

**IMPLEMENTASI WEB APPLICATION FIREWALL (WAF) DAN
CONTENT DELIVERY NETWORK (CDN) UNTUK MENINGKATKAN
KEAMANAN DAN KECEPATAN PADA WEBSITE FMIPA
UNIVERSITAS LAMPUNG**

(Skripsi)

Oleh

**IKHSAN SAPUTRA
NPM 2117051037**



**JURUSAN ILMU KOMPUTER
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2025**

**IMPLEMENTASI WEB APPLICATION FIREWALL (WAF) DAN
CONTENT DELIVERY NETWORK (CDN) UNTUK MENINGKATKAN
KEAMANAN DAN KECEPATAN PADA WEBSITE FMIPA
UNIVERSITAS LAMPUNG**

Oleh

**IKHSAN SAPUTRA
NPM 2117051037**

(Skripsi)

**Sebagai Salah Satu Syarat Untuk Mencapai
Gelar SARJANA ILMU KOMPUTER**

Pada

**Jurusan Ilmu Komputer
Fakultas MIPA Universitas Lampung**



**JURUSAN ILMU KOMPUTER
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2025**

ABSTRAK

IMPLEMENTASI WEB APPLICATION FIREWALL (WAF) DAN CONTENT DELIVERY NETWORK (CDN) UNTUK MENINGKATKAN KEAMANAN DAN KECEPATAN PADA WEBSITE FMIPA UNIVERSITAS LAMPUNG

Oleh

IKHSAN SAPUTRA

Penelitian ini bertujuan untuk mempelajari pengaruh implementasi *Web Application Firewall* (WAF) dan *Content Delivery Network* (CDN) dalam meningkatkan keamanan dan kecepatan akses situs web FMIPA Universitas Lampung. Dengan meningkatnya ancaman serangan siber seperti *Distributed Denial of Service* (DDoS), *SQL Injection*, dan *Cross-Site Scripting* (XSS), penerapan solusi efektif menjadi sangat penting. Penelitian ini berfokus pada evaluasi teknologi WAF dan CDN sebagai langkah strategis untuk meminimalisasi risiko serangan siber dan meningkatkan kinerja situs web. Metode yang digunakan dalam penelitian ini meliputi pengaktifan WAF dan integrasi CDN, yang diharapkan dapat memberikan perlindungan yang lebih baik serta mempercepat waktu akses situs web. Hasil penelitian menunjukkan bahwa penerapan WAF dan CDN secara signifikan dapat meningkatkan keamanan dan kecepatan akses situs web, sehingga memberikan pengalaman pengguna yang lebih baik. Penelitian ini diharapkan dapat menjadi referensi bagi pengembangan keamanan situs web di lembaga pendidikan dan organisasi lainnya.

Kata kunci : Keamanan Situs Web, Kecepatan Situs Web, WAF (*Web Application Firewall*), CDN (Jaringan Pengiriman Konten)

ABSTRACT

IMPLEMENTATION OF WEB APPLICATION FIREWALL (WAF) AND CONTENT DELIVERY NETWORK (CDN) TO IMPROVE SECURITY AND SPEED ON THE FMIPA WEBSITE OF LAMPUNG UNIVERSITY

By

IKHSAN SAPUTRA

This study aims to explore the influence of implementing a Web Application Firewall (WAF) and a Content Delivery Network (CDN) on enhancing the security and access speed of the FMIPA University Lampung website. With the rise in cyber threats such as Distributed Denial of Service (DDoS) attacks, SQL Injection, and Cross-Site Scripting (XSS), the need for effective security solutions has become increasingly critical. The research focuses on evaluating WAF and CDN technologies as strategic measures to minimize the risks associated with cyberattacks while improving website performance. The methods employed in this study include the activation of WAF and the integration of CDN, which are anticipated to provide better protection and accelerate website access times. The results indicate that the implementation of WAF and CDN can significantly enhance both the security and speed of website access, thereby improving the overall user experience. This study is expected to serve as a reference for the development of website security in educational institutions and other organizations.

Keywords: Website Security, Website Speed, WAF (Web Application Firewall), CDN (Content Delivery Network)

Judul Skripsi : IMPLEMENTASI WEB APPLICATION
FIREWALL (WAF) DAN CONTENT DELIVERY
NETWORK (CDN) UNTUK MENINGKATKAN
KEAMANAN DAN KECEPATAN PADA WEBSITE
FMIPA UNIVERSITAS LAMPUNG

Nama Mahasiswa : **Ikhsan Saputra**

Nomor Pokok Mahasiswa : 2117051037

Program Studi : S1 Ilmu Komputer

Jurusan : Ilmu Komputer

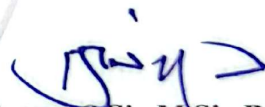
Fakultas : Matematika dan Ilmu Pengetahuan Alam

MENYETUJUI

1. Komisi Pembimbing



Muhaqiqin, S.Kom., M.T.I.
NIP. 199305252022031009



Mulyono, S.Si., M.Si., P.hd.
NIP. 197406112000031002

2. Ketua Jurusan Ilmu Komputer



Dwi Sakethi, S.Si., M.Kom.
NIP. 196806111998021001

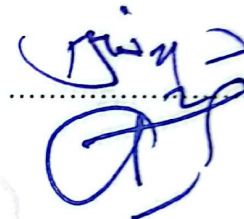
MENGESAHKAN

1. Tim Penguji

Ketua : **Muhaqiqin, S.Kom., M.T.I.**
Penguji



Sekretaris : **Mulyono, S.Si., M.Si., Ph.D.**
Penguji



Penguji : **Dr. Aristoteles, S.Si., M.Si.**
Pembahas



2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam



Dr. Eng. Heri Satria, S.Si., M.Si.
NIP. 197110012005011002

Tanggal Lulus Ujian Skripsi : **17 Juni 2025**

PERNYATAAN

Saya yang bertanda tangan di bawah ini :

Nama : Ikhsan Saputra

NPM : 2117051037

Dengan ini menyatakan bahwa skripsi saya dengan judul **“Implementasi Web Application Firewall (WAF) Dan Content Delivery Network (CDN) Untuk Meningkatkan Keamanan Dan Kecepatan Pada Website Fmipa Universitas Lampung”** merupakan karya saya sendiri dan bukan karya milik orang lain. Semua tulisan yang ada dalam skripsi ini telah mengikuti kaidah penulisan karya ilmiah Universitas Lampung. Apabila terdapat bukti di kemudian hari bahwa skripsi saya merupakan hasil plagiat atau dibuat oleh orang lain, maka saya bersedia menerima sanksi yang diberikan.

Bandar Lampung, 26 Juni 2025


Ikhsan Saputra
NPM. 2117051037

RIWAYAT HIDUP



Dilahirkan pada hari Senin, 25 Agustus 2003 di Seputih Jaya Lampung Tengah. Anak kedua dari dua bersaudara dari Bapak Madison dan Ibu Sri Astuti. Telah menyelesaikan pendidikan sekolah dasar pada tahun 2015 di SDN 1 Bandar Jaya, kemudian pendidikan menengah pertama di SMPN 3 Terbanggi Besar pada tahun 2018 dan lulus dari pendidikan menengah atas di SMAN 1 Terbanggi Besar pada tahun 2021. Kemudian pada tahun 2021 terdaftar sebagai mahasiswa Jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung melalui jalur Seleksi Nasional Masuk Perguruan Tinggi Negeri (SNMPTN) . Selama menjadi mahasiswa , kegiatan yang pernah diikuti adalah sebagai berikut.

1. Menjadi anggota bidang kaderisasi Himpunan Mahasiswa Jurusan Ilmu Komputer periode 2022/2023.
2. Menjadi Ketua Bidang kaderisasi Himpunan Mahasiswa Jurusan Ilmu Komputer periode 2023/2024.
3. Menjadi Asisten Dosen pada beberapa mata kuliah di Jurusan Ilmu Komputer dari tahun 2022-2024.
4. Melaksanakan kegiatan Kerja Praktik pada bulan Desember 2023 sampai Februari 2024 di Komisi Pemilihan Umum (KPU) Kota Metro.
5. Mengikuti program Magang Kampus Merdeka pada tahun 2024.
6. Melaksanakan Kuliah Kerja Nyata di Desa Donomulyo, Kecamatan Bumi Agung, Kabupaten Lampung Timur, Provinsi Lampung pada tahun 2024.

MOTTO

“keberhasilan bukanlah milik orang yang pintar, keberhasilan adalah kepunyaan mereka yang senantiasa selalu berusaha.”

(B.J Habibie)

“Lebih baik gagal tapi sudah mencoba, daripada tidak mencoba sama sekali.”

(Andrew Susanto)

“Tetap selalu berbuat baik, karena tidak ada salahnya yang salah adalah tidak mencoba untuk berbuat baik.”

(Ikhsan Saputra)

“Alasan bertahan dan berjuang adalah untuk melihat senyum kedua orang tua.”

(Ikhsan Saputra)

PERSEMBAHAN

Alhamdulillahirobbilalamin

Puji dan syukur kehadiran Allah Subhannahu Wa Ta'Alaa atas segala Rahmat dan Karunia-Nya sehingga saya dapat sampai di tahap untuk menyelesaikan skripsi ini. Shalawat serta salam selalu tercurahkan kepada Nabi Muhammad Shalallahu Alaihi Wasallam.

Kupersembahkan karya ini kepada :

Kedua Orang Tuaku, Kakak, Mbak Dan Ponakanku

Yang dengan kasih sayang, doa yang tak pernah putus, serta pengorbanan tanpa pamrih, telah menjadi sumber kekuatan dalam setiap langkah hidup saya. Saya menyampaikan rasa terima kasih yang tulus atas segala didikan, kasih sayang, dan pengorbanan yang tak ternilai serta tak akan pernah mampu saya balas sepenuhnya.

Faiz Muzaki, Gilang Ramadhan dan Waliid Ilham Ramadhan

Yang telah menjadi rekan tim terbaik selama proses pengerjaan skripsi ini.

Seluruh Keluarga Besar Ilmu Komputer 2021

Yang selalu memberikan semangat serta dukungan kepada saya.

Almamater Tercinta, Universitas Lampung dan Jurusan Ilmu Komputer

Tempat menimba ilmu yang telah memberikan banyak pengalaman berharga.

SANWACANA

Puji syukur kehadiran Allah Subhannahu Wa Ta'Alaa atas berkah, rahmat dan hidayah-Nya, serta petunjuk dan pedoman yang diajarkan Rasulullah Muhammad Shalallahu Alaihi Wassallam sehingga skripsi dengan judul “Implementasi Web Application Firewall (WAF) Dan Content Delivery Network (CDN) Untuk Meningkatkan Keamanan Dan Kecepatan Pada Website Fmipa Universitas Lampung” ini dapat diselesaikan dengan baik. Selama pembuatan skripsi ini, penulis mendapat banyak dukungan serta bimbingan dari berbagai pihak sehingga pada kesempatan yang baik ini penulis ingin menyampaikan rasa terimakasih yang sebanyak-banyaknya kepada :

1. Kedua orang tua tercinta, kakak, mbak dan ponakan yang sentantiasa selalu memberikan doa, dukungan dan semangat selama menjalani seluruh kegiatan yang dilakukan penulis.
2. Bapak Dr. Eng. Heri Satria, S.Si., M.Si. selaku Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.
3. Bapak Dwi Sakethi, S.Si., M.Kom. selaku Ketua Jurusan Ilmu Komputer sekaligus sebagai Dosen Pembimbing Akademik.
4. Bapak Muhaqiqin, S.Kom., M.T.I. selaku Dosen Pembimbing 1 yang telah memberikan banyak arahan, saran dan masukan dalam proses menyelesaikan skripsi, serta telah sabar membimbing penulis selama proses pengerjaan skripsi.
5. Bapak Mulyono, S.Si., M.Si., Ph.D. selaku Dosen Pembimbing 2 yang telah membimbing dan memberikan masukan dalam proses pengerjaan skripsi.
6. Bapak Dr. Aristoteles, S.Si., M.Si. selaku Dosen Pembahas yang telah membantu memberikan saran dan masukan dalam penelitian ini.

7. Bapak dan Ibu Dosen Jurusan Ilmu Komputer FMIPA Universitas Lampung yang telah memberikan ilmunya kepada penulis selama menimba ilmu semasa perkuliahan.
8. Ibu Ade Nora Maela beserta staf Jurusan Ilmu Komputer yang telah membantu penulis dalam urusan administrasi selama perkuliahan.
9. Rekan tim pengembang *Website* FMIPA Universitas Lampung, Faiz Muzaki, Gilang Ramadhan dan Waliid Ilham Ramadhan yang telah menjadi teman sekaligus sahabat untuk pengerjaan skripsi ini. Meskipun pada awalnya penulis bukan bagian dari tim, namun atas kepercayaan yang diberikan, penulis dapat bergabung dan berkontribusi dalam proses pengembangan website tersebut. Alhamdulillah, berkat kerja sama dan dukungan dari rekan-rekan semua, pengerjaan skripsi ini dapat diselesaikan dengan baik.
10. Anak-anak Ucup Family atas kesempatan yang telah diberikan dalam menjalin pertemanan, hubungan, dan komunikasi. Penulis merasa senang dan bersyukur karena telah mengenal kalian satu per satu dan menjalin hubungan pertemanan. Penulis percaya bahwa tidak ada pertemuan melainkan karena suatu sebab. Oleh karena itu penulis berharap hubungan pertemanan ini selalu terjaga hingga sampai kita tua.
11. Teman-temanku di grup Harta Tahta Nilai A yaitu Vidya, Roy, Cindy, jihan, Ayuni, Abiyyi yang telah senantiasa membantu dari awal semester dan berjuang bersama-sama hingga akhir, penulis berharap komunikasi kita tidak akan pernah putus dan selalu menjalin silaturahmi hingga akhir hayat dan penulis mendoakan kalian semua sukses dengan versi terbaik kalian masing-masing.
12. Keluarga besar Jurusan Ilmu Komputer Angkatan 2021 yang tidak dapat disebutkan satu persatu.

