

**RANCANG BANGUN *CONTENT MANAGEMENT SYSTEM (CMS)*
PORTAL BERITA BERBASIS *LARAVEL* DAN *REACT JS* DENGAN
PENDEKATAN *SINGLE PAGE APPLICATION (SPA)***

(Skripsi)

Oleh

YOS MARISON SIANIPAR

2115061074



**FAKULTAS TEKNIK
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG**

2026

**RANCANG BANGUN *CONTENT MANAGEMENT SYSTEM (CMS)*
PORTAL BERITA BERBASIS *LARAVEL* DAN *REACT* DENGAN
PENDEKATAN *SINGLE PAGE APPLICATION (SPA)***

Oleh

YOS MARISON SIANIPAR

Skripsi

**Sebagai salah satu syarat untuk mendapat gelar
SARJANA TEKNIK**

Pada

**Jurusan Teknik Elektro
Fakultas Teknik Universitas Lampung**



**FAKULTAS TEKNIK
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG**

2026

ABSTRAK

Rancang Bangun *Content Management System (CMS)* Portal Berita Berbasis *Laravel* dan *React* dengan Pendekatan *Single Page Application (SPA)*

Oleh

YOS MARISON SIANIPAR

Content Management System (CMS) merupakan sistem yang digunakan untuk mengelola konten digital, termasuk proses pembuatan, pengeditan, publikasi, dan pengaturan hak akses pengguna. Pengelolaan portal berita membutuhkan *CMS* yang mampu mendukung alur kerja redaksional. Namun, *CMS* yang digunakan pada beberapa redaksi masih memiliki keterbatasan, seperti pengelolaan hak akses yang belum optimal, ketiadaan laporan produksi artikel, serta fitur pendukung yang belum memenuhi kebutuhan operasional. Penelitian ini bertujuan untuk merancang dan membangun *CMS* portal berita berbasis *Laravel* dan *React.js* dengan pendekatan *Single Page Application (SPA)*, serta melakukan pengujian sistem menggunakan *Black Box Testing* dan *User Acceptance Test (UAT)*. Metode pengembangan yang digunakan adalah *Rapid Application Development (RAD)*, yang meliputi beberapa tahapan utama, yaitu *requirements planning*, *user design*, *rapid construction*, dan *cutover*. Hasil pengujian fungsional menggunakan *Black Box Testing* menunjukkan bahwa seluruh fitur utama, seperti manajemen artikel, *auto-save*, notifikasi *WhatsApp*, berjalan dengan baik. Pengujian *UAT* yang melibatkan tujuh responden dari lima redaksi menghasilkan rata-rata tingkat penerimaan sebesar 92,7%, yang berada pada kategori Sangat Baik. Analisis performa menunjukkan bahwa sistem berbasis *SPA* memiliki kinerja navigasi yang jauh lebih cepat dibandingkan *Multi Page Application (MPA)*. Pada pengujian *initial load*, rata-rata waktu muat *SPA* adalah 0,60 detik, sedangkan *MPA* 0,71 detik. Pada pengujian perpindahan halaman (navigasi), *SPA* memperoleh rata-rata 69,6 ms, dibandingkan *MPA* sebesar 868,4 ms, dengan hasil *SPA* 12,48× lebih cepat.

Kata kunci: *Content Management System*, *Portal Berita*, *Laravel*, *React.js*, *Single Page Application*.

ABSTRACT

Design and Development of a News Portal Content Management System (CMS) Based on Laravel and React Using the Single Page Application (SPA) Approach

By

YOS MARISON SIANIPAR

A Content Management System (CMS) is a platform used to manage digital content, including content creation, editing, publication, and user access control. News portals require a CMS capable of supporting structured editorial workflows; however, existing systems in several newsroom environments still exhibit limitations such as insufficient access control, the absence of article production reports, and supporting features that do not fully meet operational needs. This study aims to design and develop a news portal CMS based on Laravel and React.js using the Single Page Application (SPA) approach, and to evaluate the system through Black Box Testing and User Acceptance Testing (UAT). The development process adopts the Rapid Application Development (RAD) methodology, which consists of requirements planning, user design, rapid construction, and cutover phases. Functional testing using Black Box Testing shows that all core features including article management, auto-save, and WhatsApp notifications operate correctly. UAT involving seven respondents from five newsrooms yields an average acceptance level of 92.7%, categorized as Excellent. Performance analysis further demonstrates that the SPA-based system delivers significantly faster navigation compared to a traditional Multi Page Application (MPA). During initial load testing, SPA records an average load time of 0.60 seconds, while MPA records 0.71 seconds. In navigation tests, the SPA achieved 69.6 ms on average, compared to 868.4 ms for the MPA, or about 12.48× faster.

Keywords: Content Management System, News Portal, Laravel, React.js, Single Page Application.

Judul Skripsi : RANCANG BANGUN *CONTENT MANAGEMENT SYSTEM (CMS) PORTAL BERITA BERBASIS LARAVEL DAN REACT DENGAN PENDEKATAN SINGLE PAGE APPLICATION (SPA)*

Nama Mahasiswa : **Yos Marison Sianipar**

Nomor Pokok Mahasiswa : 2115061074

Program Studi : Teknik Informatika

Jurusan : Teknik Elektro

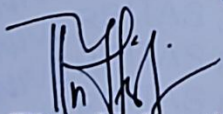
Fakultas : Teknik

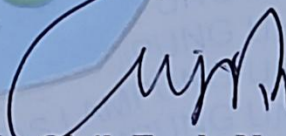


1. Komisi Pembimbing

Pembimbing Utama

Pembimbing Pendamping


Ir. Titin Yulianti, S.T., M.Eng
NIP. 198807092019032015

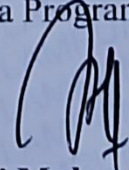

Ir. Gigih Forda Nama, S.T., M.T.I.
NIP. 198307122008121003

2. Mengetahui

Ketua Jurusan

Ketua Program Studi

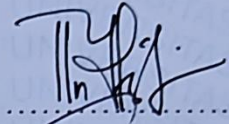

Herlinawati, S.T., M.T.
NIP. 197103141999032001


Yessi Mulyani, S.T., M.T.
NIP. 197312262000122001

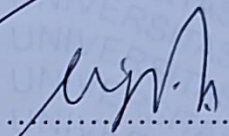
MENGESAHKAN

1. Tim Penguji

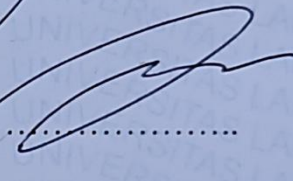
Ketua : Ir. Titin Yulianti, S.T., M.Eng.



Sekretaris : Ir. Gigih Forda Nama, S.T., M.T.I.



Penguji : Mona Arif Muda, S.T., M.T.



2. Dekan Fakultas Teknik

The signature of Dr. Ahmad Herison is written in blue ink over a blue circular stamp. The stamp contains the text "KEMENTERIAN PENDIDIKAN TINGGI, SAINS, DAN TEKNOLOGI" and "UNIVERSITAS LAMPUNG".

NIP. 196910302000031001

Tanggal Lulus Ujian Skripsi : 27 Januari 2026

SURAT PERNYATAAN

Saya yang bertanda tangan dibawah ini, menyatakan dengan sebenarnya bahwa skripsi saya yang berjudul "Rancang Bangun *Content Management System (CMS)* Portal Berita Berbasis *Laravel dan React* dengan Pendekatan *Single Page Application (SPA)*" merupakan hasil karya saya sendiri. Semua hasil yang tertuang dalam skripsi ini telah mengikuti kaidah penulisan karya ilmiah Universitas Lampung. Apabila di kemudian hari terbukti bahwa skripsi saya ini merupakan salinan atau dibuat oleh orang lain, maka saya bersedia menerima sanksi sesuai dengan ketentuan hukum atau akademik yang berlaku.

Bandar Lampung, 1 Juni 2026

at pernyataan,



Yos Marison Sianipar
2115061074

RIWAYAT HIDUP



Penulis bernama Yosa Sianipar, lahir di Siborongborong pada tanggal 5 Januari 2002. Penulis merupakan anak ke-2 dari 4 bersaudara, putra dari Bapak Bosur Sianipar dan Ibu Rushedy Sihombing. Penulis menyelesaikan Pendidikan menengah pertama di SMP Negeri 1 Siborongborong dan lulus pada tahun 2017. Penulis menyelesaikan pendidikan menengah atas di SMK Negeri 1 Siborongborong pada tahun 2020. Pada tahun 2021, penulis diterima sebagai mahasiswa Program Studi Teknik informatika, Fakultas Teknik, Universitas Lampung melalui jalur SBMPTN. Selama menempuh studi, penulis aktif dalam berbagai kegiatan kampus, di antaranya sebagai Asisten Laboratorium di Laboratorium Teknik Komputer pada tahun 2024, serta Panitia di Unit Kegiatan Mahasiswa (UKM) Kristen pada tahun 2023. Penulis juga pernah mengikuti program Magang dan Studi Independen Bersertifikat (MSIB) Angkatan 5 di PT. Gama Multi Usaha Mandiri sebagai peserta Program Studi Independen dengan kegiatan *Cloud Computing Security Engineer (Cybersecurity)* yang diselenggarakan tahun 2023. Penulis juga pernah mengikuti kegiatan pengabdian masyarakat dalam bentuk pembuatan website. Selain itu, penulis bekerja sebagai *Fullstack Developer* di situs berita Rilis.id sejak tahun 2024. Pada tahun 2024, penulis berhasil masuk sebagai finalis dalam Kompetisi *Healthkathon 2024* yang diselenggarakan oleh BPJS Kesehatan.

MOTTO

“Ask and it will be given to you; seek and you will find; knock and the door will be opened to you”

(Matthew 7:7)

“A system is valued not for its complexity, but for its usefulness.”

(penulis)

PERSEMBAHAN

Tuhan Yesus Kristus, atas segala berkat, kekuatan, dan hikmat yang selalu Engkau limpahkan dalam setiap langkah perjalananku. Karena kasih-Mu, aku sampai di titik ini.

Bapak dan Ibu yang tiada henti mendoakan dan mendukung demi masa depanku. Setiap tetes keringat kalian adalah motivasi terbesar dalam hidupku.

Almamater saya, Universitas Lampung, tempat saya bertumbuh, belajar, dan menempa diri menjadi pribadi yang lebih baik.

"For I know the plans I have for you," declares the Lord, "plans to prosper you and not to harm you, plans to give you hope and a future."

SAWACANA

Puji dan syukur penulis panjatkan ke hadirat Tuhan Yang Maha Esa atas segala berkat dan rahmat-Nya sehingga penulis dapat menyelesaikan skripsi yang berjudul “Rancang Bangun *Content Management System (CMS)* Portal Berita Berbasis *Laravel* dan *React JS* dengan Pendekatan *Single Page Application (SPA)*” sebagai salah satu syarat untuk memperoleh gelar Sarjana Teknik pada Program Studi Teknik Informatika, Fakultas Teknik, Universitas Lampung.

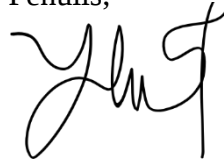
Penulis menyadari bahwa dalam proses penyelesaian skripsi ini tidak terlepas dari dukungan, bimbingan, dan bantuan berbagai pihak. Oleh karena itu, dengan segala kerendahan hati, penulis mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Bapak Dr. H. Ahmad Herison, S.T.,M.T., selaku Dekan Fakultas Teknik Universitas Lampung.
2. Ibu Herlinawati, S.T., M.T., selaku Ketua Jurusan Teknik Elektro Universitas Lampung.
3. Ibu Yessi Mulyani, S.T., M.T, selaku Ketua Program Studi Teknik Informatika Fakultas Teknik Universitas Lampung.
4. Ibu Ir. Titin Yulianti, S.T., M.Eng., selaku Dosen Pembimbing I yang telah meluangkan waktu, tenaga, dan pikiran dalam memberikan bimbingan, arahan, serta motivasi kepada penulis selama proses penyusunan skripsi ini.
5. Bapak Ir. Gigih Forda Nama, S.T., M.T.I., selaku Dosen Pembimbing II yang telah memberikan bimbingan, masukan, dan saran yang sangat berarti bagi penulis.
6. Bapak Mona Arif Muda, S.T., M.T, selaku Dosen Penguji yang telah memberikan kritik dan saran yang membangun demi penyempurnaan skripsi ini.

7. Seluruh Dosen dan Staf Program Studi Teknik Informatika Universitas Lampung yang telah memberikan ilmu, pengalaman, dan pelayanan akademik selama penulis menempuh studi.
8. Rekan-rekan mahasiswa Program Studi Teknik Informatika angkatan 2021 yang telah menemani perjalanan studi ini dengan kebersamaan, kerja sama, dan kenangan yang tidak akan terlupakan.
9. Seluruh asisten Laboratorium Teknik Komputer angkatan 2021, 2022, dan 2023 yang telah memberikan kerja sama, dukungan, dan kebersamaan selama proses penyusunan skripsi ini.
10. Semua pihak yang tidak dapat penulis sebutkan satu per satu, yang telah memberikan bantuan dan dukungan dalam penyelesaian skripsi ini.

