laravel* adalah strukturnya yang terorganisasi dengan baik, yang mendukung arsitektur berbasis *Model-View-Controller (MVC)*. Arsitektur ini membagi logika aplikasi ke dalam tiga bagian utama, yaitu Model yang bertanggung jawab atas pengelolaan data, View yang menyajikan tampilan antarmuka pengguna, dan Controller yang berperan sebagai penghubung antara keduanya. Melalui pendekatan ini, proses pembuatan aplikasi menjadi lebih terstruktur, memungkinkan tim untuk mengelola kode secara lebih mudah dan meningkatkan kolaborasi antar pengembang [33].

Arsitektur *Laravel* didasarkan pada sejumlah fitur unik, termasuk *routing*, *Eloquent ORM*, *Blade Template Engine*, dan sistem otentikasi. Perutean *Laravel* memungkinkan navigasi aplikasi yang fleksibel, sedangkan *Eloquent ORM* menyederhanakan manajemen database melalui pemetaan berbasis objek. *Blade Template Engine* memberikan kemampuan untuk menciptakan antarmuka dinamis bagi pengguna tanpa mengurangi kinerja. *Laravel* juga mendukung pengembangan cepat melalui *Artisan CLI*, yang merupakan antarmuka baris perintah yang memungkinkan pengembang membuat komponen.

Dalam konteks pengembangan *dashboard*, *Laravel* menawarkan fleksibilitas tinggi untuk mengintegrasikan berbagai komponen, seperti pengelolaan data *real-time*,

tampilan grafik, dan fitur interaktif lainnya. Dengan memanfaatkan fitur bawaan *Laravel*, pengembang dapat fokus pada pengembangan logika bisnis dan fungsionalitas utama aplikasi, sementara *framework* ini menangani aspek teknis seperti pengelolaan sesi, keamanan, dan validasi formular.

Keunggulan lainnya adalah *Laravel* memungkinkan pengembangan aplikasi skala besar dengan cepat, bahkan bagi tim dengan pengalaman terbatas, serta lebih mudah digunakan dibandingkan *framework* PHP populer lainnya, sehingga cocok bagi pemula yang ingin memulai pengembangan [34]. Oleh karena itu, *Laravel* menjadi pilihan yang pas untuk pengembangan penelitian aplikasi web, termasuk *dashboard* yang membutuhkan integrasi data *real-time* dan antarmuka pengguna yang interaktif.

2.9 Figma

Figma merupakan platform desain berbasis web yang dirancang untuk mendukung proses pembuatan dan pengujian antarmuka pengguna secara langsung. Platform ini memungkinkan kolaborasi langsung antara desainer dan pengembang dalam satu proyek secara bersamaan, sehingga mendorong peningkatan efisiensi serta produktivitas tim [35]. Figma mempermudah proses desain dan pembuatan *prototype*, sehingga tim dapat lebih cepat mengembangkan dan menguji antarmuka sebelum implementasi [36]. Fitur-fitur unggulan Figma, seperti komponen yang dapat digunakan kembali dan sistem *grid* yang fleksibel, memungkinkan desainer menciptakan antarmuka yang konsisten dan responsif, fitur ini membantu menciptakan desain yang lebih terstruktur dan mendukung pengalaman pengguna yang lebih baik. Selain itu, Figma memfasilitasi kolaborasi antaranggota tim, yang dapat meningkatkan komunikasi dan menghasilkan produk akhir yang lebih optimal.

Figma memiliki berbagai keunggulan, seperti kemampuan kolaborasi dan kerja secara *real-time*, kemudahan dalam berbagi file, fitur desain yang lengkap dan serbaguna, berbasis cloud tanpa perlu instalasi, serta dukungan banyak plugin tambahan dengan paket gratis yang tersedia. Namun, Figma juga memiliki beberapa kelemahan, antara lain tidak dapat digunakan tanpa koneksi internet dan memerlukan kapasitas RAM serta *graphic card* yang cukup tinggi untuk berjalan

dengan optimal [37]. Keterlibatan pengguna dalam proses desain dapat meningkatkan kepuasan dan pengalaman pengguna, sejalan dengan manfaat yang ditawarkan oleh Figma[38].

Pada Figma terdapat plugin yang dapat membantu developer dalam mengubah desain menjadi html dan css, yaitu menggunakan plugin anima. Secara keseluruhan, Figma adalah alat yang sangat bermanfaat untuk mendukung pengembangan antarmuka pengguna, terutama dalam penelitian seperti pengembangan *dashboard* sistem informasi perpustakaan. Dengan fitur-fitur yang memudahkan desain, prototipe, kolaborasi, dan pengujian, Figma membantu tim menghasilkan desain yang berkualitas tinggi dan sesuai kebutuhan pengguna.

2.10 Plugin Anima

Anima merupakan plugin yang dapat diintegrasikan dengan berbagai perangkat lunak desain populer seperti Figma, Adobe XD, dan Sketch. Plugin ini dirancang untuk membantu desainer dalam membuat prototipe interaktif dengan lebih mudah dan efisien. Melalui Anima, pengguna dapat mengatur efek saat kursor diarahkan ke elemen, menambahkan animasi saat halaman diakses, serta bekerja dengan berbagai media seperti video, GIF, dan animasi eksternal. Selain itu, Anima juga mendukung interaksi dengan bidang teks, tombol, navigasi halaman, dan elemen lain yang melibatkan kode eksternal. Salah satu keunggulan Anima adalah kemampuannya dalam membantu pembuatan desain responsif, cukup dengan menyiapkan desain dalam berbagai ukuran layar lalu memilih opsi *responsive*. Anima juga memungkinkan pengguna untuk menghasilkan kode frontend secara otomatis, seperti React, HTML, dan Vue, dengan cara menyinkronkan desain ke dalam aplikasi Anima itu sendiri [39]. Dengan berbagai fitur tersebut, Anima sangat mendukung kolaborasi antara tim desain dan pengembang dalam mempercepat proses pengembangan antarmuka pengguna.

2.11 Bootstrap

Bootstrap merupakan sebuah kerangka kerja CSS yang dikembangkan untuk mempermudah proses perancangan antarmuka situs web yang bersifat responsif dan memiliki tampilan modern. Dengan menyediakan beragam komponen antarmuka seperti elemen tipografi, tombol, navigasi, dan komponen lainnya, bootstrap membantu pengembang dalam menciptakan antarmuka yang menarik dan fungsional tanpa perlu menulis kode dari awal. Selain itu, Bootstrap juga dilengkapi dengan dukungan JavaScript yang memungkinkan implementasi fitur interaktif dengan lebih stabil dan efisien. Fleksibilitas dan kemudahan penggunaannya menjadikan Bootstrap sebagai salah satu *framework* yang sering digunakan dalam pengembangan web, terutama bagi pengembang yang ingin mempercepat proses desain dengan tetap mempertahankan kualitas dan konsistensi tampilan [40].

2.12 Trello

Trello adalah alat manajemen proyek berbasis web yang menggunakan sistem *Kanban* untuk membantu tim mengatur tugas dan proyek secara *visual*. Trello digunakan sebagai alat perencanaan, pencatatan aktivitas, dan pemantauan perkembangan pekerjaan stakeholder dalam proses pengembangan sistem [41]. Trello memungkinkan penggunanya untuk mengelola proyek melalui pembuatan papan (board), daftar (list), dan kartu (card) yang merepresentasikan berbagai tugas yang perlu diselesaikan. Setiap *board* berfungsi sebagai proyek atau sistem kerja yang dikelola, sementara *list* digunakan untuk mengategorikan tugas dalam berbagai tahap, seperti *To do*, *In-progress*, dan *Done*. Di dalam *list*, terdapat *cards* yang mewakili tugas spesifik dan dapat diberi label, tenggat waktu, serta daftar Periksa. Pengguna dapat dengan mudah memindahkan tugas antar daftar menggunakan fitur *drag and drop* sesuai dengan perkembangan proyek. Selain itu, Trello mendukung kolaborasi tim secara *real-time*, memungkinkan anggota tim untuk memberikan komentar, melampirkan dokumen, dan menambahkan catatan pada setiap tugas.

Trello juga dapat diintegrasikan dengan berbagai alat produktivitas lainnya, seperti Google Drive, Slack, dan Jira. Dengan antarmuka yang intuitif dan fleksibel, Trello

mempermudah pengelolaan tugas serta meningkatkan efisiensi dalam berbagai bidang, termasuk pengembangan perangkat lunak berbasis *agile Kanban*.

2.13 Electronic Library (eLib)

Electronic Library (eLib) atau biasa disebut Perpustakaan digital adalah layanan informasi yang menyediakan seluruh sumber daya informasinya dalam format yang dapat diolah oleh komputer, dengan seluruh proses pencarian, penyimpanan, pengelolaan, pengambilan, akses, dan penyajian informasi dilakukan melalui pemanfaatan teknologi digital [2]. eLib memberikan kemudahan bagi pengguna untuk mengakses, menelusuri, dan membaca berbagai sumber informasi secara fleksibel, tanpa perlu hadir langsung di perpustakaan [42]. Sebelum adanya Electronic Library (eLib), proses pencarian dan pemanfaatan sumber informasi di perpustakaan masih bersifat manual dan terbatas pada koleksi fisik, seperti buku tercetak, jurnal cetak, dan arsip yang disimpan di rak perpustakaan. Hal ini sering kali menyulitkan pengguna, terutama dalam hal ketersediaan referensi, efisiensi waktu pencarian, dan keterbatasan akses di luar jam operasional perpustakaan. Namun, setelah adanya eLib, akses terhadap sumber informasi menjadi jauh lebih mudah, cepat, dan fleksibel. Selain menyediakan berbagai koleksi elektronik, eLib juga menyimpan data penting seperti metadata dokumen, file digital, serta log aktivitas pengguna, yang dikelola dalam sistem basis data terstruktur untuk memudahkan pencarian dan pengelolaan informasi. Dengan demikian, eLib tidak hanya memperluas akses terhadap literatur ilmiah, tetapi juga meningkatkan efisiensi layanan perpustakaan dan mendukung proses pembelajaran serta penelitian di era digital.

2.14 Integrated Library System Lite (INLISLite)

INLISLite (Integrated Library System Lite) adalah perangkat lunak open source yang dikembangkan secara resmi oleh Perpustakaan Nasional Republik Indonesia untuk mendorong penerapan teknologi informasi dalam pengelolaan dan layanan perpustakaan di seluruh Indonesia [3]. Sejak pertama kali diluncurkan pada tahun 2011, INLISLite menjadi sistem otomasi standar yang diwajibkan digunakan oleh perpustakaan umum tingkat provinsi dan kabupaten/kota sebagai bagian dari

kebijakan pembinaan perpustakaan nasional, dalam rangka mendukung implementasi tersebut, Perpustakaan Nasional juga mendistribusikan perangkat server ke perpustakaan mitra daerah [43]. Seiring berjalannya waktu, INLISLite terus dikembangkan dan disempurnakan hingga kini mencapai versi 3.1, dengan fitur-fitur yang lebih komprehensif untuk mendukung kegiatan perpustakaan, mulai dari manajemen koleksi, layanan sirkulasi, hingga pelaporan. Kehadiran INLISLite sangat membantu dalam mewujudkan layanan perpustakaan yang terintegrasi, efisien, dan berbasis digital, selaras dengan kebutuhan era informasi saat ini.

2.15 PHP

PHP (Hypertext Preprocessor) merupakan bahasa pemrograman untuk pengembangan web yang beroperasi di sisi server, sehingga seluruh proses eksekusi kodenya dilakukan di server web [44]. PHP mampu menangani pemrosesan data di sisi server untuk kemudian menghasilkan halaman web yang interaktif dan responsif sesuai permintaan pengguna. Kelebihan PHP terletak pada fleksibilitas yang dapat diintegrasikan dengan berbagai sistem manajemen basis data seperti MySQL, PostgreSQL, hingga Oracle, sehingga memudahkan pengembangan dalam membangun aplikasi web yang terhubung dengan data. PHP juga mendukung berbagai framework modern seperti Laravel, CodeIgniter, dan Symfony yang semakin mempercepat proses pengembangan aplikasi web berbasis PHP. Selain itu, sintaks PHP yang relatif mudah dipahami, dokumentasi yang melimpah, dan dukungan komunitas yang besar menjadikan PHP terus digunakan oleh berbagai kalangan, mulai dari pengembang pemula hingga profesional.

2.16 MySQL

MySQL merupakan program server basis data yang dapat memproses peberimaan dan pengiriman data dengan kecepatan tinggi, mendukung penggunaan oleh banyak pengguna sekaligus, dan memanfaatkan perintah standar SQL (Structured Query Language) [45]. MySQL memiliki kapabilitas yang mendukung produktivitas developer, baik yang sudah berpengalaman mengelola basis data maupun yang masih pemula, dengan memanfaatkan bahasa SQL sebagai media aksesnya [44]. Keunggulan lain dari MySQL adalah kemudahan dalam instalasi, dokumen yang

lengkap, serta komunitas pengguna yang luas, sehingga mempermudah pengembangan dan pemeliharaan aplikasi. Dengan kemampuannya dalam menangani transaksi data yang besar secara efisien, MySQL menjadi salah satu komponen penting dalam implementasi sistem informasi.

