

**PENGEMBANGAN *FRONT-END* SISTEM INFORMASI *MONITORING SMART PARKING* FAKULTAS TEKNIK UNIVERSITAS LAMPUNG
BERBASIS NEXT.JS**

(Skripsi)

Oleh

**FANNISA NURHAFFIFI
NPM 2215061009**



**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNIK
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2026**

**PENGEMBANGAN *FRONT-END* SISTEM INFORMASI MONITORING
SMART PARKING FAKULTAS TEKNIK UNIVERSITAS LAMPUNG
BERBASIS NEXT.JS**

Oleh

FANNISA NURHAFFIFI

Skripsi

**Sebagai Salah Satu Syarat untuk Mencapai gelar
SARJANA TEKNIK**

Pada

**Jurusan Teknik Elektro
Fakultas Teknik Universitas Lampung**



**FAKULTAS TEKNIK
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG**

2026

ABSTRAK

Pengembangan *Front-End* Sistem Informasi *Monitoring Smart Parking*

Fakultas Teknik Universitas Lampung Berbasis Next.Js

Oleh

Fannisa Nurhaffifi

Permasalahan parkir kendaraan bermotor mahasiswa di lingkungan Fakultas Teknik Universitas Lampung masih belum terkelola secara optimal karena sistem yang digunakan bersifat konvensional dan menyebabkan kesulitan pemantauan parkir, serta menurunnya efisiensi pengelolaan lahan. Tidak adanya riwayat parkir menimbulkan risiko keamanan karena sulit melacak kendaraan keluar-masuk. Penelitian ini mengembangkan *front-end* Sistem Informasi *Monitoring Smart Parking* berbasis *website* menggunakan *framework* Next.js untuk menampilkan informasi parkir secara *real-time*, interaktif, dan responsif. Sistem juga dilengkapi fitur riwayat parkir yang mencatat waktu masuk, keluar, durasi parkir guna meningkatkan keamanan dan transparansi. Metode yang digunakan adalah *Rapid Application Development* (RAD) dengan durasi 2 bulan melalui dua tahapan iterasi yaitu pada iterasi pertama berfokus pada pembangunan fungsi dasar sesuai kebutuhan fungsional, sedangkan iterasi kedua pada penyempurnaan fitur dan integrasi sistem yaitu menambahkan fitur *scan* kartu RFID untuk *role* mahasiswa dan menambahkan fitur pengaturan kesempatan parkir yang sebelumnya ditetapkan sebanyak 30 kali, menjadi dapat disesuaikan sesuai kebutuhan pengguna dan kebijakan pengelola parkir serta pengaturan slot parkir untuk *role* Admin. Hasil pengujian menunjukkan sistem berjalan baik dengan tingkat keberhasilan skenario berdasarkan kebutuhan fungsional *Black Box Testing* sebesar 100% valid pada seluruh fitur. Pengujian *System Usability Scale* (SUS) memperoleh skor rata-rata 80,25 termasuk dalam kategori *Acceptable* dan berada pada tingkat *Excellent*. Nilai ini juga masuk dalam *Grade A*.

Kata Kunci : *Smart Parking*, Next.js, *Front-End*, *Rapid Application Development* (RAD), *Internet of Things* (IoT)

ABSTRACT

Front-End Development of a Smart Parking Monitoring Information System at the Faculty of Engineering, University of Lampung Using Next.js

By

Fannisa Nurhaffifi

Parking issues for students' motor vehicles at the Faculty of Engineering, University of Lampung, are still not managed optimally due to the use of conventional systems, causing difficulties in monitoring parking activities and reducing land management efficiency. In addition, the absence of parking history records creates security risks because vehicle entry and exit activities cannot be tracked properly. This study aims to develop the front-end of a web-based Smart Parking Monitoring Information System using the Next.js framework to provide real-time, interactive, and responsive parking information. The system also includes a parking history feature that records entry time, exit time, and parking duration to improve security and transparency. This research applied the Rapid Application Development (RAD) method over two months through two iterations. The first iteration focused on developing the main functions based on functional requirements, while the second iteration focused on feature enhancement and system integration, including RFID card scanning for students, adjustable parking quota settings, and parking slot management for administrators. The testing results showed that the system achieved a 100% success rate in Black Box Testing for all functional scenarios. In addition, the System Usability Scale (SUS) evaluation obtained an average score of 80.25, categorized as Acceptable, Excellent, and Grade A.

Keywords: Smart Parking, Next.js, Front-End, WebSocket, Rapid Application Development (RAD), Internet of Things (IoT),

Judul Skripsi : PENGEMBANGAN *FRONT-END* SISTEM
INFORMASI *MONITORING SMART PARKING*
FAKULTAS TEKNIK UNIVERSITAS
LAMPUNG BERBASIS NEXT.JS

Nama Mahasiswa : **Fannisa Nurhaffifi**

Nomor Pokok Mahasiswa : 2215061009

Program Studi : S1 Teknik Informatika

Jurusan : Teknik Elektro


Fakultas : Teknik

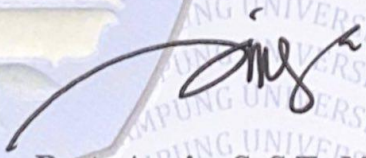
MENYETUJUI

1. Komisi Pembimbing

Pembimbing Utama

Pembimbing Pendamping


Rio Arjestia Pradipta, S.Kom, M. T.I.
NIP. 108603232019031013



Resty Annisa S, S.T., M.Kom
NIP. 199008302019032019

2. Mengetahui

Ketua Jurusan
Teknik Elektro

Ketua Program Studi
Teknik Informatika


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


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

MENGESAHKAN

1. **Tim Penguji**

Ketua : Rio Ariestia Pradipta, S.Kom, M. T.I.