Bandar Lampung, 1 Juni 2026

Penulis,



Yos Marison Sianipar
2115061074

DAFTAR ISI

DAFTAR ISI.....	i
DAFTAR TABEL	iv
DAFTAR GAMBAR	vi
I . PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Tujuan Penelitian	3
1.4 Manfaat Penelitian	4
1.5 Batasan Masalah.....	4
1.6 Sistematika Penulisan	5
II . TINJAUAN PUSTAKA.....	6
2.1 <i>Content Management System (CMS)</i>	6
2.2 Portal Berita	6
2.3 Berita	7
2.4 Alur Publikasi Berita.....	7
2.5 <i>Laravel</i>	8
2.6 <i>MySQL</i>	8
2.7 <i>React Js</i>	9
2.8 <i>Single Page Application (SPA)</i>	9
2.9 <i>Fonnte API WhatsApp Gateway</i>	9
2.10 <i>Virtual Private Server (VPS)</i>	10

2.11 <i>Black box Testing</i>	10
2.12 <i>User Acceptance Testing (UAT)</i>	11
2.13 <i>Unified Modeling Language (UML)</i>	13
2.14 <i>Rapid Application Development (RAD)</i>	14
2.15 K6.....	15
2.16 Penelitian Terkait	16
III . METODOLOGI PENELITIAN.....	30
3.1 Waktu dan Tempat	30
3.2 Alat dan Bahan Penelitian.....	31
3.2.1 Alat Penelitian.....	31
3.2.2 Bahan Penelitian.....	33
3.2.3 Bahan Pendukung Pengembangan Sistem	33
3.3 Tahapan Penelitian	35
3.3.1 <i>Requirement Planning</i>	36
3.3.2 <i>Design</i>	39
3.3.3 <i>Construction</i>	40
3.3.4 <i>Cutover</i>	41
IV . HASIL DAN PEMBAHASAN	45
4.1 Gambaran Umum Sistem	45
4.2 Arsitektur Sistem <i>Single Page Application (SPA)</i>	46
4.3 <i>Requirement Planning</i>	48
4.4 <i>Design</i> (Perancangan Sistem)	51
4.4.1 Arsitektur <i>Backend (Laravel)</i>	51
4.4.2 Arsitektur <i>Frontend (React + Inertia.js)</i>	51
4.4.3 Alur Publikasi Berita.....	53
4.4.4 <i>Use Case Diagram</i>	54

4.4.5 Activity Diagram	58
4.4.6 Entity Relational Diagram (ERD).....	72
4.4.7 Deployment diagram.....	74
4.5 Construction (Konstruksi).....	77
4.5.1 Persiapan Pembuatan sistem	77
4.5.2 Pembuatan Backend	78
4.5.3 Pembuatan Frontend (React)	99
4.5.4 Fitur-Fitur Khusus.....	104
4.5.5 Struktur Menu	108
4.5.6 Tampilan Antarmuka Sistem.....	111
4.6 Cutover (Implementasi)	126
4.6.1 Instalasi Sistem.....	126
4.6.2 Black box Testing	127
4.6.3 User Acceptance Test (UAT).....	131
4.6.4 Pengujian Performa (SPA vs MPA).....	136
4.6.5 Pengujian Kinerja Sistem.....	140
V . KESIMPULAN DAN SARAN.....	143
5.1 Kesimpulan	143
5.2 Saran.....	145
DAFTAR PUSTAKA	146

DAFTAR TABEL

Tabel 1. Skala <i>Likert</i>	11
Tabel 2. Persentase nilai interval	12
Tabel 3. Penelitian Terkait	23
Tabel 4. Waktu Pelaksanaan Penelitian	30
Tabel 5. Perangkat Keras	31
Tabel 6. Perangkat Lunak	31
Tabel 7. Bahan Penelitian	33
Tabel 8. Bahan Pendukung Pengembangan Sistem	33
Tabel 9. Skenario pengujian reporter	42
Tabel 10. Skenario pengujian editor	43
Tabel 11. Skenario pengujian admin.....	44
Tabel 12. Kebutuhan Fungsional	48
Tabel 13. Kebutuhan <i>Non-Fungsional</i>	49
Tabel 14. Indentifikasi Aktor	54
Tabel 15. Deskripsi <i>Use Case</i>	56
Tabel 16. <i>User</i>	79
Tabel 17. Kategori.....	80
Tabel 18. SubKategori	81
Tabel 19. Artikel	81
Tabel 20. Image.....	83
Tabel 21. Tag	84
Tabel 22. Artikel <i>Tag</i>	85
Tabel 23. <i>Web Setting</i>	85
Tabel 24. <i>Advert</i> (Iklan)	86
Tabel 25. <i>Advert Slot</i> (Slot Iklan)	87
Tabel 26. <i>Advert Setting</i> (Setting Iklan).....	88
Tabel 27. Daftar <i>Route</i> Utama	95

Tabel 28. Spesifikasi Server Produksi (VPS).....	126
Tabel 29. Hasil pengujian <i>black box</i> reporter	127
Tabel 30. Hasil pengujian <i>black box</i> editor.....	128
Tabel 31. Hasil pengujian <i>black box admin</i>	129
Tabel 32. Hasil Pengujian <i>Black box Testing</i>	130
Tabel 33. Daftar Pertanyaan Kuesioner <i>UAT</i>	132
Tabel 34. Hasil Kuesioner Uat.....	134
Tabel 35. Hasil Uji <i>Initial Load</i>	137
Tabel 36. Hasil Uji Perpindahan Navigasi.....	139
Tabel 37. Hasil pengujian kinerja sistem	141

DAFTAR GAMBAR

Gambar 1. Tahapan <i>Rapid application development</i>	14
Gambar 2. Tahapan Penelitian	35
Gambar 3. Arsitektur Sistem <i>Single Page Application (SPA)</i>	46
Gambar 4. Alur Proses Publikasi Berita	53
Gambar 5. <i>Use Case Diagram</i>	55
Gambar 6. <i>Activity Diagram usecase</i> Login ke Sistem	58
Gambar 7. <i>Activity Diagram</i> Membuat Berita Baru	59
Gambar 8. <i>Activity Diagram</i> Edit berita (Reporter).....	60
Gambar 9. <i>Activity Diagram</i> Edit berita (Admin/Redaktur).....	61
Gambar 10. <i>Activity Diagram</i> Melihat daftar Artikel	62
Gambar 11. <i>Activity Diagram</i> Melihat daftar Artikel	63
Gambar 12. <i>Activity Diagram</i> Manajemen Kategori	64
Gambar 13. <i>Activity Diagram</i> Manajemen SubKategori	65
Gambar 14. <i>Activity Diagram</i> Manajemen User (Tambah & Hapus).....	66
Gambar 15. <i>Activity Diagram</i> Manajemen User (Edit)	67
Gambar 16. <i>Activity Diagram</i> Laporan Berita	68
Gambar 17. <i>Activity Diagram</i> Manajemen Iklan	69
Gambar 18. <i>Activity Diagram</i> Manajemen Iklan (Pengaturan slot iklan).....	70
Gambar 19. <i>Activity Diagram</i> Manajemen Media	71
Gambar 20. <i>ERD</i> modul artikel, user, kategori, tag, gambar	72
Gambar 21. <i>ERD</i> Modul Iklan	73
Gambar 22. <i>Deployment diagram</i>	74
Gambar 23. Tampilan Terminal Lingkungan Pengembangan pada <i>WSL</i>	77
Gambar 24. Proses Instalasi Sistem	78
Gambar 25. Kode <i>Migration</i> tabel user	80

Gambar 26. Kode <i>Migration</i> tabel kategori	80
Gambar 27. Kode <i>Migration</i> tabel subkategori.....	81
Gambar 28. Kode <i>Migration</i> tabel Artikel.....	83
Gambar 29. Kode <i>Migration</i> tabel image	84
Gambar 30. Kode <i>Migration</i> tabel tag	84
Gambar 31. Kode <i>Migration</i> tabel artikel tag.....	85
Gambar 32. Kode <i>Migration</i> tabel web setting.....	86
Gambar 33. Kode <i>Migration</i> tabel <i>adverts</i>	87
Gambar 34. Kode <i>Migration</i> tabel <i>Advert Slot</i>	88
Gambar 35. Kode <i>Migration</i> tabel <i>advert setting</i>	89
Gambar 36. Kode Contoh Model Artikel.....	90
Gambar 37. Kode Artikel <i>Controller</i> fungsi <i>Index</i>	92
Gambar 38. Kode Artikel <i>Controller</i> fungsi <i>Create</i>	93
Gambar 39. Kode Artikel <i>Controller</i> Bagian <i>update</i>	94
Gambar 40. Contoh kode <i>middleware</i>	95
Gambar 41. Struktur Folder <i>frontend React</i>	99
Gambar 42. Kode Contoh <i>App Layout</i>	100
Gambar 43. kode <i>routing frontend</i>	101
Gambar 44. Kode Halaman Artikel dengan <i>Props</i> dari <i>Backend</i>	102
Gambar 45. <i>Form Handling</i>	103
Gambar 46. Kode Implementasi <i>Auto-save Frontend</i>	104
Gambar 47. Kode Implementasi <i>Auto-save Backend</i>	105
Gambar 48. Kode Implementasi <i>Fonte (Service Layer)</i>	106
Gambar 49. Kode Implementasi <i>Fonnte (Observer Layer)</i>	107
Gambar 50. Struktur Menu Admin	108
Gambar 51. Struktur Menu Editor	109
Gambar 52. Struktur Menu Reporter	110
Gambar 53. Halaman Login.....	111
Gambar 54. Halaman <i>Dashboard</i> Admin	112
Gambar 55. Halaman <i>Dashboard</i> Editor	112
Gambar 56. Halaman <i>Dashboard</i> Reporter	113
Gambar 57. Halaman Tulis Berita	113

Gambar 58. Tampilan fitur <i>Auto save</i>	114
Gambar 59. <i>Upload</i> Gambar	114
Gambar 60. Tampilan Notifikasi WhatsApp	115
Gambar 61. Halaman Artikel Saya (Index).....	115
Gambar 62. Halaman Artikel Saya (Statistik).....	116
Gambar 63. Halaman Berita terbit	116
Gambar 64. Halaman Berita Pending	117
Gambar 65. Halaman Berita Terjadwal	117
Gambar 66. Halaman Berita Dihapus	118
Gambar 67. Halaman laporan produksi berita berdasarkan penulis	118
Gambar 68. Halaman laporan keseluruhan produksi berita	119
Gambar 69. Halaman List Kategori	119
Gambar 70. Halaman List Iklan	120
Gambar 71. Halaman Pengaturan Iklan	120
Gambar 72. Tampilan Pop up Ubah Slot Iklan	121
Gambar 73. Halaman <i>List</i> Gambar	121
Gambar 74.. Halaman List User.....	122
Gambar 75. Halaman Tambah User	122
Gambar 76. Halaman Detail User	123
Gambar 77. Halaman Pengaturan Berita.....	123
Gambar 78. Homepage Demo Berita	124
Gambar 79. Halaman Demo Detail Artikel	124
Gambar 80. Tampilan Iklan pada Halaman Demo Detail Artikel	125
Gambar 81. Pengambilan Data Awal (<i>Initial Load</i>)	136
Gambar 82. Perpindahan Navigasi.....	138

I. PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi informasi telah mendorong transformasi besar dalam cara masyarakat mengakses dan menyebarkan informasi. Media daring atau portal berita menjadi sarana utama dalam mendistribusikan berita secara cepat dan langsung kepada publik. Di balik sistem distribusi berita digital tersebut, terdapat *Content Management System (CMS)* yang berperan sebagai komponen penting dalam mendukung proses editorial, karena sistem ini mengintegrasikan berbagai fungsi yang diperlukan mulai dari penulisan, revisi, hingga penerbitan artikel. Proses tersebut melibatkan peran yang beragam, seperti reporter yang bertugas untuk mencari dan mengumpulkan informasi, redaktur sebagai editor yang menyunting dan mempublikasikan artikel, serta administrator yang mengatur hak akses dan pengelolaan sistem [1].

Pengelolaan berita membutuhkan sistem manajemen konten (*CMS*) yang dapat mendukung alur kerja redaksional dengan baik. Idealnya, sistem ini tidak hanya memfasilitasi pembuatan dan penyuntingan konten terstruktur tetapi juga memerlukan fitur-fitur penting seperti manajemen konten, manajemen *user* dan manajemen iklan, serta kemampuan analitik untuk memantau produksi artikel.

Namun, implementasi *CMS* pada media lokal masih menghadapi beberapa kendala signifikan. Berdasarkan hasil wawancara dengan redaktur dan reporter di salah satu media lokal, diketahui bahwa *CMS* yang digunakan belum sepenuhnya mendukung alur kerja redaksional yang ada. Beberapa permasalahan utama yang muncul antara lain terbatasnya fitur penyimpanan otomatis yang menyebabkan artikel hilang ketika terjadi kesalahan atau koneksi terputus, format penulisan yang kurang memadai untuk kebutuhan redaksi, antarmuka *CMS* yang kurang terstruktur dan sulit digunakan, tidak adanya fitur yang menampilkan jumlah pembaca (*views*) per

artikel, belum adanya notifikasi otomatis ketika artikel baru masuk ke *CMS*, serta perlunya penerapan sistem berbasis *Single Page Application (SPA)*.

Selain itu, berdasarkan hasil observasi langsung terhadap pengelolaan *CMS* yang digunakan oleh portal berita yang saat ini beroperasi, ditemukan pula beberapa permasalahan tambahan, yaitu pengelolaan hak akses (*role-based access control*) yang hanya diterapkan pada antarmuka *frontend* saja. Akibatnya, reporter masih bisa mengakses fitur-fitur khusus redaktur dan admin jika langsung menggunakan *URL* tertentu. Di samping itu, belum tersedianya laporan atau data jumlah produksi artikel per pengguna membuat pengelolaan kinerja redaksi sulit dilakukan, serta fitur integrasi data *Google Analytics* yang tidak berfungsi sebagaimana mestinya, menyebabkan tim redaksi kesulitan dalam memperoleh analisis pengunjung dan performa artikel secara akurat.

Temuan ini diperkuat kajian literatur yang menunjukkan meskipun terdapat variasi alur kerja antar media, pola dasar proses redaksional (penulisan → penyuntingan → publikasi) pada umumnya memiliki alur yang mirip [1]. Meskipun terdapat perbedaan pada struktur organisasi atau penggunaan teknologi, sebagian besar media tetap memerlukan *CMS* yang mampu mendukung pengaturan alur kerja redaksional secara terstruktur serta mudah digunakan oleh seluruh tim yang terlibat dalam proses penerbitan berita.

Meskipun tersedia *CMS* berbasis *open source* masih banyak digunakan oleh media daring, seperti *WordPress* yang menawarkan kemudahan penggunaan dan biaya yang relatif rendah, *CMS open source* memiliki sejumlah kelemahan seperti kerentanan terhadap peretasan, keterbatasan dalam pengembangan fitur khusus, serta ketergantungan pada plugin pihak ketiga yang keamanannya tidak selalu terjamin. Masalah ini berdampak pada fleksibilitas, keamanan, dan keberlanjutan sistem dalam mendukung operasional media digital [2].

Untuk mengatasi permasalahan yang telah diidentifikasi, sistem *Content Management System (CMS)* portal berita dirancang menggunakan *framework Laravel* untuk sisi *backend* dan *React* untuk sisi *frontend*, dengan menerapkan pendekatan *Single Page Application (SPA)*. *Laravel* dipilih karena kemampuannya dalam mengelola autentikasi pengguna yang kompleks dan keamanan data,

sementara *React* memungkinkan pembuatan antarmuka *SPA* yang responsif untuk pengalaman pengguna yang lebih cepat. Sistem ini dikembangkan khusus untuk mendukung alur kerja redaksional media lokal, mencakup proses penulisan, penyuntingan, hingga publikasi artikel secara terstruktur. Sistem ini diharapkan dapat menjawab keterbatasan sistem sebelumnya serta menawarkan alternatif desain yang lebih adaptif terhadap kebutuhan redaksional.

1.2 Rumusan Masalah

1. Fitur-fitur apa saja yang dibutuhkan *Content Management System (CMS)* untuk mendukung alur kerja redaksional portal berita?
2. Bagaimana merancang dan membangun *Content Management System (CMS)* berbasis *Laravel* dan *React* dengan pendekatan *Single Page Application (SPA)* untuk mendukung alur kerja redaksional secara terstruktur?
3. Bagaimana menguji *Content Management System CMS* portal berita yang dikembangkan untuk memastikan bahwa fitur-fitur utama berjalan sesuai dengan kebutuhan pengguna?
4. Bagaimana perbandingan performa kecepatan sistem berbasis *Single Page Application (SPA)* dengan sistem berbasis *Multi Page Application (MPA)* berdasarkan waktu akses awal (*initial load*) dan waktu perpindahan navigasi halaman?