2.17 Laragon

Laragon merupakan sebuah software portable yang digunakan untuk membuat local server di komputer pengembang. Dengan Laragon, developer dapat menyiapkan lingkungan pengembangan web secara praktis tanpa perlu konfigurasi rumit. Laragon biasanya digunakan untuk menjalankan website berbasis PHP, MySQL, atau framework Laravel, dan didalamnya sudah tersedia tempat penyimpanan database, web server Apache. Keunggulan Laragon yaitu instalasinya mudah, ringan dijalankan, cepat, kemudahan pengaturan, dan cukup powerful, sehingga sangat membantu developer dalam membangun dan menguji aplikasi web secara lokal [46].

2.18 Gitlab

Gitlab adalah sebuah platform version control berbasis web yang digunakan untuk mengelola dan mengatur kode sumber (source code) dalam pengembangan perangkat lunak. Gitlab memanfaatkan sistem kontrol versi Git sehingga setiap perubahan kode dapat dicatat dengan rapi, memungkinkan pengembangan untuk melacak riwayat pengembangan secara detail. Selain sebagai repositori kode, Gitlab juga dilengkapi berbagai fitur pendukung, seperti manajemen issue untuk mencatat tugas atau perbaikan, merge request untuk melakukan peninjauan kode (code review), serta Continuous Integration/Continuous Deployment (CI/CD) yang membantu otomatisasi proses pengujian hingga rilis perangkat lunak. Keunggulan Gitlab terletak pada sifatnya yang open source dan dapat diinstal di server mandiri, sehingga organisasi memiliki kontrol penuh terhadap keamanan data dan pengelolaan proyek [47]. Dengan fitur-fitur tersebut, Gitlab mendukung kolaborasi tim pengembang agar proses pengembangan perangkat lunak dapat berjalan lebih terstruktur, efisien, dan terdokumentasi dengan baik.

2.19 Desain Antarmuka Pengguna

Desain antarmuka adalah proses yang bertujuan menciptakan tampilan dan interaksi pengguna dengan sistem secara efektif dan efisien. Perancangan antarmuka yang baik sangat berpengaruh terhadap kemudahan pengguna dalam menavigasi dan memanfaatkan fitur-fitur yang tersedia dalam suatu aplikasi atau sistem. Dalam proses desain antarmuka, kebutuhan pengguna menjadi prioritas utama, terutama dalam pengembangan aplikasi yang ditujukan untuk kelompok pengguna tertentu. Antarmuka yang ramah pengguna dapat mengurangi kesulitan yang dialami oleh pengguna, seperti lansia, dalam berinteraksi dengan teknologi, terutama di situasi di mana interaksi sosial terbatas. Hal ini menunjukkan pentingnya memahami karakteristik demografi pengguna dalam merancang antarmuka yang inklusif [48].

Proses perancangan antarmuka mengintegrasikan keterlibatan pengguna secara berkelanjutan dalam setiap tahapan pengembangan. Teknik evaluasi seperti *Usability Testing* dan *System Usability Scale (SUS)* digunakan untuk mengukur kualitas pengalaman pengguna, yang mencakup aspek efektivitas, efisiensi, serta tingkat kepuasan terhadap sistem [49]. Selain mempertimbangkan kebutuhan pengguna, desain antarmuka yang baik harus responsif terhadap berbagai perangkat dan ukuran layar maka pentingnya fleksibilitas antarmuka untuk memastikan aksesibilitas yang mudah oleh berbagai kelompok pengguna [50]. Desain yang responsif akan meningkatkan pengalaman pengguna lintas perangkat.

2.20 Kebutuhan Sistem

Kebutuhan dalam konteks pengembangan sistem informasi merujuk pada spesifikasi dan harapan pengguna terhadap sistem yang akan dibangun. Pemahaman yang baik terhadap kebutuhan pengguna menjadi kunci untuk menghasilkan sistem yang efektif dan efisien. Analisis kebutuhan sangat penting sebagai langkah awal dalam memastikan bahwa sistem yang dirancang, seperti *dashboard*, dapat memenuhi ekspektasi pengguna dengan optimal [51].

Kebutuhan sistem dalam pengembangan sistem informasi umumnya dikategorikan menjadi dua jenis utama yaitu kebutuhan *funksional* dan kebutuhan *non-fungsional*[52]. Kebutuhan *funksional* mencakup apa yang harus dilakukan oleh sistem, seperti fitur dan layanan yang diharapkan. Sementara itu, kebutuhan *non-fungsional* meliputi aspek performa, keamanan, dan kegunaan sistem. Pengembang

harus memahami kedua jenis kebutuhan ini secara mendalam untuk merancang sistem yang sesuai dengan harapan dan standar yang diinginkan[53]. Keseimbangan antara kebutuhan *funksional* dan *non-fungsional* akan meningkatkan kualitas sistem secara keseluruhan.

Proses pengumpulan kebutuhan dapat dilakukan menggunakan berbagai metode, seperti wawancara, survei, dan observasi. Pemilihan metode pengumpulan kebutuhan yang tepat sangat penting untuk memastikan informasi yang diperoleh akurat dan relevan. Metode pengumpulan yang sesuai akan memberikan gambaran yang lebih jelas mengenai kebutuhan pengguna [54]. Oleh karena itu, tim pengembang harus mempertimbangkan konteks dan karakteristik pengguna dalam menentukan metode yang akan digunakan. Setelah kebutuhan terkumpul, langkah berikutnya adalah menganalisis dan mendokumentasikan kebutuhan tersebut.

Analisis kebutuhan juga berperan dalam mengidentifikasi potensi masalah dan risiko yang mungkin muncul selama proses pengembangan. Analisis yang komprehensif dapat membantu mengurangi risiko perubahan desain yang signifikan di kemudian hari, yang seringkali memakan waktu dan biaya. Dengan memahami kebutuhan pengguna secara menyeluruh, tim pengembang dapat merancang sistem yang lebih stabil dan sesuai sejak awal pengembangan.

Secara keseluruhan, kebutuhan pengguna merupakan elemen kunci dalam pengembangan sistem informasi. Pemahaman yang baik terhadap kebutuhan ini tidak hanya mendukung keberhasilan penelitian tetapi juga memastikan bahwa sistem yang dihasilkan benar-benar bermanfaat bagi pengguna [55]. Melalui analisis kebutuhan yang matang, tim pengembang dapat menciptakan sistem yang relevan, responsif, dan berkelanjutan.

2.21 Diagram UML (Unified Modeling Language)

Diagram adalah alat *visual* yang digunakan untuk memodelkan sistem, proses, atau interaksi antara berbagai komponen dalam sistem. Dalam pengembangan perangkat lunak, diagram memiliki peran penting dalam menggambarkan kebutuhan dan fungsionalitas sistem secara jelas dan terstruktur. Diagram UML merupakan seperangkat standar teknik diagram yang menyediakan representasi visual yang

komprehensif untuk memodelkan seluruh tahapan pengembangan sistem dalam suatu proyek, mulai dari tahap analisis hingga ke tahap implementasi [56].

Penggunaan diagram membantu tim pengembang memahami sistem yang akan dibangun, serta mempermudah komunikasi antara pengembang dan pemangku kepentingan. Dalam proses pengembangan perangkat lunak, terdapat berbagai jenis diagram UML yang umum digunakan untuk memodelkan sistem, antara lain diagram *Use Case*, diagram *Activity*, *Entity-Relationship Diagram (ERD)*, dan diagram *Sequence*. Diagram-diagram ini berfungsi untuk merepresentasikan alur kerja sistem, relasi antar entitas, serta interaksi antar komponen dalam sistem secara visual dan terstruktur.

1. *Use Case* Diagram

Diagram *Use Case* menggambarkan secara visual bagaimana pengguna atau aktor berinteraksi dengan sebuah sistem. Diagram ini memiliki peran penting dalam mendokumentasikan dan menyusun kebutuhan sistem secara jelas, sehingga pengembang dapat lebih mudah memahami fungsionalitas yang diinginkan oleh pengguna [57]. *Use Case* diagram juga sangat penting dalam menggambarkan berbagai skenario penggunaan sistem yang diperlukan, terutama pada tahap analisis kebutuhan. Dengan adanya diagram ini, tim pengembang dapat dengan mudah mengidentifikasi dan mendefinisikan batasan sistem serta beragam interaksi yang terjadi di dalam sistem. Hal ini memudahkan pengembang untuk menyusun dan merencanakan pengembangan sistem secara lebih terstruktur dan terarah.

2. *Activity* Diagram

Diagram aktivitas menampilkan secara visual urutan alur kerja atau proses yang berlangsung di dalam sistem. *Activity* diagram digunakan untuk memodelkan interaksi dalam sistem dan untuk menunjukkan langkah-langkah yang terlibat dalam suatu proses atau aktivitas. Diagram ini sangat berguna dalam menggambarkan proses-proses yang kompleks dan memberikan panduan yang jelas untuk analisis serta perancangan sistem secara keseluruhan. Dengan adanya *Activity* diagram, pengembang dapat dengan mudah mengidentifikasi langkah-langkah yang diperlukan untuk mencapai tujuan atau hasil tertentu dalam sistem, sehingga

mempercepat proses pengembangan dan meningkatkan efisiensi dalam perancangan sistem [58].

3. *Entity-Relationship Diagram*

Entity-Relationship Diagram (ERD) merupakan representasi visual yang digunakan untuk memodelkan struktur data dalam suatu sistem melalui penggambaran entitas, atribut, serta relasi antar entitas. *ERD* membantu dalam memahami bagaimana data saling berhubungan dalam basis data, sehingga memudahkan pengembang dan pemangku kepentingan dalam merancang serta mengelola sistem informasi.

ERD berfungsi untuk merepresentasikan setiap entitas dan hubungan antar entitas dalam bentuk notasi grafis yang kemudian dikonversi menjadi diagram data. Dengan demikian, seluruh proses pengolahan data secara transaksional dapat divisualisasikan dengan lebih jelas dan terstruktur [59]. Dengan menggunakan *ERD*, pengembang dapat mengidentifikasi kebutuhan data, menyusun skema database yang lebih efisien, serta meningkatkan integritas dan konsistensi data dalam sistem. Selain itu, *ERD* juga membantu dalam dokumentasi dan komunikasi antar anggota tim pengembang, sehingga proses pengembangan sistem menjadi lebih terstruktur dan efektif.

4. *Sequence Diagram*

Sequence Diagram merupakan diagram yang menunjukkan bagaimana objek-objek dalam sebuah sistem saling berinteraksi sesuai dengan urutan waktu terjadinya. *Sequence diagram* memiliki peran penting dalam memodelkan perilaku dinamis suatu sistem, dengan menyajikan urutan interaksi antar objek yang terjadi seiring berjalannya waktu [60]. Diagram ini membantu pengembang untuk memahami urutan pesan atau perintah yang dikirim antar objek, sehingga memudahkan proses perancangan dan pengembangan sistem. Dengan menggunakan *sequence diagram*, tim pengembang dapat mengidentifikasi interaksi-interaksi penting yang perlu diperhatikan dalam implementasi sistem, serta memastikan bahwa alur komunikasi antara objek berjalan dengan benar sesuai dengan yang diharapkan dalam implementasi sistem.

2.22 Black-box testing

Black box testing merupakan metode pengujian perangkat lunak yang berfokus pada evaluasi fungsionalitas sistem tanpa memperhatikan struktur internal atau kode program yang digunakan. Pengujian *black-box* dilakukan dengan berfokus pada aspek-aspek eksternal aplikasi, seperti antarmuka, fungsionalitas, dan kesamaan fungsinya dengan sistem kerja yang hendak digunakan [61]. Sebagai metode pengujian berbasis fungsi, *Black-box testing* sering digunakan untuk mengidentifikasi masalah yang berkaitan dengan masalah logis, kegagalan *fungsional*, atau antarmuka yang tidak memuaskan. Pengujian jenis ini dapat digunakan pada berbagai tahapan dalam proses pengembangan perangkat lunak, termasuk pada tahap pengujian unit, integrasi, sistem, hingga tahap uji penerimaan.

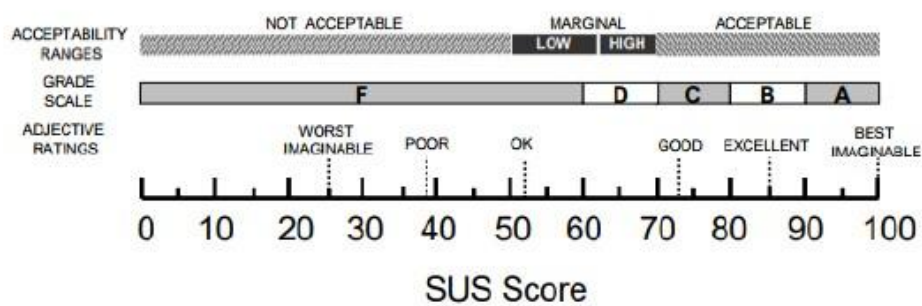
Kelebihan dari metode *black-box testing* adalah kemampuannya untuk mengidentifikasi kesalahan yang mungkin tidak terdeteksi dalam pengujian internal. Metode ini memungkinkan proses pengujian dilakukan tanpa membutuhkan pengetahuan mendalam tentang kode sumber, sehingga pengujian dapat dilakukan oleh pihak yang tidak secara langsung terlibat dalam kegiatan pengembangan perangkat lunak [62].