Bandar Lampung, 26 Juni 2025

Ikhsan Saputra
NPM. 2117051037

DAFTAR ISI

DAFTAR ISI.....	i
DAFTAR TABEL	iii
DAFTAR GAMBAR.....	v
I. PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah.....	5
1.3 Batasan Masalah	5
1.4 Tujuan	6
1.5 Manfaat	6
II. TINJAUAN PUSTAKA.....	8
2.1 Penelitian Terdahulu	8
2.2 Landasan Teori.....	12
2.2.1 Cyber Security	12
2.2.2 Keamanan Web.....	13
2.2.3 Performa Web	13
2.2.4 <i>Framework SCRUM</i>	14
2.2.5 <i>Web Applications Firewall (WAF)</i>	18
2.2.6 <i>Content Delivery Network (CDN)</i>	20
2.2.7 <i>Cloudflare</i>	21
2.2.8 <i>Distributed Denial of Service (DDoS)</i>	21

2.2.9	SQL Injection.....	22
2.2.10	Cross-Site Scripting (XSS).....	22
2.2.11	GTMetrix	23
III.	METODOLOGI PENELITIAN	24
3.1	Waktu dan Tempat Penelitian.....	24
3.1.1	Waktu dan Tempat Penelitian.....	24
3.2	Tahapan Penelitian.....	24
3.2.1	Identifikasi Masalah.....	25
3.2.2	Studi Literatur	26
3.2.3	Penerapan <i>Framework</i> Scrum.....	26
IV.	HASIL DAN PEMBAHASAN.....	68
4.1	<i>Scrum Event Sprint 1</i>	68
4.2	<i>Scrum Event Sprint 2</i>	79
4.3	<i>Scrum Event Sprint 3</i>	86
4.4	<i>Scrum Event Sprint 4</i>	96
4.5	<i>Scrum Event Sprint 5</i>	110
4.6	<i>Scrum Event Sprint 6</i>	120
4.7	Penulisan Laporan.....	128
V.	KESIMPULAN DAN SARAN	129
5.3	Kesimpulan	129
5.2	Saran	130
	DAFTAR PUSTAKA.....	131

DAFTAR TABEL

Tabel 1. Penelitian Terdahulu	8
Tabel 2. Pembagian Peran.....	27
Tabel 3. User Story	27
Tabel 4. Daftar Perangkat Keras	29
Tabel 5. Daftar Perangkat Lunak	29
Tabel 6. Product Backlog Items	31
Tabel 7. Sprint Backlog 1 (Hari 1 - Hari 18)	56
Tabel 8. Sprint Backlog 2 (Durasi : Hari 19 - Hari 36).....	58
Tabel 9. Sprint Backlog 3 (Durasi : Hari 37 - Hari 54).....	61
Tabel 10. Sprint Backlog 4 (Durasi : Hari 55 - Hari 72).....	63
Tabel 11. Sprint Backlog 5 (Durasi : Hari 73 - Hari 90).....	65
Tabel 12. Test case Penyusunan skenario pengujian	74
Tabel 13. Sprint Riview Dari Sprint 1	76
Tabel 14. Sprint Retrospective Dari Sprint 1	79
Tabel 15. Sprint Riview Dari Sprint 2	83
Tabel 16. Sprint Retrospective Dari Sprint 2	85
Tabel 17. Pengujian Halaman Beranda.....	88
Tabel 18. Pengujian Halaman Profile	89
Tabel 19. Sprint Riview dari Sprint 3	92
Tabel 20. Sprint Retrospective dari Sprint 3	95
Tabel 21. Pengujian Keamanan WAF.....	104
Tabel 22. Pengujian Kecepatan CDN	105
Tabel 23. Sprint Riview dari Sprint 4	107
Tabel 24. Sprint Retrospective dari Sprint 4.....	109
Tabel 25. Pengujian End-To-End.....	114

Tabel 26. Penyempurnaan Keamanan Sistem.....	115
Tabel 27. Sprint Riview dari Sprint 5	117
Tabel 28. Sprint Retrospective dari Sprint 5	119
Tabel 29. Pengujian Ulang Keamanan Menggunakan Black Box Testing.....	121
Tabel 30. Pengujian Ulang Kecepatan Menggunakan Black Box Testing	124
Tabel 31. Sprint Retrospective dari Sprint 6	127

DAFTAR GAMBAR

Gambar 1. 1	Performa website FMIPA UNILA menggunakan Gtmetrix.....	2
Gambar 1. 2	Performa website FMIPA UNILA menggunakan webpagetest	3
Gambar 2. 1	Tahapan Framework Scrum (Schwaber & Sutherland, 2017).....	15
Gambar 2. 2	Cara kerja WAF	19
Gambar 2. 3	Cara kerja CDN	20
Gambar 3. 1	Tahapan Penelitian.....	24
Gambar 3. 2	Aktivitas di platform trello	56
Gambar 4. 1	Daftar Tugas Sprint Backlog 1 di trello.....	69
Gambar 4. 2	Analisis Hasil Gtmetrix	73
Gambar 4. 3	Analisis Hasil SecurityHeaders	73
Gambar 4. 4	Login Valid.....	75
Gambar 4. 5	Login Email Tidak Terdaftar	75
Gambar 4. 6	Hasil Logout	75
Gambar 4. 7	Daftar Tugas Sprint Backlog 2 di trello.....	80
Gambar 4. 8	Pengujian Fitur Berita.....	82
Gambar 4. 9	Pengujian Kategori Berita.....	82
Gambar 4. 10	Daftar Tugas Sprint Backlog 3 di Trello	87
Gambar 4. 11	Daftar Tugas Sprint Backlog 4 di Trello	96
Gambar 4. 12	Implementasi WAF Cloudflare	100
Gambar 4. 13	Implementasi CDN Cloudflare	101
Gambar 4. 14	Hasil Terbaru Menggunakan Securityheaders.	102
Gambar 4. 15	Hasil Activity Cloudflare.	103
Gambar 4. 16	Hasil Analisis Kecepatan Terbaru Menggunakan Gtmetrix.	104
Gambar 4. 17	Hasil Analisis Kecepatan Terbaru Menggunakan PageSpeed. ...	104
Gambar 4. 18	Daftar Tugas Sprint Backlog 5 di Trello	108

Gambar 4. 19 Pengujian Halaman Berita	112
Gambar 4. 20 Hasil Permintaan Diblokir	113
Gambar 4. 21 Pengujian API SINTA	113
Gambar 4. 22 Diskusi Tim Menggunakan Gmeet	113
Gambar 4. 23 Hasil Pengujian SQL injection	123
Gambar 4. 24 Hasil Pengujian XSS	123
Gambar 4. 25 Hasil Pengujian Access File Sensitive	123
Gambar 4. 26 Hasil Pengujian Challenge Login	124
Gambar 4. 27 Hasil Pengujian Rate Limiting	124

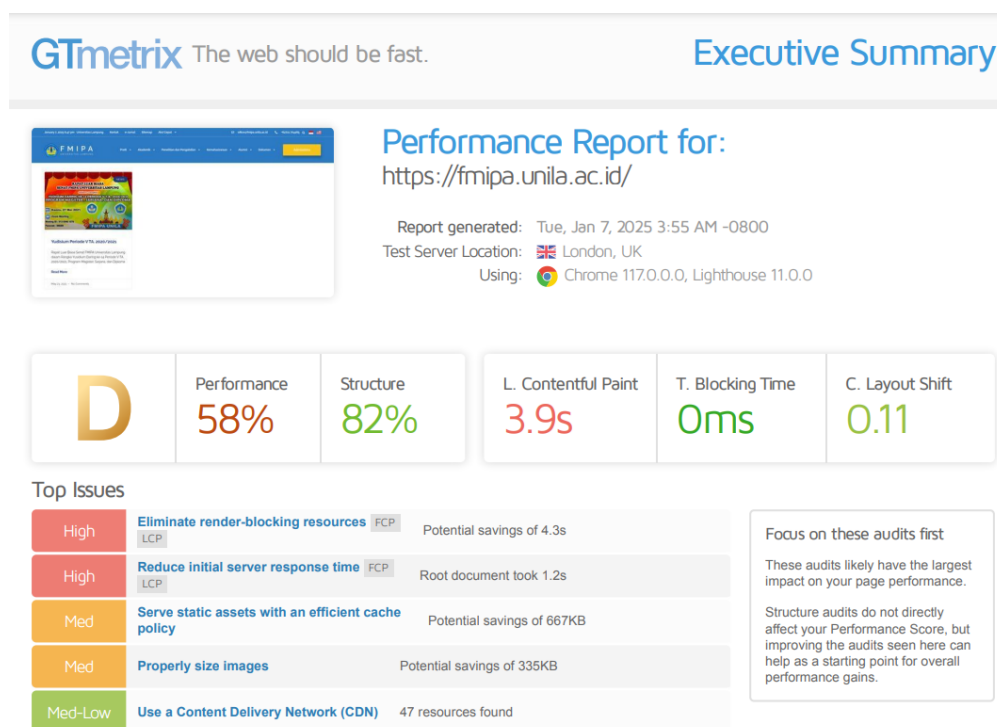
I. PENDAHULUAN

1.1 Latar Belakang

Saat ini, perkembangan teknologi informasi terus mengalami kemajuan seiring berjalannya waktu. Penggunaan internet telah menjadi bagian dari aktivitas sehari-hari, di mana berbagai kegiatan kini memanfaatkan koneksi internet dalam prosesnya (Sukmana et al., 2024). Internet menjadi kebutuhan *vital* karena menjadi tempat berkomunikasi, berbisnis, bersosialisasi, *browsing* dan lainnya (Laksmiati, 2022). Jumlah pengguna internet juga terus meningkat dengan sangat pesat. Oleh karena itu, sebuah *server* harus memiliki kapasitas yang memadai untuk melayani jutaan pengguna dari seluruh dunia secara bersamaan tanpa menambah *latensi* atau, yang lebih buruk, mengalami *downtime*. Selain memastikan performa server, perhatian terhadap keamanan jaringan dan *server* juga sangat penting untuk mencegah adanya celah atau kerentanan yang dapat dimanfaatkan oleh pihak yang tidak bertanggung jawab, seperti *threat actor* atau *hacker*. (Rahmadyanto & Chandra, 2023).

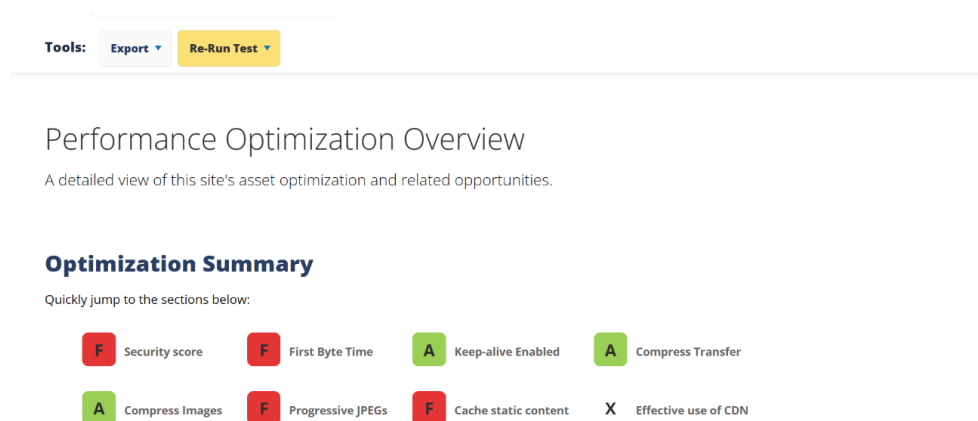
Website memungkinkan siapa saja di seluruh dunia untuk mengakses dan mengelola informasi dari berbagai sumber yang tersedia di internet. Hingga saat ini, *website* tetap memiliki peran yang sangat penting dalam mempermudah penyampaian informasi, berita, dan berbagai kebutuhan lainnya (Alim et al., 2024). Hal ini juga diterapkan oleh Fakultas Matematika dan Ilmu Pengetahuan Alam (FMIPA) Universitas Lampung dalam memanfaatkan *website* sebagai menyampaikan informasi akademik,

penelitian, kegiatan, serta layanan kepada mahasiswa, staf, dan masyarakat umum. FMIPA Universitas Lampung memerlukan *website* yang andal, aman, dan cepat. Saat ini, *website* FMIPA Universitas Lampung masih tergolong kurang aman dan memiliki performa yang kurang optimal. Berdasarkan laporan *GTmetrix*, performa *website* FMIPA Universitas Lampung mendapat skor 58% (D), yang menunjukkan bahwa kecepatan pemuatan masih kurang optimal. Struktur *website* cukup baik dengan skor 82%, tetapi LCP 3.9 detik menandakan elemen terbesar halaman dimuat terlalu lama. Sementara itu, Total *Blocking Time* 0 ms menunjukkan *interaktivitas* yang baik, dan CLS 0.11 masih dalam batas wajar tetapi perlu sedikit perbaikan. Beberapa masalah utama yang teridentifikasi termasuk *render-blocking resources*, waktu respons server lambat (1.2 detik), aset statis tanpa *caching* efisien, dan gambar berukuran besar 82%. Bisa dilihat pada gambar di bawah berikut ini.



Gambar 1. 1 Performa *website* FMIPA UNILA menggunakan *Gtmetrix*

Berdasarkan hasil analisis *Performance Optimization Overview* menggunakan *tools webpagetest*, *website* FMIPA Universitas Lampung masih memiliki beberapa aspek yang perlu diperbaiki. Skor keamanan yang rendah (*F*) menunjukkan adanya potensi celah keamanan yang dapat dimanfaatkan oleh pihak tidak bertanggung jawab. Selain itu, waktu respon server yang lambat (*F - First Byte Time*) menyebabkan penundaan dalam pemuatan halaman, yang dapat berdampak pada kenyamanan pengguna. Kurangnya implementasi *Content Delivery Network (CDN)* yang efektif juga menjadi salah satu faktor yang memperlambat akses, terutama bagi pengguna dari lokasi yang jauh dari server utama. Selain itu, strategi *caching* yang tidak optimal (*F - Cache Static Content*) mengakibatkan halaman harus dimuat ulang setiap kali diakses, sementara format gambar yang belum menggunakan *progressive JPEGs* juga berkontribusi pada lambatnya pemuatan konten visual. Bisa dilihat pada gambar di bawah berikut ini.



Gambar 1. 2 Performa *website* FMIPA UNILA menggunakan *webpagetest*

Website yang aman dan cepat sangat penting untuk memastikan akses informasi tanpa hambatan sekaligus melindungi data sensitif dari ancaman siber. Namun, kendala seperti serangan oleh *hacker* yang menyusupkan iklan ke halaman *website* kerap menjadi masalah. Banyak *website* dari berbagai kalangan masih rentan terhadap serangan semacam ini akibat adanya celah keamanan yang belum ditangani dengan baik (Ardiansyah et al., 2023). Penyusupan iklan seperti judi *online* pada *website* merupakan salah satu bentuk serangan siber yang sering terjadi, terutama pada situs dengan tingkat keamanan yang lemah. Serangan ini

umumnya disebabkan oleh *defacement*, *injeksi malware*, atau *spam SEO injection*, di mana peretas menyisipkan skrip berbahaya untuk menampilkan iklan yang tidak diinginkan atau mengarahkan pengunjung ke situs tertentu. Beberapa *website* di Universitas Lampung, seperti *V-Class UNILA*, *E-KKN UNILA*, dan *FMIPA UNILA*, telah mengalami serangan serupa, yang tidak hanya mengganggu tampilan *website* tetapi juga berpotensi membahayakan data dan kredibilitas institusi. Penyebab utama dari penyusupan ini sering kali berasal dari celah keamanan pada sistem manajemen konten (CMS), *plugin*, atau tema yang tidak diperbarui, serta konfigurasi server yang kurang aman. Selain itu, penggunaan *password* yang lemah, tidak adanya *otentikasi* dua faktor (2FA), serta kurangnya proteksi terhadap serangan seperti *SQL Injection* dan *Cross-Site Scripting (XSS)* membuat *website* lebih rentan terhadap peretasan.

Keamanan sebuah *website* merupakan aspek yang sangat penting. Untuk melindungi *website*, langkah-langkah seperti menggunakan *Web Application Firewall (WAF)*, antivirus, memperbarui CMS dan *plugin* secara berkala, atau perangkat lunak serupa dapat dilakukan pada komputer atau *router* yang terhubung secara langsung maupun berada dalam jaringan *server* tempat perangkat lunak web tersebut berjalan (Ardiansyah et al., 2023). Dengan meningkatkan sistem keamanan *website*, risiko penyusupan dapat diminimalisir sehingga *website* tetap dapat berfungsi dengan baik, memberikan pengalaman pengguna yang optimal, serta menjaga kredibilitas institusi. Oleh karena itu, perlu adanya upaya yang lebih serius dalam mengamankan *website* Universitas Lampung agar tidak terus menjadi target serangan siber.