1.3 Tujuan Penelitian

Adapun tujuan penelitian ini adalah sebagai berikut

1. Mengidentifikasi fitur-fitur utama yang dibutuhkan *Content Management System (CMS)* dalam mendukung alur kerja redaksional pada portal berita.
2. Merancang dan membangun *CMS* berbasis *Laravel (backend)* dan *React (frontend)* menggunakan pendekatan *Single Page Application (SPA)*, guna mendukung alur kerja redaksional portal berita secara terstruktur.
3. Melakukan pengujian terhadap sistem *CMS* yang dikembangkan untuk

mengevaluasi kelayakan fungsionalitas dan kesesuaian sistem dengan kebutuhan pengguna menggunakan metode *Black box Testing* dan *User Acceptance Test (UAT)*.

4. Menganalisis perbandingan performa kecepatan antara sistem berbasis *Single Page Application (SPA)* dan *Multi Page Application (MPA)* berdasarkan pengujian waktu *initial load* dan navigasi antar halaman.

1.4 Manfaat Penelitian

Adapun manfaat penelitian ini adalah sebagai berikut

1. Hasil penelitian ini diharapkan dapat menyediakan solusi berupa *Content Management System (CMS)* yang sesuai dengan kebutuhan proses redaksional portal berita, dilengkapi fitur pendukung kolaborasi antar peran dalam tim redaksi.
2. Menambah referensi penelitian di bidang rekayasa perangkat lunak, khususnya terkait *CMS* dan aplikasi web berbasis *Laravel* dan *React JS*.
3. Dapat digunakan sebagai landasan atau referensi bagi pengembangan lanjutan jika sistem serupa akan diadaptasi oleh media lain yang memiliki kebutuhan serupa di masa mendatang.

1.5 Batasan Masalah

Batasan masalah dalam penelitian ini adalah sebagai berikut:

Fokus penelitian dibatasi pada identifikasi, perancangan, dan pengembangan fitur-fitur *Content Management System (CMS)* yang dibutuhkan oleh tim redaksi portal berita, dengan menggunakan *framework Laravel* versi 12 sebagai *backend* dan *React JS* sebagai *frontend* dengan pendekatan *Single Page Application (SPA)*.

1.6 Sistematika Penulisan

Sistematika yang digunakan dalam penulisan penelitian ini adalah sebagai berikut

BAB I : Pendahuluan

Bab ini memuat latar belakang dibuatnya *Content Management System (CMS)*, rumusan masalah penelitian, tujuan penelitian, manfaat penelitian, batasan masalah penelitian dan sistematika penulisan

BAB II : TINJAUAN PUSTAKA

Bab ini berisi teori-teori yang mendukung penelitian, termasuk kajian terkait *CMS*, portal berita, teknologi *Laravel* dan *React*, pendekatan *Single Page Application (SPA)*, serta penelitian terdahulu yang relevan sebagai dasar pengembangan sistem.

BAB III : METODOLOGI PENELITIAN

Bab ini menjelaskan metode yang digunakan dalam penelitian, yaitu pendekatan *Rapid Application Development (RAD)* yang terdiri dari tahapan perencanaan kebutuhan (*requirement planning*), desain (*User design*), konstruksi sistem (*Construction*), dan implementasi (*Cutover*).

BAB IV : HASIL DAN PEMBAHASAN

Bab ini menyajikan hasil implementasi sistem *CMS* yang dikembangkan, mencakup fitur-fitur utama yang telah dirancang dan dibangun sesuai dengan kebutuhan pengguna. Selain itu, dilakukan pembahasan mengenai hasil pengujian dan kesesuaian sistem terhadap tujuan yang telah ditetapkan.

BAB V : KESIMPULAN DAN SARAN

Bab ini memuat kesimpulan dari hasil penelitian yang telah dilakukan serta saran-saran untuk pengembangan sistem.

II. TINJAUAN PUSTAKA

2.1 *Content Management System (CMS)*

Content Management System (CMS) adalah aplikasi komputer (baik berbasis desktop maupun cloud) yang memungkinkan pengguna untuk mengatur, membuat, mengedit, dan mempublikasikan konten secara mudah di sebuah situs *web*. *CMS* menggunakan basis data dan sistem *file* untuk menyimpan konten, sehingga memungkinkan skalabilitas mulai dari situs kecil hingga platform berita besar dengan jutaan halaman. *CMS* memiliki struktur kerja yang menyerupai produksi buku, dengan beberapa peran utama seperti *Author* (Penulis): membuat artikel atau konten yang akan ditampilkan di situs. *Editor* (Penyunting): menyunting konten agar sesuai dengan gaya dan konteks situs, termasuk memperbaiki ejaan dan tata bahasa. *Editor* juga biasanya bertindak sebagai publisher yang menandai konten siap untuk dipublikasikan. *Administrator*: mengelola sisi *backend* situs, seperti pengaturan peran pengguna, pemeliharaan konten, backup, dan update sistem. Dalam beberapa kasus, admin juga merangkap sebagai *developer* [3].

2.2 Portal Berita

Portal berita adalah situs digital yang menyajikan berbagai informasi dan berita terbaru kepada pembaca. Di Indonesia, pertumbuhan portal berita sangat pesat seiring dengan perkembangan teknologi informasi dan komunikasi, yang membuat masyarakat lebih mudah mengakses informasi. Menurut Dewan Pers (2019), ada lebih dari 43.000 situs berita di Indonesia, meskipun hanya sekitar 2.700 yang sudah terverifikasi. Hal ini menunjukkan betapa cepatnya pertumbuhan media berita online, yang dipengaruhi oleh meningkatnya kebebasan pers sejak era Reformasi dan meluasnya penggunaan perangkat seluler di masyarakat [4].

2.3 Berita

Berita adalah suatu peristiwa yang terpengaruh oleh unsur waktu yang menarik perhatian masyarakat. Berita tidak hanya dipahami secara sempit sebagai sekadar laporan peristiwa, melainkan sebagai hasil dari kerja jurnalistik yang jujur, kritis, dan bermakna. Berita adalah hasil dari proses verifikasi yang ketat, riset mendalam, dan wawancara yang serius, bukan sekadar menyalin pernyataan atau mengikuti arus informasi. Berita harus memiliki makna bagi publik, bukan hanya menyampaikan apa yang terjadi, tetapi juga menjelaskan mengapa hal itu penting [5].

2.4 Alur Publikasi Berita

Alur publikasi berita pada media daring diawali dengan tahap perencanaan liputan yang dilakukan oleh tim redaksi. Pada beberapa media, perencanaan tersebut dilaksanakan secara terstruktur melalui rapat redaksi mingguan yang membahas agenda liputan mendatang. Namun, terdapat pula media yang menerapkan pola perencanaan yang lebih reaktif, yaitu dengan menyesuaikan liputan berdasarkan isu-isu aktual yang sedang menjadi perbincangan di masyarakat (*trending topic*).

Setelah agenda liputan ditetapkan, wartawan ditugaskan untuk melaksanakan proses peliputan. Informasi dikumpulkan melalui berbagai metode, seperti reportase langsung di lapangan, serta pengumpulan data dari siaran pers atau rilis resmi. Berikutnya, wartawan menyusun naskah berita berdasarkan hasil liputan. Setelah selesai ditulis, naskah dikirim ke redaksi melalui sistem manajemen konten atau email untuk menjalani tahap penyuntingan (*editing*). Pada tahap ini, redaktur sebagai *editor* memiliki peran penting dalam meninjau, memperbaiki, dan menyempurnakan isi berita. Setelah melalui proses penyuntingan dan dinyatakan layak tayang, redaktur bertanggung jawab dalam menerbitkan (*publish*) berita ke platform media daring agar dapat diakses oleh publik [1].

2.5 Laravel

Laravel adalah sebuah *framework* pengembangan aplikasi web berbasis bahasa pemrograman *PHP* yang bersifat *open-source*. *Framework* ini dirancang untuk membantu pengembang membangun aplikasi web modern secara lebih cepat dan efisien, dengan menyediakan berbagai fitur yang menyederhanakan proses pengembangan. *Laravel* mendukung pembuatan aplikasi berskala kecil hingga besar dengan pendekatan *full-stack*, serta mengadopsi arsitektur *Model-View-Controller (MVC)* untuk memisahkan logika bisnis dan tampilan aplikasi. Beberapa fitur utama *Laravel* antara lain adalah *routing URL*, sistem autentikasi dan otorisasi, *Eloquent ORM* untuk interaksi basis data, serta sistem penjadwalan tugas (*scheduling*) dan antrean pesan (*queue system*). *Laravel* juga mendukung pengiriman email melalui berbagai layanan (seperti *SMTP*, *Mailgun*, dan *Amazon SES*), serta memungkinkan penggunaan *seeders* untuk mengisi data awal ke dalam tabel basis data [6].

2.6 MySQL

MySQL adalah sistem manajemen basis data relasional (*Relational Database Management System/RDBMS*) yang populer karena kecepatan, keandalan, dan kemudahan penggunaannya. *MySQL* menggunakan arsitektur *client-server*, di mana server bertugas mengelola *Database* dan menjalankan perintah *SQL*, sementara *client* berfungsi sebagai antarmuka untuk mengirim perintah dan menerima hasilnya. *MySQL* dapat berjalan pada berbagai sistem operasi, termasuk berbagai varian *Linux* dan *Windows*, serta dapat diakses menggunakan berbagai bahasa pemrograman seperti *PHP*, *Python*, *Java*, dan lainnya.

MySQL sangat cocok digunakan untuk berbagai skala aplikasi, mulai dari proyek kecil hingga sistem berskala besar yang digunakan oleh perusahaan-perusahaan besar dunia. Sistem ini mendukung fitur-fitur utama basis data relasional seperti indeks, *join*, dan transaksi, yang membuatnya handal dalam mengelola data secara efisien. Selain itu, *MySQL* juga bersifat *open source*, sehingga banyak pengembang dan perusahaan yang menggunakannya secara bebas dan dapat berkontribusi dalam pengembangannya [7].

2.7 React Js

React Js adalah sebuah *library JavaScript* yang digunakan untuk membangun antarmuka pengguna (*user interfaces*) pada aplikasi web. Dikembangkan oleh *Facebook*, *React* menjadi sangat populer karena kemampuannya dalam membuat aplikasi web yang dinamis dan responsif. Berbeda dari *framework* besar lainnya, *React* tidak menyediakan banyak komponen antarmuka seperti tombol, *grid*, atau *slider* secara default, juga tidak memaksakan struktur aplikasi tertentu. *React* justru sangat minimalis dan fleksibel. Salah satu fitur utama dari *React* adalah konsep komponen, yang memungkinkan pengembang untuk membangun *UI* dengan cara yang modular dan terstruktur. Dengan menggunakan komponen, pengembang dapat membagi antarmuka menjadi bagian-bagian yang lebih kecil dan dapat digunakan kembali, sehingga mempermudah pengelolaan dan pengembangan aplikasi kompleks [8].

2.8 Single Page Application (SPA)

Single Page Application (SPA) adalah jenis aplikasi web yang dijalankan di *browser* dan tidak memuat ulang seluruh halaman selama penggunaannya. Tidak seperti aplikasi web tradisional yang sering berpindah antar halaman dan melakukan pemuatan ulang dari server, *SPA* hanya memuat satu halaman *HTML* utama dan secara dinamis memperbarui kontennya melalui *JavaScript* saat pengguna berinteraksi. Tujuan utama *SPA* adalah membantu pengguna menyelesaikan tugas, seperti menulis dokumen atau mengelola server, dengan pengalaman yang cepat dan mulus layaknya aplikasi desktop [9].

2.9 Fonnte API WhatsApp Gateway

Fonnte merupakan layanan *Unofficial WhatsApp API Gateway* yang memungkinkan integrasi otomatis pengiriman pesan *WhatsApp* melalui protokol *REST API*. Sistem ini bekerja dengan memanfaatkan mekanisme *WhatsApp Web* untuk mengirim pesan secara otomatis, tanpa perlu perangkat lunak tambahan atau server *WhatsApp* resmi [10].

Penggunaan *Fonnte* dalam sistem informasi seperti *Content Management System (CMS)* memungkinkan notifikasi dikirim secara instan ketika peristiwa tertentu terjadi, misalnya saat artikel baru diajukan. Pengembang cukup mengirim permintaan POST ke *endpoint API* milik *Fonnte* dengan menyertakan parameter penting seperti nomor tujuan (*target*) dan isi pesan (*message*), serta *token otentikasi* yang diperoleh dari akun *Fonnte*.

2.10 Virtual Private Server (VPS)

Virtual Private Server (VPS) adalah layanan server virtual yang memungkinkan pengguna mengelola sistem layaknya server fisik, namun tanpa perlu menangani perangkat keras secara langsung. *VPS* hadir sebagai solusi lebih terjangkau dibanding menyewa ruang di data center secara fisik, yang membutuhkan biaya besar dan dukungan teknis tinggi. Sejak tahun 2005, *VPS* menjadi pilihan populer karena menawarkan kontrol penuh terhadap server, akses jarak jauh, dan efisiensi dalam pengelolaan aplikasi berbasis web [11].

2.11 Black box Testing

Black box Testing merupakan salah satu metode pengujian perangkat lunak yang berfokus pada fungsi sistem berdasarkan masukan (*input*) dan keluaran (*output*), tanpa memeriksa struktur internal atau kode sumber aplikasi. Penguji tidak perlu mengetahui bagaimana sistem dikembangkan secara teknis, melainkan hanya memverifikasi bahwa sistem merespons masukan sebagaimana mestinya. Beberapa teknik umum dalam *Black box Testing* meliputi *equivalence partitioning*, di mana input dibagi ke dalam kelompok valid dan invalid, *boundary value analysis*, yaitu pengujian pada nilai batas input, *cause-effect graphing*, yang menghubungkan kondisi input dengan hasil output melalui logika *boolean*, *decision table based testing*, untuk menguji kombinasi kondisi dan keputusan dalam sistem, serta *error guessing*, yaitu pengujian berdasarkan asumsi dan pengalaman penguji terhadap potensi kesalahan yang mungkin terjadi [12].

2.12 User Acceptance Testing (UAT)

User Acceptance Testing (UAT) adalah salah satu fase krusial dalam pengujian perangkat lunak yang dilakukan untuk memastikan bahwa sistem yang dikembangkan memenuhi kebutuhan dan ekspektasi pengguna akhir. *UAT* biasanya dilakukan oleh pengguna sebenarnya yang berinteraksi langsung dengan sistem, termasuk staf atau pegawai perusahaan. Tujuannya adalah untuk memverifikasi apakah fungsi yang ada beroperasi sesuai dengan keinginan pengguna, dengan fokus pada pengalaman dan kebutuhan nyata dari pengguna. *UAT* memberikan beberapa keuntungan, antara lain menjaga objektivitas, memahami interaksi manusia, dan menyediakan lingkungan pengujian yang realistis. Dalam konteks ini, pengguna memberikan umpan balik berharga tentang fungsionalitas sistem dan apakah sistem tersebut sesuai dengan harapan mereka [13].