Sekretaris : Resty Annisa S, S.T., M.Kom

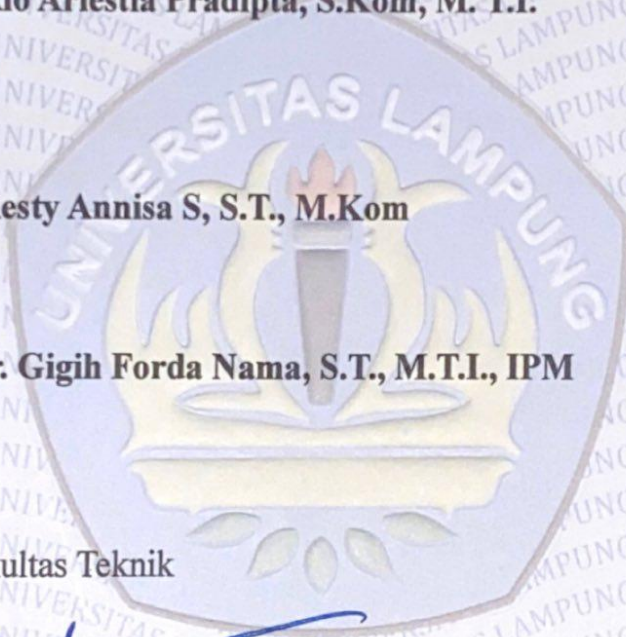
Penguji : Ir. Gigih Forda Nama, S.T., M.T.I., IPM

2. **Dekan Fakultas Teknik**

Dr. Ahmad Herison, S.T., M.T.

NIP. 196910302000031001

Tanggal Lulus Ujian Skripsi : 20 Mei 2026



Rio Ariestia Pradipta

Resty Annisa S
Ir. Gigih Forda Nama

SURAT PERNYATAAN

Saya yang bertanda tangan dibawah ini, menyatakan dengan sebenarnya bahwa skripsi saya yang berjudul “Pengembangan *Front-End* Sistem Informasi *Monitoring Smart Parking* Fakultas Teknik Universitas Lampung Berbasis Next.js” merupakan hasil karya saya sendiri. Semua hasil yang tertuang dalam skripsi ini telah mengikuti kaidah penulisan karya ilmiah Universitas Lampung. Apabila di kemudian hari terbukti bahwa skripsi saya ini merupakan salinan atau dibuat oleh orang lain, maka saya bersedia menerima sanksi sesuai dengan ketentuan hukum atau akademik yang berlaku.

Bandar Lampung, 25 Mei 2026
Pembuat Pernyataan



Fannisa Nurhaffifi
2215061009

RIWAYAT HIDUP



Penulis bernama Fannisa Nurhaffifi, dilahirkan di Bandar Lampung pada tanggal 18 Agustus 2004, sebagai anak dari pasangan Bapak Jumhani Ali dan Ibu Jamilah Maulida. Penulis memulai pendidikan di TK Qurotta Ayun, kemudian melanjutkan pendidikan dasar di SD Negeri 5 Talang dan lulus pada tahun 2016. Pendidikan menengah pertama diselesaikan di SMP Negeri 4 Bandar Lampung pada tahun 2019, dan pendidikan menengah atas di SMA S Yp Unila yang diselesaikan pada tahun 2022. Selanjutnya, penulis melanjutkan pendidikan tinggi di Universitas Lampung pada Program Studi S1 Teknik Informatika, Jurusan Teknik Elektro, Fakultas Teknik melalui jalur SNMPTN (Seleksi Nasional Masuk Perguruan Tinggi Negeri) pada tahun 2022. Selama menjalani masa perkuliahan, penulis aktif berpartisipasi dalam berbagai kegiatan akademik dan organisasi, antara lain:

1. Menjadi anggota dan pengurus Himpunan Mahasiswa Teknik Elektro (HIMATRO) pada periode 2022-2024.
2. Mengikuti program Merdeka Belajar Kampus Merdeka (MBKM) Batch 7 pada tahun 2024 melalui program Studi Independen sebagai Data Analyst & Gen AI di PT Revolusi Cita Edukasi (RevoU).
3. Melaksanakan Kuliah Kerja Nyata (KKN) di Desa Uman Agung Mataram, Kecamatan Bandar Mataram, Kabupaten Lampung Tengah, Provinsi Lampung pada tahun 2025.
4. Melaksanakan Kerja Praktik (KP) di Diskominfotik, Bandar Lampung pada bulan Juni hingga Agustus 2025.
5. Berperan sebagai Asisten Praktikum di Laboratorium Teknik Komputer Universitas Lampung pada tahun 2024-2026.