Content Delivery Network (CDN) adalah sistem jaringan *server* yang berfungsi mendistribusikan *server* secara global. Dengan adanya CDN, permintaan dari *client* tidak perlu langsung menuju *server* utama yang mungkin berlokasi jauh dan memerlukan waktu lama untuk mengunduh konten. Sebagai gantinya, CDN akan mencegat permintaan tersebut dan mengarahkannya ke *server* CDN terdekat, sehingga mengurangi *latensi*. Salah satu contoh CDN, yaitu *Cloudflare*, juga dilengkapi dengan fitur *Web Application Firewall (WAF)* yang dapat digunakan

untuk meningkatkan keamanan server dan *website* (Rahmadyanto & Chandra, 2023). WAF adalah metode perlindungan untuk *website* yang berfungsi mencegah serangan dari pihak-pihak yang tidak bertanggung jawab, seperti *attacker* atau *hacker*. Keamanan *website* menjadi salah satu aspek yang sangat penting untuk diperhatikan. Serangan siber, seperti *SQL injection*, *Cross-Site Scripting (XSS)*, dan *Distributed Denial of Service (DdoS)*, dapat merusak reputasi serta mengurangi keandalan sebuah *website* (Alim dkk., 2024).

Implementasi WAF dan CDN dalam pengembangan *website* FMIPA Universitas Lampung diharapkan dapat menjawab tantangan terkait keamanan dan kecepatan. Dengan demikian, penelitian ini bertujuan untuk merancang, mengimplementasikan, dan mengevaluasi penggunaan Web WAF dan CDN pada *website* FMIPA Universitas Lampung, serta menilai dampaknya terhadap aspek keamanan dan kecepatan akses *website* tersebut. Hasil dari penelitian ini diharapkan memberikan kontribusi positif.

1.2 Rumusan Masalah

Rumusan masalah pada penelitian ini adalah sebagai berikut.

Bagaimana pengaruh implementasi WAF dan CDN untuk meningkatkan keamanan dalam menghadapi ancaman serangan siber seperti *Distributed Denial of Service (DDoS)*, *SQL Injection*, dan *Cross-Site Scripting (XSS)* dan peningkatan kecepatan akses *website* FMIPA Universitas Lampung.

1.3 Batasan Masalah

Batasan masalah pada penelitian ini adalah sebagai berikut.

1. Penelitian ini hanya akan fokus pada implementasi dan evaluasi teknologi WAF dan CDN sebagai solusi untuk meningkatkan keamanan dan kecepatan

akses *website*.

2. Penelitian mengenai keamanan dalam penelitian ini akan terbatas pada beberapa jenis serangan umum terhadap aplikasi web, seperti DDoS, SQL *Injection*, dan XSS.
3. Pengujian kecepatan akses hanya akan dilakukan berdasarkan waktu pemuatan halaman (*page load time*) dan *latency* yang diukur sebelum dan sesudah implementasi CDN.
4. Implementasi dan pengujian WAF dan CDN hanya akan dilakukan pada lingkungan *website* FMIPA Universitas Lampung yang berbasis pada teknologi dan infrastruktur yang sudah ada (misalnya *framework Laravel* atau teknologi pendukung lainnya).

1.4 Tujuan

Tujuan dari penelitian ini adalah sebagai berikut.

Tujuan Penelitian ini adalah untuk mengimplementasikan WAF dan CDN pada *website* FMIPA Universitas Lampung guna meningkatkan keamanan terhadap ancaman serangan siber seperti DDoS, SQL *Injection*, dan XSS serta meningkatkan kecepatan akses *website*.

1.5 Manfaat

Manfaat dari penelitian ini adalah sebagai berikut.

1. Memberikan kontribusi terhadap pengembangan ilmu di bidang keamanan dan optimasi kecepatan *website*, terutama dalam konteks penerapan teknologi WAF dan CDN pada *website* institusi pendidikan.
2. Memberikan pengalaman akses yang lebih baik bagi pengguna *website* FMIPA, terutama mahasiswa, dosen, dan pihak eksternal yang memerlukan akses cepat dan aman ke berbagai informasi akademik.

3. Penerapan teknologi WAF dan CDN yang efektif dapat meningkatkan citra dan kredibilitas institusi, karena *website* yang aman dan responsif mencerminkan komitmen terhadap kualitas layanan digital yang diberikan kepada pengguna.

II. TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Tabel 1. Penelitian Terdahulu

No	Penulis & Tahun	Judul Penelitian	Tujuan Penelitian	Metode Penelitian	Hasil Penelitian
1.	Sukmana et al., 2024	Analisis Web Performance Load Test Setelah Menggunakan Cloud Web Application Firewall (WAF) Studi Kasus Pada Situs Web PT Neptus Teknologi Indonesia	Menganalisis pengaruh penggunaan <i>Cloud Web Application Firewall (WAF)</i> terhadap kinerja situs web dan untuk menghindari serangan peretas pada situs web PT Neptus Teknologi Indonesia	Observasi, pengujian performa, dan analisis data	Penggunaan <i>Cloud WAF</i> mempercepat akses situs web (<i>Performance Score</i> meningkat dari 61% ke 98%) dan mencegah serangan, memberikan perlindungan dari <i>DDoS</i> dan <i>hacking tool</i> . Situs web menjadi lebih aman dan lebih cepat diakses

No	Penulis & Tahun	Judul Penelitian	Tujuan Penelitian	Metode Penelitian	Hasil Penelitian
2.	(Alim et al., 2024).	Monitoring dan Pencegahan Serangan Judi Online (Slot Gacor) pada Website	Mencegah dan memantau serangan <i>hacker</i> yang menyisipka n iklan judi online pada website	Deskriptif, Implement asi <i>Cloudflare</i> (<i>WAF</i>)	<i>Cloudflare</i> berhasil memblokir 126 <i>URL path</i> yang berisi iklan judi <i>online</i> (Slot Gacor)
3.	(Rahma dyanto & Chandra, 2023)	Analisis keamanan Content Delivery Network (CDN) <i>Cloudflare</i> (Studi kasus: Web Hakazon)	Menganalisis s keamanan <i>website</i> yang menggunakan an CDN <i>Cloudflare</i> dan melakukan perbandinga n dengan <i>website</i> yang tidak menggunakan an CDN <i>Cloudflare</i> .	Perbandin gan <i>Website</i> yang telah diimpleme ntasikan CDN dengan <i>website</i> yang belum diimpleme ntasikan.	Keamanan situs web yang menggunakan <i>Cloudflare</i> CDN lebih baik dibandingkan yang tidak menggunakan, dengan WAF memblokir serangan.

Selama proses penelitian ini, data diperoleh melalui kajian pustaka dengan merujuk pada beberapa buku atau penelitian sebelumnya yang memiliki topik yang relevan dengan topik penelitian ini. Tujuan dari langkah ini adalah untuk memperkuat proses penelitian serta menghindari terjadinya *plagiarisme*. Pada

bagian ini, akan dibahas berbagai penelitian terdahulu yang berhubungan dengan topik yang sedang diteliti.

Penelitian lain yang berkaitan yaitu “Analisis Web Performance Load Test Setelah Menggunakan Cloud Web Application Firewall (WAF) Studi Kasus Pada Situs Web PT Neptus Teknologi Indonesia” oleh Sukmana dkk pada tahun 2024. Penelitian ini membahas pentingnya implementasi Cloud WAF dalam meningkatkan performa dan keamanan situs web perusahaan PT Neptus Teknologi Indonesia. Penelitian ini diawali dengan observasi terhadap performa situs web perusahaan, yang menunjukkan bahwa website tersebut memiliki kinerja yang rendah menurut Pingdom Tools dan GTMetrix, serta tidak dilengkapi perlindungan keamanan. Untuk mengatasi masalah tersebut, diterapkan solusi berbasis cloud menggunakan WAF, khususnya platform Imperva Cloud. Langkah-langkah implementasi meliputi pemasangan, konfigurasi, hingga pengujian performa sebelum dan sesudah penggunaan WAF. Hasil penelitian menunjukkan adanya peningkatan signifikan pada skor performa situs, dari 61% menjadi 98%, dengan penurunan waktu Largest Contentful Paint dari 3,2 detik menjadi 906 Millisecond (MS). Selain itu, pengujian simulasi serangan siber memperlihatkan efektivitas WAF dalam mencegah akses tidak sah dengan menampilkan pesan "Access Denied" dan memblokir permintaan berbahaya. Hal ini membuktikan bahwa WAF mampu menyaring lalu lintas internet, memisahkan lalu lintas sah dari ancaman, serta melindungi data penting situs.

Penelitian oleh Endi Sjaiful Alim dkk pada tahun 2024 bulan juni dengan judul “*Monitoring dan Pencegahan Serangan Judi Online (Slot Gacor) pada Website*” Jurnal ini berfokus pada keamanan *website* sebagai upaya mencegah serangan *hacker* yang menyisipkan konten ilegal seperti judi *online*, yang dapat merugikan reputasi *website*. Penelitian ini menggunakan pendekatan metode deskriptif untuk memberikan gambaran rinci tentang bagaimana memantau dan mencegah serangan *hacker* dengan memanfaatkan layanan *Cloudflare*, khususnya fitur WAF. *Cloudflare* diterapkan sebagai sistem keamanan pada

website bpti.uhamka.ac.id. Studi ini mencakup beberapa tahapan, termasuk identifikasi masalah, instalasi dan konfigurasi *Cloudflare*, serta implementasi aturan WAF untuk memblokir serangan. Data dikumpulkan melalui observasi langsung pada *Cloudflare* dan dianalisis dalam bentuk grafik serta jumlah *path URL* yang berhasil diblokir. Selama delapan hari pengujian, *Cloudflare* berhasil memblokir 126 *path URL* terkait konten judi online, dengan sebagian besar berkaitan dengan "Slot Gacor.". Penelitian ini menyoroti peran penting *Cloudflare* dalam meningkatkan keamanan *website* dengan cara memantau lalu lintas dan mencegah ancaman sebelum mencapai target. Implementasi WAF *Cloudflare* terbukti efektif dalam melindungi *website* dari serangan *hacker*. Hal ini menjadikan *website* bpti.uhamka.ac.id sebagai contoh penerapan sistem keamanan yang baik di lingkungan Universitas Muhammadiyah Prof. DR. HAMKA. Namun, penelitian ini memiliki beberapa kelemahan. Salah satunya adalah kurangnya perbandingan dengan platform keamanan lain untuk memberikan evaluasi yang lebih komprehensif. Selain itu, penjelasan teknis mengenai konfigurasi *Cloudflare* kurang mendetail, sehingga sulit di replikasi oleh pengelola *website* lainnya. Kesimpulan penelitian ini menyatakan bahwa WAF *Cloudflare* adalah solusi yang efektif untuk mencegah serangan siber, terutama penyisipan konten ilegal seperti judi *online*. Penelitian ini merekomendasikan adanya kajian lanjutan yang membandingkan berbagai platform keamanan serta dokumentasi konfigurasi teknis yang lebih terperinci. Secara keseluruhan, jurnal ini memberikan kontribusi signifikan terhadap literatur keamanan *website*, terutama dalam konteks serangan judi *online*, yang masih jarang dibahas dalam penelitian sebelumnya.

Terdapat penelitian yang dilakukan oleh Eko Putra Rahmadyanto & Dian Widiyanto Chandra pada tahun 2023 yang berjudul "*Analisis keamanan Content Delivery Network (CDN) Cloudflare (Studi kasus: Web Hakazon)*". Penelitian ini membahas tentang efektivitas penggunaan CDN *Cloudflare*, terutama fitur WAF dalam meningkatkan keamanan dan performa suatu *website*. Dengan semakin pesatnya pertumbuhan jumlah *website* dan pengguna internet, kebutuhan akan sistem yang dapat menangani *trafik* tinggi dan

mengurangi *latensi* tanpa mengurangi performa *server* semakin meningkat. CDN, termasuk *Cloudflare*, membantu mendistribusikan permintaan ke *server* terdekat, mengurangi beban pada server utama, dan mempercepat akses data. Selain itu, fitur WAF pada *Cloudflare* berfungsi untuk mencegah serangan siber seperti *SQL Injection*, *Cross-Site Scripting (XSS)*, dan *Brute Force*, yang sering menjadi ancaman bagi *website*. Metode penelitian yang digunakan adalah penelitian berbasis *grey box testing*, di mana pengujian dilakukan pada *website* yang belum dan sudah menggunakan CDN serta WAF. Serangan yang dilakukan meliputi simulasi berbagai jenis ancaman keamanan untuk membandingkan efektivitas fitur tersebut dalam mencegah kerentanan. Hasil penelitian menunjukkan bahwa penggunaan *Cloudflare* CDN dan WAF secara signifikan memperkuat keamanan *website*, terutama melalui fitur pencegahan serangan otomatis pada tingkat *server*. Penelitian ini juga menekankan pentingnya evaluasi keamanan secara berkala, implementasi *secure coding*, dan pembaruan sistem untuk menjaga integritas dan keamanan dari potensi ancaman baru, seperti *zero-day vulnerabilities*.

2.2 Landasan Teori

Landasan Teori merupakan bagian dalam suatu penelitian atau laporan yang berisi konsep, teori, dan prinsip yang mendukung serta menjadi dasar dalam pelaksanaan penelitian atau proyek yang dilakukan. Landasan Teori yang terdapat pada penelitian ini di antaranya sebagai berikut

2.2.1 Cyber Security

Keamanan siber (*cyber security*) mencakup sekumpulan alat, kebijakan, prinsip perlindungan, strategi pengendalian, pelatihan, serta teknologi yang dirancang untuk melindungi dunia digital, organisasi, dan pengguna dari berbagai ancaman. Keamanan siber melibatkan fungsi-fungsi yang terkait dengan komputasi, individu, infrastruktur, aplikasi, layanan, jaringan

telekomunikasi, serta semua data yang masuk dan keluar melalui internet. Keamanan siber bertujuan untuk melindungi *privasi*, menjaga integritas, dan memastikan ketersediaan informasi. Proses ini mencakup langkah-langkah untuk melindungi sistem informasi dari ancaman, baik serangan fisik maupun serangan digital (*cyber attack*), sehingga risiko terhadap data dan sistem dapat diminimalkan (Indah et al., 2022).

2.2.2 Keamanan Web

Keamanan web adalah upaya untuk melindungi aplikasi web dari kerentanan yang dapat memberikan akses atau memungkinkan modifikasi data oleh pihak yang tidak berwenang. Kerentanan pada *website* dapat menjadi ancaman serius karena dapat dieksploitasi oleh penyerang untuk merusak sistem. Tujuan utama dari keamanan aplikasi web mencakup kerahasiaan (*confidentiality*), integritas (*integrity*), dan ketersediaan informasi (*availability*). Ketiga elemen ini, yang dikenal sebagai *Triad CIA*, merupakan dasar penting dalam menjaga keamanan data, informasi, dan layanan komputasi (Harahap, 2021). Keamanan aplikasi web adalah aspek krusial yang harus diperhatikan. Langkah-langkah untuk melindungi aplikasi web dapat mencakup penggunaan *firewall*, antivirus, atau perangkat lunak serupa pada komputer atau router yang terhubung langsung atau berada dalam jaringan yang sama dengan *server* aplikasi web tersebut (Riska & Alamsyah, 2021).

2.2.3 Performa Web

Performa web berkaitan dengan memastikan situs web berjalan dengan cepat, termasuk memberikan kesan kecepatan pada proses yang mungkin memakan waktu (Laksmiati, 2022). Sebuah *website* dikatakan berkualitas baik jika mampu menampilkan halaman dengan cepat saat diakses, sehingga pengunjung dapat memperoleh informasi secara efektif. Jika proses pemuatan halaman lambat, hal ini dapat menghambat pengunjung dalam menerima

informasi dengan optimal. Keberadaan *website* juga berkontribusi dalam meningkatkan kualitas organisasi atau perusahaan yang menggunakannya. Waktu muat (*loading time*) memiliki pengaruh besar terhadap kecepatan akses dan kehandalan *web server*, yang berdampak pada kenyamanan pengguna. Oleh karena itu, performa *website* menjadi salah satu indikator utama keberhasilan dalam menyampaikan informasi melalui platform tersebut (Nurul Hima Hidayati, 2022).