Dalam penelitian ini, pengujian UAT dilakukan menggunakan instrumen kuesioner berbasis Skala *Likert* 1–5 untuk mengukur tingkat penerimaan dan persepsi pengguna terhadap sistem yang dikembangkan. Skala *Likert* yang digunakan dalam penelitian ini ditunjukkan pada Tabel 1, terdiri dari lima tingkat penilaian mulai dari Sangat Tidak Setuju (1) hingga Sangat Setuju (5).

Tabel 1. Skala *Likert*

Skor	Kategori
5	Sangat Setuju (SS)
4	Setuju (S)
3	Netral (N)
2	Tidak Setuju (TS)
1	Sangat Tidak Setuju (STS)

Rumus Menghitung *User Acceptance Testing (UAT)*[14].

Perhitungan nilai *UAT* dilakukan menggunakan Persamaan (1), sedangkan persentase tingkat penerimaan sistem dihitung menggunakan Persamaan (2).

$$Qn = \sum_{i=1}^5 F(i) * scale(i) \quad (1)$$

$$P = \left(\frac{Total\ Qn}{N} \right) / 5 \times 100\% \quad (2)$$

Keterangan:

Qn = Pertanyaan (1,2,3.....n)

n = 1,2,3...10

F = Frekuensi Jawaban

Scale = Skala Likert

P = Persentase

N = Total Responden

Untuk rentang nilai kriteria interval ditunjukkan pada Tabel 2 di bawah ini:

Tabel 2. Persentase nilai interval

Skor	Keterangan
0% - 19,99%	Sangat Tidak Baik
20% - 39,99%	Kurang Baik
40% - 59,99%	Netral
60% - 79,99%	Baik
80% -100,00%	Sangat Baik

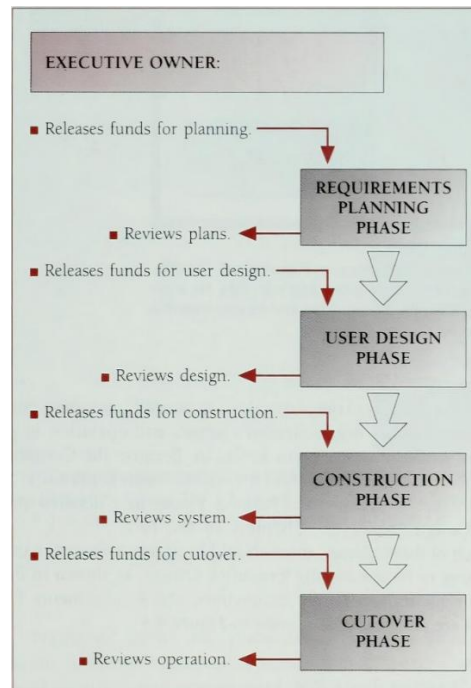
2.13 Unified Modeling Language (UML)

Unified Modeling Language (UML) adalah sebuah bahasa pemodelan standar yang digunakan dalam rekayasa perangkat lunak untuk memvisualisasikan, mendefinisikan, membangun, dan mendokumentasikan sistem secara terstruktur. *UML* memungkinkan para pengembang dan pemangku kepentingan untuk berkomunikasi secara lebih efektif tentang desain dan fungsionalitas sistem. Di dalam *UML* terdapat berbagai jenis diagram, di antaranya adalah *Use Case Diagram* dan *Activity diagram*, yang masing-masing memiliki fungsi spesifik dalam konteks pengembangan perangkat lunak [15].

1. *Use Case Diagram* digunakan untuk menggambarkan interaksi antara aktor dengan sistem yang sedang dikembangkan. Diagram ini merepresentasikan fungsionalitas-fungsionalitas utama sistem dari sudut pandang pengguna dan berfungsi sebagai dasar dalam merancang antarmuka serta fitur sistem. Komponen utama dalam *Use Case Diagram* meliputi aktor yaitu pihak yang berinteraksi dengan sistem, *use case* sebagai fungsi atau layanan yang disediakan oleh sistem dan relasi sebagai koneksi antara aktor dan *use case*, yang menunjukkan interaksi yang terjadi.
2. *Activity Diagram* digunakan untuk menggambarkan alur kerja atau proses aktivitas dalam sistem, baik dari sisi pengguna maupun sistem internal. Diagram ini bersifat dinamis karena menampilkan urutan aktivitas, percabangan logika, serta kondisi awal dan akhir dari suatu proses. Komponen penting dalam *Activity Diagram* meliputi *initial node* (titik awal), *activity/action* (langkah-langkah yang dilakukan), *decision node* (percabangan logika), *merge node*, dan *join* (*proses paralel*) serta *final node* (titik akhir).

2.14 Rapid Application Development (RAD)

Rapid Application Development (RAD) adalah suatu siklus pengembangan perangkat lunak yang dirancang untuk menghasilkan sistem dengan waktu yang jauh lebih cepat dan kualitas yang lebih tinggi dibandingkan pendekatan pengembangan tradisional [16].



Gambar 1. Tahapan *Rapid application development*

RAD terdiri dari empat tahapan utama, yaitu: *Requirements Planning*, *User Design*, *Construction*, dan *Cutover*. Setiap tahapan dirancang untuk mempercepat siklus pengembangan tanpa mengorbankan kualitas dan fleksibilitas sistem.

1. *Requirements Planning*

Requirements Planning merupakan tahap awal yang berfungsi untuk menyelaraskan kebutuhan bisnis dengan solusi teknis. Martin menjelaskan bahwa fase ini melibatkan diskusi terstruktur antara pengguna akhir dan profesional sistem informasi untuk mendefinisikan tujuan dan fungsi utama sistem.

2. *User Design Phase*

Pada tahap ini, pengguna tidak sekadar menjadi objek uji coba, tetapi turut aktif

dalam merancang prototipe sistem. Martin menekankan pentingnya kolaborasi melalui *Joint Application Design (JAD) workshop*, di mana output desain direpresentasikan langsung dalam alat bantu seperti *I-CASE tool*.

3. *Construction Phase*

Tahap ini berfokus pada pengembangan sistem secara cepat berdasarkan desain yang telah dibuat sebelumnya. Tidak seperti model tradisional yang memisahkan antara desain dan pengkodean, *RAD* mengintegrasikan keduanya. Martin menyoroti bahwa pengguna tetap dilibatkan dalam validasi tampilan dan fungsi transaksi selama pembangunan berlangsung.

4. *Cutover*

Cutover merupakan tahap implementasi akhir, termasuk pelatihan pengguna, perpindahan sistem, serta pengujian menyeluruh. Martin menekankan bahwa karena pembangunan dilakukan secara cepat, maka perencanaan *Cutover* juga harus dimulai sejak awal, terutama untuk memastikan transisi yang mulus dari sistem lama ke sistem baru.

2.15 K6

K6 merupakan salah satu tools load testing yang digunakan untuk melakukan pengujian kinerja (performance testing) pada aplikasi berbasis web, termasuk API dan website. Pengujian menggunakan k6 dilakukan dengan cara mensimulasikan sejumlah pengguna virtual (*Virtual Users/VU*) yang mengirim permintaan (*request*) ke *server* secara bersamaan untuk mengukur ketahanan dan performa sistem pada kondisi beban tertentu.

Penggunaan k6 sebagai alat pengujian kinerja juga diterapkan dalam penelitian yang membandingkan performa *web server Nginx* dan *Caddy* untuk aplikasi *PHP*. Pada penelitian tersebut, pengujian dilakukan menggunakan tiga skenario beban kerja, yaitu beban ringan (*50 VU*), beban sedang (*500 VU*), dan beban berat (*2000 VU*). Pengujian tersebut mengukur metrik kinerja seperti latensi, waktu respon, *throughput*, serta penggunaan sumber daya *server (CPU dan RAM)* untuk menganalisis performa dan stabilitas web server pada berbagai tingkat beban [17].

2.16 Penelitian Terkait

Penelitian ini mengacu pada beberapa penelitian sebelumnya yang berkaitan dengan pengembangan sistem manajemen konten (*CMS*).

Penelitian oleh Anwar dan Sarjono (2022) membahas perancangan sistem portal berita untuk mendukung kebutuhan penyampaian informasi secara cepat oleh stasiun televisi lokal JEKTV. Penelitian ini menggunakan metode *Prototype* untuk pengembangan sistem, yang memungkinkan adanya interaksi langsung antara pengembang dan stakeholder dalam proses iteratif perancangan sistem. Sistem yang dikembangkan memiliki fitur-fitur utama seperti manajemen berita, pengelolaan *user*, serta kolom komentar bagi pengunjung yang sudah terdaftar. Fitur-fitur ini mendukung proses penyajian berita yang sebelumnya melalui 8 tahapan prosedur menjadi hanya 3 tahapan, sehingga meningkatkan efisiensi dan kecepatan distribusi informasi. Selain itu, sistem dirancang dengan memperhatikan kebutuhan fungsional (seperti kemampuan menambah, mengedit, dan menghapus berita) dan non-fungsional (*usability, security, flexibility*) [18].

Penelitian oleh Roby et al. (2022) membahas pengembangan *CMS* berbasis *Laravel* menggunakan metode *Rational Unified Process (RUP)* yang digunakan oleh perangkat daerah Kabupaten Garut. Sistem yang dikembangkan memiliki fitur manajemen berita, galeri, tema, dan identitas situs yang dapat diubah tanpa mengedit kode sumber, serta fitur pesan untuk meningkatkan interaksi antara masyarakat dan instansi pemerintahan. Meskipun sistem ini fleksibel secara tampilan, namun masih berbasis server-side rendering dan belum menerapkan arsitektur *Single Page Application (SPA)* serta belum menggunakan *framework frontend* modern seperti *React* sebagaimana disarankan di bagian saran penelitian mereka. [19] .

Penelitian oleh Yuricha dan Phan (2023) bertujuan untuk mengimplementasikan pengamanan akses data dalam sistem *SCM* menggunakan konsep *Role-Based Access Control (RBAC)*. Penelitian ini menggunakan metode *Waterfall* dan dikembangkan dengan *framework Laravel 9* serta *library Laravel SPatie* untuk mengelola hak akses berdasarkan peran pengguna. Hasilnya, sistem mampu memberikan pembatasan akses yang detail dan dinamis bagi setiap peran dalam

berbagai modul seperti pembelian, penjualan, gudang, dan keuangan [20].

Penelitian oleh Santoso (2021) membahas penerapan teknik *SPA* dalam merancang layout halaman web agar lebih cepat dan efisien. Penelitian ini bertujuan mengoptimalkan kecepatan render halaman web dengan memanfaatkan *ReactJS* sebagai *library frontend* dan *Bootstrap* sebagai *framework CSS responsif*. Dengan menggunakan metode studi literatur dan implementasi langsung, sistem yang dibangun menunjukkan perbedaan signifikan dari sisi performa antara layout berbasis *SPA* dan layout tradisional. Hasil pengujian menunjukkan bahwa penggunaan *React* dengan teknik *SPA* mampu mempercepat perpindahan halaman karena tidak memuat ulang seluruh halaman, melainkan hanya memperbarui konten yang diperlukan. Kelebihan dari penelitian ini adalah keberhasilan menerapkan arsitektur *SPA* yang modern dan ramah pengguna (*user-friendly*) [21].

Penelitian oleh Devianto dan Dwiasnati (2021) bertujuan untuk membangun portal berita *E-News* yang menyajikan informasi seputar hasil pertanian dari wilayah pedesaan. Penelitian ini menggunakan metode gabungan *SDLC* dan *prototyping*, serta dirancang khusus untuk mendukung penyebaran informasi pertanian secara luas kepada masyarakat. Sistem dikembangkan dengan fitur-fitur *frontend* dan *backend*, termasuk manajemen berita, kategori, galeri, otorisasi pengguna. Kelebihan dari sistem ini adalah adanya proses verifikasi berita oleh editor, dan penyajian konten spesifik (bidang pertanian [22].

Penelitian oleh Ali et al. (2022) bertujuan untuk membangun *CMS* yang mendukung pengelolaan konten hasil riset kendaraan listrik otonom oleh mahasiswa dan dosen. Sistem dikembangkan menggunakan metode *Software Development Life Cycle (SDLC)* dengan pendekatan *prototyping*, serta mengimplementasikan fitur *dashboard*, manajemen artikel, dan pengelolaan kontributor. Pengujian dilakukan menggunakan teknik *black-box testing* untuk memastikan fungsionalitas sistem berjalan sesuai harapan. Kelebihan dari sistem ini adalah kemampuan pengelolaan konten dinamis yang cukup lengkap dan *user-friendly*, termasuk fitur registrasi, *login*, dan pengelolaan data multimedia seperti video dan artikel [23].

Penelitian oleh Priskila et al. (2022) bertujuan untuk merancang dan membangun portal berita berbasis web yang lebih efektif dibandingkan sistem sebelumnya yang masih menggunakan *WordPress* dengan pelaporan berita melalui *WhatsApp*. Penelitian ini menggunakan metode *Waterfall*, dengan tahapan sistematis mulai dari analisis kebutuhan, desain sistem, implementasi, hingga pengujian dan pemeliharaan. Sistem yang dihasilkan menyediakan fitur *login* peran (admin, redaktur, wartawan, admin iklan), pengelolaan berita, manajemen kategori, serta pengaturan iklan. Kelebihan dari sistem ini adalah proses pengelolaan berita menjadi lebih sistematis dan terintegrasi, serta mampu menggantikan proses manual sebelumnya [24].

Penelitian oleh Maulana (2025) bertujuan menganalisis kelebihan dan kekurangan penggunaan *CMS open source*, khususnya *WordPress*, dalam mendukung bisnis portal berita. Menggunakan metode kualitatif deskriptif, peneliti mengumpulkan data primer melalui wawancara dengan redaksi media serta menganalisis 10 portal berita yang menggunakan *CMS open source* dan *close source*. Hasil penelitian menunjukkan bahwa meskipun *CMS open source* seperti *WordPress* murah dan mudah digunakan, sistem ini memiliki keterbatasan dari sisi pengembangan, keamanan, manajemen data sponsor, dan manajemen risiko bisnis jangka panjang [2].

Penelitian oleh Urmila et al. (2023) bertujuan untuk meningkatkan jumlah pengunjung portal berita Pikiran-Rakyat.com melalui optimasi mesin pencari. Penelitian ini menggunakan metode *SEO Web Design Methodology* yang mencakup enam tahap: *planning*, *analysis*, *Design*, *implementation*, *testing*, dan *maintenance*, dengan penerapan teknik *On-Page SEO* (judul, konten, *meta tag*) dan *Off-Page SEO* (backlink, traffic monitoring dengan *Google Analytics*). Hasil penelitian menunjukkan bahwa penerapan strategi *SEO* yang tepat secara signifikan dapat menaikkan peringkat situs pada halaman pertama mesin pencari *Google* [25].