2.23 System Usability Scale (SUS)

System Usability Scale (SUS) merupakan metode pengujian sederhana namun efektif untuk menilai seberapa baik suatu sistem, seperti aplikasi, situs web, atau perangkat lunak perangkat, bekerja. Metode pengukuran kegunaan bernama SUS ini dibuat oleh John Brooke pada tahun 1986 [63]. *SUS* merupakan salah satu alat evaluasi kegunaan yang paling banyak digunakan karena fleksibilitas dan kemudahan penggunaannya dalam penelitian. Metode ini membantu pengguna memahami bagaimana sistem memenuhi kebutuhan pengguna dalam hal kenyamanan dan keramahan pengguna. *SUS* terdiri dari 10 pernyataan yang dievaluasi menggunakan skala Likert 5 poin, menghasilkan skor antara 0 dan 100 [64].

System Usability Scale (SUS) mengelompokkan skor kegunaan sistem ke dalam beberapa kategori berdasarkan tingkat penerimaan (*acceptability ranges*), skala

nilai (*grade scale*), dan penilaian deskriptif (*adjective ratings*). Skor SUS di 0-49 dikategorikan sebagai "*Not Acceptable*" atau tidak dapat diterima, sedangkan skor antara 50-59 masuk dalam kategori "*Marginal*" tingkat rendah (*low*) dan skor antara 60-69 masuk kategori "*Marginal*" tinggi (*high*). Sementara itu, skor di atas 70 dianggap "*Acceptable*", menunjukkan bahwa sistem memiliki kegunaan yang baik. Dengan demikian, skor *SUS* dapat memberikan gambaran yang jelas mengenai tingkat kegunaan suatu sistem, di mana skor tinggi menunjukkan pengalaman pengguna yang lebih baik, sedangkan skor rendah mengindikasikan perlunya perbaikan signifikan dalam aspek *usability*. Adapun penjelasan skor *System Usability Scale (SUS)* disajikan pada gambar 2.3 sebagai berikut:



Gambar 2.3 Skor *System Usability Scale (SUS)*[65]

2.24 Penelitian Terdahulu

Penelitian terdahulu dilakukan untuk memberikan wawasan yang lebih mendalam mengenai pengembangan sistem yang digunakan. Dengan meninjau berbagai penelitian sebelumnya, dapat dianalisis tujuan penelitian, metode yang digunakan, serta hasil yang diperoleh, sehingga relevansi dan kontribusinya terhadap penelitian ini dapat lebih dipahami dengan baik.

Pada penelitian yang dilakukan oleh Diebold et al. (2019) berjudul "*Stepwise Transition to Agile: From Three Agile Practices to Kanban Adaptation*" bertujuan untuk menganalisis bagaimana perusahaan kecil dan menengah (*Small and Medium Enterprises* atau SME) dapat bertransisi secara bertahap ke metode *Agile*, khususnya dalam mengadopsi *Kanban*, guna meningkatkan efisiensi manajemen proyek. Studi ini berfokus pada perusahaan Jerman, TQsoft GmbH, yang

sebelumnya telah menerapkan beberapa praktik *Agile*, seperti *User Stories*, *Definition of Ready*, dan *Definition of Done*, sebelum memperluas penerapannya ke *Kanban Board* dan sistem *pull*. Metode yang digunakan dalam penelitian ini adalah transisi *Agile* bertahap (*evolutionary transition*), dengan pendekatan studi kasus yang melibatkan wawancara kualitatif terhadap karyawan untuk mengevaluasi dampak dari perubahan tersebut. Hasil penelitian menunjukkan bahwa pendekatan bertahap ini mampu meningkatkan transparansi proyek, efektivitas komunikasi tim, serta efisiensi dalam pengelolaan tugas tanpa mengganggu operasional bisnis. Penelitian ini relevan untuk pengembangan sistem yang menggunakan metode *Agile Kanban*, karena membuktikan efektivitas *Kanban* dalam mengelola *backlog*, tugas, dan pemantauan progres dalam pengembangan perangkat lunak [66] .

Selanjutnya penelitian yang dilakukan oleh Wael Zayat dan Ozlem Senvar (2019) berjudul "*Framework Study for Agile Software Development Via Scrum and Kanban*". Tujuan penelitian ini adalah membandingkan metodologi *Agile Scrum* dan *Kanban* dengan meninjau kelebihan serta kekurangannya dalam berbagai lingkungan kerja. Metode yang digunakan adalah studi literatur dan survei kepada praktisi *Agile* untuk memahami pengalaman mereka dalam menerapkan kedua metodologi tersebut. Hasil penelitian menunjukkan bahwa *Scrum* lebih efektif untuk proyek yang memiliki tujuan jelas dan membutuhkan perencanaan mendetail, sementara *Kanban* lebih sesuai untuk lingkungan kerja yang membutuhkan fleksibilitas tinggi dan alur kerja yang berkelanjutan. Penelitian ini menyimpulkan bahwa pemilihan antara *Scrum* dan *Kanban* bergantung pada kompleksitas proyek dan tingkat perubahan yang dihadapi. Metode yang digunakan dalam penelitian ini relevan sebagai acuan untuk pengembangan sistem yang menggunakan *Agile Kanban*, terutama dalam memahami keunggulan dan keterbatasannya dibandingkan dengan metode *Agile* lainnya. Studi ini memberikan wawasan bahwa *Kanban* adalah pilihan yang tepat untuk sistem yang membutuhkan fleksibilitas dalam pengelolaan tugas serta peningkatan berkelanjutan [7].

Lebih lanjut, penelitian yang dilakukan oleh Saltz dan Heckman (2020) yang berjudul "*Exploring Which Agile Principles Students Internalize When Using a Kanban Process Methodology*." Penelitian mengenai metode *Agile Kanban* dalam pengelola mahasiswa saat menggunakan metode *Kanban* dalam proyek berbasis

tim. Metode penelitian yang digunakan adalah studi kasus dengan analisis berdasarkan prinsip-prinsip *Agile* yang terdapat dalam *Agile Manifesto*. Data dikumpulkan melalui survei terbuka kepada mahasiswa yang telah menggunakan *Kanban* dalam proyek mereka. Hasil penelitian menunjukkan bahwa dua prinsip *Agile* yang paling banyak diinternalisasi adalah *self-organizing teams* dan *reflection at regular intervals*, yang meningkatkan komunikasi serta koordinasi tim. Penelitian ini menunjukkan bahwa metode *Kanban* efektif dalam membantu tim proyek mengelola tugas dan meningkatkan pemahaman terhadap konsep *Agile*. Oleh karena itu, metode ini dapat dijadikan acuan dalam penelitian pengembangan sistem berbasis *Agile Kanban*, termasuk dalam proyek skripsi ini yang menerapkan *framework* serupa untuk mengelola dan memantau perkembangan sistem [6].

Kajian lain yang relevan dilakukan oleh Mazni Omar, Hamzah Alaidaros, dan Rohaida Romli (2020) berjudul "*An Improved Software Project Monitoring Task Model of Agile Kanban Method: A Practitioners' Perspective*" bertujuan untuk meningkatkan pemantauan proyek perangkat lunak dalam metode *Agile Kanban* melalui pengembangan model *i-KAM (Improved Kanban Monitoring)*. Metode penelitian menggunakan *Expert Review* dengan enam praktisi untuk mengevaluasi model berdasarkan lima dimensi utama. Hasil penelitian menunjukkan bahwa *i-KAM* dapat meningkatkan tracking progres, mengoptimalkan batas *Work-in-Progress (WIP)*, dan memberikan wawasan yang lebih bermanfaat terkait *workflow proyek*. Model ini dinilai relevan dan layak diterapkan dalam pengelolaan proyek berbasis *Agile Kanban*, sehingga dapat menjadi referensi dalam penelitian skripsi ini untuk meningkatkan transparansi dan efisiensi dalam pengembangan sistem .

Studi lain yang membahas metode *kanban* yaitu oleh penelitian Azizah Rahmat dan Nur Aisyah Mohd Hanifah (2020) berjudul "*Usability Testing in Kanban Agile Process for Club Management System*" bertujuan untuk mengembangkan sistem manajemen klub berbasis web yang memfasilitasi komunikasi dan pengelolaan keanggotaan di lingkungan universitas. Sistem ini dikembangkan menggunakan metode *Agile Kanban* sebagai bagian dari pendekatan Lean untuk meningkatkan efisiensi dan transparansi dalam proses pengembangan. *Framework* yang digunakan dalam pengembangan sistem adalah *Laravel PHP*. Pengujian dilakukan dengan *functional testing* dan *usability testing* untuk memvalidasi kebutuhan

pengguna. Hasil penelitian menunjukkan bahwa sistem yang dikembangkan memenuhi lima faktor kegunaan (*usability factors*), yang membuktikan bahwa pendekatan *Kanban* efektif dalam mengelola proyek pengembangan perangkat lunak berbasis web [67].

Dalam penelitian lainnya, ditemukan bahwa penelitian yang dilakukan oleh Eric Weflen, Cameron A. MacKenzie, dan Iris V. Rivero (2021) yang berjudul "*An Influence Diagram Approach to Automating Lead Time Estimation in Agile Kanban Project Management*". Penelitian ini bertujuan untuk mengembangkan metode estimasi lead time dalam proyek *Agile Kanban* menggunakan *influence diagram* dan Bayesian belief network, guna membantu tim proyek dalam mengelola *backlog* yang dinamis dan memprediksi durasi penyelesaian tugas secara lebih akurat. Pengujian dilakukan melalui analisis sensitivitas dan evaluasi data dari tim *Kanban* profesional. Hasil penelitian menunjukkan bahwa faktor utama yang memengaruhi estimasi *lead time* adalah kecepatan tim dan ukuran rata-rata tugas, serta bahwa metode berbasis *influence diagram* dapat memberikan perkiraan yang lebih realistis dibandingkan dengan metode tradisional seperti *story points* dan *velocity*. Penelitian ini lebih berfokus pada aspek estimasi waktu dalam *Agile Kanban*, pendekatan yang digunakan tetap relevan sebagai acuan dalam pengelolaan *backlog* dan perencanaan tugas pada proyek berbasis *Agile*, termasuk dalam pengembangan sistem berbasis web yang memerlukan fleksibilitas tinggi [5].

Selain penelitian diatas, terdapat penelitian yang juga relevan yaitu penelitian oleh Rudi Hartono (2022) yang berjudul "Penerapan *Kanban* Model Sebagai Metode Perancangan Sistem Informasi (Studi Kasus: Pemetaan Sekolah SMA/K/MA Kota Tasikmalaya)" bertujuan untuk menerapkan model *Kanban* dalam perancangan sistem informasi berbasis GIS untuk pemetaan sekolah di Tasikmalaya. Metode yang digunakan dalam penelitian ini adalah *Kanban*, yang memungkinkan visualisasi alur kerja dan pemantauan progres proyek secara lebih terstruktur. Proses desain sistem menggunakan pemodelan *UML* untuk menggambarkan hubungan antar entitas dan perancangan *user interface*. Hasil penelitian menunjukkan bahwa penerapan metode *Kanban* dalam perancangan sistem informasi mampu meningkatkan efisiensi proses pengembangan, memberikan hasil yang lebih cepat dengan kualitas tinggi, serta mengoptimalkan manajemen proyek.

Model *Kanban* terbukti menjadi alternatif yang efektif dalam pengembangan sistem berbasis web, sehingga relevan sebagai referensi dalam penelitian ini yang juga menerapkan metode *Agile Kanban* untuk pengelolaan proyek pengembangan *dashboard* monitoring perpustakaan.

Adapun penelitian lain yang mendukung adalah penelitian oleh Alton Gunawan Purwanto et al. (2022) berjudul "*Website System Design Using Agile Kanban Based On QR Code*" bertujuan untuk mengembangkan sistem pemesanan restoran berbasis web menggunakan *QR Code* guna meningkatkan efisiensi layanan. Pengembangan sistem dilakukan menggunakan metode *Agile Kanban*, yang memungkinkan pemantauan progres kerja secara transparan melalui *kanban board*. Sistem ini dikembangkan dengan PHP dan MySQL serta dimodelkan menggunakan *UML*. Pengujian dilakukan dengan *Black-box testing* untuk memastikan fungsi sistem berjalan dengan baik. Hasil penelitian menunjukkan bahwa sistem yang dibangun mampu meningkatkan efisiensi pemesanan, mengurangi penggunaan kertas, dan mempercepat layanan restoran. Metode *Agile Kanban* terbukti efektif dalam pengelolaan proyek pengembangan web.