2.2.4 Framework SCRUM

Scrum adalah kerangka kerja ringan yang membantu individu, tim, dan organisasi menghasilkan nilai melalui solusi adaptif untuk masalah kompleks. Scrum bukanlah proses atau metode yang kaku, melainkan kerangka kerja yang memungkinkan penerapan berbagai proses dan teknik. Dengan menyoroti efektivitas relatif dari teknik manajemen produk dan pekerjaan yang digunakan, Scrum mendorong peningkatan berkelanjutan pada produk, tim, dan lingkungan kerja (Schwaber & Sutherland, 2017).

Tim Scrum terdiri dari *Product Owner*, *Development Team*, dan *Scrum Master*. Mereka bekerja secara mandiri dan memiliki berbagai keahlian yang diperlukan untuk menyelesaikan tugas tanpa bergantung pada pihak luar. Struktur tim ini dirancang untuk meningkatkan fleksibilitas, kreativitas, dan produktivitas. Tim Scrum telah terbukti efektif dalam berbagai aplikasi, termasuk pekerjaan yang kompleks (Schwaber & Sutherland, 2017).

Berikut penjelasan lebih lanjut mengenai Tim Scrum :

1. *Product Owner*

Product Owner memiliki peran utama dalam memastikan bahwa produk yang dikembangkan memiliki nilai maksimal. Ia bertanggung jawab atas *Product Backlog*, yaitu daftar tugas yang harus dilakukan oleh Tim Pengembang. Dalam mengelola *Product Backlog*, *Product Owner* harus

menjelaskan tugas dengan jelas, menentukan prioritas, memastikan daftar tersebut transparan, dan membantu Tim Pengembang memahami tugas mereka. Walaupun ia bisa berbagi tugas dengan tim, tanggung jawab utama tetap ada padanya (Schwaber & Sutherland, 2017).

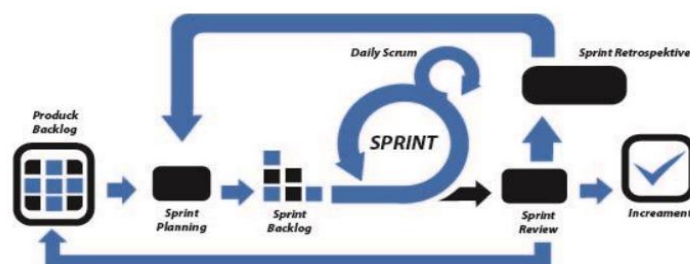
2. *Development Team*

Development Team adalah kelompok profesional yang bertanggung jawab untuk menghasilkan bagian produk yang sudah selesai dan siap rilis di akhir setiap Sprint. Mereka bekerja secara mandiri untuk merencanakan dan mengelola pekerjaan mereka, tanpa campur tangan pihak lain. Sinergi dan kerja-sama yang terjadi dalam tim ini membantu mereka bekerja lebih efisien dan efektif (Schwaber & Sutherland, 2017).

3. *Scrum Master*

Scrum Master adalah orang yang bertanggung jawab untuk memastikan Scrum diterapkan dengan benar dalam tim. Ia membantu tim dan pihak luar memahami cara kerja Scrum dan nilai-nilai yang ada di dalamnya. *Scrum Master* bekerja untuk membantu orang lain di luar tim memahami bagaimana mereka bisa berinteraksi dengan tim secara lebih efektif, serta memperbaiki interaksi yang tidak mendukung kerja tim. Hal ini bertujuan untuk meningkatkan nilai yang dihasilkan oleh tim (Schwaber & Sutherland, 2017).

Selain itu, Adapun beberapa tahapan yang terdapat dalam proses pengembangan *website* menggunakan *framework* Scrum seperti pada gambar.



Gambar 2. 1 Tahapan *Framework* Scrum (Schwaber & Sutherland, 2017)

1. *Product Backlog*

Product Backlog adalah daftar yang mencakup semua kebutuhan dan persyaratan untuk produk, yang menjadi dasar bagi setiap perubahan yang akan diterapkan. Daftar ini mencakup berbagai fitur, fungsi, perbaikan, dan peningkatan yang perlu dilakukan pada produk (Schwaber & Sutherland, 2017).

2. *Sprint Planning*

Sprint Planning adalah kegiatan untuk merencanakan pekerjaan yang akan dilakukan dalam Sprint. Semua anggota *Scrum Team* bekerja sama untuk membuat rencana ini. Acara *Sprint Planning* memiliki waktu terbatas, yaitu delapan jam untuk Sprint satu bulan. Jika Sprint lebih singkat, waktu acara akan lebih pendek. *Scrum Master* memastikan acara ini berlangsung dengan baik dan peserta memahami tujuannya, serta membantu *Scrum Team* agar acara tetap sesuai dengan waktu yang telah ditentukan (Schwaber & Sutherland, 2017).

3. *Sprint Backlog*

Sprint Backlog adalah kumpulan item *Product Backlog* yang dipilih untuk Sprint tertentu, beserta rencana untuk menyelesaikan *Increment* produk dan mencapai tujuan Sprint. Ini adalah perkiraan dari Tim Pengembang tentang fungsionalitas yang akan ada di *Increment* berikutnya, serta pekerjaan yang diperlukan untuk menyelesaikan *Increment* yang “Selesai”. Daftar ini menunjukkan semua pekerjaan yang diperlukan untuk mencapai *Sprint Goal*. Selain itu, *Sprint Backlog* mencakup perbaikan proses yang prioritas dari hasil *Retrospective* sebelumnya.

4. *Daily Scrum*

Daily Scrum adalah pertemuan harian yang berlangsung selama 15 menit untuk Tim Pengembang. Setiap hari selama Sprint, tim merencanakan pekerjaan mereka untuk 24 jam ke depan,

mengoptimalkan kerja sama dan kinerja tim dengan meninjau pekerjaan yang telah dilakukan dan memprediksi pekerjaan yang akan datang. *Daily Scrum* diadakan pada waktu dan tempat yang sama setiap hari untuk memudahkan proses. Rapat ini membantu Tim Pengembang mengevaluasi kemajuan menuju *Sprint Goal* dan memastikan apakah pekerjaan dalam *Sprint Backlog* dapat selesai tepat waktu. Dengan begitu, Tim Pengembang memiliki pemahaman yang jelas setiap hari mengenai cara mereka bekerja bersama sebagai tim swakelola untuk mencapai *Sprint Goal* dan menghasilkan Increment yang diharapkan pada akhir Sprint (Schwaber & Sutherland, 2017).

5. *Sprint Review*

Sprint Review adalah pertemuan yang diadakan di akhir Sprint untuk memeriksa hasil pekerjaan (*Increment*) dan jika diperlukan, memperbarui *Product Backlog*. Dalam pertemuan ini, Tim Scrum dan pemangku kepentingan berdiskusi tentang apa yang telah dicapai selama *Sprint* dan apa yang bisa dilakukan selanjutnya untuk meningkatkan nilai produk. Ini adalah rapat yang tidak bersifat formal, tujuannya bukan untuk melaporkan status, tetapi untuk mendapatkan masukan dan meningkatkan kolaborasi antar tim dan pemangku kepentingan. Acara ini berlangsung maksimal empat jam untuk *Sprint* satu bulan, dan lebih singkat untuk Sprint yang lebih pendek. *Scrum Master* memastikan acara berjalan dengan baik dan peserta memahami tujuannya. *Scrum Master* juga membantu agar acara tetap sesuai dengan waktu yang telah ditentukan (Schwaber & Sutherland, 2017).

6. *Sprint Retrospective*

Sprint Retrospective adalah pertemuan di mana Tim Scrum melakukan evaluasi diri dan menyusun rencana perbaikan yang akan diterapkan pada Sprint selanjutnya. Pertemuan ini diadakan setelah *Sprint Review* dan sebelum *Sprint Planning* berikutnya. Durasi pertemuan ini maksimal tiga jam untuk Sprint satu bulan, dan lebih pendek untuk

Sprint yang lebih singkat. *Scrum Master* bertanggung jawab untuk memastikan acara ini berjalan dan memastikan peserta memahami tujuannya. *Scrum Master* berperan menjaga agar pertemuan ini tetap positif dan produktif serta memastikan agar pertemuan selesai tepat waktu sesuai dengan batasan yang ada. Meskipun berpartisipasi dalam pertemuan, *Scrum Master* tetap bertanggung jawab atas kelancaran proses Scrum (Schwaber & Sutherland, 2017).

7. *Increment*

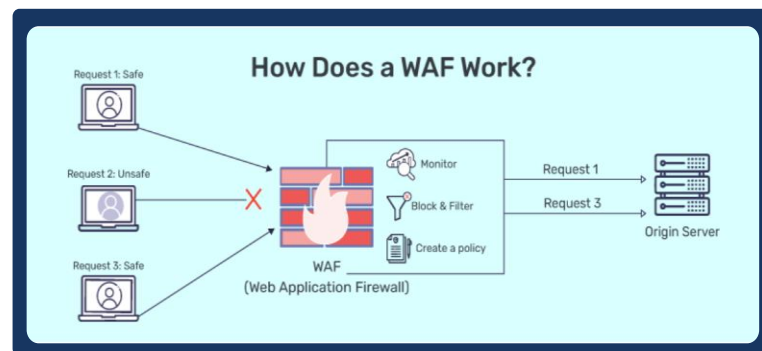
Increment adalah total dari semua item *Product Backlog* yang telah diselesaikan selama Sprint, ditambah dengan nilai *Increment* dari Sprint-sprint sebelumnya. Pada akhir Sprint, *Increment* baru harus dinyatakan “*Done*,” yang berarti dalam kondisi siap pakai dan memenuhi kriteria “*Done*” yang ditetapkan oleh Tim Scrum. *Increment* merupakan sekumpulan pekerjaan yang sudah selesai dan dapat diperiksa yang mendukung pendekatan empiris pada akhir Sprint. *Increment* ini adalah langkah maju menuju pencapaian tujuan atau visi produk. *Increment* harus berada dalam kondisi yang siap pakai, meskipun *Product Owner* memilih untuk tidak merilisnya (Schwaber & Sutherland, 2017).

2.2.5 Web Applications Firewall (WAF)

Web Application Firewall (WAF) adalah mekanisme keamanan yang dirancang untuk melindungi aplikasi web dengan memantau dan memblokir lalu lintas *Hypertext Transfer Protocol (HTTP)* yang masuk dan keluar. Berbeda dengan *firewall* konvensional yang berfungsi sebagai pengaman antara *server*, WAF memiliki kemampuan untuk memfilter dan menganalisis konten spesifik dari aplikasi web. Dengan memeriksa lalu lintas HTTP, WAF dapat mencegah serangan yang memanfaatkan kerentanan pada keamanan aplikasi web (Hidayah & Saptono, 2017).

Web Application Firewall (WAF) memiliki berbagai fungsi penting, seperti memantau lalu lintas (*traffic monitoring*), melindungi direktori (*secure directory*), memfilter string tertentu (*string filtering*), dan memberikan perlindungan terhadap serangan seperti *SQL Injection*, *Cross-Site Scripting (XSS)*, serta *Unrestricted File Upload*. WAF menciptakan lapisan keamanan tambahan yang mampu mendeteksi dan mencegah ancaman terhadap aplikasi web. Beberapa tindakan yang dapat dilakukan WAF meliputi menghentikan permintaan dengan status 403 *Forbidden* serta melakukan *virtual patching* untuk memperbaiki kerentanan secara langsung (Riska & Alamsyah, 2021).

Adapun cara kerja WAF seperti pada gambar.



Gambar 2. 2 Cara kerja WAF

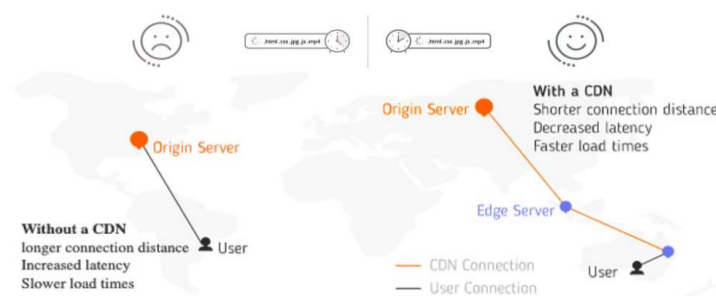
Setiap permintaan (*request*) dari pengguna akan dianalisis terlebih dahulu oleh WAF sebelum diteruskan ke server asal (*origin server*). Permintaan yang dianggap aman akan dilanjutkan ke server untuk diproses, sedangkan permintaan yang terdeteksi berpotensi membahayakan akan diblokir. Proses ini dilakukan melalui mekanisme *monitoring* lalu lintas, penyaringan dan pemblokiran permintaan, serta penerapan kebijakan keamanan (*policy*). Dengan implementasi WAF, risiko serangan siber seperti *SQL Injection*, *Cross-Site Scripting (XSS)*, dan serangan *Distributed Denial of Service (DDoS)* dapat di minimalisasi, sehingga keamanan dan kinerja aplikasi web dapat terjaga dengan optimal.

2.2.6 Content Delivery Network (CDN)

Content Delivery Network (CDN) adalah sekumpulan *server* yang tersebar secara geografis untuk mempercepat pengiriman konten web dengan mendekatkannya ke lokasi pengguna. Dengan menggunakan *caching*, yaitu proses menyimpan salinan *file* sementara, CDN memungkinkan pengunjung mengakses konten web lebih cepat melalui server yang terdekat dengan mereka. CDN menyimpan berbagai jenis konten, seperti halaman web, gambar, dan video, di server *proxy* yang berada dekat dengan lokasi pengguna. Hal ini memungkinkan pengunjung melakukan aktivitas seperti menonton film, mengunduh perangkat lunak, memeriksa saldo bank, posting di media sosial, atau melakukan pembelian tanpa harus menunggu lama untuk memuat konten. Selain meningkatkan kecepatan akses, CDN juga memberikan perlindungan tambahan terhadap ancaman keamanan, termasuk serangan *Distributed Denial of Service (DDoS)*, sehingga meningkatkan keandalan dan keamanan situs web (Laksmiati, 2022).

Adapun cara kerja CDN seperti pada gambar.

CDN improves page load times



Gambar 2.3 Cara kerja CDN

Berdasarkan gambar, tanpa penggunaan CDN, pengguna harus mengakses langsung ke origin server yang jaraknya bisa sangat jauh, sehingga menyebabkan peningkatan *latency* dan waktu muat halaman menjadi lebih lambat. Dengan menggunakan CDN, *file-file* statis seperti HTML, CSS,

gambar, *JavaScript*, dan video disimpan di *edge* server yang lokasinya lebih dekat dengan pengguna. Ketika pengguna melakukan permintaan, konten akan dikirimkan dari *edge* server terdekat, sehingga jarak koneksi lebih pendek, *latency* berkurang, dan waktu muat halaman menjadi lebih cepat. Dengan demikian, implementasi CDN dapat secara signifikan meningkatkan kinerja *website* dan memberikan pengalaman pengguna yang lebih baik.

2.2.7 Cloudflare

Cloudflare adalah jaringan pengiriman konten (*Content Delivery Network*) yang berfungsi sebagai *proxy* untuk menghubungkan *website* dengan pengunjung. Dalam perannya, *Cloudflare* menjadi perantara antara *server website* dan pengguna, membantu mengoptimalkan performa dan keamanan. Banyak pengguna memilih *Cloudflare* karena kemampuannya dalam mempercepat waktu muat (*loading*) *website* sekaligus meningkatkan perlindungan terhadap berbagai ancaman keamanan (Laksmiati, 2022). *Cloudflare* menggunakan teknologi yang baik untuk memantau lalu lintas web dan mengidentifikasi potensi ancaman, sehingga dapat menghalangi serangan sebelum mencapai *website*. *Cloudflare* digunakan untuk melakukan implementasi *Content Delivery Network (CDN)* dan *Web Application Firewall (WAF)* (Alim et al., 2024).