Penelitian oleh Thoha et al (2025) bertujuan untuk membuat Sistem Informasi Geografis (SIG) yang memanfaatkan teknologi web untuk memetakan sekolah di

Kecamatan Entikong. Relevansi penelitian ini dengan topik *CMS* terletak pada penggunaan teknologi *Laravel* dan *React.Js* untuk membangun aplikasi web. Thoha et al. menggunakan *Laravel* sebagai *backend* dan *React.Js* sebagai *frontend*, yang juga merupakan teknologi yang dipakai dalam topik penelitian *CMS*. Selain itu, penelitian ini juga menerapkan pendekatan pengembangan perangkat lunak dengan metode seperti perencanaan sistem menggunakan *flowchart* dan *UML*, serta membuat desain wireframe dengan mockflow tools. Hasil dari penelitian ini adalah suatu sistem yang memudahkan pengguna untuk mencari informasi tentang sekolah dan menampilkan lokasi geografisnya. Dalam proses pengembangan, Thoha et al. juga melakukan pengujian sistem menggunakan *blackbox testing* [26].

Penelitian oleh Susanto et al. mengembangkan sistem pengaduan masyarakat menggunakan metode *Rapid Application Development (RAD)* dan *framework Laravel*. Penelitian ini bertujuan untuk meningkatkan efisiensi dan efektivitas pengelolaan data pengaduan di Desa Logung. Hasil penelitian menunjukkan bahwa sistem yang dikembangkan dapat mempermudah masyarakat dalam mengajukan pengaduan dan membantu perangkat desa dalam mengelola data pengaduan. Sistem ini juga dilengkapi dengan form pengaduan yang memungkinkan pengguna untuk mengunggah foto dan memberikan deskripsi lengkap tentang pengaduan mereka [27].

Penelitian yang dilakukan oleh Mogeab et al. bertujuan untuk meningkatkan kinerja, efisiensi, dan keandalan platform *e-commerce*, serta mengembangkan desain responsif yang kompatibel dengan semua jenis perangkat, ramah pengguna, tampilan menarik, dan mudah dinavigasi untuk memberikan pengalaman pengguna yang terbaik. Untuk mencapai tujuan tersebut, penelitian ini menggunakan perpustakaan modern seperti *React*, *Redux*, dan kerangka kerja *TailwindCSS* untuk membangun antarmuka pengguna, serta kerangka kerja *Laravel* untuk *backend* dan *MySQL* sebagai *Database*. Platform yang dikembangkan terdiri dari tiga bagian: bagian administrasi yang memungkinkan admin mengendalikan platform dan mendapatkan statistik toko, mengontrol produk, pesanan, dan pelanggan, bagian karyawan yang memungkinkan penjual yang berbeda menambahkan produk

mereka ke toko dan menampilkan statistik serta pesanan yang datang kepada mereka dan bagian pelanggan yang menyediakan antarmuka untuk menampilkan berbagai produk serta memberikan informasi yang membantu pembeli memilih produk yang sesuai [28].

Penelitian yang dilakukan oleh Sahid et al. berfokus pada pengembangan sistem informasi manajemen untuk mendukung kegiatan akademik di perpustakaan Universitas Lampung. Masalah yang diidentifikasi dalam penelitian ini mencakup proses manual yang masih berlangsung dalam pengumpulan berkas, pembuatan akun, serta koordinasi dengan mahasiswa yang memerlukan akses ke perpustakaan, yang mengakibatkan ketidakefisienan. Hasil penelitian ini menunjukkan bahwa sistem yang dikembangkan menggunakan kerangka kerja *Laravel* telah berhasil menyediakan layanan unggah berkas untuk persyaratan pendaftaran akun serta pengunggahan tautan repositori yang telah divalidasi oleh admin. Fitur-fitur kunci dari sistem termasuk proses analisis kebutuhan sistem, desain, pengembangan, dan pengujian menggunakan metode Kanban. Selain itu, pengujian sistem dilakukan menggunakan metode blackbox, yang menghasilkan output yang diharapkan, sehingga dapat disimpulkan bahwa aplikasi tersebut lulus dalam pengujian [29].

Penelitian yang dilakukan oleh Nikita Minj membahas secara menyeluruh mengenai perkembangan dan peranan sistem manajemen konten (*CMS*) dalam era digital saat ini. Dalam penelitiannya, Minj menyoroti bahwa *CMS* telah berevolusi dari sistem sederhana berbasis *HTML* menjadi platform kompleks yang mencakup fitur seperti pengelolaan multimedia, manajemen pengguna berbasis peran (*role-based*), alur kerja terstruktur (*workflow management*), serta kemampuan untuk menerbitkan konten ke berbagai saluran digital [30].

Penelitian oleh Rayputri dan Rusli bertujuan untuk mengetahui bagaimana manajemen editorial yang dilakukan oleh Okezone.com dalam menyajikan berita yang layak dan disenangi pembaca. Metode penelitian yang digunakan adalah metode kualitatif dengan pengumpulan data melalui wawancara, dokumentasi, dan observasi. Hasil penelitian menunjukkan bahwa strategi Okezone.com dalam

menyajikan berita yang layak adalah melalui implementasi manajemen *editorial* (*POAC*). *POAC*, yang merupakan singkatan dari *Planning, Organizing, Actuating, dan Controlling*, adalah konsep manajemen editorial yang penting dalam menghasilkan berita yang berkualitas. Sementara itu, strategi Okezone untuk mendapatkan pembaca yang loyal adalah dengan bekerja sama, menempatkan iklan di situs web, menentukan waktu publikasi berita yang tepat, dan menciptakan keterlibatan dengan pembaca melalui media sosial. Penelitian ini menekankan pentingnya manajemen editorial yang baik dalam menghasilkan berita yang berkualitas dan mampu menarik perhatian pembaca [31]

Penelitian oleh Defni dan Tri Lestari bertujuan untuk menerapkan kerangka kerja *Laravel* dalam pengembangan *e-commerce* pada Toko Tani Indonesia Center (TTIC) di Kota Padang. Metode yang digunakan adalah penggabungan model prototipe dengan metode siklus hidup pengembangan sistem (*SDLC*), menggunakan pendekatan *Waterfall*. Hasil penelitian menunjukkan bahwa penggunaan *Laravel framework* efektif dalam mengembangkan *e-commerce* yang dapat memudahkan manajemen barang, transaksi pembelian, dan penjualan di TTIC Padang. Sistem yang dihasilkan memiliki tiga tingkatan akses: admin, karyawan, dan pengguna, dengan fitur-fitur seperti pengelolaan stok, pembelian, penjualan, pembayaran, dan pengiriman barang [32].

Penelitian yang dilakukan oleh Karolina Kowalczyk dan Tomasz bertujuan untuk meningkatkan visibilitas dan pengalaman pengguna pada aplikasi web satu halaman (*Single-Page Application/SPA*) melalui penerapan teknik optimasi mesin pencari (*SEO*). Dalam penelitian ini, dibandingkan tingkat kesulitan dan tantangan optimasi *SEO* antara *SPA* dan aplikasi web -halaman (*Multi-Page Application/MPA*). Metode yang digunakan meliputi implementasi teknik optimasi pada tiga jenis proyek web, yaitu *SPA*, *MPA*, dan pendekatan gabungan menggunakan *JavaScript Isomorfik*. Hasil penelitian menunjukkan bahwa dengan menggabungkan teknik rendering sisi klien pada muatan awal halaman, praktik *SEO* tradisional, serta peningkatan performa sistem, *SPA* dapat dioptimalkan hingga memiliki kemampuan *SEO* yang setara dengan *MPA* [33].

Penelitian yang dilakukan oleh Santoso et al. memfokuskan pada penerapan platform *open source* dalam pengembangan sistem *e-commerce*. Masalah utama yang diangkat adalah bagaimana memanfaatkan teknologi *open source* untuk meningkatkan efisiensi dan efektivitas dalam bisnis fashion online. Hasil penelitian menunjukkan bahwa dengan menggunakan *framework Laravel*, yang mengusung desain arsitektur *Model-View-Controller (MVC)*, pengembangan aplikasi menjadi lebih terstruktur dan efisien. Selain itu, fitur-fitur seperti integrasi dengan metode pembayaran, manajemen produk, serta akurasi data pengiriman yang diperoleh dari RajaOngkir, memperkuat sistem yang dibangun. Peneliti merekomendasikan agar pengembang terus mempertimbangkan penggunaan teknologi *open source* untuk mempercepat pengembangan dan menawarkan solusi yang lebih fleksibel dalam dunia bisnis digital [34].

Penelitian yang dilakukan oleh Kusuma et al. bertujuan untuk mengembangkan sistem informasi berbasis web untuk mengelola administrasi terkait dengan kegiatan magang dan tugas akhir mahasiswa di Program Studi Teknik Elektro. Sistem informasi yang dikembangkan mencakup berbagai fitur yang dapat membantu mahasiswa, dosen, dan staf dalam mengelola kegiatan magang dan tugas akhir, mulai dari panduan pelaksanaan, penulisan proposal, hingga penyusunan laporan akhir. Sistem ini dikembangkan dengan menggunakan metode *Rapid Application Development (RAD)* dan konsep *Model-View-Controller (MVC)* yang memungkinkan pemisahan komponen aplikasi. Hasil pengujian menunjukkan bahwa sistem informasi ini telah berhasil diterapkan di Program Studi Teknik Elektro Universitas Sebelas Maret (UNS) dan direncanakan untuk diimplementasikan di beberapa program studi lainnya [35].

Penelitian yang dilakukan oleh Chakvetadze et al. mengidentifikasi masalah dalam pengelolaan pembaruan komponen pada aplikasi *React*, di mana pembaruan sebuah properti di bagian atas tree menyebabkan pembaruan pada semua child components dan elemen terkait, yang berpotensi mengakibatkan beban berlebih pada sumber daya server. Penelitian ini mencakup tinjauan literatur mengenai pendekatan dalam pengelolaan state untuk aplikasi berbasis *React* dan alat pengembangan yang dapat

digunakan untuk memodernisasi proses pengembangan aplikasi. Penelitian ini menyajikan berbagai metode dan alat yang dirancang untuk mengoptimalkan aplikasi *React*, termasuk *optimisasi event handlers*, penggunaan *React.memo*, pemanfaatan *React.Profiler API* dan *lazy loading* untuk komponen, serta optimasi pembaruan state menggunakan *Immer* dan teknik *throttling* serta *debouncing* untuk input handlers. Kesimpulan dari penelitian ini menyarankan pengembang untuk lebih memperhatikan penerapan metode optimasi yang tepat dalam proses pengembangan untuk meningkatkan efisiensi dan performa aplikasi tanpa menghadapi masalah yang signifikan [36].

Tabel 3. Penelitian Terkait

No	Peneliti (Tahun)	Framework/Tools	Objek Penelitian	Metode	Hasil
1	Anwar dan Sarjono (2022) [18]	<i>UML (Use Case, Class, Activity Diagram), Prototype Model</i>	PT. Jambi Ekspres Televisi (JEKTV)	<i>Prototype</i>	Sistem portal berita dirancang untuk mempercepat penyampaian informasi dan meningkatkan kinerja perusahaan. Sistem mencakup fitur manajemen berita, komentar, dan akun pengguna.
2	Roby et al. (2022) [19]	<i>Laravel (PHP Framework)</i>	Instansi Pemerintahan di Kabupaten Garut	<i>Rational Unified Process (RUP)</i>	Aplikasi CMS berbasis web untuk instansi pemerintah, dengan fitur pengelolaan konten, penggantian identitas dan tema tanpa mengubah kode sumber, serta fitur pesan publik. Sistem diuji menggunakan

					<i>black-box testing.</i>
3	Yuricha & Phan (2023) [20]	Laravel 9, Laravel SPAtie (RBAC Library)	Sistem Supply Chain Management (SCM) berbasis cloud	SDLC (Waterfall)	Penerapan RBAC pada sistem SCM menggunakan Laravel dan library SPAtie berhasil membatasi akses berdasarkan peran. Setiap user hanya bisa mengakses modul sesuai perannya. Sistem diuji dengan skenario dan berhasil sesuai ekspektasi.
4	Santoso (2021) [21]	ReactJS, NodeJS, Bootstrap, Visual Studio Code	Teknik layout website berbasis SPA (Single Page Application)	Studi Literatur + Implementasi Eksperimen	Implementasi teknik SPA menggunakan ReactJS terbukti mempercepat render halaman dan meningkatkan efisiensi navigasi web. Dibanding teknik tradisional, SPA tidak perlu reload ulang sehingga lebih cepat dan user-friendly.
5	Devianto & Dwiasnati (2021) [22]	Web-based E-News Portal, Deployment Diagram, Backend + Frontend CMS	Web portal berita pertanian untuk desa (E-News)	Applied Research, SDLC (modifikasi dengan prototyping & component-based development)	Sistem portal berita E-News dikembangkan untuk menyebarkan informasi hasil pertanian dari desa ke masyarakat luas. Sistem terdiri dari frontend, backend, dan pengaturan user. Fitur mencakup

					berita, kategori, komentar, galeri, dan lain-lain.
6	Ali et al. (2022) [23]	<i>PHP, MySQL, Visual Studio Code, XAMPP</i>	Website riset Fakultas Teknik Universitas Nurtanio	<i>SDLC – Prototype + Black-box testing</i>	Sistem CMS berhasil dibangun untuk mengelola konten riset kendaraan listrik otonom. Sistem mendukung login kontributor, pengelolaan artikel, video, dan profil. Pengujian black-box menunjukkan semua fitur berjalan sesuai.
7	Priskila et al. (2022) [24]	<i>PHP, MySQL,</i>	Aplikasi portal berita Lintasberita1.com (PT. Inti Berita Kalimantan)	<i>Waterfall</i>	Sistem portal berita berbasis web dirancang untuk menggantikan proses pelaporan manual (via <i>WhatsApp</i>) menjadi terintegrasi digital. Aplikasi mendukung peran terpisah (admin, redaktur, wartawan, admin iklan), dengan fitur pengelolaan berita, iklan, dan pengguna.
8	Maulana (2024) [2]	<i>CMS Open source</i> (WordPress, Joomla)	Dampak penggunaan <i>CMS open source</i> pada industri portal berita online di Indonesia	Kualitatif Deskriptif	<i>CMS Open source</i> seperti WordPress banyak digunakan karena murah dan mudah, tetapi memiliki keterbatasan dalam keamanan,