MOTTO

إِنَّ اللَّهَ مَعَ الصَّابِرِينَ

“*Sesungguhnya Allah bersama orang-orang yang sabar.*”

(QS. Al-Baqarah: 153)

خُسْرًا لَفِي الْإِنْسَانِ إِنَّ

بِالصَّبْرِ وَتَوَاصَوْا بِالْحَقِّ وَتَوَاصَوْا الصُّلِحَاتِ وَعَمِلُوا أَمْثُلًا الَّذِينَ إِلَّا

“*sesungguhnya manusia benar-benar berada dalam kerugian,*”

“*kecuali orang-orang yang beriman dan beramal saleh serta saling menasihati untuk kebenaran dan kesabaran.*”

(QS. Al-Ashr: 2-3)

“*Youth is happy because it has the capacity to see beauty. Anyone who keeps the ability to see beauty never grows old.*”

(Franz Kafka)

PERSEMBAHAN

Bismillahirrahmanirrahim

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

Kupersembahkan skripsi ini kepada:

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

“Seluruh dosen dan civitas akademik serta rekan rekan seperjuangan prodi Teknik Informatika, Jurusan Teknik Elektro Universitas Lampung, yang telah memberikan arahan dan bimbingan kepada penulis”

“Kepada teman dan sahabat tercinta, yang senantiasa memberikan dukungan, semangat, doa, serta kebersamaan dalam setiap proses perjuangan, terima kasih atas motivasi dan kenangan yang telah menjadi bagian berarti dalam perjalanan penulis menyelesaikan skripsi ini.”

SANWACANA

Penulis mengucapkan puji syukur ke hadirat Allah SWT, Tuhan Yang Maha Esa, atas segala nikmat dan karunia-Nya, terutama nikmat kehidupan dan ketenangan batin dalam menjalaninya. Berkat rahmat dan ridha-Nya, penulis akhirnya dapat menyelesaikan skripsi yang berjudul: "Pengembangan *Front-End* Sistem Informasi *Monitoring Smart Parking* Fakultas Teknik Universitas Lampung Berbasis Next.Js" Skripsi ini disusun sebagai salah satu syarat untuk memperoleh gelar Sarjana Teknik pada Program Studi S1 Teknik Informatika, Fakultas Teknik, Universitas Lampung. Dalam proses tersebut, penulis menghadapi berbagai tantangan dan hari-hari yang tidak mudah, sehingga terselesaikannya skripsi ini menjadi pencapaian yang sangat berarti. Oleh karena itu, penulis ingin menyampaikan ucapan terima kasih kepada:

1. Bapak Dr. Ahmad Herison, S.T., M.T., selaku Dekan Fakultas Teknik Universitas Lampung.
2. Ibu Herlinawati, S.T., M.T., selaku Ketua Jurusan Teknik Elektro Fakultas Teknik Universitas Lampung.
3. Ibu Yessi Mulyani, S.T., M.T., selaku Ketua Program Studi Teknik Informatika Fakultas Teknik Universitas Lampung.
4. Bapak Ir. Gigih Forda Nama, S.T., M.T.I., IPM, selaku Dosen Penguji, yang telah memberikan kritik, saran, serta masukan yang membangun demi penyempurnaan skripsi ini sehingga menjadi lebih baik dan sistematis.
5. Bapak Rio Ariestia P, S.Kom., M.T.I., selaku Dosen Pembimbing Utama, yang telah meluangkan waktu, tenaga, dan pikiran untuk memberikan arahan, bimbingan, serta masukan yang sangat berarti kepada penulis selama proses penyusunan skripsi ini.

6. Ibu Resty Annisa, S.ST., M.Kom., selaku Dosen Pembimbing Pendamping, yang telah memberikan arahan, bimbingan, serta nasihat kepada penulis selama proses penyusunan skripsi ini.
7. Kedua orang tua tercinta, yaitu Bapak Jumhani Ali dan Ibu Jamilah Maulida, serta seluruh keluarga yang senantiasa memberikan doa, kasih sayang, dukungan moral maupun materi, serta semangat yang tiada henti kepada penulis.
8. Tim Skripsi yaitu tim *website* dan tim IoT atas kerja sama dan perjuangannya hingga akhir dalam penelitian ini
9. Arya Adhitama Al-Abyan terima kasih atas segala perhatian, dukungan yang selalu diberikan dalam setiap proses perjuangan penulis. Kehadiran dan semangat yang diberikan selama menyelesaikan skripsi ini.
10. Seluruh Asisten Laboratorium Teknik Komputer angkatan 2022–2023 atas pengalaman dan kebersamaan yang berharga.
11. Teman-temanku Annisa Tri Hapsari, Alya Ananda Putri, Jena Feronika atas dukungan dan persahabatan dalam kuliah ini
12. Himpunan Mahasiswa Teknik Elektro Universitas Lampung (HIMATRO UNILA) yang telah menjadi wadah bagi penulis dalam berorganisasi, mengembangkan diri, serta memberikan pengalaman berharga selama masa perkuliahan.
13. Seluruh mahasiswa dan civitas akademika Teknik Informatika, Jurusan Teknik Elektro Universitas Lampung atas dukungan dan bantuan yang diberikan.
14. Seluruh teman-teman Rectical 22 sebagai angkatan yang telah memberikan kebersamaan, dukungan, serta kenangan selama masa perkuliahan.
15. Pihak Jurusan Teknik Geodesi Fakultas Teknik Universitas Lampung yang telah memberikan izin serta dukungan dalam pelaksanaan penelitian.
16. Semua pihak yang tidak dapat disebutkan satu per satu, yang telah memberikan kontribusi dalam penyusunan skripsi ini