Selain itu, penelitian yang dilakukan oleh Baginda Oloan Lubis dkk. (2023) juga membahas metode *kanban* dalam jurnal berjudul "*Implementasi Metode Kanban pada Rancangan Sistem Informasi HelpDesk Pada Kantor Imigrasi Berbasis Website*" bertujuan untuk mengembangkan sistem informasi berbasis web yang dapat mengelola dan menangani keluhan di kantor imigrasi secara lebih efektif dan efisien. Pengembangan sistem ini menggunakan metode *Kanban* untuk meningkatkan manajemen alur kerja dalam penanganan keluhan serta memastikan bahwa data keluhan dapat diakses dengan lebih mudah di masa mendatang. Metode penelitian yang diterapkan mencakup pengamatan langsung, wawancara, serta studi pustaka, sementara pengujian sistem dilakukan dengan pendekatan *black-box testing*. Hasil penelitian menunjukkan bahwa penerapan metode *Kanban* pada sistem informasi *helpdesk* berhasil meningkatkan efisiensi dalam pengelolaan keluhan, mempercepat proses pencatatan dan penanganan keluhan, serta memberikan transparansi dalam pelacakan setiap laporan yang masuk. Penelitian ini menunjukkan bahwa metode *Kanban* dapat menjadi pendekatan yang relevan dalam sistem berbasis web yang membutuhkan manajemen tugas dan pemantauan

progres secara terstruktur [68]..

Selain penelitian-penelitian tersebut, terdapat pula studi yang dilakukan oleh Winarni (2024), yang relevan dengan membahas metode *agile* yang berjudul "Sistem Informasi Monitoring Kerja Praktek Dan Skripsi (Studi Kasus: STMIK Bandung)." Penelitian ini bertujuan untuk membangun sistem informasi yang dapat memonitor kerja praktek dan skripsi mahasiswa dengan menggunakan *framework Laravel*. Pengembangan sistem dilakukan menggunakan metode *Agile* dengan pendekatan sistem berorientasi objek yang dimodelkan melalui *Unified Modeling Language (UML)*. Pengujian sistem dilakukan menggunakan metode *black-box testing*. Hasil penelitian menunjukkan bahwa sistem informasi yang dibangun mampu berfungsi dengan baik dalam memonitor kegiatan kerja praktek dan skripsi mahasiswa serta memenuhi kebutuhan pengguna. Meskipun penelitian ini tidak secara khusus menggunakan *Kanban*, penerapan metode *Agile* yang fleksibel tetap relevan sebagai acuan dalam pengembangan sistem berbasis web yang membutuhkan pemantauan data secara *real-time*.

Sementara itu, selain membahas metode *Agile Kanban*, penelitian terdahulu juga mengkaji literatur terkait pengembangan *framework Laravel* serta penerapan pengujian *Black-box testing*. Dalam penelitian yang dilakukan oleh Majida Laaziri et al berjudul "A Comparative Study of PHP Frameworks Performance". Penelitian ini bertujuan membandingkan performa *Laravel*, *Symfony*, dan *CodeIgniter* dalam pengembangan aplikasi web. Sistem dikembangkan menggunakan pola desain Model-View-Controller (MVC), dan pengujian dilakukan dengan metode benchmarking menggunakan Apache Benchmark. Hasil penelitian menunjukkan bahwa *Laravel* memiliki performa terbaik, menangani hingga 3000 request per detik, dengan penggunaan memori sekitar 518 KB, dan waktu respons tercepat 4,46 ms. *Laravel* dinilai lebih unggul untuk pengembangan aplikasi skala besar dibandingkan *Symfony* dan *CodeIgniter*. Penelitian ini relevan sebagai referensi dalam proyek sistem dashboard monitoring perpustakaan karena mendukung pemilihan *Laravel* sebagai framework utama yang efisien, cepat, dan terstruktur untuk pengembangan web skala besar.

Lalu penelitian lain oleh Supriatmaja (2024) dengan judul "Pengembangan Sistem

Informasi Monitoring dan Pengarsipan Administrasi Keuangan Menggunakan *Framework Laravel* pada BPS Buleleng." Penelitian ini bertujuan mengembangkan sistem informasi yang dapat membantu monitoring dan pengarsipan administrasi keuangan secara lebih efisien di BPS Buleleng, yang sebelumnya masih dilakukan secara manual. Pengembangan sistem menggunakan metode *Agile development* untuk memastikan fleksibilitas dan adaptasi terhadap kebutuhan pengguna. *Framework Laravel* dipilih karena kemampuannya dalam pengelolaan basis data yang efisien serta arsitektur *Model-View-Controller (MVC)*. Sistem diuji menggunakan metode *Black-box testing*, yang menunjukkan bahwa sistem berfungsi sesuai harapan dan mampu meningkatkan efektivitas pengelolaan administrasi keuangan di BPS Buleleng. Metode *Agile* yang diterapkan dalam penelitian ini menunjukkan hasil yang baik dalam pengembangan sistem berbasis web dan dapat dijadikan referensi untuk penelitian pengembangan sistem *dashboard* monitoring [69].

Penelitian sejenis yang dilakukan oleh Sugiharto (2024) dengan judul "Rancang Bangun Platform Kolaborasi *Creative Hub* dan *Client* Berbasis Web dengan *Agile* dan *Scrum*." Penelitian ini bertujuan untuk merancang platform kolaborasi yang efektif antara *Creative Hub* dan klien menggunakan *framework Laravel*. Evaluasi sistem dilakukan melalui *System Usability Scale (SUS)* dan pengujian *black-box testing*. Hasil evaluasi menunjukkan bahwa platform kolaborasi ini mendapatkan skor SUS sebesar 85,12 yang masuk dalam kategori "*Excellent*." Pengujian black box memastikan bahwa fungsi platform telah sesuai dengan spesifikasi yang dirancang, sehingga platform ini dinilai efektif dalam mengatasi hambatan utama dalam kolaborasi antara *Creative Hub* dan klien. Penelitian ini relevan sebagai referensi dalam pengembangan sistem berbasis *Agile* dengan pengujian menggunakan *Black-box testing* [70].

Penelitian terakhir yang relevan adalah karya Dimas Ibnu Rahmadhani dan Maryam (2024) berjudul "*Pengembangan Sistem Point of Sale Berbasis Web pada Toko Faafoo Menggunakan Framework Laravel*." Penelitian ini bertujuan untuk mengembangkan sistem *point of sale (POS)* berbasis web yang efisien menggunakan *framework Laravel*. Metode *Agile* digunakan untuk pengembangan sistem, dan pengujian dilakukan dengan *black-box testing* serta evaluasi *System*

Usability Scale (SUS). Hasil penelitian menunjukkan bahwa sistem yang dikembangkan berfungsi dengan baik dan mendapat skor SUS 75,5, yang dikategorikan sebagai "*acceptable*". Pendekatan *Agile* dalam penelitian ini relevan sebagai referensi dalam pengembangan sistem berbasis web yang fleksibel dan mudah disesuaikan [71]. Adapun ringkasan penjelasan penelitian terdahulu terdapat pada tabel 2.1 sebagai berikut:

Tabel 2.1 Penelitian Terdahulu

No	Peneliti (Tahun)	Objek Penelitian	Frame work	Metode	Pengujian	Hasil
1	Diebold et al. (2019)	Perusahaan Kecil di Jerman, TQsoft GmbH, yang sedang melakukan transisi bertahap ke metode <i>Agile</i> khususnya dalam mengadopsi <i>Kanban</i> dalam pengembangan perangkat lunak.	-	<i>Agile</i>	Studi Kasus	Menunjukkan bahwa pendekatan bertahap meningkatkan transparansi, komunikasi tim, dan efisiensi tugas tanpa mengganggu operasional bisnis.
2	Wael Zayat dan Ozlem Senvar (2019)	Perbandingan <i>Agile</i> Scrum dan <i>Kanban</i> .	-	Studi literatur	Survei kepada 40 praktisi <i>Agile</i>	Scrum lebih unggul dalam perencanaan dan estimasi waktu, sedangkan <i>Kanban</i> lebih baik dalam pengelolaan tugas yang terus berlangsung dan peningkatan proses secara bertahap.
3	Saltz dan Heckman (2020)	Mahasiswa yang menggunakan metode <i>Agile</i> <i>Kanban</i> dalam proyek berbasis tim di kelas data science.	-	Studi kasus dan survei terbuka kepada mahasiswa	Analisis kualitatif terhadap tanggapan mahasiswa	Ditemukan bahwa prinsip <i>Agile</i> yang paling banyak diinternalisasi mahasiswa adalah self-organizing teams dan reflection at regular intervals. Hal ini menunjukkan

No	Peneliti (Tahun)	Objek Penelitian	Frame work	Metode	Pengujian	Hasil
						bahwa penggunaan <i>Kanban</i> membantu meningkatkan koordinasi dan komunikasi tim, sehingga meningkatkan efektivitas kerja dalam proyek berbasis <i>Agile</i> .
4	Mazni Omar, Hamzah Alaidaros, dan Rohaida Romli (2020)	Pemantauan proyek perangkat lunak dalam metode <i>Agile Kanban</i> melalui pengembangan model <i>i-KAM</i> (Improved <i>Kanban</i> Monitoring)	-	<i>Expert Review</i>	Evaluasi <i>Expert Review Method</i>	Model <i>i-KAM</i> terbukti mampu meningkatkan pemantauan progres proyek dengan memperluas mekanisme pelacakan, mengontrol batas <i>Work-In-Progress (WIP)</i> , serta memberikan wawasan yang lebih baik dalam pengelolaan proyek.
5	Azizah Rahmat dan Nur Aisyah Mohd Hanifah (2020)	Sistem manajemen klub berbasis web untuk universitas.	<i>Laravel</i>	<i>Agile Kanban</i>	<i>functional testing</i> dan <i>usability testing</i>	Sistem yang dikembangkan berhasil memenuhi lima faktor <i>usability</i> , yang membuktikan bahwa pendekatan <i>Kanban</i> efektif dalam pengelolaan proyek pengembangan perangkat lunak berbasis web
6	Eric Weflen, Cameron A. MacKenzie, dan Iris V. Rivero (2021)	Metode estimasi lead time dalam proyek <i>Agile Kanban</i>	-	Metode estimasi lead time dalam proyek <i>Agile Kanban</i>	Analisis sensitivitas dan evaluasi	Faktor utama yang memengaruhi estimasi lead time adalah kecepatan tim dan ukuran rata-rata tugas. Metode <i>influence diagram</i> memberikan hasil lebih akurat dibandingkan story

No	Peneliti (Tahun)	Objek Penelitian	Frame work	Metode	Pengujian	Hasil
						<i>points</i> dan <i>velocity</i> . Penelitian ini fokus pada aspek estimasi waktu dalam <i>Agile Kanban</i> , pendekatan yang digunakan tetap relevan sebagai acuan dalam pengelolaan <i>backlog</i> dan perencanaan tugas pada proyek berbasis <i>Agile</i> .
7	Rudi Hartono (2022)	Pengembangan sistem informasi berbasis GIS untuk pemetaan sekolah SMA/K/MA di Kota Tasikmalaya.	-	<i>Agile Kanban</i>	-	- Implementasi metode <i>Kanban</i> dalam perancangan sistem informasi terbukti memberikan proses desain yang lebih terstruktur. - <i>Kanban Board</i> membantu membatasi iterasi yang sedang dikerjakan sesuai dengan kebutuhan sistem.
8	Alton Gunawan Purwanto et al. (2022)	Sistem pemesanan restoran berbasis web menggunakan <i>QR Code</i> .	PHP dan MySQL	<i>Agile Kanban</i>	<i>Black-box testing</i>	Sistem yang dikembangkan mampu meningkatkan efisiensi dalam pemesanan, mengurangi penggunaan kertas, serta mempercepat layanan restoran. <i>Agile Kanban</i> terbukti efektif dalam mengelola proyek pengembangan web, memberikan transparansi terhadap progres kerja, dan

No	Peneliti (Tahun)	Objek Penelitian	Frame work	Metode	Pengujian	Hasil
						membantu tim dalam manajemen tugas.
9	Baginda Oloan Lubis dkk. (2023)	Sistem Informasi <i>HelpDesk</i> berbasis website untuk pengelolaan keluhan di Kantor Imigrasi.	-	<i>Agile Kanban</i>	<i>Black-box testing</i>	Sistem yang dikembangkan mampu mempercepat pencatatan dan penanganan keluhan, meningkatkan efisiensi pengelolaan data, serta memberikan transparansi dalam pelacakan keluhan. Penerapan metode <i>Kanban</i> terbukti efektif dalam mengelola alur kerja secara terstruktur.
10	Winarni (2024)	Sistem informasi monitoring kerja praktek dan skripsi di STMIK Bandung.	<i>Laravel</i>	<i>Agile development</i>	<i>Black-box testing</i>	Sistem informasi yang dikembangkan mampu memudahkan pemantauan progres kerja praktek dan skripsi mahasiswa, membantu program studi dalam mengambil keputusan terkait penyelesaian kendala yang dihadapi mahasiswa, serta meningkatkan efisiensi dalam proses administrasi akademik.
11	Majida Laaziri et al (2019)	Perbandingan kinerja tiga framework PHP yang paling umum digunakan dalam pengembangan web, yaitu	<i>Laravel</i> , <i>Symfony</i> , dan <i>CodeIgniter</i> .	Metode <i>QSOS</i>	Pengujian performa berbasis <i>Apache Benchmark</i>	<i>Laravel</i> memiliki performa terbaik dalam menangani request per detik dan waktu respons tercepat, sedangkan <i>CodeIgniter</i>