2.2.8 Distributed Denial of Service (DDoS)

Distributed Denial of Service (DDoS) adalah bentuk serangan keamanan yang merupakan pengembangan dari *Denial of Service (DoS)*. Serangan ini menargetkan *server*, layanan, atau jaringan dengan mengirimkan sejumlah besar permintaan palsu secara bersamaan, sehingga menciptakan gangguan lalu lintas dan memberikan beban berlebih pada *server*. Akibatnya, kinerja sistem menjadi lambat, bahkan dapat menyebabkan kegagalan total pada *server* (Bhayangkara & Ashari, 2024). Serangan *DDoS* juga dapat membuat layanan

menjadi tidak tersedia bagi pengguna yang sah, menghalangi akses klien atau pelanggan ke sistem yang seharusnya berfungsi dengan baik. Serangan ini memberikan tantangan serius bagi keamanan jaringan karena menghambat ketersediaan layanan bagi pengguna legal (Ardiansyah et al., 2023).

2.2.9 SQL Injection

SQL *Injection* adalah teknik serangan keamanan yang memanfaatkan celah dalam kode aplikasi dengan menyisipkan perintah SQL ke dalam kueri yang dikirimkan melalui formulir web, nama domain, atau permintaan halaman. Serangan ini terjadi ketika input pengguna yang tidak divalidasi digunakan untuk membangun pernyataan SQL dinamis yang terhubung ke *database*. Selain itu, SQL *Injection* juga dapat terjadi pada penggunaan *stored procedure* jika input yang diteruskan berupa *string* tanpa penyaringan yang memadai (Haikal Muhammad et al., 2023). Teknik ini biasanya muncul karena kelemahan dalam kode program serta kurangnya proteksi keamanan pada aplikasi yang dikelola oleh administrator. Penyerang hanya membutuhkan *port* 80 untuk melakukan serangan ini, bahkan pada server yang sudah sering dilakukan *patching*. Serangan ini dilakukan dengan menginjeksikan kode SQL ke dalam kueri melalui halaman web, memungkinkan penyerang mengakses halaman administrator tanpa perlu memindai *port* terbuka, tanpa terdeteksi oleh *firewall*, dan tanpa menggunakan alat bantu khusus. SQL *Injection* mulai menjadi perhatian publik sejak *database* SQL pertama kali terhubung dengan aplikasi web, yang diperkirakan terjadi sekitar tahun 1998. Sejak saat itu, para peneliti keamanan telah mengembangkan berbagai teknik untuk mengeksploitasi celah ini (Elu, 2013).

2.2.10 Cross-Site Scripting (XSS)

Cross-Site Scripting (XSS) adalah jenis eksploitasi keamanan di mana penyerang menyisipkan kode berbahaya ke dalam halaman HTML yang

ditampilkan kepada pengguna. Jika kode tersebut berhasil dijalankan, perilaku atau fungsi sistem dan situs web dapat diubah sepenuhnya, memungkinkan penyerang untuk mengakses data pengguna yang seharusnya bersifat rahasia dan tidak dapat diakses (Haikal Muhammad et al., 2023). Dalam serangan XSS, peretas biasanya menyisipkan *payload* berisi kode *JavaScript* ke situs web. Kode tersebut memungkinkan peretas mengakses korban dari jarak jauh dan mencuri informasi penting. Informasi yang diperoleh sering kali diarahkan ke *platform* seperti *XSS Hunter*. Setelah korban terjebak, *payload* dapat membuat browser mereka tetap terhubung secara *online* dan menampilkan *pop-up*. Selanjutnya, korban sering kali diarahkan ke situs web *phishing* yang dirancang untuk mencuri data lebih lanjut (Risky & Yuhandri, 2021).

2.2.11 GTMetrix

GTmetrix adalah alat uji otomatis yang dikembangkan oleh perusahaan *Gossamer Threads*, yang berbasis di Amerika Serikat. Alat ini digunakan untuk menganalisis performa sebuah *website* guna menentukan apakah kinerjanya sudah optimal atau masih perlu perbaikan. Dalam pengujian, *GTmetrix* menampilkan *GTmetrix Grade*, yaitu skor yang menunjukkan kualitas performa website yang diuji. Hingga saat ini, *GTmetrix* telah menganalisis lebih dari 900 juta halaman *website* dan memiliki sekitar 1 juta pelanggan. Selain itu, perusahaan ini memiliki tenaga kerja, baik langsung maupun tidak langsung, yang tersebar di 82 negara. Audit performa website di *GTmetrix* dilakukan menggunakan Google Lighthouse, yang memberikan laporan performa serta rekomendasi mengenai kelebihan dan kekurangan website yang diuji (Nurul Hima Hidayati, 2022).

III. METODOLOGI PENELITIAN

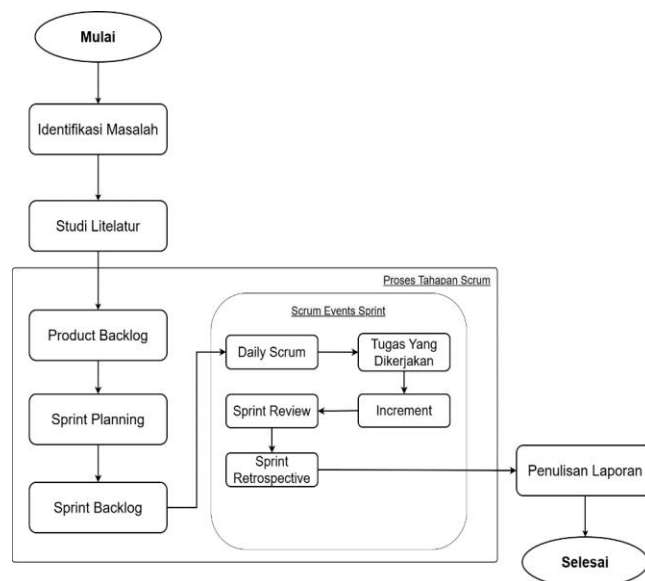
3.1 Waktu dan Tempat Penelitian

Waktu dan Tempat Penelitian adalah bagian yang menjelaskan kapan dan di mana suatu penelitian atau proyek dilakukan.

3.1.1 Waktu dan Tempat Penelitian

Pada penelitian ini, waktu pelaksanaan penelitian pada semester ganjil 2024/2025 yang dimulai dari bulan Oktober 2024 hingga bulan Juni 2025. Penelitian dilakukan di Universitas Lampung Fakultas Matematika dan Ilmu Pengetahuan Alam yang terletak di Jl. Prof. Dr. Ir. Sumantri Brojonegoro No.1, Gedong Meneng, Kec. Rajabasa, Kota Bandar Lampung, Lampung 35141.

3.2 Tahapan Penelitian



Gambar 3. 1 Tahapan Penelitian

3.2.1 Identifikasi Masalah

Identifikasi masalah adalah proses untuk menemukan dan memahami masalah utama yang sedang dihadapi. Dalam proses ini, kita mencari tahu apa yang tidak berjalan dengan baik, mengapa hal tersebut terjadi, dan bagaimana dampaknya. Tujuannya adalah untuk mengetahui masalah secara jelas agar bisa dirumuskan solusi yang tepat. Pada tahap Identifikasi Masalah, fokus utama adalah menemukan kendala atau kelemahan yang ada pada *website* FMIPA Universitas Lampung sehingga dapat dirumuskan solusi yang sesuai. Proses ini dimulai dengan analisis kondisi *website* saat ini, seperti meninjau kecepatan akses, celah keamanan, dan kemudahan pengelolaan. Dari hasil peninjauan, masalah yang sering ditemukan meliputi waktu muat halaman yang lambat, *website* yang rentan terhadap serangan seperti DDoS atau *hacking*.

Selanjutnya, data pendukung dikumpulkan melalui berbagai cara, seperti observasi langsung terhadap *website* yang ada saat ini, wawancara dengan pihak fakultas untuk mengumpulkan informasi mengenai kekurangan dan kebutuhan apa saja yang diperlukan untuk pengembangan *website*, dan pengujian teknis menggunakan alat seperti *Gtmetrix*, *Webpagetest*, dan *securityheaders* untuk mengevaluasi keamanan dan performa. Hasil dari pengumpulan data ini digunakan untuk merumuskan masalah utama secara spesifik, seperti kurangnya perlindungan terhadap serangan web dan kinerja *website* yang tidak optimal dalam memuat konten.

Berdasarkan hasil identifikasi, solusi yang relevan kemudian mulai dipertimbangkan. Beberapa solusi yang diusulkan antara lain adalah implementasi WAF untuk meningkatkan keamanan *website*, penggunaan CDN untuk mempercepat waktu akses dengan mendistribusikan konten ke server terdekat.

3.2.2 Studi Literatur

Studi literatur adalah proses mencari, membaca, dan menganalisis berbagai sumber informasi yang relevan untuk mendukung suatu penelitian atau proyek. Studi literatur ini bertujuan untuk memahami konsep-konsep yang akan digunakan, seperti WAF dan CDN. Jenis referensi yang dicari meliputi artikel, jurnal ilmiah, buku, laporan penelitian, serta dokumentasi teknis. Artikel dan jurnal ilmiah biasanya berisi penelitian terkait keamanan web dan optimasi performa menggunakan teknologi WAF dan CDN, sedangkan buku memberikan pemahaman mendalam mengenai teori keamanan jaringan dan pengembangan web.

Sumber-sumber tersebut diperoleh melalui berbagai platform. Untuk jurnal ilmiah, akses dilakukan melalui *database* seperti *Google Scholar* dan *ResearchGate*. Proses pencarian juga melibatkan penggunaan kata kunci seperti "Implementasi *Web Application Firewall* (WAF)", "Manfaat CDN untuk kecepatan *website*" dan "Pengembangan *website* menggunakan *Cloudflare*". Selain itu, peneliti juga melakukan kunjungan ke *website* serupa dari fakultas MIPA di berbagai universitas, seperti *website* FMIPA Universitas Indonesia, *Universiti Malaya*, FMIPA Universitas Negeri Semarang, dan FMIPA Universitas Padjadjaran.

Hasil dari studi literatur ini adalah pemahaman mendalam mengenai konsep dan teknologi yang akan digunakan, praktik terbaik dari kasus serupa, dukungan ilmiah untuk memperkuat penelitian, dan panduan implementasi yang jelas. Tahap ini menjadi dasar penting untuk memastikan setiap langkah dalam penelitian dapat dilakukan dengan efektif dan terarah.

3.2.3 Penerapan *Framework Scrum*

Untuk mengembangkan *website* dengan menerapkan *framework Scrum*, langkah-langkah yang dilakukan adalah sebagai berikut.

a. Peran dan Pengerjaan

Dalam pengembangan *website* menggunakan *framework* Scrum, langkah pertama yang harus dilakukan adalah menetapkan pembagian peran utama. Peran tersebut meliputi *Product Owner*, *Scrum Master*, dan *Scrum Team*. Pembagian peran dapat dilihat pada tabel 2 berikut.

Tabel 2. Pembagian Peran

Peran	Nama Lengkap
<i>Product Owner</i>	• Mulyono, S.Si., M.Si., Ph.D. • Dr. Aristoteles, S.Si., M.Si.
<i>Scrum Master</i>	• Muhaqiqin, S.Kom., M.T.I.
<i>Scrum Team</i>	• Faiz Muzaki • Gilang Ramadhan • Ikhsan Saputra • Waliid Ilham Ramadhan

b. User Stories

User story adalah deskripsi singkat tentang kebutuhan atau keinginan pengguna terhadap sebuah sistem, yang ditulis dari sudut pandang pengguna. Biasanya, *user story* membantu tim pengembang memahami fitur apa yang harus dibuat dan bagaimana fitur tersebut akan digunakan. *User story* yang di dapat dari pengumpulan data melalui wawancara dan observasi terhadap pengguna sistem dapat dilihat pada tabel berikut.

Tabel 3. *User Story*

No	Pengguna	<i>User Story</i>
1.	Mahasiswa	Sebagai mahasiswa FMIPA Universitas Lampung, saya ingin bisa mengakses semua aktivitas fakultas dalam satu <i>website</i> agar memudahkan saya dalam mendapatkan

No	Pengguna	User Story
		informasi.
2.	Mahasiswa	Sebagai mahasiswa, saya ingin pengalaman pengguna (UX) yang lebih baik saat mengakses <i>website</i> .
3.	Mahasiswa	Sebagai mahasiswa, saya ingin berita di <i>website</i> selalu diperbarui agar mendapatkan informasi terkini.
4.	Admin Berita	Sebagai admin berita Fakultas, saya ingin tampilan untuk mengelola berita di fakultas yang dinamis dan mudah diakses.
5.	Admin Fakultas	Sebagai admin <i>website</i> Fakultas, saya ingin bisa melihat berapa lama orang membuka <i>website</i> dan jumlah akses agar dapat mengevaluasi performa <i>website</i> .
6.	Pihak Fakultas	Sebagai pihak fakultas, saya ingin <i>website</i> yang memiliki pilihan tampilan utama yang dapat diusulkan dan disesuaikan dengan identitas fakultas agar <i>website</i> tersebut merepresentasikan FMIPA Universitas Lampung dengan baik. Dan saya ingin <i>website</i> ini dilindungi dari serangan peretasan, seperti pemasangan iklan judi slot, agar pengalaman pengguna tetap aman dan terjamin.
7.	Pengunjung Website	Sebagai pengunjung <i>website</i> , saya ingin <i>website</i> yang cepat diakses, sehingga saya tidak mengalami waktu tunggu yang lama. Dan memiliki desain yang elegan dan modern, serta memuat video pendek yang otomatis diputar agar pengalaman menjelajah menjadi menyenangkan.

Berdasarkan *user story* yang diperoleh dari beberapa calon pengguna *website*, dapat disimpulkan bahwa penelitian ini memerlukan alat penelitian dengan spesifikasi sebagai berikut.

Tabel 4. Daftar Perangkat Keras

Perangkat Keras	Fungsi
LAPTOP-FLJG8B91 10 th Gen Intel(R) Core(TM) i3-1005G1 CPU @ 1.20Ghz	Membuat seluruh rangkaian mulai dari <i>coding</i> sampai implementasi sistem keamanan di <i>website</i> FMIPA UNILA.
Mouse	Mempermudah pergerakan kursor

Tabel 5. Daftar Perangkat Lunak

Perangkat Lunak	Fungsi
Sistem Operasi Windows 10 Pro 64 bit	Sistem pengoperasian dalam mengelola <i>file</i> dan aplikasi yang sedang berjalan
Visual Studio Code versi 1.75.1	<i>Tools text</i> editor untuk menulis kode program dalam proses pengembangan <i>website</i> FMIPA
PHP versi 8.1.10 dan <i>framework Laravel 10</i>	Sebagai bahasa pemrograman dalam pengembangan situs web.
<i>Laragon Full</i> 6.0 220916	<i>Tools</i> untuk server lokal dalam mengembangkan situs web.
Draw.io.	<i>Tools</i> untuk membuat

Perangkat Lunak	Fungsi
	diagram, flowchart, dan mind map.
Trello	<i>Tools</i> untuk mengorganisir tugas dan proyek.
Figma	<i>Tools</i> untuk merancang antarmuka pengguna (UI) dan pengalaman pengguna (UX)
GTMetrix	<i>Tools</i> untuk mengukur kecepatan <i>website</i> .
Cloudflare	Layanan berbasis <i>cloud</i> yang membantu meningkatkan kinerja dan keamanan situs web
Webpagetest	<i>Tools</i> menguji performa <i>website</i> dengan berbagai lokasi dan kecepatan koneksi dan keamanan <i>website</i> .
securityheaders	<i>Tools</i> yang digunakan untuk menganalisis dan menguji pengaturan header keamanan pada sebuah <i>website</i> .
Burpsuite	<i>Tools</i> untuk memeriksa kerentanan keamanan
Web Browser Google Chrome versi 110.0.5481.177	<i>Tools</i> untuk browser dalam pengujian situs web

c. *Product Backlog*

Setelah memperoleh user story, langkah berikutnya adalah menyusun product backlog, yaitu daftar kebutuhan fungsional dan non-fungsional untuk website FMIPA Universitas Lampung. Proses ini melibatkan wawancara serta diskusi dengan pihak fakultas guna mengumpulkan data dan mengidentifikasi fitur yang akan diterapkan. Setelah kebutuhan teridentifikasi, fitur-fitur tersebut disusun berdasarkan tingkat prioritasnya. Selain itu, dokumentasi yang rinci juga diperlukan agar setiap fitur memiliki deskripsi yang jelas dan dapat dipahami oleh seluruh tim pengembang.