Penulis menyadari skripsi ini masih jauh dari sempurna karena keterbatasan pengetahuan dan pengalaman. Oleh sebab itu, kritik dan saran yang membangun sangat diharapkan. Semoga karya ini dapat memberikan manfaat, khususnya dalam bidang

Frontend dan pengembangan website, serta menjadi amal jariyah bagi semua pihak yang terlibat. Aamiin.

Bandar Lampung, 25 Mei 2026

Penulis

A handwritten signature in black ink, appearing to be 'Fannisa Nurhaffifi', written over a faint yellow rectangular stamp.

Fannisa Nurhaffifi

DAFTAR ISI

DAFTAR ISI	ix
DAFTAR GAMBAR	i
DAFTAR TABEL.....	i
I. PENDAHULUAN	2
1.1 Latar Belakang	2
1.2 Rumusan Masalah	6
1.3 Tujuan Penelitian.....	7
1.4 Manfaat Penelitian	7
1.5 Batasan Masalah.....	8
1.6 Sistematika Penulisan	8
II. TINJAUAN PUSTAKA.....	10
2.1. <i>Website</i>	10
2.2. Sistem Informasi <i>Monitoring</i> Parkir	11
2.3. Next.Js.....	12
2.4. Tailwind.....	12
2.5. Unified Model Language (UML).....	13
2.6. <i>Rapid Application Development (RAD)</i>	15
2.7. <i>Black Box Testing</i>	17
2.8. <i>System Usability Scale</i>	18
2.9. Penelitian Terkait.....	20
III. METODOLOGI PENELITIAN.....	26
3.1 Waktu dan Tempat.....	26
3.2 Tim Penelitian	27

3.3	Alat dan Bahan Penelitian.....	29
3.4	Arsitektur Sistem.....	32
3.5	Tahapan Penelitian	34
3.5.1	<i>Requirements Planning</i>	35
3.5.2	<i>User Design</i>	37
3.5.3	<i>Construction</i>	38
3.5.4	<i>Cutover</i>	39
IV.	HASIL DAN PEMBAHASAN.....	41
4.1	<i>Requirements Planning</i>	41
4.1.1	Kuesioner	41
4.1.2	Identifikasi Kebutuhan	46
4.2	<i>User Design</i>	48
4.2.1	Iterasi Pertama.....	48
4.2.2	Iterasi Kedua	83
4.3	<i>Construction</i>	93
4.3.1	Struktur Folder	93
4.3.2	Implementasi Kode	94
4.3.3	<i>Testing</i>	135
4.4	<i>Cutover</i>	145
4.4.1	Implementasi Sistem	145
4.4.2	Analisa dan Pembahasan.....	148
V.	KESIMPULAN DAN SARAN.....	151
5.1	Kesimpulan	151
5.2	Saran.....	152
	DAFTAR PUSTAKA	153

DAFTAR GAMBAR

Gambar 1. Hasil Kuesioner Penggunaan Parkir Fakultas Teknik	2
Gambar 2. Lahan Parkir Jurusan Teknik Geodesi	4
Gambar 3. Struktur <i>Unified Model Language</i> [15]	13
Gambar 4. Tahapan <i>Black Box Testing</i> [21]	17
Gambar 5. Kategori Penilaian System Usability Scale [25]	19
Gambar 6 Arsitektur Sistem	32
Gambar 7. Tahapan Penelitian	35
Gambar 8. Grafik Hasil Kuesioner Mahasiswa Jurusan Teknik Geodesi	43
Gambar 9 Jawaban Frekuensi Parkir Setiap Bulan	45
Gambar 10. <i>Use Case Diagram</i>	49
Gambar 11. <i>Activity Diagram</i> Memantau Data Parkir	53
Gambar 12. <i>Activity Diagram</i> Mengelola Pengguna Parkir	54
Gambar 13. Memverifikasi Pendaftaran	55
Gambar 14. Melihat akses Mahasiswa	56
Gambar 15. Membatasi akses pengguna	58
Gambar 16. Menghapus pengguna	59
Gambar 17. Daftar	60
Gambar 18. Melihat informasi parkir	61
Gambar 19. Melihat ketersediaan parkir	62
Gambar 20. Melihat kesempatan parkir	63
Gambar 21. Mengubah profil	64
Gambar 22. Masuk	65
Gambar 23. Masuk dan Keluar	66