No	Peneliti (Tahun)	Objek Penelitian	Frame work	Metode	Pengujian	Hasil
		Laravel, Symfony, dan CodeIgniter.				menggunakan jumlah file paling sedikit. Symfony memiliki penggunaan memori tertinggi dibandingkan dua framework lainnya.
12	Supriatmaja (2024)	sistem informasi monitoring dan pengarsipan administrasi keuangan di Badan Pusat Statistik (BPS) Buleleng.	Laravel	Agile development	Black-box testing	Penelitian menunjukkan bahwa sistem yang dikembangkan mampu meningkatkan efisiensi dan efektivitas dalam pengelolaan administrasi keuangan, memfasilitasi kolaborasi antarpegawai, serta memungkinkan monitoring administrasi secara <i>real-time</i>
13	Sugiharto (2024)	Pengembangan platform berbasis web bernama Kolab Aja	Laravel	Agile Scrum	SUS dan Black-box testing	Platform Kolab Aja mampu mengatasi hambatan utama dalam industri kreatif, seperti inkonsistensi proyek dan ekspektasi client, serta memperkuat peran manajerial dalam ekosistem Creative Hub. Dengan Pengujian menggunakan <i>Black-box testing</i> dan SUS memastikan bahwa platform berfungsi sesuai dengan

No	Peneliti (Tahun)	Objek Penelitian	Frame work	Metode	Pengujian	Hasil
						spesifikasi yang telah dirancang, sehingga dinilai efektif dalam mengatasi hambatan utama dalam kolaborasi antara <i>Creative Hub</i> dan klien
14	Dimas Ibnu Rahmadhani dan Maryam (2024)	Sistem <i>Point of Sale (POS)</i> berbasis web untuk toko Faafoo.	<i>Laravel</i>	<i>Agile development</i>	SUS dan <i>Black-box testing</i>	Sistem POS berbasis web ini berhasil meningkatkan efisiensi transaksi, mengurangi kesalahan dalam perhitungan pembayaran, serta membantu dalam pengelolaan inventaris toko

Penelitian-penelitian ini menunjukkan kesamaan dalam penggunaan Metode *Agile Kanban* dan *Framework Laravel* sebagai pendekatan pengembangan yang diuji menggunakan *Black-box testing* dan *System Usability Scale (SUS)*, yang memungkinkan fleksibilitas dan efisiensi dalam adaptasi terhadap perubahan kebutuhan. Dengan mengintegrasikan hasil penelitian terdahulu ini, penelitian yang sedang dilakukan diharapkan dapat memberikan kontribusi pada penerapan metode *Agile Kanban* dalam pengembangan *dashboard* monitoring berbasis *Laravel*, dengan pengujian *Black-box testing* dan *SUS*, serta mengeksplorasi tantangan dan praktik terbaik yang dapat diimplementasikan.

III. METODOLOGI PENELITIAN

3.1 Waktu dan Tempat

Penelitian ini dilaksanakan pada waktu dan tempat sebagai berikut:

1. Waktu Penelitian : Januari 2025 – Juni 2025
2. Tempat Penelitian : Universitas Lampung

3.2 Jadwal Penelitian

Tabel 3.1 Jadwal Penelitian

No	Nama	2025						
		Januari	Februari	Maret	April	Mei	Juni	Juli
1	Requirement Gathering							
2	Pembuatan <i>User story</i>							
3	Penyusunan <i>Backlog</i>							
4	Desain dan Perancangan sistem							

5	<i>Development</i>							
6	Pengujian Sistem							
7	<i>Deployment</i>							

3.3 Alat dan Bahan Penelitian

Berikut merupakan alat dan bahan yang digunakan dalam penelitian ini :

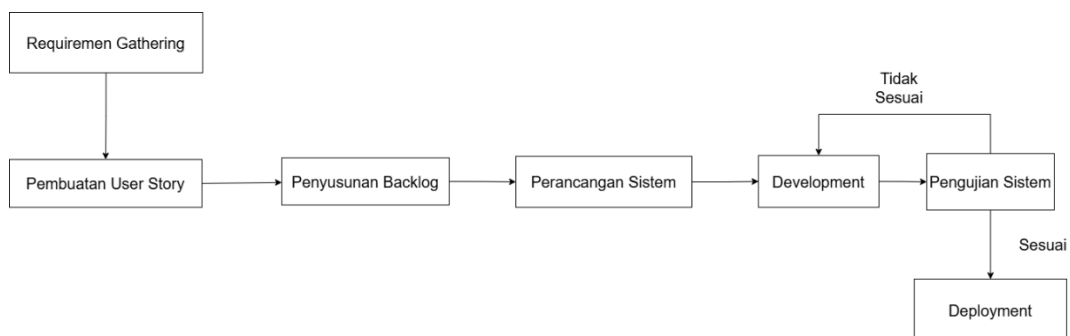
Tabel 3.2 Alat dan Bahan pengembangan sistem

NO	Perangkat	Kegunaan
1	Laptop	Perangkat utama untuk pengembangan, pengujian, dan implementasi antarmuka pengguna (UI).
2	<i>Framework Laravel</i>	<i>Framework</i> yang digunakan untuk integrasi <i>front-end</i> seperti routing, templating, dan validasi form.
3	<i>Visual Studio Code</i>	Perangkat lunak teks editor yang mendukung pengembangan <i>front-end</i> dengan fitur seperti debugging dan ekstensi.
4	Figma	Alat desain untuk membuat prototipe dan mockup antarmuka pengguna sebelum implementasi.
5	Browser (Chrome, Edge)	Alat untuk menguji tampilan dan performa <i>dashboard</i> pada berbagai perangkat.
6	Git & GitLab	Digunakan untuk versi kontrol kode dan kolaborasi selama pengembangan antarmuka.
7	Draw.io	Alat yang digunakan untuk membuat berbagai jenis diagram, seperti <i>Use Case Diagram</i> dan <i>Activity Diagram</i> , yang membantu dalam menggambarkan alur sistem dan interaksi pengguna
8	Trello	Alat yang digunakan untuk mengelola tugas dan proyek secara visual menggunakan sistem kanban, mendukung kolaborasi tim, serta integrasi dengan berbagai tools.

NO	Perangkat	Kegunaan
9	Black-Box Testing	Alat pengujian <i>funksional</i> pada sistem apakah berjalan dengan baik.
10	<i>System Usability Scale (SUS)</i>	Untuk pengukuran tingkat kepuasan pengguna.

3.4 Tahapan Penelitian

Penelitian ini dilakukan menggunakan metode *Agile development* dengan kerangka kerja *Kanban*, yang berfokus pada pengembangan secara bertahap dan interaktif guna memastikan kebutuhan pada pengguna terpenuhi dengan baik. Tahapan metode yang diterapkan dalam penelitian ini, yang melibatkan Requirement Gathering, Pembuatan *User story*, Penyusunan *Backlog*, Perancangan Sistem, *Development*, Pengujian sistem, Deployment & Monitoring serta, Pelaporan.



Gambar 3. 1 Tahapan Penelitian

Berdasarkan Gambar 3.1 Tahapan Penelitian, yang menunjukkan alur kerja penelitian sebagai berikut:

1. Tahap pertama adalah *Requiremen Gathering* sebagai proses mengumpulkan kebutuhan penelitian dan dokumentasi dari pihak perpustakaan yang berada pada bagian divisi Litabang dan Kerja sama. Tujuannya yaitu untuk memahami apa saja yang dibutuhkan oleh sistem yang akan dibuat.
2. Tahap kedua adalah *User story* yakni kebutuhan yang telah dikumpulkan, dibuat *user story* yang menjelaskan fitur atau fungsi yang diinginkan dari perspektif pengguna. *User story* mempermudah tim untuk memahami kebutuhan *user*

secara spesifik.

3. Tahap ketiga adalah *Backlog* pada tahapan ini semua kebutuhan dan *user story* disusun menjadi *backlog*. *Backlog* adalah daftar tugas atau fitur yang perlu dikerjakan dalam penelitian, yang nantinya akan menjadi panduan dalam tahap pengembangan.
4. Tahap keempat adalah Perancangan Sistem, dilakukannya pembuatan desain sistem secara menyeluruh, meliputi desain aritektur(misalnya alur data dan komunikasi) dan desain tampilan (mockup atau prototipe UI/UX)
5. Tahap kelima adalah *Development* proses pengembangan berdasarkan desain yang telah dibuat. Pada tahap ini, pengkodean dilakukan dengan menulis HTML dan CSS ke dalam *blade template* pada bagian *view* di Laravel, sehingga antarmuka sistem dapat dibangun sesuai rancangan.
6. Tahap keenam adalah Pengujian sisem , setelah proses pengembangan selesai, setiap fitur atau modul diuji menggunakan metode *Black-box Testing* guna memastikan bahwa fungsionalitasnya telah berjalan sesuai dengan yang diharapkan. Selain itu, dilakukan pula evaluasi tingkat kepuasan pengguna melalui pendekatan System Usability Scale (SUS). Jika hasil pengujian menunjukkan ketidaksesuaian atau nilai kepuasan rendah, maka sistem akan dikembalikan ke tahap *Development* untuk diperbaiki. Namun, jika hasil pengujian sudah sesuai, maka proses dapat dilanjutkan ke tahap *Development*.
7. Tahap ketujuh adalah *Deployment*, Sistem yang telah diuji dan dinyatakan siap akan *dideploy* diunggah ke server lokal pada jaringan perpustakaan , agar bisa digunakan oleh pengguna.

3.4.1 Requirement Gathering

Requirement Gathering dalam penelitian ini diperoleh melalui wawancara dengan pustakawan Universitas Lampung, Ibu Angelino Vinanti Sonjaya, S.I.Pust., yang bertugas di divisi Litbang dan Kerjasama. Wawancara ini bertujuan untuk mengidentifikasi kebutuhan utama dalam pengembangan sistem website *dashboard* monitoring. Dari hasil wawancara, diperoleh berbagai kebutuhan sistem yang harus dipenuhi guna mendukung fungsionalitas dan efektivitas *dashboard* monitoring dalam penelitian ini. Kebutuhan sistem inilah yang akan dijadikan pedoman dalam pembuatan sistem guna meminimalkan kesalahan pada tahapan selanjutnya.

Berdasarkan analisis kebutuhan tersebut, dapat diidentifikasi kebutuhan fungsional dan non-fungsional yang harus diterapkan dalam sistem yang akan dikembangkan.

3.4.2 Pembuatan *User story*

User story berfokus pada apa yang diinginkan oleh pengguna dan mengapa fitur tersebut diperlukan, tanpa menjelaskan detail teknis implementasinya. Selain itu, *User story* nantinya akan menjadi dasar dalam menyusun *backlog*. Setiap *User story* akan dikonversi menjadi tugas-tugas yang lebih kecil (*task*) dan dikelola dalam bentuk card di *backlog* pada papan *Kanban*. *User story* dalam penelitian ini mencakup dua jenis pengguna, yaitu *Super admin* dan *Admin*. *Super admin* merupakan pustakawan Universitas Lampung, yaitu Ibu Angelino Vinanti Sonjaya, S.I.Pust. *Admin* terdiri dari staf setiap divisi yang ada di perpustakaan. *User story* ditulis dalam format yang mencakup tiga elemen utama: *siapa* (*role* pengguna), *apa* (tujuan atau kebutuhan), dan *mengapa* (alasan kebutuhan tersebut penting).

Contoh format umum *user story*:

"Sebagai [*role*], saya ingin [tujuan/kebutuhan] sehingga [alasan/manfaat]."

User story memiliki peran penting dalam metode *Agile Kanban* karena membantu tim untuk fokus pada apa yang benar-benar dibutuhkan oleh pengguna, bukan hanya berdasarkan asumsi teknis. Selain itu, *user story* memastikan bahwa fitur yang dibuat benar-benar bermanfaat bagi pengguna, sehingga hasil akhir produk lebih sesuai dengan kebutuhan pengguna. Perlu dibedakan antara *user story* dan *backlog*. *Backlog* merupakan daftar semua pekerjaan yang harus dilakukan untuk menyelesaikan proyek, termasuk *user story*, tugas teknis, bug, dan elemen lainnya. Sementara itu, *user story* merupakan card tugas di dalam *backlog* yang berfokus pada kebutuhan pengguna. *Backlog* mencakup semua elemen pekerjaan yang diperlukan dalam proyek secara keseluruhan, sedangkan *user story* lebih terfokus pada apa yang dibutuhkan pengguna dan manfaatnya.

3.4.3 Penyusunan *Backlog*

Setelah menentukan *User Story*, tahap selanjutnya adalah menyusun *backlog*, yang berisi daftar tugas yang harus diselesaikan dalam perancangan dan pengembangan sistem. *Backlog* divisualisasikan dalam papan *Kanban* untuk memudahkan

pemantauan status tugas dari tahap perencanaan hingga selesai.

Sebagai bagian awal dari alur kerja *Agile Kanban*, *backlog* berisi kumpulan tugas yang kemudian diprioritaskan sesuai kebutuhan. Tugas yang telah diprioritaskan akan masuk ke dalam tahap Kanban *Queue* atau *To Do*, kemudian tugas yang sedang dikerjakan akan dipindahkan ke tahap *In Progress* yang berisi analisis, desain, development, testing, dan deploy. Setelah semua tugas selesai dikerjakan akan berpindah ke tahap *Done*, yang menandakan bahwa tugas telah selesai.