Tabel 6. *Product Backlog Items*

No	Nama Backlog	Detail Tugas	Prioritas	Definition of Done
1.	<i>Setup repository dan workflow Git</i>	<i>Setup repository Git dan CI/CD workflow.</i>	Tinggi	- Repository <i>Git</i> aktif dengan <i>branch</i> yang sesuai (development). - CI/CD pipeline (GitHub Actions) terintegrasi untuk otomatisasi build, test, dan deploy. - Menambahkan semua anggota tim ke dalam repository <i>Git</i>.
2.	<i>Setup database</i>	Konfigurasi <i>database</i> awal dan struktur dasar.	Tinggi	<i>Database</i> telah dibuat. - Skema tabel sesuai kebutuhan.

No	Nama Backlog	Detail Tugas	Prioritas	Definition of Done
				• <i>Database</i> dapat diakses tanpa <i>error</i>.
3.	Penyusunan <i>test plan</i>	Menyusun rencana pengujian awal.	Tinggi	• Pengujian keamanan menggunakan Security Headers • Pengujian kecepatan menggunakan GTmetrix. • Pengujian kecepatan menggunakan WebPageTest. • Dokumentasi dari hasil pengujian website FMIPA.
4.	Desain <i>wireframe</i> awal bagian <i>Login</i> admin	Membuat desain <i>wireframe</i> untuk <i>login</i> dan <i>dashboard</i> admin	Tinggi	• Tahapan <i>emphatize</i> dengan melakukan wawancara dengan <i>stakeholder</i> dan melakukan pengumpulan data kebutuhan pengguna dengan kuisisioner. • Membuat <i>User</i> Persona admin. • Melakukan tahapan <i>define</i> dengan dengan

No	Nama Backlog	Detail Tugas	Prioritas	Definition of Done
				membuat <i>table Point Of View</i> dan <i>table How Might We</i>. • Tahapan <i>Ideate</i> dengan membuat <i>wireframe login</i> dan <i>dashboard admin</i> berdasarkan ide ide dari tahapan <i>define</i>..
5.	Implementasi Autentikasi User	Implementasi <i>login</i> dan <i>middleware</i> autentikasi.	Tinggi	• <i>Login & register</i> berfungsi. • <i>Middleware</i> diterapkan. • <i>User</i> dapat <i>logout</i>. • Validasi <i>input</i> sudah diterapkan.
6.	Dokumentasi Sistem Autentikasi	Menyusun dokumentasi sistem login.	Tinggi	• Dokumentasi mencakup alur kerja autentikasi, mekanisme keamanan dan testing hasil implementasi autentikasi. • Tersedia didalam <i>repository</i> dalam <i>file</i> dengan format <i>.txt</i>.

No	Nama Backlog	Detail Tugas	Prioritas	Definition of Done
7.	Penyusunan skenario pengujian	Menyiapkan <i>test case</i> untuk autentikasi dan <i>database</i> .	Tinggi	• Pengujian kondisi sukses dan gagal pada proses <i>login</i>. • Penanganan <i>error</i> jika terjadi <i>input</i> yang tidak valid. • Pengujian relasi antar tabel di <i>database</i>. • Mengidentifikasi celah keamanan dan <i>bug</i>.
8.	Implementasi UI <i>login</i> dan dashboard	Mengembangkan tampilan <i>login</i> dan dashboard admin.	Tinggi	• Implementasi Wireframe login dan dashboard admin menggunakan Figma. • Konten pada halaman dashboard sesuai dengan keinginan Stakeholder dan admin. • Tampilan mudah dipahami. • Tampilan UI login berisikan kolom Username dan Password.

No	Nama Backlog	Detail Tugas	Prioritas	<i>Definition of Done</i>
				• Tampilan Dashboard Admin berisikan 6 menu yang berada di sebelah kiri. • pada masing masing halaman berisikan kolom-kolom untuk mengisi konten yang akan ditampilkan pada halaman guest.
9.	Pengujian autentikasi	Menguji <i>login</i> , dan <i>relasi database</i> .	Tinggi	• menguji secara langsung fungsi login. • menguji middleware autentikasi. • menguji struktur serta relasi database
10.	Laporan Analitik	Membuat dashboard analitik statistik pengunjung (harian/bulanan).	Tinggi	• Membuat grafik interaksi (<i>bar chart</i>) untuk melihat jumlah pengunjung <i>website</i> harian. • Membuat tren pengunjung bulanan dalam bentuk <i>line chart</i>.
11.	Pengujian UI	Pengujian UI	Tinggi	• Halaman <i>login</i> dan <i>dashboard</i> admin

No	Nama Backlog	Detail Tugas	Prioritas	Definition of Done
	Berdasarkan <i>Feedback</i>	berdasarkan hasil pengujian awal.		sudah sesuai dengan keinginan <i>Stakeholder</i>. • Halaman <i>dashboard</i> dan <i>login</i> admin telah diuji oleh Admin Fakultas. • UI sudah sesuai dan mudah di pahami oleh admin dan tidak ada perbaikan yang perlu dilakukan.
12.	Pengelolaan Banner	Membuat CRUD banner untuk ditampilkan pada <i>website</i> .	Tinggi	• CRUD banner berfungsi dengan baik. • Banner muncul dengan baik di <i>landing page</i>.
13.	Optimalisasi <i>Database</i>	Optimasi indeks dan query <i>database</i>	Tinggi	• Query lebih cepat dibanding sebelum optimasi. • <i>Indexing</i> diterapkan di tabel <i>database</i> yang diperlukan.

No	Nama Backlog	Detail Tugas	Prioritas	<i>Definition of Done</i>
				Sistem lebih efisien dalam penggunaan sumber daya.
14.	<i>Role-based access control</i>	Implementasi <i>role</i> admin, editor, dan user biasa.	Tinggi	- Hak akses sesuai role (admin: full <i>access</i>, editor: edit/hapus berita, user: baca). <i>Middleware</i> membatasi akses ke <i>endpoint</i> sesuai <i>role</i>.
15.	Pengelolaan Berita	Membuat CRUD untuk menambah, mengedit, menghapus berita.	Tinggi	- Admin dapat menambahkan, mengedit, dan menghapus berita. - Validasi input diterapkan untuk memastikan data berita sesuai standar. - Tidak ada error pada proses CRUD berita. - Pengguna dapat mengelola berita dengan editor <i>What You See Is What You Get</i>.

No	Nama Backlog	Detail Tugas	Prioritas	Definition of Done
16.	Pengelolaan Daftar Pimpinan	Membuat fitur kelola untuk manajemen pimpinan.	Tinggi	• Admin dapat <i>mengupdate</i> pimpinan dengan <i>field</i> wajib: nama, jabatan, nip dan foto baru. • Data tersimpan di <i>database</i>. • Admin dapat melihat daftar pimpinan dalam tabel dengan kolom: nama, jabatan dan aksi.
17.	Pengujian Fitur Berita	Menguji fitur pengelolaan berita untuk memastikan validasi data.	Tinggi	• pengecekan validasi <i>input</i>. • pengujian konsistensi data. • verifikasi bahwa setiap aksi (tambah, <i>edit</i>, hapus) berfungsi sesuai dengan spesifikasi.
18.	Implementasi Fitur Generasi Konten	Mengembangkan fitur AI yang menghasilkan teks berita	Tinggi	• API ChatGPT telah terintegrasi dengan sistem dan dapat menghasilkan berita

No	Nama Backlog	Detail Tugas	Prioritas	Definition of Done
		berdasarkan parameter 5W+1H dan kategori.		sesuai format jurnalistik. • Parameter 5W+1H diterapkan untuk memastikan kelengkapan informasi. • Admin dapat melakukan editing setelah berita otomatis dihasilkan. • Dokumentasi penggunaan fitur telah dibuat untuk admin.
19.	Desain Wireframe Awal Bagian Pengguna	Membuat desain wireframe halaman beranda dan halaman - halaman profile bagian pengguna.	Tinggi	• Tahapan <i>emphasize</i> dengan melakukan wawancara dengan <i>stakeholder</i> dan melakukan pengumpulan data kebutuhan pengguna dengan <i>kuisisioner</i>. • Membuat <i>User Persona</i> pengguna. • Melakukan tahapan <i>define</i> dengan

No	Nama Backlog	Detail Tugas	Prioritas	Definition of Done
				membuat <i>table Point Of View</i> dan <i>table How Might We</i>. • Tahapan <i>Ideate</i> dengan membuat <i>wireframe login</i> dan <i>dashboard</i> admin berdasarkan ide - ide dari tahapan <i>define</i>.
20.	Pengelolaan Kategori Berita	Membuat fitur untuk menambah, mengedit, menghapus kategori berita.	Tinggi	• CRUD kategori berita berfungsi dengan validasi. • Validasi kategori unik (tidak duplikat). • Kategori terhubung ke tabel <i>database</i>.
21.	Manajemen Profil Admin	Membuat Fitur manajemen admin.	Tinggi	• Admin dapat memperbarui informasi profil. • Validasi input diterapkan. • Password dapat diperbarui dengan enkripsi yang aman. • Gambar profil dapat diunggah dan

No	Nama Backlog	Detail Tugas	Prioritas	<i>Definition of Done</i>
				diperbarui dengan validasi ukuran dan format file. • Perubahan profil langsung tersimpan dan ditampilkan tanpa perlu refresh manual. • Hanya admin yang bersangkutan atau superadmin yang dapat mengubah profil. • Tidak ada error atau bug saat melakukan pembaruan profil. • Pengujian telah dilakukan untuk memastikan fitur bekerja dengan baik. • Perubahan telah dicommit dan terdokumentasi dalam repository.
22.	Pengujian Kategori Berita	Menguji fitur	Tinggi	• kategori dapat ditambah, diedit, atau

No	Nama Backlog	Detail Tugas	Prioritas	Definition of Done
		kategori berita		dihapus dengan lancar.
23.	Kelola Profil Departemen	Membuat fitur CRUD (Create, Read, Update, Delete) untuk pengelolaan data departemen.	Tinggi	• Tampilan pada dashboard admin sudah terintegrasi berupa tabel yang menampilkan kolom: Title, Gambar, Departemen, dan kolom aksi (untuk edit dan hapus). • Data dapat dimodifikasi melalui form CRUD tanpa error. • Data tersimpan di database dengan validasi yang tepat.
24.	Pengelolaan Daftar Senat Fakultas	Mengembangkan fitur bagi admin untuk menambahkan dan mengelola data senat fakultas.	Sedang	• Admin dapat menambahkan data senat yang akan ditampilkan pada <i>website</i>. • Tabel tampilan senat terdiri dari kolom: No, Nama, NIP, Email, Telepon, dan Jabatan.

No	Nama Backlog	Detail Tugas	Prioritas	Definition of Done
				• Setiap data senat dapat <i>diinput</i>, <i>diedit</i>, dan dihapus dengan tampilan yang konsisten. • Validasi data berjalan baik.
25.	Manajemen Sejarah Untuk Fakultas dan Jurusan	Membuat fitur CRUD untuk penambahan, pengeditan, dan penghapusan konten sejarah pada fakultas dan jurusan.	Sedang	• Tersedianya tampilan tabel pada <i>dashboard</i> admin yang terdiri dari kolom: Departemen, Sejarah, dan Aksi. • Data sejarah dapat di <i>input</i> dan dimodifikasi melalui antarmuka CRUD. • Validasi berjalan dengan baik tanpa <i>error</i>. • Tampil secara konsisten di <i>website</i>.
26.	Manajemen Visi, Misi, dan Tujuan untuk	Mengembangkan fitur CRUD bagi admin untuk	Sedang	• Tampilan <i>dashboard</i> admin menampilkan tabel dengan kolom: Departemen, Visi,

No	Nama Backlog	Detail Tugas	Prioritas	Definition of Done
	Fakultas dan Jurusan	mengelola konten visi, misi, dan tujuan pada fakultas dan masing-masing jurusan.		Misi, Tujuan, dan Aksi. • Fitur CRUD berjalan lancar. • Validasi <i>input</i> dan tampilan yang responsif. • Integrasi data yang terjamin di <i>website</i>.
27.	Fitur Multibahasa	Menambahk an dukungan multibahasa ke seluruh konten (statis dan dinamis) sehingga pengguna bisa memilih antara bahasa Indonesia dan Inggris.	Sedang	• Tersedia <i>icon</i> bendera Indonesia dan Inggris untuk pilihan bahasa di tampilan pengguna. • Seluruh konten sudah diterjemahkan dan ditampilkan sesuai pilihan bahasa tanpa mengganggu <i>layout</i>.
28.	Optimasi SEO	Implementa si optimasi	Sedang	• <i>Website</i> mudah terindeks oleh mesin

No	Nama Backlog	Detail Tugas	Prioritas	Definition of Done
	(Search Engine Optimizati on)	SEO onpage dan off-page		pencari (<i>search engine</i>). • Telah memenuhi standar SEO dan struktur URL. • <i>Markup</i> telah diterapkan secara konsisten pada seluruh halaman
29.	Implementasi UI Halaman Landing Page Dan Halaman - Halaman Profil Bagian Pengguna	Mendesain UI halaman <i>landing page</i> dan halaman-halaman profil bagian pengguna	Sedang	• Implementasi <i>Wireframe login</i> dan <i>dashboard</i> admin menggunakan Figma. • Konten pada halaman <i>dashboard</i> sesuai dengan keinginan <i>Stakeholder</i> dan admin. • Tampilan mudah dipahami. • Tampilan UI <i>login</i> berisikan kolom <i>Username</i> dan <i>Password</i>. • Tampilan <i>Dashboard</i> Admin berisikan 6

No	Nama Backlog	Detail Tugas	Prioritas	Definition of Done
				menu yang berada di sebelah kiri. • pada masing - masing halaman berisikan kolom-kolom untuk mengisi konten yang akan ditampilkan pada halaman <i>guest</i>.
30.	Implementasi Dashboard <i>Business Intelligence</i>	Mengembangkan modul analisis data pengguna berdasarkan interaksi berita (jumlah klik, waktu baca, komentar).	Sedang	• <i>Dashboard</i> analitik telah menampilkan data kunjungan, durasi baca, dan <i>engagement</i>. • Data dapat di filter berdasarkan rentang waktu dan kategori berita. • Sistem telah diuji dengan data <i>dummy</i> dan data produksi.
31.	Integrasi API Whatsapp	Integrasi API WhatsApp untuk sistem notifikasi.	Sedang	• Terhubung ke API WhatsApp <i>Business</i> (WhatsApp Cloud API) dengan token autentikasi valid • Template pesan mencantumkan placeholder [Judul

No	Nama Backlog	Detail Tugas	Prioritas	Definition of Done
				Berita], [Link Berita], dan [Tanggal] yang otomatis terisi. • Template disetujui oleh <i>Product Owner</i>.
32.	Pengujian UI Halaman Beranda Dan Halaman - Halaman Profile Bagian Pengguna	Pengujian UI berdasarkan hasil pengujian awal.	Sedang	• Uji responsif di beberapa <i>device</i>. • Konfirmasi <i>stakeholder</i> terkait hasil pengujian UI yang dibuat. • Memperbaiki <i>bug</i> yang ada pada <i>layout</i>.
33.	Implementasi UI Halaman Jurusan	Mendesain dan mengembangkan tampilan halaman jurusan.	Sedang	• Daftar jurusan lengkap dengan deskripsi. • Tautan ke berita terkait jurusan. • Memastikan responsif untuk semua <i>device</i>.
34.	Tampilan Detail Berita	Mendesain halaman detail berita	Sedang	• Menampilkan gambar, penulis, & tanggal publikasi.