Gambar 24. Melihat statistik kendaraan yang terparkir	66
Gambar 25.. RFID.....	67
Gambar 26. RFID Scan Masuk.....	68
Gambar 27. RFID <i>Scan</i> Keluar	69
Gambar 28 <i>Wireframe</i> Halaman Masuk.....	70
Gambar 29. <i>Wireframe</i> Halaman Daftar	71
Gambar 30. <i>Wireframe</i> Halaman Beranda Mahasiswa	72
Gambar 31. <i>Wireframe</i> Halaman Informasi Pengguna	73
Gambar 32. <i>Wireframe</i> Halaman Statistik Kendaraan	74
Gambar 33. <i>Wireframe</i> Halaman Profil Pengguna.....	75
Gambar 34. <i>Wireframe</i> Halaman Beranda Admin	76
Gambar 35. <i>Wireframe</i> Halaman Statistik Kendaraan Admin	77
Gambar 36. <i>Wireframe</i> Halaman Pengguna Parkir Admin	78
Gambar 37. <i>Wireframe</i> Halaman Data Parkir Admin	79
Gambar 38. <i>Wireframe</i> Halaman Lupa Kata Sandi.....	80
Gambar 39. <i>Wireframe</i> Halaman Verifikasi OTP.....	81
Gambar 40. <i>Wireframe</i> Halaman <i>Reset</i> Kata Sandi	82
Gambar 41. <i>Wireframe</i> Halaman Kata Sandi Berhasil Diubah.....	82
Gambar 42. <i>Use Case Diagram</i> Iterasi Kedua.....	84
Gambar 43. <i>Activity Diagram</i> mengubah Slot Kendaraan.....	86
Gambar 44. <i>Activity Diagram</i> Menambah Informasi RFID.....	87
Gambar 45. <i>Activity Diagram</i> Mengubah Kesempatan Parkir.....	88
Gambar 46. <i>Wireframe</i> Halaman Beranda Admin	89
Gambar 47. <i>Wireframe</i> Halaman Pengguna.....	90
Gambar 48. <i>Wireframe</i> Halaman <i>Scan</i> RFID.....	91
Gambar 49. Struktur Folder	93
Gambar 50. Implementasi Halaman Daftar	96
Gambar 51. Implementasi Halaman Sukses Daftar	98
Gambar 52. Halaman Masuk Sistem.....	100
Gambar 53. Implementasi Halaman Lupa Kata Sandi.....	102

Gambar 54. Implementasi Tampilan Halaman Verifikasi OTP.....	104
Gambar 55. Implementasi Tampilan <i>Reset</i> Kata Sandi.....	106
Gambar 56. Implementasi Tampilan Halaman Sukses Reset Kata Sandi.....	107
Gambar 57. Halaman Masuk Sistem.....	109
Gambar 58. Halaman Beranda Mahasiswa.....	111
Gambar 59. Halaman Informasi Parkir Mahasiswa.....	114
Gambar 60. Halaman Statistik Kendaraan Mahasiswa.....	116
Gambar 61. Implementasi Tampilan Profil.....	118
Gambar 62. Halaman Beranda Admin.....	121
Gambar 63. Halaman Statistik Pengguna Admin.....	123
Gambar 64. Halaman Pengguna Parkir Admin.....	127
Gambar 65. Implementasi Tampilan Individu.....	129
Gambar 66. Implementasi Tampilan Scan Kartu RFID.....	132
Gambar 67. Halaman Data Parkir.....	134
Gambar 68. Skor System Usability Testing.....	142
Gambar 69. Jumlah Responden.....	144
Gambar 70. Konfigurasi DNS.....	146

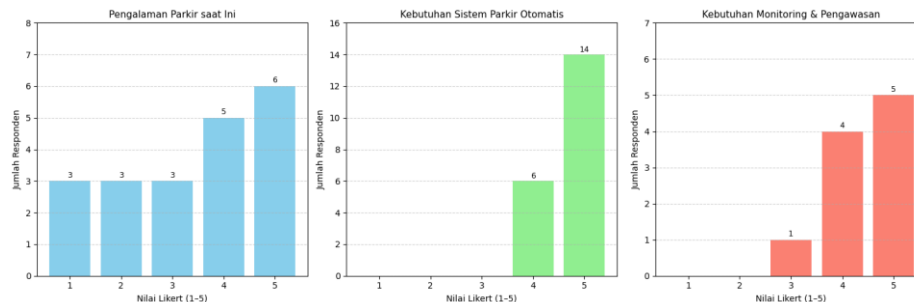
DAFTAR TABEL

Tabel 1 Pertanyaan System Usability Scale [24]	19
Tabel 2. Penelitian Terkait.....	20
Tabel 3. Waktu Pelaksanaan Penelitian	26
Tabel 4 Tim Penelitian	27
Tabel 5 Alat dan Bahan	29
Tabel 7. Pertanyaan Kuesioner.....	41
Tabel 8. Kebutuhan Fungsional.....	46
Tabel 9. Kebutuhan Non-Fungsional	47
Tabel 10 Deskripsi <i>Use Case</i> Diagram	50
Tabel 11. Deskripsi Use Case Diagram.....	85
Tabel 12. Skenario Pengujian Fitur Autentikasi.....	136
Tabel 13. Skenario Pengujian Fitur Admin	137
Tabel 14. Skenario Pengujian Fitur Mahasiswa.....	139
Tabel 15. Pertanyaan <i>System Usability Testing</i> (SUS).....	141
Tabel 16. Hasil Jawaban.....	143