Dalam *backlog*, *user story* dan kebutuhan sistem dikumpulkan bersama dengan daftar prioritas lainnya, seperti fitur yang harus dikembangkan dan tugas-tugas pendukung. *Backlog* tidak hanya mencerminkan daftar pekerjaan yang perlu dilakukan, tetapi juga membantu tim dalam menentukan prioritas sebelum tugas dipindahkan ke kolom lain yang mencerminkan progres, seperti *To Do*, *In Progress*, atau *Done*.

3.4.4 Desain dan Perancangan Sistem

Pada tahap desain dan perancangan sistem, dilakukan pembuatan rancangan teknis yang menjadi dasar dalam pengembangan sistem. Tahap ini bertujuan untuk memastikan kebutuhan *funksional* dan *non-fungsional* dalam penelitian ini dirancang untuk mendukung pengembangan sistem yang mampu memenuhi kebutuhan pengguna, seperti *super admin*, *admin*, dan *user*, melalui fitur-fitur yang disediakan oleh sistem, yang telah dianalisis sebelumnya dapat diterapkan secara efektif dalam sistem yang akan dibangun.

Proses ini mencakup pembuatan berbagai diagram berbasis *UML (Unified Modeling Language)* seperti *Use Case* diagram untuk memetakan peran dan interaksi aktor dengan sistem, *Activity* diagram untuk menggambarkan alur kerja proses, dan *Entity Relationship diagram (ERD)* untuk merancang struktur serta hubungan antar entitas dalam basis data, diagram *UML* ini dibuat menggunakan *draw.io* sebagai alat bantu perancangannya .

Selain itu, rancangan antarmuka berupa mockup juga dibuat untuk memberikan gambaran awal mengenai tampilan dan fungsi *dashboard* monitoring serta *landing page*, rancangan ini dibuat menggunakan *figma* sebagai alat bantu perancangan.

Keseluruhan proses ini dirancang dengan memperhatikan aspek keterbacaan, aksesibilitas, dan pengalaman pengguna (UX) agar sistem yang dihasilkan tidak hanya *funksional* tetapi juga *user-friendly*.

3.4.5 Development

Pada tahap *Development*, pekerjaan yang telah dirancang dalam tahap sebelumnya, seperti perancangan UI/UX menggunakan Figma, mulai diimplementasikan dalam bentuk kode menggunakan *framework Laravel* untuk pengembangan *front-end*. Proses pengembangan ini mengikuti metode *Kanban*, di mana setiap tugas diprioritaskan dalam *Kanban Board*. Tugas yang siap dikerjakan akan dipindahkan dari kolom *Kanban Queue* atau *To Do* ke *In-Progress*, yang menandakan bahwa pekerjaan sedang dalam tahap pengembangan.

Dalam pengembangan *front-end* sistem ini, terdapat beberapa langkah utama yang dilakukan. Pertama, pembuatan komponen UI dan memastikan setiap elemen UI sesuai dengan hasil desain di Figma. Kedua, implementasi struktur *front-end*, yang melibatkan pembangunan tampilan dasar sistem menggunakan *Blade Template Engine (View)* dalam *Laravel*, penerapan *Bootstrap* sebagai *framework CSS* untuk styling dan desain responsif pada *Dashboard*, serta pemanfaatan *HTML*, *CSS*, dan *JavaScript* untuk meningkatkan tampilan dan interaktivitas sistem. Kemudian bagian *front-end* akan terhubung secara otomatis dengan backend melalui pengiriman data yang dikelola oleh *Controller Laravel*, yaitu dari *View* ke *Model*. Selain itu, *JavaScript* digunakan untuk mendukung fungsionalitas dinamis dan interaksi pengguna.

Setelah pengembangan fitur dilakukan, tahap selanjutnya adalah *testing dan debugging* untuk menguji tampilan serta fungsionalitas dari setiap halaman guna memastikan sistem berjalan sesuai harapan. Jika fitur telah lolos uji, maka akan dipindahkan ke kolom *Done* dalam *Kanban Board*. Namun, jika ditemukan kekurangan atau revisi, tugas akan dikembalikan ke tahap *In-Progress* untuk diperbaiki. Untuk memastikan seluruh pekerjaan terdokumentasi dengan baik dan memvisualisasikan setiap tugas dalam pengembangan sistem, digunakan Trello sebagai alat bantu dalam mencatat seluruh aktivitas, mulai dari yang akan dikerjakan, sedang berlangsung, hingga yang telah diselesaikan.

3.4.6 Pengujian Sistem

Pada tahapan ini, pengujian sistem dilakukan dalam dua tahap, yaitu *Testing Pre-Production* dan *Testing Production*, untuk memastikan bahwa sistem yang dikembangkan memenuhi standar fungsionalitas dan *usability* yang diharapkan. *Testing Pre-Production* merupakan tahap pengujian awal yang bertujuan untuk mengevaluasi hasil dari *backlog* yang telah dikerjakan selama tahap *development*. Pengujian ini menggunakan metode *Black-box testing*, yang dilakukan oleh tim pengembang dan pustakawan Universitas Lampung pada divisi Litbang dan Kerjasama, yaitu Ibu Angelino Vinanti Sonjaya, S.I.Pust. Pengujian ini memastikan bahwa fitur yang ada dalam sistem berfungsi sesuai dengan spesifikasi yang direncanakan. Pengujian *Black Box testing* ini dihitung dengan jumlah *test case* benar dibagi dengan jumlah total *test case* lalu dibagi 100%, untuk mencari presentase keberhasilan. Jika ditemukan ketidaksesuaian atau *bug* dalam fitur yang diuji, maka fitur tersebut akan dikembalikan ke tahap *development* untuk diperbaiki sebelum masuk ke tahap berikutnya.

Setelah seluruh fitur dalam backlog berhasil diuji dan diperbaiki pada tahap *pre-production*, sistem kemudian memasuki tahap *Testing Production*. Pengujian ini bertujuan untuk mengevaluasi produk yang telah selesai dan memastikan bahwa seluruh fitur berfungsi dengan baik serta siap digunakan oleh pengguna akhir. Pada tahap ini, selain menggunakan *Black-box testing*, dilakukan juga pengujian dengan metode *System Usability Scale (SUS)* untuk mengukur tingkat kenyamanan dan kemudahan penggunaan sistem. Pengujian SUS dilakukan dengan melibatkan 30 *responden*, terdiri dari pustakawan sebagai pengguna utama sistem, serta dosen, tenaga kependidikan, dan mahasiswa sebagai responden tambahan. Keterlibatan berbagai kalangan ini bertujuan untuk mengukur kelayakan dan kemudahan pengguna website secara umum, sekaligus memperoleh masukan dari berbagai perspektif pengguna yang berpotensi berinteraksi dengan sistem secara tidak langsung. Responden nantinya akan diberikan tautan menuju form yang berisi 10 pertanyaan *System Usability Scale (SUS)* untuk mengukur tingkat kemudahan pengguna dalam menggunakan sistem. Adapun daftar pertanyaan dari *System Usability Scale (SUS)* sebagai berikut :

Tabel 3.3 Pertanyaan *System Usability Scale* (SUS)

No	Pertanyaan	Skala
1	Saya merasa ingin sering menggunakan sistem ini.	1 s/d 5
2	Saya menemukan sistem ini rumit untuk digunakan.	1 s/d 5
3	Saya merasa sistem ini mudah digunakan.	1 s/d 5
4	Saya rasa saya membutuhkan bantuan teknis untuk dapat menggunakan sistem ini.	1 s/d 5
5	Fitur-fitur dalam sistem ini terintegrasi dengan baik.	1 s/d 5
6	Saya merasa banyak hal yang tidak konsisten dalam sistem ini.	1 s/d 5
7	Sebagian besar orang akan dapat mempelajari sistem ini dengan cepat.	1 s/d 5
8	Sistem ini terasa membingungkan untuk digunakan.	1 s/d 5
9	Saya percaya diri saat menggunakan sistem ini.	1 s/d 5
10	Saya harus belajar banyak sebelum bisa menggunakan sistem ini dengan baik.	1 s/d 5

Berdasarkan tabel 3.3 menjelaskan terdapat 10 pertanyaan untuk pengujian SUS. Setiap pertanyaan memiliki 5 opsi jawaban, yaitu sangat tidak setuju, tidak setuju, netral, setuju, dan sangat setuju. Masing - masing jawaban memiliki bobot nilai dengan rentang skor dari 1 sampai 5. Skor ini digunakan untuk menghitung Tingkat kebergunaan atau kemudahan sistem berdasarkan persepsi pengguna. Adapun jawaban dan skor disajikan pada tabel berikut:

Tabel 3.4 Jawaban dan Skor Metode SUS

No	Jawaban	Skor
1	Sangat Setuju	5

2	Setuju	4
3	Ragu - Ragu	3
4	Tidak Setuju	2
5	Sangat Tidak Setuju	1

Berdasarkan tabel 3.4 skor digunakan untuk menghitung nilai total SUS, yang nantinya akan menggambarkan tingkat kebergunaan (usability) sistem menurut persepsi pengguna. Berikut penjelasan penilaian SUS:

1. Skala penilaian yang digunakan berkisar dari sangat tidak setuju (strongly disagree) hingga sangat setuju (strongly agree), dengan rentang nilai mulai dari 1 sampai 5.
2. Pernyataan ganjil (positif) → skor dihitung : jawaban responden -1
Contoh: kalau pilih 4, maka skornya $4 - 1 = 3$
3. Pertanyaan genap (negatif) → skor dihitung : $5 -$ jawaban responden
Contoh: kalau pilih 2, maka skornya $5 - 2 = 3$
4. Jumlahkan seluruh nilai dari responden, kemudian dikalikan 2,5 untuk
5. mendapatkan nilai SUS dalam rentang 0 – 100.

$$\text{Skor SUS Responden} = (\text{Total skor poin}) \times 2.5$$

Contoh :

Jika total skor hasil penjumlahan adalah 32, maka nilai SUS = $32 \times 2.5 = 80$.

6. Jika ada banyak responden, setelah mendapat total skor , nilai dibagi dengan jumlah responden untuk mendapatkan rata- rata skor SUS akhir.

$$\tilde{x} = \frac{\sum x}{n}$$

$\tilde{x}$ = Skor SUS per responden (sudah dikali 2,5)

$\sum x$ = Jumlah total skor SUS seluruh responde

n = Jumlah responden

Rumus di atas digunakan untuk menghitung rata – rata skor SUS dari semua responden. Pertama, total seluruh skor SUS dijumlahkan lalu dikali 2.5, kemudian dibagi dengan jumlah responden. Nilai rata – rata ini nantinya akan digunakan untuk menentukan tingkat kebergunaan atau usability sistem, yang akan dibandingkan dengan kriteria acceptability atau rentang skor SUS . Adapun berikut tabel Acceptable Ranges:

Tabel 3.5 Acceptable Ranges

No	Acceptability	Range
1	Acceptable	70-100
2	Marginal (Hight)	60-69
3	Marginal (Low)	50-69
4	Not Acceptable	0-49

Berdasarkan tabel 3.5 Acceptable Ranges menjelaskan rentang nilai akhir SUS yang kemudian dikategorikan ke dalam tinglat kelayakan (acceptability), tabel ini berasal dari gambar 3. Berikut penjelasan mengenai Acceptable Ranges:

1. Nilai 70-100 menunjukkan sistem sangat diterima dan layak digunakan, (Acceptable).
2. Nilai 60 – 69 menunjukkan bahwa sitem cukup layak digunakan, tapi sebaiknya dilakukan perbaikan ,(Marginal – Hight).
3. Nilai 50 – 59 berarti sistem kurang nyaman digunakan, perlu perbaikan (Marginal – Low).
4. Nilai 0 – 49 menunjukkan bahwa sistem tidak dapat diterima (Not Acceptable) Karena usability rendah.

Rentang ini membantu untuk menafsirkan skor SUS agar dapat disimpulkan seberapa baik sistem diterima oleh pengguna.

Tabel 3. 6 Skor SUS

No	Skor	Peringkat	Grade
1	Skor ≥ 86	Best Imaginable	A
2	Skor ≥ 72 dan <86	Excellent	B
3	Skor ≥ 52 dan <72	Good	C
4	Skor ≥ 38 dan <52	Ok/Fair	D
5	Skor >25 dan <52	Poor	E
6	< 25	Worst Imaginable	F

Berdasarkan tabel 3.6 Skor SUS, menjelaskan tingkatan peringkat skala penilaian (grade). Setelah skor akhir diperoleh, tabel ini digunakan untuk mengetahui peringkat hasil pengujian SUS pada sistem dashboard monitoring layanan perpustakaan, sehingga dapat diketahui sistem tersebut berada pada kategori penilaian yang sesuai, tabel ini berasal dari gambar 3.

Dengan adanya dua tahap pengujian ini, sistem dapat diuji secara menyeluruh sebelum diimplementasikan secara penuh. Jika masih terdapat kekurangan pada tahap *Pre-Production*, maka perbaikan dilakukan di tahap *development* sebelum kembali diuji pada tahap *Production*. Pendekatan ini memastikan bahwa sistem yang dihasilkan memiliki kualitas yang optimal, baik dari segi fungsionalitas maupun pengalaman pengguna.