No	Nama Backlog	Detail Tugas	Prioritas	Definition of Done
		dengan komentar.		• Related news berdasarkan kategori.
35.	Pengelolaan Manajemen Kepegawaian	Membuat fitur untuk menambah, mengedit, menghapus kepegawaian dari fakultas dan semua jurusan.	Sedang	• Admin dapat menambah data kepegawaian dengan <i>field</i> wajib: nama, jabatan, NIP, foto, dan unit kerja (fakultas/jurusan). • Admin dapat mengedit dan menghapus data kepegawaian tanpa <i>error</i>. • Validasi <i>input</i> diterapkan untuk memastikan data sesuai standar. • Data tersimpan di <i>database</i> dengan struktur tabel yang sesuai. • Daftar kepegawaian ditampilkan dalam tabel dengan kolom: nama, jabatan, unit

No	Nama Backlog	Detail Tugas	Prioritas	<i>Definition of Done</i>
				kerja, dan aksi (edit/hapus).
36.	Notifikasi Berita Baru	Implementasi fitur notifikasi berita untuk user dengan API WhatsApp	Sedang	• Log notifikasi (nomor tujuan, status pengiriman, <i>timestamp</i>) tersimpan di tabel. • Admin dapat melihat riwayat notifikasi di <i>dashboard</i> dengan filter tanggal/nomor. • Berhasil mengirim notifikasi ke 50+ user secara simultan. • Waktu pengiriman rata-rata
37.	Pencarian Berita	Implementasi pencarian berita berdasarkan judul & kategori	Sedang	• Pencarian bekerja dengan akurat. • Hasil pencarian ditampilkan dengan <i>sorting</i> relevan. • Tidak ada <i>error</i> dalam proses pencarian.

No	Nama Backlog	Detail Tugas	Prioritas	<i>Definition of Done</i>
38.	Optimasi Tampilan Mobile	Memastikan UI responsif di semua <i>device</i> .	Sedang	• Diuji di 5 ukuran layar yang berbeda. • Navigasi mudah di mobile.
39.	Pengujian Notifikasi Berita	Menguji sistem notifikasi berita yang sudah terintegrasi dengan API WhatsApp	Rendah	• Kirim notifikasi ke 10+ user secara manual. • Cek log status pengiriman. • Validasi waktu rata-rata
40.	UI Pencarian Berita	Mendesain tampilan pencarian berita.	Rendah	• Input field dengan placeholder "Cari berita..." • Hasil pencarian dalam <i>grid/card</i>. • <i>Pagination</i> 10 item/halaman.
41.	Keamanan Sistem	Menambahk an proteksi WAF keamanan pada sistem.	Rendah	• Proteksi keamanan berhasil diimplementasikan. • Proteksi WAF berjalan dengan baik

No	Nama Backlog	Detail Tugas	Prioritas	<i>Definition of Done</i>
42.	Kecepatan Sistem	Menambahkan proteksi CDN kecepatan pada sistem.	Rendah	• Proteksi Kecepatan berhasil diimplementasikan. • Proteksi CDN berjalan dengan baik.
43.	Pengujian Keamanan Sistem	keamanan testing untuk memastikan sistem aman saat sudah diimplementasikan proteksi keamanan.	Rendah	• WAF dapat menangani serangan seperti terhadap SQL injection, XSS, dan Ddos
44.	Pengujian Performa Sistem	<i>Load testing</i> untuk memastikan kestabilan sistem	Rendah	• <i>Load</i> sistem pada setiap halaman menjadi cepat. • Sistem berjalan dengan cepat.
45.	Penyempurnaan UI	Menyempurnakan tampilan UI berdasarkan <i>feedback testing</i> .	Rendah	• Implementasi <i>feedback</i> pengguna. • Peningkatan skor aksesibilitas (>90). • Dokumentasi perubahan

No	Nama Backlog	Detail Tugas	Prioritas	<i>Definition of Done</i>
46.	Integrasi <i>Frontend - Backend</i>	Menghubungkan fitur backend dengan tampilan frontend.	Rendah	• API terhubung ke semua <i>endpoint</i> CRUD. • Data tampil konsisten di UI. • <i>Error handling</i> untuk koneksi gagal.
47.	Pengujian End-To End.	Melakukan pengujian <i>end-to-end</i> pada sistem.	Rendah	• Uji alur lengkap (<i>login</i> → buat berita → <i>publish</i> → notifikasi). • Membuat dokumentasi bug & <i>fix critical issues</i>.
48.	Penyempurnaan Keamanan Sistem.	Menyempurnakan keamanan sistem dengan validasi tambahan.	Rendah	• Validasi tambahan untuk <i>input</i> sensitif. • Audit keamanan oleh tim.
49.	Integrasi API SINTA Untuk Scraping Data	Setup koneksi ke API SINTA dan implementasi mekanisme	Rendah	• API berhasil diintegrasikan ke sistem. • Data berhasil diambil dan ditampilkan dalam sistem.

No	Nama Backlog	Detail Tugas	Prioritas	Definition of Done
	Penelitian Dosen	scraping data penelitian dosen		• Tidak ada <i>error</i> pada proses pengambilan data API.
50.	Pengujian API SINTA	Melakukan pengujian pada mekanisme <i>scraping</i> data penelitian dosen melalui API SINTA.	Rendah	• data yang ditarik akurat. • koneksi dengan API stabil. • data yang diterima sesuai dengan format yang diharapkan.
51.	Dokumentasi Sistem	Menyusun dokumentasi sistem dan panduan penggunaan admin & <i>user</i> .	Rendah	• Membuat panduan yang jelas untuk admin & user dalam format .txt. • Update file README.md.
52.	Pengujian Sistem Produksi	Menguji aplikasi dalam lingkungan produksi, Deploy	Rendah	• Deploy ke server staging. • Uji load dengan 100+ user simultan. • Monitor error rate <1%.

No	Nama Backlog	Detail Tugas	Prioritas	<i>Definition of Done</i>
		staging & load testing		
53.	Finalisasi UI Dan UX	Finalisasi tampilan dan interaksi pengguna.	Rendah	• Persetujuan desain dari <i>stakeholder</i>. • <i>Update style guide</i>. • Dokumentasi perubahan terakhir
54.	<i>Review</i> Sistem Oleh Tim	<i>Review</i> dan finalisasi sistem sebelum <i>go-live</i>	Rendah	• Dokumen review tersedia.

d. *Sprint Planning*

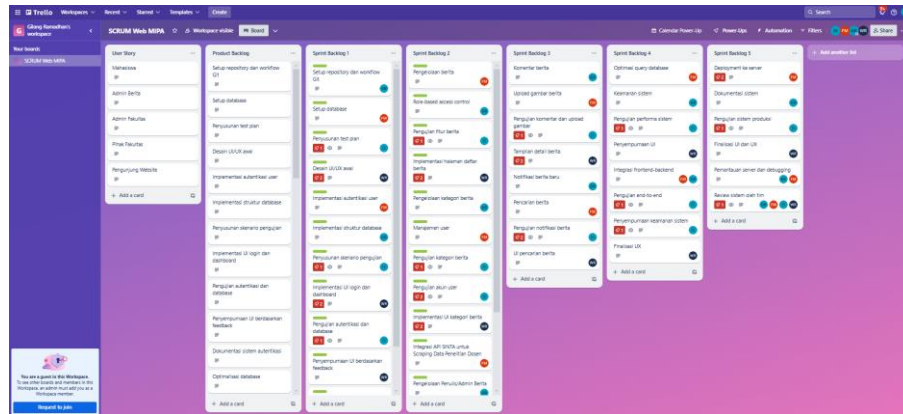
Setelah *product backlog* tersusun, tahap berikutnya dalam pengembangan *website* dengan *framework* Scrum adalah melakukan *sprint planning*. *Sprint planning* adalah proses pertemuan di mana seluruh tim, bersama dengan *product owner*, menetapkan tujuan sprint dan memilih *backlog items* yang akan dikerjakan berdasarkan prioritas, estimasi waktu, dan ketergantungan antar tugas. Pada Tahapan ini, tim Scrum mengadakan pertemuan untuk merinci pekerjaan yang akan dikerjakan selama satu sprint, dengan dasar acuan *product backlog* yang sudah tersedia. Setiap anggota tim dapat memilih tugas-tugas yang telah disusun di *platform* Trello, di mana seluruh *backlog* item ditampilkan dalam bentuk kartu (*card*) dengan deskripsi, detail tugas, dan tingkat prioritas (tinggi, sedang, rendah).

Pemilihan *product backlog items* yang akan dikerjakan dalam *sprint* dilakukan berdasarkan beberapa faktor yaitu tingkat prioritas yang telah ditetapkan oleh *product owner*, estimasi waktu pengerjaan untuk masing-masing fitur, serta kemampuan untuk mengerjakan beberapa tugas secara *fleksibel* tanpa menimbulkan ketergantungan yang kompleks antar anggota tim. Tahapan *sprint planning* diawali dengan penentuan *sprint goal*, yaitu tujuan spesifik yang ingin dicapai pada akhir *sprint*, sehingga seluruh tim memiliki acuan yang jelas mengenai target yang harus dicapai. Setelah tujuan *sprint* ditetapkan, tim memilih *product backlog items* dengan prioritas tinggi yang realistis untuk diselesaikan dalam batas waktu *sprint*. Setiap *product backlog items* yang terpilih kemudian dipecah menjadi *sprint backlog*, yaitu daftar tugas-tugas terperinci yang harus diselesaikan untuk mencapai item tersebut.

Dalam proses penyusunan *sprint backlog*, tim pengembang mempertimbangkan logika pengerjaan dan ketergantungan antar fitur serta membuat estimasi waktu pengerjaan setiap tugas berdasarkan diskusi bersama. Tugas-tugas tersebut kemudian dipublikasikan di *platform Trello* sehingga setiap anggota tim dapat memilih sesuai dengan keahlian masing-masing. Jika ditemukan tugas yang terlalu besar atau rumit, tugas tersebut dipecah lebih lanjut menjadi bagian-bagian yang lebih kecil untuk memudahkan pengelolaan dan pemantauan kemajuannya.

Setelah semua tugas dinyatakan selesai, langkah selanjutnya adalah melakukan rekap kembali di *platform trello* untuk menandai bahwa pengerjaan telah selesai. Pada tahap ini, status setiap tugas diubah menjadi "selesai" atau "*done*" sehingga seluruh tim dapat memantau progres secara *real-time*. Proses rekap ini tidak hanya memastikan bahwa setiap *backlog items* telah memenuhi standar dan persyaratan yang ditetapkan, tetapi juga menyediakan data evaluasi untuk membandingkan hasil yang dicapai dengan *sprint goal* yang telah ditetapkan sebelumnya. Dengan adanya pencatatan transparan di *trello*, tim dapat dengan mudah mengidentifikasi

pencapaian maupun fitur yang perlu perbaikan untuk sprint-sprint berikutnya, sehingga meningkatkan efisiensi dan efektivitas proses pengembangan secara keseluruhan.



Gambar 3. 2 Aktivitas di *platform* trello

Dapat dilihat pada gambar di atas merupakan hasil kerja tim pada *platform* trello yang di mana menunjukkan secara langsung bagaimana tim pengembang mengelola seluruh aktivitas dan daftar tugas, baik yang telah diselesaikan maupun yang masih dalam proses pengerjaan. Berdasarkan aktivitas yang tercatat di trello, diperoleh hasil berupa tabel sprint backlog berikut ini.

Tabel 7. *Sprint Backlog* 1 (Hari 1 - Hari 18)

Tujuan Sprint			
Melakukan perencanaan, setup infrastruktur, autentikasi dan desain awal.			
No	Nama Backlog	Detail Tugas	Pengerjaan (Hari)
1.	Setup repository dan workflow Git	Setup repository Git dan CI/CD workflow.	2

Tujuan Sprint		Melakukan perencanaan, setup infrastruktur, autentikasi dan desain awal.	
No	Nama Backlog	Detail Tugas	Pengerjaan (Hari)
2.	Setup database	Konfigurasi database awal dan struktur dasar.	3
3.	Penyusunan test plan	Menyusun rencana pengujian awal.	2
4.	Desain Wireframe Awal Bagian Login Admin	Membuat desain wireframe untuk login dan dashboard admin.	5
5.	Implementasi autentikasi user	Implementasi login dan middleware autentikasi.	3
6.	Dokumentasi Sistem Autentikasi	Menyusun dokumentasi sistem login.	3
7.	Penyusunan Skenario Pengujian	Menyiapkan test case untuk autentikasi dan database.	3
8.	Implementasi UI Login Dan Dashboard Bagian Admin	Mengembangkan tampilan login dan dashboard admin.	6
9.	Pengujian Autentikasi	Menguji login dan relasi database	2

Tujuan Sprint		Melakukan perencanaan, setup infrastruktur, autentikasi dan desain awal.	
No	Nama Backlog	Detail Tugas	Pengerjaan (Hari)
10.	Laporan Analitik	Membuat <i>dashboard</i> analitik statistik pengunjung (harian/bulanan).	3
11.	Pengujian UI Berdasarkan <i>Feedback</i>	Pengujian UI berdasarkan hasil pengujian awal.	3
12.	Pengelolaan Banner	Membuat fitur CRUD banner untuk ditampilkan pada <i>website</i> .	5

Sprint 1 difokuskan pada perencanaan dan penyiapan proyek, dengan tujuan akhir memastikan lingkungan kerja tim siap digunakan. Tahapan ini mencakup perancangan serta pembuatan *database*, implementasi sistem *autentikasi*, dan pengembangan desain awal untuk fitur utama *website*.

Tabel 8. *Sprint Backlog* 2 (Durasi : Hari 19 - Hari 36)

Tujuan Sprint		Implementasi fitur utama <i>website</i> , <i>database</i> dan membuat <i>role management</i> .	
No	Nama Backlog	Detail Tugas	Pengerjaan (Hari)
1.	Optimalisasi <i>Database</i>	Optimasi indeks dan <i>query database</i> .	3

Tujuan Sprint		Implementasi fitur utama <i>website</i> , <i>database</i> dan membuat <i>role management</i> .	
No	Nama Backlog	Detail Tugas	Pengerjaan (Hari)
2.	<i>Role-based access control</i>	Implementasi <i>role</i> admin, <i>editor</i> , dan user biasa.	4
3.	Pengelolaan Berita	Membuat CRUD untuk menambah, mengedit, menghapus berita.	4
4.	Pengelolaan Daftar Pimpinan	Membuat fitur kelola untuk manajemen pimpinan.	5
5.	Pengujian Fitur Berita	Menguji fitur pengelolaan berita untuk memastikan validasi data.	3
6.	Implementasi Fitur Generasi Konten	Mengembangkan fitur AI yang menghasilkan teks berita berdasarkan parameter 5W+1H dan kategori.	5
7.	Desain <i>Wireframe</i> Awal Bagian Pengguna	Membuat desain <i>wireframe</i> halaman beranda dan halaman-halaman <i>profile</i> bagian pengguna.	10

Tujuan Sprint		Implementasi fitur utama <i>website</i> , <i>database</i> dan membuat <i>role management</i> .	
No	Nama Backlog	Detail Tugas	Pengerjaan (Hari)
8.	Pengelolaan Kategori Berita	Membuat fitur untuk menambah, mengedit, menghapus kategori berita.	5
9.	Manajemen Profil Admin	Membuat Fitur manajemen admin.	5
10.	Pengujian Kategori Berita	Menguji fitur kategori berita.	2
11.	Kelola Profil Departemen	Membuat fitur CRUD (<i>Create, Read, Update, Delete</i>) untuk pengelolaan data departemen.	4

Sprint 2 difokuskan pada pengembangan fitur inti untuk halaman utama *website* serta *implementasi* fitur utama pada *backend* dan *frontend* secara bersamaan, memungkinkan proses pengembangan berjalan secara paralel tanpa ketergantungan yang menghambat satu sama lain.