I. PENDAHULUAN

1.1 Latar Belakang

Sistem parkir seringkali menjadi salah satu permasalahan yang muncul di berbagai tempat, termasuk di lingkungan kampus. Peningkatan jumlah mahasiswa baru setiap tahun menyebabkan bertambahnya jumlah kendaraan pribadi yang digunakan oleh civitas akademika. Kondisi tersebut menimbulkan tantangan tersendiri dalam hal pengaturan serta penyediaan lahan parkir yang memadai di area kampus. Permasalahan ini berdampak pada berbagai aspek, seperti berkurangnya kenyamanan mahasiswa saat memarkir kendaraan, meningkatnya kasus parkir di area yang tidak semestinya, serta potensi gangguan terhadap ketertiban dan keamanan lingkungan kampus. [1] Pengelolaan lahan parkir secara manual sudah tidak efektif lagi dalam mengatasi permasalahan parkir. Akibatnya, kesulitan menemukan lokasi parkir yang sesuai dan sering kali memilih untuk memarkirkan kendaraannya di tempat yang tidak semestinya.[2]

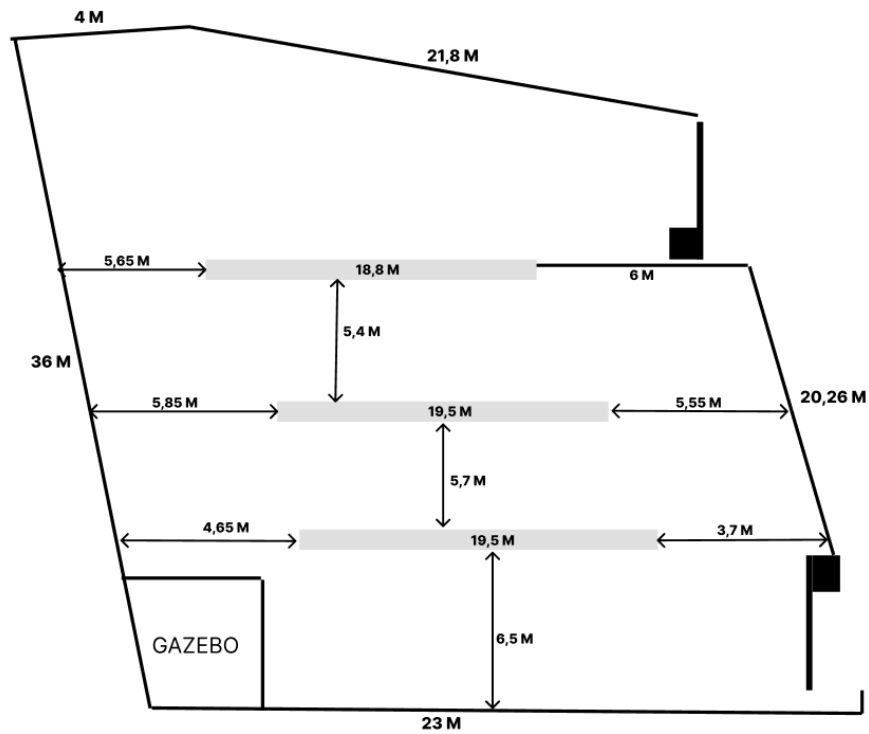


Gambar 1. Hasil Kuesioner Penggunaan Parkir Fakultas Teknik

Berdasarkan Gambar 1. secara keseluruhan, hasil kuesioner yang telah dilakukan menunjukkan bahwa permasalahan parkir di Fakultas Teknik Universitas Lampung masih menjadi isu yang memerlukan perhatian serius dan membutuhkan solusi berbasis teknologi. Kuesioner ini melibatkan dua kelompok responden utama, yaitu pihak pimpinan fakultas (ketua dan wakil dekanat) sebagai pengambil keputusan, serta petugas keamanan (satpam) yang terlibat langsung dalam pengawasan dan pengelolaan parkir di lapangan. Berdasarkan penilaian kedua pihak tersebut, diperoleh kesimpulan bahwa sistem parkir konvensional yang masih mengandalkan pengawasan manual tidak lagi memadai untuk mengatasi kepadatan kendaraan, keterbatasan informasi ketersediaan informasi slot parkir, serta risiko keamanan yang mungkin muncul.

Penelitian sistem *Smart Parking* ini pada dasarnya dikembangkan untuk dapat diterapkan di seluruh area Fakultas Teknik Universitas Lampung sebagai solusi terpadu dalam pengelolaan parkir tetapi sistem parkir di Jurusan Teknik Geodesi dipilih sebagai objek studi kasus karena telah memiliki fasilitas lahan parkir yang memadai serta portal parkir yang sudah terintegrasi dengan teknologi. Namun, sistem yang berjalan belum menyediakan akses informasi terkait kondisi lahan parkir secara menyeluruh. Pengguna belum dapat mengetahui jumlah slot yang tersedia, riwayat parkir, maupun statistik penggunaan parkir secara *real-time*. Dengan demikian, sistem yang ada masih terbatas pada kontrol akses masuk, sementara aspek monitoring dan penyajian informasi parkir belum terintegrasi dalam satu sistem digital yang terpadu.