3.4.7 Deployment

Pada tahap *Deployment*, merupakan proses di mana sistem yang telah berhasil melalui pengujian diimplementasikan ke dalam lingkungan produksi untuk dapat diakses dan digunakan oleh pengguna akhir. *Deployment* dilakukan setelah seluruh fitur dalam *backlog* berhasil diuji dan dinyatakan siap melalui *Testing Production*. Proses ini mencakup pemasangan sistem pada *server*, konfigurasi basis data, serta penyesuaian lingkungan kerja agar sistem dapat berjalan dengan optimal. Dalam

penelitian ini, tugas *deployment* sepenuhnya menjadi tanggung jawab tim *back-end*, yang bertanggung jawab atas implementasi *server*, pengelolaan basis data, serta integrasi dengan sistem yang telah dikembangkan.

Sementara itu, tim *front-end* hanya berfokus pada tahap *development*, yang mencakup perancangan antarmuka pengguna dan pengimplementasian komponen UI sesuai dengan desain yang telah dibuat. Setelah pengembangan *front-end* selesai, hasilnya akan diserahkan kepada tim *back-end* untuk dilakukan integrasi dan *deployment*. Dengan pendekatan *Agile Kanban* yang digunakan dalam penelitian ini, setiap fitur yang telah diuji dan dianggap siap akan segera diterapkan dalam sistem melalui proses *deployment* oleh tim *back-end*. Jika terdapat kendala atau revisi yang diperlukan setelah pengujian, fitur tersebut akan dikembalikan ke tahap *development* untuk diperbaiki sebelum akhirnya *dideploy* ulang. Dengan demikian, proses *deployment* berjalan secara terstruktur sesuai dengan tanggung jawab masing-masing tim, memastikan bahwa sistem dapat digunakan secara optimal oleh pengguna akhir.

V KESIMPULAN DAN SARAN

5.1 Kesimpulan

Adapun kesimpulan berdasarkan penelitian yang telah dilakukan sebagai berikut:

1. Front-end dashboard monitoring berhasil dirancang dan dikembangkan menggunakan pendekatan metode Agile dengan kerangka kerja Kanban. Sistem ini mampu menampilkan data pengunjung dan peminjaman buku secara real-time melalui grafik dan tabel informatif yang mudah diakses oleh pustakawan. Penerapan Kanban melalui Trello board dengan pembagian kolom kerja (Backlog, To Do, In Progress, Done) serta batasan maksimal tiga task aktif (WIP limit) membuat alur pengerjaan lebih terstruktur dan terkontrol. Total waktu efektif pengembangan frontend tercatat sebesar 219,5 jam, yang dialokasikan secara realistis sesuai kompleksitas fitur. Sistem ini dinilai berhasil mendukung proses pelaporan dan pengambilan keputusan yang lebih efisien.
2. Berdasarkan Pengujian sistem menggunakan metode Black Box dilakukan terhadap 13 fitur utama dengan total 19 test case. Pada iterasi pertama, terdapat 1 test case yang gagal sehingga tingkat keberhasilan mencapai 94,74%. Setelah dilakukan perbaikan, iterasi kedua menunjukkan seluruh test case berhasil (100%), yang berarti semua fitur telah berjalan sesuai fungsinya. Hasil ini menunjukkan bahwa sistem telah memenuhi kebutuhan pengguna dan siap untuk digunakan.

3. Berdasarkan hasil pengujian *System Usability Scale* (SUS) terhadap 30 responden memperoleh skor rata-rata 77,42, yang termasuk dalam kategori acceptable dan berada pada tingkat grade B. Hal ini menunjukkan bahwa dashboard yang dikembangkan sudah dinilai cukup mudah digunakan, layak, serta memberikan pengalaman pengguna yang positif.

5.2 Saran

Berdasarkan hasil penelitian yang telah dilakukan, berikut beberapa saran yang dapat menjadi acuan untuk penelitian selanjutnya:

1. Pengujian fungsionalitas website yang telah dilakukan melalui blackbox testing, dapat dilanjutkan dengan whitebox testing untuk meningkatkan kualitas dan cakupan hasil pengujian.
2. Penambahan fitur unduh data ke dalam format Excel agar mempermudah pengguna dalam menyimpan dan mengelola data pengunjung atau peminjaman buku untuk laporan dengan lebih mudah.

DAFTAR PUSTAKA

- [1] Saharia, “Evaluasi Tingkat Kepuasan Pengguna Aplikasi Smart Library Menggunakan Metode *System Usability Scale*(SUS),” *Juktisi*, 2023, doi: 10.62712/juktisi.v2i1.60.
- [2] F. Wood, N. Ford, and C. Walsh, “The effect of postings information on searching behaviour,” *J. Inf. Sci.*, vol. 20, no. 1, pp. 29–40, Feb. 1994, doi: 10.1177/016555159402000104.
- [3] Ach. N. R. Nizam, W. Hariyanto, F. S. Bahtiar, F. K. R. Mahfud, and F. J. Pamungkas, “INLISLite (Integrated Library System) Version 3 Based Library Management Training in School Libraries within the Scope of the Dinas Perpustakaan dan Kearsipan Kota Batu: Manajemen Pengelolaan Perpustakaan Berbasis INLISlite (Integrated Library System) Versi 3 di Perpustakaan Sekolah dalam Lingkup Binaan Dinas Perpustakaan dan Kearsipan Kota Batu,” *Din. J. Pengabd. Kpd. Masy.*, vol. 8, no. 3, pp. 778–792, Jun. 2024, doi: 10.31849/dinamisia.v8i3.18691.
- [4] H. Hendra, Yulia Wahyuningsih, and Fernandi Mahendrasusila, “Rancang Bangun Sistem Proses Transaksi Perusahaan Berbasis Website dengan Metode Agile Development,” *PROSISKO J. Pengemb. Ris. Dan Obs. Sist. Komput.*, vol. 11, no. 1, pp. 10–19, Mar. 2024, doi: 10.30656/prosisko.v11i1.7809.
- [5] E. Weflen, C. A. MacKenzie, and I. V. Rivero, “An influence diagram approach to automating lead time estimation in Agile Kanban project

management,” *Expert Syst. Appl.*, vol. 187, p. 115866, Jan. 2022, doi: 10.1016/j.eswa.2021.115866.

[6] J. Saltz and R. Heckman, “Exploring Which Agile Principles Students Internalize When Using a Kanban Process Methodology,” vol. 31, 2020.

[7] W. Zayat and O. Senvar, “Framework Study for Agile Software Development Via Scrum and Kanban,” *Int. J. Innov. Technol. Manag.*, vol. 17, no. 04, p. 2030002, Jun. 2020, doi: 10.1142/S0219877020300025.

[8] A. Saputra, M. A. Nugroho, F. D. Astuti, and D. Kriestanto, “Rancangan Dan Implementasi Aplikasi Mentoring Menggunakan Framework Laravel,” *Juti J. Teknol. Inf.*, 2023, doi: 10.26798/juti.v1i2.811.

[9] A. Lelitasari, “Penerapan Metode Design Thinking Dan Agile Development Pembangunan Aplikasi Marketplace ‘Hijab,’” *J. Inf. Syst. Res. Josh*, 2023, doi: 10.47065/josh.v4i4.3695.

[10] A. C. Hutauruk and A. F. Pakpahan, “Perancangan Sistem Informasi Organisasi Kemahasiswaan Berbasis Web Pada Universitas Advent Indonesia Menggunakan Metode Agile Development (Studi Kasus: Universitas Advent Indonesia),” *Cogito Smart J.*, 2021, doi: 10.31154/cogito.v7i2.328.315-328.

[11] R. Bustomi, I. Kusumah, S. Supriyadi, and Z. Zaharuddin, “Analisis Manajemen Pengadaan Buku Digital Untuk Perpustakaan Sekolah Atau Madrasah Dan Perguruan Tinggi Tahun 2023,” *Arus J. Sos. Dan Hum.*, vol. 4, no. 2, pp. 802–808, Aug. 2024, doi: 10.57250/ajsh.v4i2.559.

- [12] R. A. Ag, Z. Zulfitria, and H. I. Dewi, "Pengolahan Bahan Pustaka Perpustakaan Tingkat Sekolah Dasar Desa Iwul, Parung," *Nat. J. Kaji. Penelit. Pendidik. Dan Pembelajaran*, vol. 4, no. 2a, pp. 553–561, Jul. 2020, doi: 10.35568/naturalistic.v4i2a.794.
- [13] "Persepsi Pustakawan terhadap Transformasi Perpustakaan Berbasis Inklusi Sosial pada Dinas Perpustakaan dan Kearsipan Provinsi Maluku," *J. Ilmu Inf. Perpust. Dan Kearsipan*, vol. 23, no. 2, Oct. 2021, doi: 10.7454/JIPK.v23i2.002.
- [14] T. Aziz and S. Asmiatun, "Penerapan Algoritma Bubble Sort Pada Sistem Perpustakaan Menggunakan Framework Codeigniter," *Inf. Sci. Libr.*, 2022, doi: 10.26623/jisl.v3i1.5104.
- [15] D. R. Aryanto, Q. H. Hidayah, and B. A. Sekti, "Sistem dashboard untuk visualisasi data pendapatan dan pengeluaran berbasis web PT. XYZ," 2024.
- [16] B. E. A. Edi and Lutfan Lazuardi, "Pemanfaatan Digital Dashboard Rumah Sakit Dalam Pengambilan Keputusan dan Penentuan Strategi di Rumah Sakit Anugerah Kota Pekalongan," *J. Manaj. Pelayanan Kesehat. Indones. J. Health Serv. Manag.*, vol. 25, no. 02, Oct. 2022, doi: 10.22146/jmpk.v25i02.5034.
- [17] I. Rahyadi, M. Duncik, N. Jannah, and I. Ghifari, "Look Who's Talking: Means of Interpersonal Communication Between Librarians and Library Users," 2021, doi: 10.4108/eai.17-7-2021.2312391.
- [18] Z. Yunita, E. R. Susanto, and F. Ulum, "Sistem Informasi Manajemen Monitoring Kemajuan Pekerjaan Konstruksi pada PT PLN UP3 Kota Metro," *J. Teknol. DAN Sist. Inf.*, vol. 4, no. 2, 2023.

- [19] S. Rathor, W. Xia, and D. Batra, “Achieving Software Development Agility: Different Roles of Team, Methodological and Process Factors,” *Inf. Technol. People*, 2023, doi: 10.1108/itp-10-2021-0832.
- [20] N. S. Aminah, A. S. Ichwanda, D. D. Djamal, Y. B. Wijaya Budiharto, and M. Budiman, “A Low-Cost Indoor Navigation and Tracking System Based on Wi-Fi-Rssi,” 2023, doi: 10.21203/rs.3.rs-2469968/v1.
- [21] R. T. Aldisa and M. A. Abdullah, “Penerapan Agile Development Methodology dalam Sistem Penjualan Buku dengan Fitur Kategori dan Pencarian,” *Build. Inform. Technol. Sci. BITS*, vol. 3, no. 4, pp. 547–553, Mar. 2022, doi: 10.47065/bits.v3i4.1434.
- [22] Tofid, Eddy Julianto, and Y. Harjoseputro, “Revamp Aplikasi Teman Bumi Lebih Interaktif Dengan Pendekatan Agile,” *J. RESTI Rekayasa Sist. Dan Teknol. Inf.*, vol. 4, no. 5, pp. 923–929, Oct. 2020, doi: 10.29207/resti.v4i5.2325.
- [23] K. Brock, E. den Ouden, F. Langerak, and K. Podoyntsyna, “Front End Transfers of Digital Innovations in a Hybrid Agile-Stage-Gate Setting,” *J. Prod. Innov. Manag.*, 2020, doi: 10.1111/jpim.12556.
- [24] H. Alaidaros, M. Omar, and R. Romli, “The state of the art of agile kanban method: challenges and opportunities,” *Indep. J. Manag. Prod.*, vol. 12, no. 8, pp. 2535–2550, Dec. 2021, doi: 10.14807/ijmp.v12i8.1482.
- [25] H. Shen, B. Wall, M. Zaremba, Y. Chen, and J. Browne, “Integration of business modelling methods for enterprise information system analysis and user

requirements gathering,” *Comput. Ind.*, vol. 54, no. 3, pp. 307–323, Aug. 2004, doi: 10.1016/j.compind.2003.07.009.

[26] S. Sharma and D. Kumar, “Exploring Story Cards for Evaluating Requirement Understanding in Agile Software Development,” *J. Inf. Technol. Manag.*, vol. 14, no. Security and Resource Management challenges for Internet of Things, Mar. 2022, doi: 10.22059/jitm.2022.85646.

[27] R. P. Aji, A. D. Pritama, and B. D. Putranto, “Implementasi Metode Agile Scrum pada Perancangan Aplikasi Pengecekan Penyakit Diabetes Berdasarkan Assesment,” vol. 13, 2024.

[28] M. Zulfikar and A. Qoiriah, “Perancangan Dan Pembangunan Sistem Informasi Keuangan Yayasan Umar Faruq Al Arifin Berbasis Website Menggunakan Metode Agile Pendekatan Kanban,” vol. 06, 2025.