					pengembangan, dan manajemen bisnis. Untuk portal berita profesional, penggunaan CMS berlisensi disarankan.
9	Urmila et al. (2023) [25]	<i>SEO Web Design Methodology, Google Tools (Keyword Planner, Google Trends, Google Analytics)</i>	Website Pikiran-rakyat.com	<i>SEO Web Design Methodology</i>	Penerapan teknik <i>SEO</i> (on-page & off-page) berhasil menempatkan website di halaman pertama Google. Teknik mencakup optimasi konten, judul, meta tag, backlink, dan pemantauan traffic melalui Google Analytics.
10	Thoha et al. (2025) [26]	<i>Laravel, React.js, Mapbox, Progressive Web Apps (PWA), UML, Mockflow</i>	Sistem Informasi Geografis (<i>SIG</i>) pemetaan sekolah di Kecamatan Entikong	Perancangan Sistem + <i>Black-box testing</i>	Sistem <i>SIG</i> berhasil dibangun untuk menampilkan lokasi dan informasi sekolah (SD, SMP, MTS, SMK) di Entikong. Sistem mendukung fitur peta interaktif, open direction, kategori sekolah, login admin, serta <i>dashboard</i> pengelolaan data. Beberapa fitur mengalami bug kecil.
11	Susanto, Widyandayani & Pramono (2024) [27]	<i>Laravel, UML</i>	Sistem Pengaduan Masyarakat di Desa Logung	<i>Rapid Application Development (RAD)</i>	Sistem <i>SIPMAS</i> berbasis web yang mempermudah masyarakat membuat

					pengaduan secara real-time; mempermudah petugas desa menangani pengaduan; hasil diuji dengan metode RAD dan visualisasi UML
12	Shehab, Saeed & Amr (2023) [28]	<i>ReactJS, Redux, Laravel, REST API</i>	Platform E-Commerce berbasis web	Pengembangan SPA dengan arsitektur <i>frontend - backend</i> terpisah	Platform e-commerce berbasis SPA yang responsif, efisien, dan kompatibel dengan semua jenis perangkat. Memiliki performa tinggi, <i>REST API</i> aman, serta fitur admin dan customer lengkap.
13	Sahid, Nama, et al. (2022) [29]	<i>Laravel, MySQL, WhatsApp API, Laravel Excel, SMTP, Kanban</i>	Sistem Manajemen Repositori Perpustakaan Universitas Lampung	<i>Kanban, Blackbox testing</i>	Sistem berhasil menyediakan fitur unggah karya ilmiah secara digital, validasi admin, notifikasi otomatis via <i>WhatsApp</i> & email, dan diuji dengan sukses.
14	Minj (2024) [30]	CMS umum, konsep <i>Headless, AI, Web Apps</i>	Sistem Manajemen Konten (CMS)	Studi literatur dan analisis model	Memberikan tinjauan komprehensif tentang evolusi, fitur, tantangan, serta tren CMS modern seperti <i>AI, headless CMS</i> , dan personalisasi berbasis data.
15	Rayputri & Rusli (2023)	Editorial Management (POAC),	Manajemen Redaksi Okezone.com	Kualitatif (wawancara,	Okezone.com menjadi media online terpopuler

	[31]	CMS, Sosial Media		observasi, dokumentasi)	di Indonesia dengan strategi editorial <i>POAC</i> , penggunaan <i>CMS</i> , penjadwalan berita yang tepat, serta engagement melalui sosial media (<i>Instagram</i> , <i>TikTok</i> , <i>LINE Today</i>).
16	Lestari & Defni (2020) [32]	<i>Laravel</i> , <i>MySQL</i>	Sistem <i>E-Commerce</i> di Toko Tani Indonesia Center (TTIC) Padang	<i>SDLC</i> (<i>Waterfall</i>), <i>UML</i> , <i>Blackbox testing</i>	Sistem e-commerce berbasis <i>Laravel</i> membantu TTIC Padang mengelola transaksi pembelian dan penjualan barang kebutuhan pokok secara efisien dan terstruktur.
17	Kowalczyk & Szandala (2024) [33]	<i>React</i> , <i>Next.js</i> , <i>Flask</i> , <i>SEO tools</i> (<i>SEObility</i> , <i>PageSpeed</i>)	Optimasi <i>SEO</i> pada <i>Single Page Application</i> (<i>SPA</i>) dibanding <i>Multi Page Application</i> (<i>MPA</i>)	Eksperimen implementasi dan komparatif	Penerapan teknik seperti <i>isomorphic JS</i> , <i>SSR</i> , <i>lazy loading</i> , <i>robots.txt</i> , <i>dsb.</i> mampu menyamakan tingkat optimasi <i>SEO SPA</i> dengan <i>MPA</i> berdasarkan hasil pengujian nyata.
18	Santoso et al. (2021) [34]	<i>Laravel</i> , <i>MVC</i> , <i>UML</i> , <i>Midtrans</i>	Aplikasi e-commerce berbasis web	<i>SDLC</i> (<i>Waterfall</i>), <i>OOA</i> dengan <i>UML</i>	<i>Laravel</i> mempermudah pengembangan aplikasi e-commerce yang terstruktur (dengan konsep <i>MVC</i>), fitur lengkap termasuk <i>dashboard</i>

					admin, customer, pembayaran, dan notifikasi status pesanan.
19	Kusuma, Sutrisno & Ibrahim (2020) [35]	<i>Laravel, MVC, RAD, Blackbox testing</i>	Sistem Informasi Manajemen Magang dan Tugas Akhir di Prodi Teknik Elektro UNS	<i>Rapid Application Development (RAD), Blackbox testing</i>	Sistem berhasil mengintegrasikan proses administrasi magang dan tugas akhir dalam satu aplikasi berbasis web, dan telah diterapkan di lingkungan Prodi Teknik Elektro.
20	Ruslanovich (2024) [36]	<i>React, React.memo, React.Profile, React-window, Next.js</i>	Aplikasi berbasis <i>React</i>	Studi literatur dan analisis implementasi metode optimasi <i>React</i>	Penggunaan metode dan tools optimasi tertentu secara tepat dapat secara signifikan meningkatkan performa aplikasi <i>React</i> dan efisiensi pengembangan

Dalam penelitian ini dilakukan pembuatan *Content Management System (CMS)* Portal Berita berbasis web yang ditujukan untuk mendukung proses manajemen redaksi media online, mulai dari penulisan berita, proses editorial, hingga publikasi. Sistem dibangun menggunakan *framework Laravel* sebagai *backend* dan *React JS* dengan pendekatan *Single Page Application (SPA)* sebagai *frontend*, sehingga mampu memberikan pengalaman penggunaan yang cepat, responsif, dan interaktif. Hasil penelitian berupa implementasi CMS Portal Berita yang memiliki fitur utama seperti manajemen artikel (*draft, Pending, terjadwal, publikasi, dan dihapus*), manajemen kategori dan tag, editor penulisan berita dengan *auto-save*, notifikasi status artikel, pengelolaan media, manajemen user dan role (admin, redaktur, reporter), statistik dan analitik portal berita, serta pengelolaan iklan digital.

III. METODOLOGI PENELITIAN

3.1 Waktu dan Tempat

Penelitian ini dilaksanakan di Laboratorium Komputer Universitas Lampung, yang berlokasi di Jl. Prof. Dr. Ir. Sumantri Brojonegoro, Gedong Meneng, Kec. Rajabasa, Kota Bandar Lampung, Lampung. Penelitian ini dijadwalkan berlangsung pada bulan Juni hingga Desember 2025.

Tabel 4. Waktu Pelaksanaan Penelitian

No	Kegiatan	2025																											
		Juni				Juli				Agust				Sept				Okt				Nov				Des			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	<i>Studi Literatur</i>																												
2	<i>Planning (Observasi, Wawancara)</i>																												
3	<i>Perancangan</i>																												
4	<i>Konstruksi</i>																												
5	<i>Implementasi</i>																												

3.2 Alat dan Bahan Penelitian

3.2.1 Alat Penelitian

Penelitian ini menggunakan berbagai alat bantu yang berperan penting dalam mendukung proses pengembangan serta setiap tahapan penelitian. Alat-alat tersebut diklasifikasikan ke dalam tiga kategori utama, yaitu perangkat keras (*hardware*), perangkat lunak (*software*) dan bahan pendukung pengembangan sistem.

Tabel 5. Perangkat Keras

No	Nama Alat	Spesifikasi	Keterangan
1	Laptop Thinkpad T14	Prosesor <i>Amd Ryzen 5 4650u</i>	Digunakan sebagai perangkat utama untuk menjalankan software pengembangan
2	Server Uji Coba	<i>4vCPU/8GB RAM/80GBSSD (Cloud VPS)</i>	Deployment staging sistem
3	<i>Smartphone</i>	<i>Android</i>	Uji responsivitas antarmuka

Tabel 6. Perangkat Lunak

No	Nama Alat	Spesifikasi	Keterangan
1	<i>Visual Studio Code</i>	Versi 1.99	Editor kode yang digunakan untuk menulis dan mengelola seluruh source code.
2	<i>Laravel</i>	Versi 12.4	<i>Framework PHP</i> yang digunakan sebagai kerangka <i>backend</i> dalam pengembangan aplikasi.
3	<i>React Js</i>	Versi 19.0	Pembangunan antarmuka <i>SPA</i> responsif
4	<i>Node.js</i>	Versi 22.3	Runtime environment untuk <i>React</i> build tools

5	<i>Laragon</i>	Versi 3.0	Menyediakan berbagai tools yang dibutuhkan untuk pengembangan web, seperti server web (<i>Apache</i> atau <i>Nginx</i>), Database (<i>MySQL</i>)
6	<i>MySQL</i>	Versi 8.0	Digunakan sebagai sistem manajemen basis data untuk menyimpan data artikel, pengguna, dan metadata lainnya.
7	<i>Git dan GitHub</i>	Versi 0.4	Digunakan untuk manajemen versi kode selama proses pengembangan.
8	<i>draw.io</i>	-	Digunakan untuk merancang antarmuka pengguna (<i>UI Design</i>) dan diagram alur sistem.
9	<i>Microsoft Edge</i>	-	Digunakan sebagai browser utama untuk pengujian antarmuka <i>frontend</i> .

3.2.2 Bahan Penelitian

Dalam pelaksanaan penelitian ini, digunakan beberapa bahan pendukung untuk memperoleh informasi yang relevan dan mendukung proses pengembangan sistem.

Tabel 7. Bahan Penelitian

No	Bahan	Keterangan
1	Data Wawancara	Hasil wawancara dengan redaksi media lokal mengenai kekurangan <i>CMS</i> yang mereka gunakan saat ini.
2	<i>Observasi CMS</i> Eksisting	Observasi pada pengelolaan <i>CMS</i> yang sudah diproduksi dan <i>CMS</i> populer seperti <i>WordPress</i> .
3	Jurnal dan Buku Penelitian Terkait	Referensi dari jurnal dan buku nasional dan internasional pengembangan <i>CMS</i> dan sistem informasi portal berita.
4	Dokumentasi Teknis <i>Laravel</i> dan <i>React</i>	Rujukan teknis dalam proses integrasi <i>frontend - backend</i> dan penggunaan <i>framework</i> secara optimal.

3.2.3 Bahan Pendukung Pengembangan Sistem

Dalam proses pengembangan sistem *CMS*, digunakan sejumlah komponen dan alat bantu yang mendukung tercapainya fungsionalitas yang dibutuhkan. Bahan pendukung ini berfungsi untuk mempercepat pengembangan, memastikan keamanan, serta mempermudah manajemen hak akses dan navigasi sistem. Setiap komponen dipilih berdasarkan kebutuhan teknis yang relevan dengan tujuan sistem

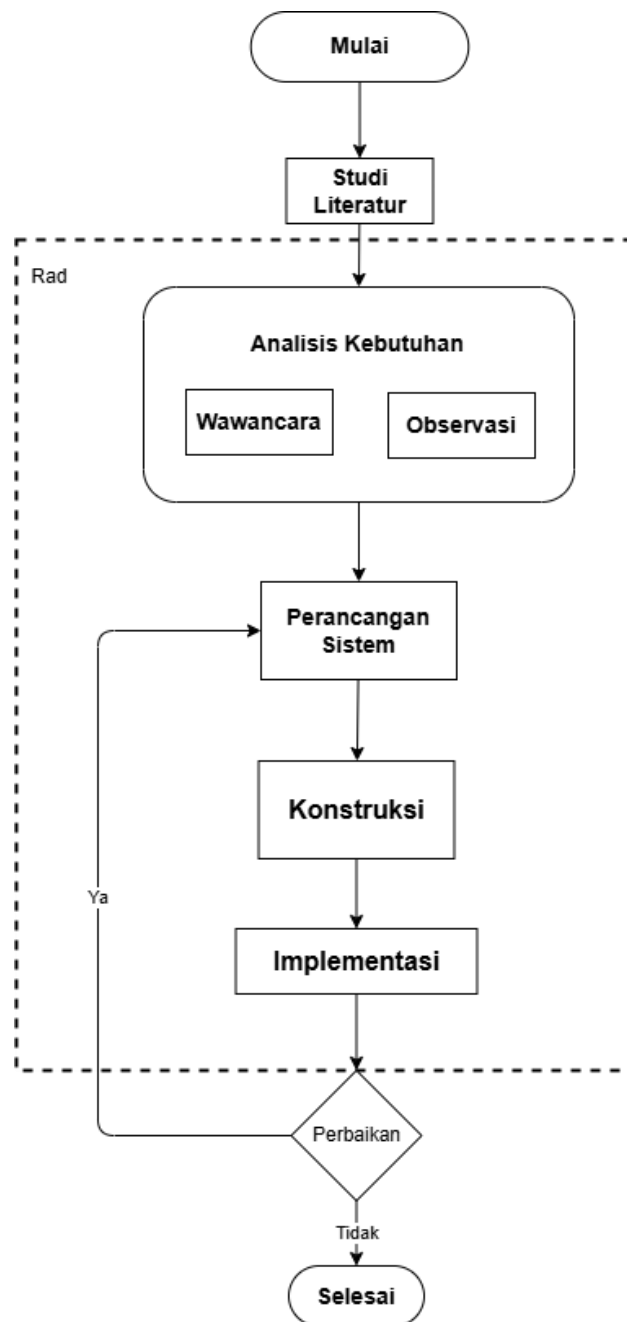
Tabel 8. Bahan Pendukung Pengembangan Sistem

No	Komponen	Keterangan
1	<i>Laravel Breeze</i>	<i>Starter kit autentikasi (login, registrasi, reset password)</i>
2	<i>React Router</i>	Manajemen navigasi <i>SPA</i>

3	<i>SPAtie/Laravel-Permission</i>	Paket <i>Laravel</i> untuk mengelola hak akses dan peran pengguna dalam sistem.
5	<i>Inertia Js</i>	Perantara antara <i>Laravel</i> dan <i>React</i> yang memungkinkan pembuatan <i>SPA</i> tanpa API.
6	<i>Tailwind CSS</i>	<i>Framework</i> CSS utilitas untuk mendesain antarmuka yang responsive.
7	<i>TypeScript</i>	<i>Superset JavaScript</i> yang menambahkan fitur pengetikan statis untuk meningkatkan keandalan kode.
8	<i>Lucide-React</i>	<i>Library</i> ikon berbasis <i>React</i> yang digunakan untuk mempercantik tampilan antarmuka.
9	<i>TypeScript-Eslint</i>	Alat linting untuk menjaga konsistensi dan kualitas kode <i>TypeScript</i> .
10	<i>Shadcn</i>	Komponen UI siap pakai berbasis <i>Tailwind</i> dan <i>Radix</i> UI untuk mempercepat pengembangan <i>frontend</i> .
11	<i>Tiptap Editor</i>	<i>Editor</i> teks berbasis web yang bersifat modular, fleksibel, dan dapat dikustomisasi
12	<i>Fonnte</i>	<i>Fonnte</i> merupakan layanan Unofficial <i>WhatsApp API Gateway</i> yang memungkinkan integrasi otomatis pengiriman pesan <i>WhatsApp</i>

3.3 Tahapan Penelitian

Penelitian ini menggunakan metode *Rapid Application Development (RAD)* yang terdiri dari empat tahapan utama, yaitu: perencanaan kebutuhan (*Requirements Planning*), perancangan sistem secara cepat (*User Design*), pembangunan sistem (*Construction*), dan implementasi (*Cutover*).