Hal ini membebani petugas serta berisiko menimbulkan ketidakteraturan parkir, keterbatasan pemantauan, hingga potensi kehilangan kendaraan. Penelitian di Shahid Bahonar University of Kerman (SBUK), Iran, membahas penerapan sistem pengelolaan parkir berbasis *website* untuk mengatasi keterbatasan lahan dan tingginya jumlah pengguna di lingkungan kampus. Sistem yang diusulkan dirancang menggunakan model optimasi agar alokasi tempat parkir dapat dilakukan secara *real-time* dan adil bagi berbagai kelompok pengguna, seperti dosen, pegawai, dan mahasiswa.[3]



Gambar 2. Lahan Parkir Jurusan Teknik Geodesi

Berdasarkan Gambar 2. lahan parkir diperkirakan sekitar $\pm 878 \text{ m}^2$. Tidak seluruh area tersebut dapat digunakan secara maksimal karena sebagian lahan dimanfaatkan untuk jalur jalan kendaraan. Oleh karena itu, luas lahan parkir yang benar-benar dapat digunakan diperkirakan sekitar $\pm 720 \text{ m}^2$. Dengan kebutuhan ruang parkir sepeda motor sekitar $2,5 \text{ m}^2$ per unit, area parkir tersebut diperkirakan dapat menampung sekitar ± 280 sepeda motor apabila kendaraan tertata dengan baik. Namun, sistem parkir yang diterapkan saat ini masih bersifat konvensional dan belum terintegrasi dengan sistem *monitoring* berbasis *website* yang dimana sistem parkir belum mampu menyediakan informasi kondisi parkir secara menyeluruh dan *real-time*, baik terkait ketersediaan lahan, jumlah kendaraan yang terparkir, maupun riwayat penggunaan parkir melalui sistem terpusat. Peralihan dari sistem parkir konvensional menuju *smart parking* menjadi langkah penting untuk meningkatkan efisiensi pengelolaan area parkir di lingkungan Jurusan Teknik Geodesi Universitas Lampung. Pada sistem sebelumnya, proses keluar-masuk kendaraan sepenuhnya bergantung pada pengawasan manual, sehingga ketersediaan informasi slot tidak dapat diketahui secara langsung oleh

pengguna maupun pengelola. Untuk mengatasi keterbatasan tersebut, sistem *smart parking* mulai diterapkan dengan memanfaatkan teknologi yang lebih terintegrasi.

Selain aspek kapasitas dan pengelolaan lahan, pengaturan parkir juga berkaitan dengan pembatasan kesempatan parkir bagi setiap mahasiswa sebagai upaya mengurangi dampak lingkungan. Dengan keterbatasan akses parkir, pengguna didorong untuk mempertimbangkan alternatif transportasi yang lebih ramah lingkungan, seperti berjalan kaki, atau menggunakan transportasi umum yang sudah disediakan. Pada penelitian di UNNES ditemukan bahwa sebagian besar mahasiswa UNNES masih bergantung pada sepeda motor untuk mobilitas sehari-hari, karena alasan fleksibilitas dan kenyamanan. Tingginya penggunaan kendaraan pribadi ini berkontribusi pada peningkatan emisi gas rumah kaca dan polusi udara di sekitar kampus.[4] Selain berdampak pada lingkungan, kebijakan pembatasan kesempatan parkir juga memiliki manfaat dari sisi kesehatan. Berkurangnya ketergantungan terhadap kendaraan bermotor mendorong peningkatan aktivitas fisik ringan, seperti berjalan kaki dari titik parkir menuju gedung tujuan, yang secara tidak langsung dapat meningkatkan kesehatan pengguna. Oleh karena itu, pengelolaan parkir tidak hanya berfungsi sebagai penyedia ruang kendaraan, tetapi juga sebagai pendukung terciptanya lingkungan kampus yang lebih sehat.

Smart Parking merupakan sistem pengelolaan parkir berbasis teknologi informasi yang menyediakan informasi ketersediaan slot secara *real-time* serta mengatur akses pengguna melalui perangkat seperti sensor kendaraan dan kartu RFID. Sistem ini memandu pengguna menuju slot yang tersedia melalui indikator lampu, sehingga proses parkir menjadi lebih cepat, tertata, dan efisien dalam pemanfaatan ruang. [5] Penerapan teknologi dimulai dari penggunaan kartu RFID sebagai identitas pengguna parkir. Setiap kendaraan yang masuk atau keluar akan menggunakan RFID pada modul pembaca (RFID *reader*). Perangkat ini bekerja sebagai sensor yang mendeteksi keberadaan pengguna secara otomatis tanpa perlu input manual dari petugas. Data hasil pembacaan sensor kemudian dikirimkan ke *back-end* melalui mikrokontroler, yang

bertugas memproses informasi mengenai status kendaraan, seperti waktu masuk, waktu keluar, serta perubahan jumlah slot yang tersedia. *back-end* selanjutnya melakukan pembaruan pada basis data dan mengirimkan status terbaru ke *front-end*. Pada tahap ini, sistem *front-end* menampilkan informasi ketersediaan slot parkir secara *real-time* melalui antarmuka *website*, sehingga pengguna dapat mengetahui keadaan parkir sebelum memasuki area kampus. Melalui alur ini, pengelolaan parkir menjadi lebih terstruktur, informatif, dan minim kesalahan, sekaligus membantu menciptakan lingkungan parkir yang lebih tertata dan efisien dibandingkan metode konvensional.