[29] F. Amsury, I. Kurniawati, Heriyanto, and M. Rizki Fahdia, “Pelatihan Pemanfaatan Aplikasi Trello Untuk Meningkatkan Efektifitas Manajemen Proyek Pada Karyawan PT Jaya Persada Indonesia,” *Abdi Teknayasa*, Dec. 2022, doi: 10.23917/abditeknayasa.v3i2.1249.

[30] W. Welda, D. M. Dwi Putra, and A. M. Dirgayusari, “Usability Testing Website Dengan Menggunakan Metode *System Usability Scale* (Sus)s,” *Int. J. Nat. Sci. Eng.*, 2020, doi: 10.23887/ijnse.v4i2.28864.

[31] Z. Tianyi, “A Cloud Monitoring Management Architecture Based on Artificial Intelligence,” 2022, doi: 10.4108/eai.20-12-2021.2315040.

- [32] I. A. Alfarisi, A. T. Priandika, and A. S. Puspaningrum, “Penerapan Framework Laravel Pada Sistem Pelayanan Kesehatan (Studi Kasus: Klinik Berkah Medical Center),” *J. Ilm. Comput. Sci.*, vol. 2, no. 1, pp. 1–9, Jul. 2023, doi: 10.58602/jics.v2i1.11.
- [33] D. P. Sari and R. Wijanarko, “Implementasi Framework Laravel Pada Sistem Informasi Penyewaan Kamera (Studi Kasus Di Rumah Kamera Semarang),” *J. Inform. Dan Rekayasa Perangkat Lunak*, 2020, doi: 10.36499/jinrpl.v2i1.3190.
- [34] M. Laaziri, K. Benmoussa, S. Khouliji, and M. L. Kerkeb, “A Comparative study of PHP frameworks performance,” *Procedia Manuf.*, vol. 32, pp. 864–871, 2019, doi: 10.1016/j.promfg.2019.02.295.
- [35] U. Hairah *et al.*, “Peningkatan Kompetensi Siswa Dalam Merancang UI/UX Aplikasi Mobile Menggunakan Figma,” *Jompa Abdi J. Pengabd. Masy.*, vol. 3, no. 4, pp. 59–65, Dec. 2024, doi: 10.57218/jompaabdi.v3i4.1211.
- [36] M. P. Utami, “Pemanfaatan Desain Interaksi Antar Muka Pengguna Dengan Implementasi Model Goms Pada Aplikasi Mobile Elma,” *Rabit J. Teknol. Dan Sist. Inf. Univrab*, 2023, doi: 10.36341/rabit.v8i1.2967.
- [37] S. Sudjiran, M. Saefudin, and S. A. Perdana, “Digital System UI/UX Design Management Submission Of Agricultural Cost Loans Using Figma Software,” *J. Inf. Syst. Inform. Comput.*, vol. 7, no. 1, p. 74, Jun. 2023, doi: 10.52362/jisicom.v7i1.1090.
- [38] R. T. Ginting, “Restorasi E-Library Berbasis Perpustakaan Digital Di SMA Negeri 2 Kuta Selatan,” *J. Servite*, 2023, doi: 10.37535/102004220225.

- [39] R. Sekulić and N. M. Keresteš, “Analiza Softvera Koji Ubrazavaju Proces Izrade Veb I Mobilnih Aplikacija ,” *Zb. Rad. Fak. Teh. Nauka U Novom Sadu*, vol. 37, no. 05, pp. 786–789, May 2022, doi: 10.24867/17EF08Sekulic.
- [40] G. A. Supriatmaja, K. D. D. Widyaputra, J. Deva, and G. S. Mahendra, “Sistem Informasi Perpustakaan Menggunakan Framework Bootstrap Dengan PHP Native dan Database MySQL Berbasis Web Pada SMP Negeri 2 Dawan,” vol. 1, no. 1, 2022.
- [41] Z. Alawi, “Perencanaan Manajemen Proyek dalam Pengembangan Learning Management System Menggunakan Trello,” 2022.
- [42] H. Al-Bazar, H. Abdel-Jaber, E. Labib, and M. Al-Madi, “Impacts of Blended Learning Systems on AOU Students’ Satisfaction: An Investigational Analysis of KSA’s Branch,” *Turk. Online J. Distance Educ.*, pp. 213–235, Jul. 2021, doi: 10.17718/tojde.961839.
- [43] R. Sudradjat, “Pemanfaatan Aplikasi Otomasi Perpustakaan Inlislite pada Perpustakaan Umum Provinsi dan Kabupaten / Kota di Provinsi Jambi,” *J. Khazanah Intelekt.*, vol. 3, no. 2, pp. 489–504, Jan. 2020, doi: 10.37250/newkiki.v3i2.50.
- [44] M. Raharjo, M. Napiah, and R. S. Anwar, “Perancangan Sistem Informasi Dengan PHP Dan MYSQL Untuk Pendaftaran Sekolah Di Masa Pandemi,” *Comput. Sci. CO-Sci.*, vol. 2, no. 1, pp. 50–58, Jan. 2022, doi: 10.31294/coscience.v2i1.689.

- [45] M. Ahmadar, P. Perwito, and C. Taufik, "Perancangan Sistem Informasi Penjualan Berbasis Web pada Rahayu Photo Copy dengan Database MySQL," *Dharmakarya*, vol. 10, no. 4, p. 284, Dec. 2021, doi: 10.24198/dharmakarya.v10i4.35873.
- [46] I. R. Afandi, N. Pratiwi, A. A. Rizki, M. Irva, and M. F. Aulia, "Perancangan Sistem Informasi Pelayanan Pembuatan Surat Online di Desa Ciangsana Berbasis Website," *JATI J. Mhs. Tek. Inform.*, vol. 6, no. 2, pp. 571–577, Aug. 2022, doi: 10.36040/jati.v6i2.5318.
- [47] H. E. Wahanani, W. S. J. Saputra, and B. H. V. Wahono, "Perancangan Infrastruktur Server VCS (Version Control System) dengan GitLab Berbasis Git," *SCAN - J. Teknol. Inf. Dan Komun.*, vol. 14, no. 2, Jun. 2019, doi: 10.33005/scan.v14i2.1490.
- [48] D. Sebastián, H. Hermawan, Restyandito, and K. A. Nugraha, "Perancangan Desain Antarmuka Buku Kontak Untuk Lansia," *Aiti*, 2022, doi: 10.24246/aiti.v19i2.167-184.
- [49] M. Tambunan, "Pembuatan Desain Antarmuka Informasi Magang Prodi Informatika Universitas Kristen Duta Wacana Berbasis Web," *J. Inform. Dan Tek. Elektro Terap.*, 2023, doi: 10.23960/jitet.v11i3.3295.
- [50] I. G. A Indrayani, I. P. Agung Bayupati, and I. M. Suwija Putra, "Analisis Usability Aplikasi iBadung Menggunakan Heuristic Evaluation Method," *J. Ilm. Merpati Menara Penelit. Akad. Teknol. Inf.*, 2020, doi: 10.24843/jim.2020.v08.i02.p03.

- [51] R. D. Rusdiyan Yusron and M. M. Huda, "Analisis Perancangan Sistem Informasi Perpustakaan Menggunakan Model Waterfall Dalam Peningkatan Inovasi Teknologi," *J. Autom. Comput. Inf. Syst.*, 2021, doi: 10.47134/jacis.v1i1.4.
- [52] S. Kosasi and S. M. Kuway, "Studi Analisis Persyaratan Kebutuhan Sistem dalam Menghasilkan Perangkat Lunak yang Berkualitas".
- [53] R. D. Gunawan, R. Napianto, R. I. Borman, and I. R. Uswatul Hanifah, "Penerapan Pengembangan Sistem Extreme Programming Pada Aplikasi Pencarian Dokter Spesialis Di Bandarlampung Berbasis Android," *Format J. Ilm. Tek. Inform.*, 2020, doi: 10.22441/format.2019.v8.i2.008.
- [54] M. Misinem, T. B. Kurniawan, A. Astried, and J. A. Widiars, "Model Simulasi Penyelesaian Masalah Perjalanan Penjual Menggunakan Pendekatan Kecerdasan Buatan, Optimisasi Koloni Semut," *J. Bina Komput.*, 2020, doi: 10.33557/binakomputer.v2i2.974.
- [55] F. S. Andini, D. Apandi, and A. A. Muris, "Analisis Dan Perancangan Sistem Informasi Perpustakaan Pada Pesantren Lukmanul Hakim Batumarta II," *Intech*, 2020, doi: 10.54895/intech.v1i2.637.
- [56] T. Arianti, A. Fa'izi, S. Adam, and M. Wulandari, "Perancangan Sistem Informasi Perpustakaan Menggunakan Diagram UML (Unified Modelling Language)," vol. 1, 2022.
- [57] E. M. Bouzian, "Use Case and UML Scenarios Engineering for Building B2B E-Commerce," *Int. J. Adv. Trends Comput. Sci. Eng.*, 2020, doi: 10.30534/ijatcse/2020/180942020.

- [58] R. I. Melyani, “Pengembangan Sistem Informasi Penggajian Berbasis Web Menggunakan Framework Laravel Dengan Metode Agile Software Development,” *J. Sist. Inf. Akunt. Jasika*, 2023, doi: 10.31294/jasika.v3i01.2195.
- [59] D. Edi and S. Betshani, “Analisis Data dengan Menggunakan ERD dan Model Konseptual Data Warehouse,” *J. Inform.*, no. 1.
- [60] A. Kusumadewi, “Implementation of Laravel Framework on Online Presence App Design for Internship Employees (Case Study: PT. XYZ),” *E3s Web Conf.*, 2023, doi: 10.1051/e3sconf/202338802007.
- [61] B. Hardika *et al.*, “Pengujian Blackbox Testing Website Garuda Farm Menggunakan Teknik Equivalence Partitioning,” *J. Kridatama Sains dan Teknol.*, vol. 6, no. 02, pp. 740–753, Dec. 2024, doi: 10.53863/kst.v6i02.1420.
- [62] R. Pramudita, “Pengujian Black Box Pada Aplikasi Ecampus Menggunakan Metode Equivalence Partitioning,” *Inform. Educ. Prof. J. Inform.*, 2020, doi: 10.51211/itbi.v4i2.1347.
- [63] P. Vlachogianni and N. Tselios, “Perceived usability evaluation of educational technology using the *System Usability Scale* (SUS): A systematic review,” *J. Res. Technol. Educ.*, vol. 54, no. 3, pp. 392–409, May 2022, doi: 10.1080/15391523.2020.1867938.
- [64] M. Junaidi, “Evaluasi Usability Website Badan Pusat Statistik (BPS) Kabupaten Indragiri Hilir Menggunakan Metode *System Usability Scale* (SUS),” *J. Sist. Inf.*, vol. 2, no. 7, 2024.

- [65] A. Bangor, "Determining What Individual SUS Scores Mean: Adding an Adjective Rating Scale," vol. 4, no. 3, 2009.
- [66] P. Diebold, S. Theobald, J. Wahl, and Y. Rausch, "Stepwise transition to agile: From three agile practices to Kanban adaptation," *J. Softw. Evol. Process*, vol. 31, no. 5, p. e2167, May 2019, doi: 10.1002/smr.2167.
- [67] A. Rahmat and N. A. M. Hanifah, "Usability Testing in Kanban Agile Process for Club Management System," in *2020 6th International Conference on Interactive Digital Media (ICIDM)*, Bandung, Indonesia: IEEE, Dec. 2020, pp. 1–6. doi: 10.1109/ICIDM51048.2020.9339668.
- [68] B. O. Lubis, I. Carolina, A. Supriyatna, A. Sudradjat, H. Destiana, and R. Komarudin, "Implementasi Metode Kanban pada Rancangan Sistem Informasi HelpDesk Pada Kantor Imigrasi Berbasis Website," *J. Infortech*, vol. 5, no. 2, pp. 106–117, Dec. 2023, doi: 10.31294/infortech.v5i2.17163.
- [69] G. A. Supriatmaja, K. Mahendra, I. G. B. Widi Atmaja, D. M. J. Putra, N. K. A. T. Wahyuni, and P. Y. Prawati, "Pengembangan Sistem Informasi Monitoring dan Pengarsipan Administrasi Keuangan Menggunakan Framework Laravel pada BPS Buleleng," *Infomatek*, vol. 26, no. 2, pp. 239–252, Dec. 2024, doi: 10.23969/infomatek.v26i2.19231.
- [70] S. Sugiharto, S. Holanda, W. William, E. Fortino, J. Lim, and G. Gunawan, "Rancang Bangun Platform Kolaborasi Creative Hub dan Client Berbasis Web dengan Agile dan Scrum," *J. Pendidik. Dan Teknol. Indones.*, vol. 4, no. 12, pp. 565–583, Dec. 2024, doi: 10.52436/1.jpti.488.

[71] Universitas Muhammadiyah Surakarta, D. I. Rahmadhani, and M. Maryam, "Pengembangan Sistem Point of Sale Berbasis Web pada Toko Faafoo Menggunakan Framework Laravel," *Methomika J. Manaj. Inform. Dan Komputerisasi Akunt.*, vol. 8, no. 2, pp. 241–249, Oct. 2024, doi: 10.46880/jmika.Vol8No2.pp241-249.