Gambar 2. Tahapan Penelitian

Berdasarkan diagram alir penelitian pada Gambar 2, tahapan dalam penelitian ini menggunakan pendekatan *rapid application development (rad)* yang telah dimodifikasi untuk menyesuaikan kebutuhan pengembangan sistem, terdapat beberapa langkah penelitian yaitu:

3.3.1 Requirement Planning

Tahap ini merupakan langkah awal dalam proses pengembangan sistem yang bertujuan untuk mengidentifikasi dan merumuskan kebutuhan serta fitur yang diperlukan dalam membangun *Content Management System (CMS)* portal berita. Kegiatan pada tahap ini dilakukan melalui wawancara, studi literatur, dan observasi terhadap sistem pengelolaan berita yang telah berjalan sebelumnya.

1. Wawancara

Wawancara dilakukan dengan pihak salah satu media lokal yang terdiri atas tiga peran utama dalam alur kerja redaksional, yaitu wartawan, redaktur, dan admin sistem. Dari proses ini diperoleh informasi mengenai pola kerja penulisan artikel, proses penyuntingan, serta hambatan yang dialami dalam penggunaan *CMS* yang ada. Beberapa permasalahan yang teridentifikasi antara lain:

- a) Tidak tersedianya fitur penyimpanan otomatis (*autosave*) sehingga draf berita dapat hilang saat terjadi kesalahan atau gangguan koneksi
- b) Tidak adanya notifikasi otomatis ketika artikel baru diunggah ke *CMS*.
- c) Sulitnya memantau statistik produksi artikel per pengguna.
- d) Gambar iklan terkadang tidak muncul dan kode iklan adsense harusnya bisa dimasukkan dari *CMS*
- e) Kebutuhan akan arsitektur *Single Page Application (SPA)* untuk meningkatkan responsivitas antarmuka.

2. Observasi CMS Eksisting (CMS Internal)

Observasi langsung dilakukan terhadap *CMS* internal yang saat ini digunakan redaksi. Observasi meliputi pemantauan struktur halaman *backend*, alur penggunaan fitur, dan pengujian akses berdasarkan peran. Temuan tambahan dari observasi ini meliputi:

- a) *Role-based access control* hanya diterapkan dibagian *frontend*, sehingga reporter masih dapat menjangkau fungsi khusus redaktur atau admin melalui *URL* langsung.
- b) Tidak tersedianya laporan jumlah produksi artikel per pengguna, mempersulit evaluasi kinerja individu.
- c) Fitur integrasi *google analytics* tidak berfungsi, sehingga data pengunjung dan performa artikel tidak dapat diakses.

3. Observasi CMS Populer (WordPress)

Sebagai bahan pembandingan, dilakukan juga pengamatan terhadap CMS populer seperti *wordpress*. *WordPress* dipilih karena merupakan salah satu CMS paling banyak digunakan, termasuk oleh beberapa portal berita. Beberapa keunggulan *WordPress* yang diamati seperti tersedia banyak plugin tambahan untuk mendukung berbagai kebutuhan redaksional, antar muka cukup ramah bagi pengguna umum dan mendukung sistem peran pengguna secara default (*Author, Editor, Administrator.*). Namun, dari sisi kelemahan seperti keamanan menjadi isu utama, terutama karena banyak *plugin* pihak ketiga yang tidak terjamin keandalannya, pengembangan kustom fitur sering kali membutuhkan *plugin* tambahan yang kompleks dan tidak cocok untuk alur kerja berita yang kompleks atau membutuhkan sistem modular tersendiri.

4. Studi Literatur Proses Penerbitan Berita

Selain wawancara dan observasi langsung, dilakukan juga studi literatur terhadap beberapa jurnal nasional dan internasional yang membahas pengembangan sistem informasi portal berita, sistem manajemen konten, serta proses editorial digital. Tujuan dari studi literatur ini adalah untuk memperoleh pemahaman yang lebih luas mengenai alur kerja redaksional secara umum dan mengidentifikasi praktik terbaik (*best practices*) yang dapat diadopsi ke dalam sistem yang dikembangkan.

Dari beberapa literatur yang ditinjau, umumnya proses penerbitan berita mengikuti pola sebagai berikut:

- a) Wartawan ditugaskan untuk melaksanakan proses peliputan untuk mengumpulkan informasi melalui berbagai metode, seperti reportase langsung di lapangan, serta pengumpulan data dari siaran pers atau rilis resmi.
- b) Wartawan menyusun naskah awal berita berdasarkan hasil liputan kemudian naskah dikirim ke redaksi melalui sistem manajemen konten atau email untuk menjalani tahap penyuntingan (*editing*).
- c) Redaktur sebagai editor memiliki peran penting dalam meninjau, memperbaiki, dan menyempurnakan isi berita. Setelah melalui proses penyuntingan dan dinyatakan layak tayang, redaktur bertanggung jawab dalam menerbitkan (*publish*) berita ke platform media daring agar dapat diakses oleh publik.

6. Kebutuhan Fungsional dan *Non Fungsional*

Kebutuhan sistem merupakan hasil dari proses perencanaan kebutuhan yang dilakukan melalui wawancara, observasi, dan studi terhadap sistem *CMS* yang sudah ada. Kebutuhan ini dibagi menjadi dua kategori, yaitu kebutuhan fungsional yang menggambarkan layanan yang harus disediakan sistem, dan kebutuhan non-fungsional yang menggambarkan karakteristik atau kualitas sistem yang dibangun.

7. Target Pengguna

Target pengguna dari *CMS* Portal Berita ini adalah individu yang terlibat dalam proses editorial media daring, yaitu reporter (penulis artikel), redaktur (penyunting dan pengawas konten), serta administrator (pengelola sistem dan manajemen pengguna). Setiap jenis pengguna memiliki peran dan hak akses yang berbeda, yang menjadi dasar dalam perancangan kebutuhan fungsional dan struktur hak akses pada sistem.

3.3.2 Design

Tahap ini mencakup proses perancangan sistem berdasarkan kebutuhan yang telah diidentifikasi pada tahap analisis kebutuhan. Tujuan utama dari tahap ini adalah menyusun rancangan struktur sistem, alur proses, dan antarmuka pengguna agar sistem mudah dipahami dan dikembangkan. Perancangan dilakukan melalui penyusunan *Use Case Diagram*, *Activity Diagram* serta *Entity Relationship Diagram (ERD)*

1. Perancangan Use Case Diagram

Perancangan *Use Case Diagram* dilakukan untuk menggambarkan relasi antara aktor dan fungsi-fungsi sistem yang dapat diakses berdasarkan peran masing-masing pengguna. Diagram ini berperan penting sebagai dasar dalam perancangan interaksi, hak akses, serta pemahaman proses bisnis yang berjalan di dalam sistem. Tahapan perancangan dimulai dengan mengidentifikasi aktor, yaitu entitas eksternal yang berinteraksi langsung dengan sistem. Selanjutnya dilakukan identifikasi *use case* yang merepresentasikan fungsi utama yang dibutuhkan oleh masing-masing aktor.

2. Perancangan Activity diagram

Activity Diagram menggambarkan alur proses dari setiap *use case* yang telah dirancang sebelumnya. Diagram ini berfungsi untuk memvisualisasikan urutan aktivitas yang terjadi dalam sistem, sesuai dengan peran masing-masing aktor. *Activity Diagram* membantu memodelkan logika proses dalam sistem detail, sehingga mendukung pemahaman yang lebih jelas terhadap alur kerja serta pengambilan keputusan dalam proses pengembangan sistem.

3. Perancangan Entity relationship diagram (ERD)

Entity relationship diagram (ERD) digunakan untuk memetakan struktur data yang akan disimpan dalam sistem, beserta hubungan antar entitas yang saling terhubung. Perancangan ERD dilakukan dengan tujuan agar sistem mampu mengelola data secara efisien dan terstruktur.

3.3.3 Construction

Tahap *construction* atau pembuatan sistem merupakan proses penerapan desain sistem ke dalam bentuk program yang dapat dijalankan. Seluruh kebutuhan pengguna yang telah dianalisis dan dirancang sebelumnya diterjemahkan ke dalam bahasa pemrograman.

1. Pengembangan *backend* dibangun menggunakan *framework Laravel*. Fungsionalitas utama yang dibangun mencakup proses autentikasi, manajemen data pengguna, pengelolaan artikel, pengelolaan kategori, pengelolaan iklan serta pengaturan hak akses berdasarkan peran pengguna. Seluruh data disimpan dan dikelola melalui basis data yang telah dirancang pada tahap sebelumnya.
2. Pengembangan *frontend* dibuat menggunakan *framework React* dengan pendekatan *Single Page Application (SPA)*. Pendekatan ini memungkinkan interaksi antarpengguna dan sistem berjalan secara cepat dan responsif tanpa perlu memuat ulang halaman secara keseluruhan. Komponen-komponen antarmuka disusun modular untuk mempermudah pengelolaan, pengujian, dan pengembangan lebih lanjut.
3. Pengujian awal dilakukan terhadap fitur-fitur utama sistem untuk memastikan bahwa seluruh fungsi berjalan sesuai dengan rancangan dan tidak terjadi kesalahan fungsional yang signifikan.

3.3.4 Cutover

Tahap *Cutover* merupakan proses implementasi sistem ke dalam lingkungan produksi setelah tahap pengembangan selesai. Tujuan dari tahap ini adalah untuk memastikan sistem siap digunakan oleh pengguna akhir serta berfungsi sesuai dengan kebutuhan. Adapun aktivitas utama dalam tahap ini meliputi.

1. Instalasi Sistem

Instalasi *Content Management System (CMS)* yang telah dikembangkan diinstal pada server *Virtual Private Server (VPS)*, baik untuk keperluan pengujian maupun produksi. Proses ini mencakup konfigurasi domain, basis data, serta pemasangan dependensi sistem *frontend* dan *backend*.

2. Pengujian Fungsional (*Black box Testing*)

Pengujian fungsional dilakukan menggunakan metode *Black box Testing*, yaitu dengan menguji setiap fitur berdasarkan kombinasi input dan output tanpa memperhatikan struktur kode internal. Pengujian ini dilakukan oleh pengembang untuk memverifikasi bahwa seluruh fungsi utama berjalan sesuai dengan skenario yang dirancang.

3. Uji Penerimaan Pengguna (*User Acceptance Test – UAT*)

Setelah pengujian fungsional selesai, sistem diuji langsung oleh pengguna akhir yang mewakili peran utama dalam proses redaksional, seperti admin, redaktur, dan reporter. Pengujian dilakukan melalui metode *User Acceptance Test (UAT)* dengan tujuan untuk mengevaluasi apakah sistem telah memenuhi kebutuhan pengguna secara nyata dan mudah digunakan sesuai dengan alur kerja redaksi. Dalam penelitian ini, UAT dilaksanakan pada dua redaksi media yang berbeda untuk memastikan bahwa sistem yang dikembangkan dapat beradaptasi dengan variasi kebutuhan dan struktur organisasi redaksional. Hasil umpan balik dari kedua pengguna tersebut dijadikan acuan untuk melakukan penyempurnaan sistem sebelum diterapkan secara penuh di lingkungan operasional.

Skenario Pengujian

Tabel 9. Skenario pengujian reporter

No	Fitur/Skenario Uji	Langkah Pengguna	Hasil yang Diharapkan
1	Login	Masukkan <i>Email</i> dan <i>password</i>	Berhasil masuk ke <i>dashboard</i>
2	Buat berita baru	Klik menu tulis artikel	Tampil <i>text editor</i> berisi data berita baru
3	Upload media	Klik tambah media dan <i>Upload</i> gambar <i>via drag & drop/URL</i>	Gambar tersimpan & bisa ditampilkan
4	Sisipkan media	Pilih media yang diinginkan dan klik sisipkan ke postingan	Media berhasil disisipkan ke postingan
5	<i>Auto save</i>	Melakukan perubahan data berita	Berita tersimpan otomatis
6	Simpan Berita	Klik tombol simpan berita	Berita berhasil disimpan dengan status yang sama
7	Simpan <i>Pending</i>	Klik tombol save <i>Pending</i>	Berita masuk status <i>Pending</i>
8	Pratinjau berita	Klik tombol pratinjau	Berita tampil sesuai format portal
9	Artikel saya	Klik menu artikel saya	Tampil semua list berita dan statistik produksi berita berdasarkan Pengguna
10	Edit berita	Klik menu artikel saya	Berita bisa diedit jika statusnya masih draft
11	Pulihkan berita yang dihapus	Klik menu artikel dihapus pada halaman artikel saya	Tampi list berita yang dihapus dan bisa di pulihkan
12	Ubah nama, email dan <i>password</i>	Klik menu profile	Nama, email dan <i>password</i> bisa diubah

Tabel 10. Skenario pengujian editor

No	Fitur/Skenario Uji	Langkah Pengguna	Hasil yang Diharapkan
1	Login	Masukkan Email dan <i>password</i>	Berhasil masuk ke <i>dashboard</i>
2	Notifikasi berita <i>Pending</i> via <i>WhatsApp</i>	1. Reporter menyimpan berita dengan status " <i>Pending</i> ". 2. Admin/editor membuka grup <i>WhatsApp</i> redaksi.	Pesan notifikasi otomatis dikirim ke grup <i>WhatsApp</i> berisi informasi berita <i>Pending</i> (judul & penulis).
3	Edit berita <i>Pending</i>	Masuk ke halaman artikel <i>Pending</i> dengan klik menu artikel <i>Pending</i> , klik tombol edit pada salah satu berita	Menampilkan halaman edit pada berita yang dipilih
4	Upload media	Klik tambah media dan Upload gambar via drag & drop/URL	Gambar tersimpan & bisa ditampilkan
5	Sisipkan media	Pilih media yang diinginkan dan klik sisipkan ke postingan	Media berhasil disisipkan ke postingan
6	<i>Auto save</i>	Melakukan perubahan data berita	Berita tersimpan otomatis
7	Simpan Berita	Klik tombol simpan berita	Berita berhasil disimpan dengan status yang sama
8	Pratinjau berita	Klik tombol pratinjau	Berita tampil sesuai format berita
9	Terbitkan berita	Masuk ke halaman <i>Pending</i> , klik tombol publikasi dan klik publikasi sekarang	Status artikel (Berubah dari " <i>Pending</i> " menjadi " <i>Publikasi</i> ")
10	Jadwalkan terbit	Masuk ke halaman <i>Pending</i> , klik tombol publikasi dan atur jadwal publikasi	Status artikel (Berubah dari " <i>Pending</i> " menjadi " <i>Scheduled</i> ")
11	Artikel saya	Klik menu artikel saya	Tampil semua list berita dan statistik produksi berita berdasarkan Pengguna
12	Kelola kategori dan subkategori	Masuk ke halaman list Kategori / Subkategori dan tambahkan 1 data	list kategori dan sub kategori berhasil ditambahkan