Tabel 9. Sprint Backlog 3 (Durasi : Hari 37 - Hari 54)

Tujuan Sprint		Pengembangan UI/UX, integrasi sistem, dan manajemen konten.	
No	Nama Backlog	Detail Tugas	Pengerjaan (Hari)
1.	Pengelolaan Daftar Senat Fakultas	Mengembangkan fitur bagi admin untuk menambahkan dan mengelola data senat fakultas.	5
2.	Manajemen Sejarah Untuk Fakultas Dan Jurusan	Membuat fitur CRUD untuk penambahan, pengeditan, dan penghapusan konten sejarah pada fakultas dan jurusan.	5
3.	Manajemen Visi, Misi, Dan Tujuan Untuk Fakultas Dan Jurusan	Mengembangkan fitur CRUD bagi admin untuk mengelola konten visi, misi, dan tujuan pada fakultas dan masing-masing jurusan.	5
4.	Fitur Multibahasa	Menambahkan dukungan multibahasa ke seluruh konten (statis dan dinamis) sehingga pengguna bisa memilih antara bahasa Indonesia dan Inggris.	5

Tujuan Sprint		Pengembangan UI/UX, integrasi sistem, dan manajemen konten.	
No	Nama Backlog	Detail Tugas	Pengerjaan (Hari)
5.	Optimasi Seo (<i>Search Engine Optimization</i>)	Implementasi optimasi SEO <i>on-page</i> dan <i>off-page</i>	3
6.	Implementasi UI Halaman <i>Landing Page</i> Dan Halaman - Halaman Profil Bagian Pengguna	Mendesain UI halaman <i>landing page</i> dan halaman-halaman profil bagian pengguna.	10
7.	Implementasi <i>Dashboard Business Intelligence</i> .	Mengembangkan modul analisis data pengguna berdasarkan interaksi berita (jumlah klik, waktu baca, komentar).	5
8.	Integrasi API <i>WhatsApp</i>	Integrasi API <i>WhatsApp</i> untuk sistem notifikasi.	3
9.	Pengujian UI Halaman Beranda Dan Halaman -Halaman <i>Profile</i> Bagian Pengguna	Pengujian UI berdasarkan hasil pengujian awal.	5
10.	Implementasi UI Halaman Jurusan	Mendesain dan mengembangkan tampilan halaman jurusan.	8

Sprint 3 difokuskan pada penyempurnaan fitur manajemen berita, yang meliputi pengembangan fungsi CRUD (*Create, Read, Update, Delete*) berita melalui *dashboard* admin. Aktivitas dalam sprint ini mencakup perbaikan pada fitur berita, pengujian sistem, serta penyesuaian tampilan pada sisi *frontend*. Fokus utama dari sprint ini adalah untuk meningkatkan kualitas dan stabilitas fitur inti sebelum melanjutkan ke tahap pengembangan berikutnya.

Tabel 10. *Sprint Backlog 4* (Durasi : Hari 55 - Hari 72)

Tujuan Sprint		Integrasi sistem notifikasi, keamanan, dan optimasi.	
No	Nama Backlog	Detail Tugas	Pengerjaan (Hari)
1.	Tampilan Detail Berita	Mendesain halaman detail berita dengan komentar.	7
2.	Pengelolaan Manajemen Kepegawaian	Membuat fitur untuk menambah, mengedit, menghapus kepegawaian dari fakultas dan semua jurusan.	5
3.	Notifikasi Berita Baru	Implementasi fitur notifikasi berita untuk user dengan API WhatsApp.	4
4.	Pencarian Berita	Implementasi pencarian berita berdasarkan judul & kategori.	3
5.	Optimasi Tampilan Mobile	Memastikan UI responsif di semua device.	5

Tujuan Sprint		Integrasi sistem notifikasi, keamanan, dan optimasi.	
No	Nama Backlog	Detail Tugas	Pengerjaan (Hari)
6.	Pengujian Notifikasi Berita	Menguji sistem notifikasi berita yang sudah terintegrasi dengan API WhatsApp.	3
7.	UI Pencarian Berita	Mendesain tampilan pencarian berita.	5
8.	Keamanan Sistem	Menambahkan proteksi WAF keamanan pada sistem.	5
9.	Kecepatan Sistem	Menambahkan proteksi CDN kecepatan pada sistem.	5
10.	Pengujian Keamanan Sistem	keamanan testing untuk memastikan sistem aman saat sudah diimplementasikan proteksi keamanan.	3
11.	Pengujian Performa Sistem	<i>Load testing</i> untuk memastikan kestabilan sistem.	3

Sprint 4 berfokus pada pengembangan sistem notifikasi berbasis API WhatsApp, integrasi keamanan, serta optimalisasi kecepatan *website*. Pada

sprint ini, aktivitas yang dilakukan mencakup pembuatan form input data notifikasi, integrasi API WhatsApp, dan pengujian pengiriman pesan. Tujuan utama dari sprint ini adalah mengimplementasikan sistem notifikasi otomatis melalui WhatsApp untuk mendistribusikan informasi berita, meningkatkan performa dan keamanan *website*, serta mengoptimalkan tampilan antarmuka pengguna berdasarkan masukan dari sejumlah pengguna.

Tabel 11. *Sprint Backlog 5* (Durasi : Hari 73 - Hari 90)

Tujuan Sprint		Finalisasi, dokumentasi, dan persiapan <i>go-live</i> .	
No	Nama Backlog	Detail Tugas	Pengerjaan (Hari)
1.	Penyempurnaan UI	Menyempurnakan tampilan UI berdasarkan <i>feedback</i> testing.	7
2.	Integrasi Frontend-Backend	Menghubungkan fitur <i>backend</i> dengan tampilan <i>frontend</i> .	5
3.	Pengujian <i>End-To-End</i>	Melakukan pengujian <i>end-to-end</i> pada sistem.	3
4.	Penyempurnaan Keamanan Sistem	Menyempurnakan keamanan sistem dengan validasi tambahan	5

Tujuan Sprint		Finalisasi, dokumentasi, dan persiapan <i>go-live</i> .	
No	Nama Backlog	Detail Tugas	Pengerjaan (Hari)
5.	Integrasi API SINTA Untuk <i>Scraping</i> Data Penelitian Dosen	<i>Setup</i> koneksi ke API SINTA dan implementasi mekanisme <i>scraping</i> data penelitian dosen.	5
6.	Pengujian API SINTA	Melakukan pengujian pada mekanisme <i>scraping</i> data penelitian dosen melalui API SINTA.	2
7.	Dokumentasi Sistem	Menyusun dokumentasi sistem dan panduan penggunaan admin & <i>user</i> .	3
8.	Pengujian Sistem Produksi	Menguji aplikasi dalam lingkungan produksi, <i>deploy staging & load testing</i> .	7
9.	Finalisasi UI Dan UX	Finalisasi tampilan dan interaksi pengguna.	10
10.	<i>Review</i> Sistem Oleh Tim	<i>Review</i> dan finalisasi sistem sebelum <i>go-live</i> .	7

Sprint 5 berfokus pada tahap *finalisasi* dan *deployment*, di mana tim akan menyelesaikan penyempurnaan akhir sistem serta memastikan seluruh fitur berfungsi dengan optimal sebelum dirilis. Selain itu, sprint ini juga mencakup pembuatan dokumentasi untuk memastikan setiap aspek sistem terdokumentasi dengan baik, sehingga *website* siap digunakan secara maksimal.

V. KESIMPULAN DAN SARAN

5.3 Kesimpulan

Berdasarkan hasil implementasi dan pengujian sistem, serta mengacu pada rumusan masalah dan tujuan penelitian yang telah dijabarkan pada Bab I, maka dapat disimpulkan bahwa:

Implementasi *Web Application Firewall* (WAF) dan *Content Delivery Network* (CDN) pada *website* FMIPA Universitas Lampung menunjukkan hasil yang signifikan dalam meningkatkan keamanan dan kecepatan akses. WAF mampu memberikan perlindungan terhadap berbagai serangan siber umum seperti *Distributed Denial of Service* (DDoS), *SQL Injection*, dan *Cross-Site Scripting* (XSS). Efektivitas ini dibuktikan melalui pengujian keamanan menggunakan metode *black box testing*, di mana sistem berhasil memblokir atau merespons dengan tepat terhadap upaya simulasi serangan. Di sisi lain, penggunaan CDN berhasil mempercepat waktu muat halaman (*page load time*) serta meningkatkan efisiensi pengiriman konten melalui *cache*. Hasil pengujian performa menggunakan *GTmetrix* dan *Google PageSpeed Insights* menunjukkan peningkatan skor performa dan penurunan *latency* setelah implementasi.

Secara keseluruhan, implementasi teknologi WAF dan CDN ini telah berhasil mencapai tujuan utama penelitian, yakni meningkatkan keamanan dan kecepatan akses *website* FMIPA Universitas Lampung, yang pada akhirnya juga memberikan dampak positif terhadap pengalaman pengguna serta citra digital institusi.

5.2 Saran

Berdasarkan hasil penelitian dan proses pengembangan yang telah dilakukan, beberapa saran yang dapat diberikan untuk pengelolaan dan pengembangan sistem selanjutnya adalah sebagai berikut.

- a. Meskipun implementasi *Web Application Firewall* (WAF) telah berhasil menahan serangan umum seperti *DDoS*, *SQL Injection*, dan *XSS*, tetap diperlukan evaluasi rutin terhadap konfigurasi *firewall* dan *rules* untuk menyesuaikan dengan pola serangan terbaru.
- b. Pemantauan performa dan keamanan melalui *dashboard Cloudflare*. Disarankan agar tim pengelola secara aktif memanfaatkan fitur analitik dan *firewall event* log di *cloudflare* guna memantau *trafik* mencurigakan serta menilai efektivitas pemblokiran oleh WAF dan CDN.
- c. Untuk penelitian atau pengembangan berikutnya, disarankan agar sistem dilengkapi dengan fitur keamanan tambahan yang lain, serta dapat mengevaluasi perbandingan beberapa layanan CDN lainnya (misalnya, *Akamai*, dan *Fastly*) agar diperoleh *insight* lebih luas.

DAFTAR PUSTAKA

- Alim, E. S., Nuroji, N., Rizkiawan, M. A., Anhari, T., & Sobari, B. (2024). Monitoring dan Pencegahan Serangan Judi Online (Slot Gacor) pada Website. *Edumatic: Jurnal Pendidikan Informatika*, 8(1), 75–83. <https://doi.org/10.29408/edumatic.v8i1.25267>
- Ardiansyah, Y., Sunandar, M. A., & Muhyidin, Y. (2023). Implementasi Keamanan Website Dengan Metode Firewall Aplikasi Web (WAF) Studi kasus: Web Desa Wantilan. *Jurnal Mahasiswa Teknik Informatika*, 7(3), 2018–2025. <https://ejournal.itn.ac.id/index.php/jati/article/view/7048>
- Bhayangkara, D. S., & Ashari, W. M. (2024). Pengaruh Load Balancing Pada Serangan DDoS Menggunakan Nginx. *Indonesian Journal of Computer Science*, 13(4), 6275–6287. <http://ijcs.stmikindonesia.ac.id/ijcs/index.php/ijcs/article/view/3135>
- Elu, A. M. (2013). Rancang Bangun Aplikasi Pendeteksian Vulnerability Structured Query Language (Sql) Injection Untuk Keamanan Website. *Respati*, 8(22), 111–124. <https://doi.org/10.35842/jtir.v8i22.53>
- Haikal Muhammad, H., Id Hadiana, A., & Ashaury, H. (2023). Pengamanan Aplikasi Web Dari Serangan Sql Injection Dan Cross Site Scripting Menggunakan Web Application Firewall. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 7(5), 3265–3273. <https://doi.org/10.36040/jati.v7i5.7320>

Harahap, B. (2021). Penerapan Keamanan Owasp Terhadap Aplikasi GTFW Pada Website Universitas Battuta. *Jurnal Informatika Dan Teknologi Pendidikan*, 1(2), 80–86.
<https://doi.org/10.25008/jitp.v1i2.15>

Hidayah, T., & Saptono, H. (2017). PENERAPAN HIGH AVAILABILITY WEB SERVER MENGGUNAKAN NGINX DAN MODSECURITY. *Jurnal Informatika Terpadu*, 3(2), 95–102.
<https://journal.nurulfikri.ac.id/index.php/JIT>

Indah, F., Sidabutar, A., & Annisa, N. (2022). Peran Cyber Security Terhadap Keamanan Data Penduduk Negara Indonesia (Studi Kasus: Hacker Bjorka). *Jurnal Bidang Penelitian Informatika*, 1(1), 1–8.
<https://ejournal.kreatifcemerlang.id/index.php/jbpi>

Laksmiati, D. (2022). PENGUJIAN OPTIMASI PERFORMA WEBSITE MENGGUNAKAN CLOUDFLARE DENGAN METODE STRESS TEST. *Jurnal AKRAB JUARA*, 7(1), 261–272.

Hidayati, N. H. (2022). Analisis Performa Website Kantor Pencarian Dan Pertolongan Palembang Menggunakan Gtmetrix. *Seminar Nasional Teknologi Dan Multidisiplin Ilmu (SEMNASTEKMU)*, 2(2), 228–233.
<https://doi.org/10.51903/semnastekmu.v2i1.173>

Rahmadyanto, E. P., & Chandra, D. W. (2023). Analisis keamanan Content Delivery Network (CDN) Cloudflare (Studi kasus: Web Hakazon). *Jutisi: Jurnal Ilmiah Teknik Informatika Dan Sistem Informasi*, 12(3), 1818–1929

Riska, R., & Alamsyah, H. (2021). Penerapan Sistem Keamanan Web Menggunakan Metode Web Application Firewall. *Jurnal Amplifier : Jurnal Ilmiah Bidang Teknik Elektro Dan Komputer*, 11(1), 37–42.
<https://doi.org/10.33369/jamplifier.v11i1.16683>

Risky, M. A. Z., & Yuhandri, Y. (2021). Optimalisasi dalam Penetrasi Testing Keamanan Website Menggunakan Teknik SQL Injection dan XSS. *Jurnal Sistim Informasi Dan Teknologi*, 3, 215–220.
<https://doi.org/10.37034/jsisfotek.v3i4.68>

Schwaber, K., & Sutherland, A. (2017). *The Scrum Guide The Definitive Guide to Scrum: The Rules of the Game*.
<https://doi.org/10.1088/1751-8113/44/8/085201>

Sukmana, S. H., Saputra, D., Puspitasari, D., Azizah, Q. N., Sikumbang, E. D., & Ramanda, K. (2024). Analisis Web Performance Load Test Pada Situs Web PT Neptus Teknologi Indonesia Jakarta Setelah Menggunakan Cloud Web Application Firewall (WAF). *Jurnal Sains Komputer & Informatika (J-SAKTI)*, 8(1), 12–24.