Tabel 11. Skenario pengujian admin

No	Fitur/Skenario Uji	Langkah Pengguna	Hasil yang Diharapkan
1	Login	Masukkan <i>Email</i> dan <i>password</i>	Berhasil masuk ke <i>dashboard</i>
2	Kelola <i>user & role</i>	Menambahkan pengguna baru & ubah <i>role</i>	pengguna terdaftar & akses sesuai <i>role</i>
3	Statistik pengguna	Klik menu manajemen pengguna	Tampil Jumlah user aktif/inaktif tampil, role dan status <i>offline/online</i>
4	Edit pengguna	Klik salah satu pengguna	Bisa mengubah nama, kata sandi, role dan status pengguna
5	Kelola iklan	Klik amenu daftar iklan	Tampil daftar iklan
6	Tambah iklan	Klik tambah iklan pada menu daftar iklan dan tambahkan satu data iklan	Iklan berhasil ditambahkan dengan mengisi data iklan sesuai format (bisa <i>banner/code</i>)
7	Edit slot iklan	Klik menu pengaturan iklan dan tambahkan iklan pada salah satu slot iklan	Iklan berhasil ditambahkan berdasarkan slot yang dipilih
8	Statistik produksi berita berdasarkan pengguna	Klik menu artikel per user	Tampil grafik dan list pengguna berdasarkan jumlah berita yang diterbitkan/diedit
9	Statistik produksi berita	Klik menu statistik berita	Tampil total berita dan grafik produksi berita
10	Pengaturan website	Klik menu web setting	Bisa mengubah judul, icon dan logo website

V. KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil perancangan, implementasi, dan pengujian sistem yang telah dilakukan, maka dapat diambil kesimpulan sebagai berikut:

1. Penelitian ini berhasil mengidentifikasi kebutuhan fitur utama yang diperlukan dalam mendukung alur kerja redaksional portal berita, meliputi penulisan dan pengelolaan artikel, pengaturan kategori dan tag, manajemen pengguna dan peran, pengelolaan media dan iklan, serta fitur pendukung seperti *auto-save* dan notifikasi *WhatsApp*.
2. *Content Management System (CMS)* portal berita berhasil dirancang dan dibangun menggunakan *framework Laravel* pada sisi *backend* dan *React.js* pada sisi *frontend* dengan pendekatan *Single Page Application (SPA)*.
3. Pengujian fungsional menggunakan metode *Black Box Testing* menunjukkan bahwa seluruh fitur utama berjalan *valid* sesuai skenario pengujian, sementara hasil *User Acceptance Test (UAT)* terhadap tujuh responden dari lima redaksi menghasilkan rata-rata tingkat penerimaan sebesar 92,7%, yang berada pada kategori sangat baik berdasarkan persentase nilai interval. Selanjutnya, berdasarkan pengujian kinerja sistem, sistem portal berita mampu beroperasi secara stabil hingga 275 *Virtual Users (VU)* yang mengakses sistem secara bersamaan. Pada beban di atas nilai tersebut, khususnya mulai 300 *VU*, sistem mulai mengalami penurunan performa dan memasuki kondisi *overload*, sehingga kapasitas layanan yang direkomendasikan adalah hingga sekitar 275 pengguna bersamaan.
4. Analisis performa membuktikan bahwa pendekatan *Single Page Application (SPA)* memiliki kinerja perpindahan halaman yang jauh lebih cepat dibandingkan *Multi Page Application (MPA)*. Pada pengujian *initial*

load, SPA memperoleh rata-rata waktu 0,60 detik, sedangkan MPA 0,71 detik. Pada pengujian navigasi, SPA menghasilkan waktu rata-rata 69,6 ms, sedangkan MPA 868,4 ms, atau sekitar 12,48 kali lebih cepat.

5.2 Saran

Berdasarkan hasil penelitian dan pengujian sistem yang telah dilakukan, terdapat beberapa saran yang dapat dipertimbangkan untuk pengembangan sistem di masa mendatang, yaitu sebagai berikut:

1. Pengembangan *frontend* publik menggunakan *framework SSR* (misalnya *Next.js*), Untuk meningkatkan performa *SEO* dan kecepatan *indexing* mesin pencari, halaman publik disarankan dibangun menggunakan *framework* yang mendukung *server-side rendering (SSR)*, sehingga konten lebih mudah diindeks oleh mesin pencari.
2. Sistem dapat diperluas menjadi arsitektur multi-portal sehingga satu *backend* mampu mengelola banyak situs berita sekaligus. Hal ini memungkinkan efisiensi pengelolaan konten bagi perusahaan media dengan beberapa kanal berita.

DAFTAR PUSTAKA

- [1] H. R. P. Sianturi, “Proses Gatekeeping dalam Produksi Berita di Media Daring,” *JPI*, vol. 8, no. 1, pp. 24–45, Jun. 2023, doi: 10.35706/jpi.v8i1.9407.
- [2] Y. Maulana, “Analisis Dampak Penggunaan Content Management System (CMS) Open Source Terhadap Industri Portal Berita Media Online,” *Jurnal Algoritma*, vol. 21, no. 1, pp. 1–8, May 2024, doi: 10.33364/algoritma/v.21-1.1141.
- [3] N. George, *Beginning django CMS*. Berkeley, CA: Apress, 2015. doi: 10.1007/978-1-4842-1669-9.
- [4] E. Maryani, P. Yudhaprarnesti, L. Nurhajati, and X. A. Wijayanto, Eds., *Idealisme Jurnalis & Inovasi Model Bisnis Industri Media*. Lembaga Penelitian, Publikasidan Pengabdian Masyarakat (LP3M) LSPR, 2022. doi: 10.37535/20320220100.
- [5] G. Mohamad, *Seandainya saya wartawan Tempo*, Revisi. Jakarta: Tempo Publishing, 2014.
- [6] D. Jain, *Ultimate Laravel for Modern Web Development: Build Robust and Interactive Enterprise-Grade Web Apps Using Laravel’s MVC, Authentication, APIs, and Cloud Deployment (English Edition)*, 1st ed. Delhi: Orange Education PVT Ltd, 2024.
- [7] S. Smirnova and A. Tezuysal, *MySQL Cookbook*, 4th Edition. Sebastopol, CA: O’Reilly, 2022.
- [8] F. Zammetti, *Modern Full-Stack Development: Using TypeScript, React, Node.js, Webpack, Python, Django, and Docker*. Berkeley, CA: Apress, 2022. doi: 10.1007/978-1-4842-8811-5.

- [9] M. S. Mikowski and G. D. Benson, *Single page web applications: Javascript end-to-end*. Shelter Island, NY: Manning, 2014.
- [10] Fonnte, "API WhatsApp Gateway," Fonnte. Accessed: May 11, 2025. [Online]. Available: <https://fonnte.com>
- [11] J. Westfall, *Set Up and Manage Your Virtual Private Server: Making System Administration Accessible to Professionals*. Berkeley, CA: Apress, 2021. doi: 10.1007/978-1-4842-6966-4.
- [12] A. Verma, "A Comparative Study of Black Box Testing and White Box Testing," *ijcse*, vol. 5, no. 12, pp. 301–304, Dec. 2017, doi: 10.26438/ijcse/v5i12.301304.
- [13] Zulkarnaini, A. Firdhayanti, T. Taufik, and B. Bachry, "User Acceptance Testing through Blackbox Evaluation for Corn Distribution Information System," *bit-Tech*, vol. 6, no. 2, pp. 208–215, Dec. 2023, doi: 10.32877/bt.v6i2.1065.
- [14] D. Wijaya, B. Daniawan, and Y. Gunawan, "Search Engine Optimization (SEO) As A Promotional Media On Google Search," *bit-Tech*, vol. 4, no. 1, pp. 31–39, Aug. 2021, doi: 10.32877/bt.v4i1.237.
- [15] G. Booch, J. Rumbaugh, and I. Jacobson, *The unified modeling language user guide*. in The Addison-Wesley object technology series. Reading Mass: Addison-Wesley, 1999.
- [16] J. Martin, *Rapid Application Development*. New York: Macmillan Publishing, 1991. [Online]. Available: <https://archive.org/details/rapidapplication00mart/page/81>
- [17] M. Amin, "Perbandingan Kinerja Nginx dan Caddy sebagai Web Server untuk Aplikasi PHP," *INSECT*, vol. 11, no. 1, pp. 88–96, Mar. 2025, doi: <https://doi.org/10.33506/insect.v11i1.4223>.

- [18] S. Anwar and S. Sarjono, "Portal Berita Berbasis Web Pada PT. Jambi Ekspres Televisi (JEKTV)," *Jurnal MSI UNAMA*, vol. 7, no. 3, pp. 407–420, Sep. 2022, doi: 10.33998/jurnalmsi.2022.7.3.178.
- [19] A. I. Roby, R. Cahyana, R. Kurniawati, and L. Fitriani, "Rancang Bangun Content Management System Untuk Situs Web Instansi Pemerintahan," *Jurnal Algoritma*, vol. 18, no. 2, pp. 367–375, Jan. 2022, doi: 10.33364/algoritma/v.18-2.959.
- [20] Y. Yuricha and I. K. Phan, "Penerapan Role Based Access Control dalam Sistem Supply Chain Management Berbasis Cloud: The Implementation of Role Based Access Control in a Cloud-Based Supply Chain Management System," *MALCOM*, vol. 3, no. 2, pp. 339–348, Nov. 2023, doi: 10.57152/malcom.v3i2.1259.
- [21] M. F. Santoso, "Teknik Single Page Application (Spa) Layout Web Dengan Menggunakan React Js Dan Bootstrap," *JKI*, vol. 9, no. 2, Dec. 2021, doi: 10.31294/jki.v9i2.11357.
- [22] Y. Devianto and S. Dwiasnati, "Rancang Bangun Web Portal Berita Sebagai Sumber Informasi Berita Tentang Pertanian," *JATISI*, vol. 8, no. 2, pp. 534–546, Jun. 2021, doi: 10.35957/jatisi.v8i2.792.
- [23] A. L. Ali, A. S. Satyawan, I. Y. Wulandari, and H. Puspita, "Rancang Bangun Content Management System Pada Website Riset Fakultas Teknik Universitas Nurtanio Menggunakan Bahasa Pemrograman PHP Dan MySQL," *senastindo*, vol. 4, pp. 42–51, Oct. 2022, doi: 10.54706/senastindo.v4.2022.168.
- [24] R. Priskila, N. N. K. Sari, and M. R. Arief, "Aplikasi Portal Berita Berbasis Website (Studi Kasus: Lintasberita1.Com)," *JCOMS*, vol. 2, no. 3, pp. 199–208, Jun. 2022, doi: 10.47111/jointecom.v2i3.8864.
- [25] Urmila Urmila, Erwan Sastrawan, and Harnida Wahyuni Adda, "Penerapan Teknik Seo (Search Engine Optimization) pada Website Pikiran-Rakyat.Com

dalam Strategi Pemasaran,” *JUMEK*, vol. 1, no. 3, pp. 50–56, Jun. 2023, doi: 10.59024/jumek.v1i3.111.

- [26] A. M. Thoha, A. Pratama, and A. Hidayati, “Rancang Bangun Sistem Informasi Geografis Pemetaan Sekolah di Kecamatan Entikong Berbasis Website Menggunakan Laravel, React.Js, Mapbox dan Progressive Web Apps,” *Jurnal Riset Sistem Informasi dan Teknik Informatika*, vol. 3, no. 2, pp. 173–194, Mar. 2025.
- [27] S. Susanto, B. A. Pramono, A. G. Widyandayani, and P. Patmawati, “Implementasi Metode RAD pada Sistem Pengaduan Masyarakat (SIPMAS) di Desa Logung Menggunakan Framework Laravel,” *JTEKSIS*, vol. 6, no. 3, pp. 494–506, Jul. 2024, doi: 10.47233/jteksis.v6i3.1376.
- [28] M. Shehab, “Electronic Commerce Platform As Single Page Application Using React Js And Laravel Framework,” *Taiz University Research Journal (TURJ)*, vol. 38, no. 38, pp. 116–135, 2023.
- [29] A. Sahid and G. F. Nama, “Design and Development of Management Information Systems at the University of Lampung Library Repository Using the Laravel Framework,” *JESR*, vol. 4, no. 2, Dec. 2022, doi: 10.23960/jesr.v4i2.110.
- [30] N. Minj, “Content Management Systems: A Comprehensive Review and Analysis,” *GIMRJ*, pp. 137–141, Jun. 2024, doi: 10.69758/GIMRJ2406I8V12P017.
- [31] V. A. Rayputri and F. Rusli, “The Editorial Management Of Okezone.Com In Presenting Worthy And Likable News,” *IJASSH*, vol. 1, no. 1, pp. 824–833, Feb. 2023, doi: 10.24912/ijassh.v1i1.25949.
- [32] T. Lestari, “Implementing Laravel Framework for E-Commerce: Case Study at Indonesian Farmer Shop Center,” *International Journal of Advanced Computing Science and Engineering*, vol. 2, no. 1, p. 20, 2020.

- [33] K. Kowalczyk and T. Szandala, "Enhancing SEO in Single-Page Web Applications in Contrast With Multi-Page Applications," *IEEE Access*, vol. 12, pp. 11597–11614, 2024, doi: 10.1109/ACCESS.2024.3355740.
- [34] G. B. Santoso, T. M. Sinaga, and A. Zuhdi, "MVC Implementation In Laravel Framework For Development Web-Based E-Commerce Applications," *Intelect. matics.*, vol. 1, no. 1, Jan. 2021, doi: 10.25105/itm.v1i1.7867.
- [35] Y. P. Kusuma, S. Sutrisno, and M. H. Ibrahim, "Information System for Internship and Final Project Management Based on Laravel Framework," *JEEICT*, vol. 2, no. 2, p. 42, Oct. 2020, doi: 10.20961/jeeict.2.2.43972.
- [36] C. V. Ruslanovich, "Performance Analysis of React Applications: Optimization Methods and Tools," *IJLEMR*, vol. 9, no. 5, pp. 51–57, May 2024, doi: 10.56581/IJLEMR.9.05.51-57.