Penelitian ini berfokus pada pengembangan *front-end* Sistem Informasi *Monitoring Smart Parking* berbasis *website* di Fakultas Teknik Jurusan Teknik Geodesi. *Front-end* sistem akan dirancang untuk menyajikan data parkir secara *real time* dengan memanfaatkan *framework* Next.js, dengan adanya sistem ini, diharapkan proses monitoring parkir dapat dilakukan secara terpusat, efisien, dan informatif, sehingga membantu pihak fakultas dalam melakukan pengawasan, analisis, serta pengambilan keputusan terkait manajemen lahan parkir. Selain itu, penerapan sistem berbasis *website* ini juga dapat meningkatkan transparansi dan akurasi data parkir, mendukung kebijakan pembatasan hak parkir bagi pengguna yang melanggar aturan.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan, maka rumusan masalah dalam penelitian ini adalah sebagai berikut:

1. Bagaimana mengimplementasikan *framework* Next.js untuk menghasilkan tampilan antarmuka (*front-end*) yang interaktif, responsif, dan mudah digunakan oleh pengguna?
2. Bagaimana proses perancangan dan pembangunan sistem informasi *monitoring Smart Parking* berbasis web dengan menerapkan metode *Rapid Application Development* (RAD) agar mampu menampilkan data ketersediaan parkir secara *real-time*?

3. Bagaimana kinerja fungsional dan kepuasan penggunaan sistem *Smart Parking* ini dapat dievaluasi melalui metode *Blackbox Testing* dan *System Usability Scale* (SUS) untuk menilai efektivitas penggunaan sistem dalam menggunakan lahan parkir di Jurusan Teknik Geodesi Universitas Lampung?

1.3 Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah:

1. Mengembangkan *front-end* Sistem Informasi *Monitoring Smart Parking* berbasis website menggunakan *framework* Next.js sebagai antarmuka utama yang digunakan oleh pengguna.
2. Mengimplementasikan sistem informasi monitoring *Smart Parking* berbasis website dengan menerapkan metode *Rapid Application Development* (RAD) guna menghasilkan sistem yang mampu menampilkan data ketersediaan parkir secara *real-time*, akurat, dan responsif sesuai dengan kebutuhan pengguna.
3. Melakukan evaluasi terhadap kinerja fungsional serta tingkat *usability* sistem *Smart Parking* dengan menggunakan metode *Blackbox Testing* dan *System Usability Scale* (SUS) dalam pengelolaan lahan parkir di Jurusan Teknik Geodesi Universitas Lampung.

1.4 Manfaat Penelitian

Hasil penelitian ini diharapkan memberikan beberapa manfaat sebagai berikut:

1. Membantu dalam pengelolaan dan pemantauan lahan parkir secara *real time*.
2. Meningkatkan keamanan area parkir melalui pemantauan akses pengguna, verifikasi identitas, serta pencegahan penggunaan slot parkir oleh pihak yang tidak berwenang.
3. Memudahkan pengguna dalam mengetahui ketersediaan slot parkir sebelum memasuki area kampus.

1.5 Batasan Masalah

Agar penelitian ini lebih terarah dan fokus, maka batasan masalah yang ditetapkan adalah sebagai berikut:

1. Sistem berfokus pada *monitoring* ketersediaan parkir kendaraan roda dua di lingkungan Jurusan Teknik Geodesi.
2. Pengembangan sistem hanya mencakup bagian *front-end*, tanpa mencakup implementasi *back-end* maupun integrasi perangkat IoT.
3. Penelitian ini dibatasi pada pengembangan sistem informasi *monitoring smart parking* dalam skala simulasi yang dirancang sebagai representasi sistem parkir kampus yang sebenarnya, namun belum diimplementasikan secara langsung di lingkungan kampus.

1.6 Sistematika Penulisan

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

BAB I : PENDAHULUAN

Bab ini berisi latar belakang yang menjelaskan alasan pengembangan Sistem Informasi *Monitoring Smart Parking* di Jurusan Teknik Geodesi Universitas Lampung, rumusan masalah yang ingin diselesaikan, tujuan penelitian, manfaat penelitian, batasan masalah, serta sistematika penulisan laporan.

BAB II : TINJAUAN PUSTAKA

Bab ini berisi tinjauan pustaka yang menjadi dasar dalam pengembangan Sistem Informasi *Monitoring Smart Parking*. Pembahasan dalam bab ini meliputi peran teknologi informasi dalam sistem transportasi, konsep *smart parking*, pengembangan sistem informasi berbasis website, teknologi *database*, pemodelan

sistem menggunakan UML, *framework* Next.js, metode pengembangan *Rapid Application Development* (RAD), serta pengujian perangkat lunak. Selain itu, bab ini juga mencakup penelitian terdahulu yang relevan sebagai referensi dalam penulisan laporan ini.

BAB III : METODOLOGI PENELITIAN

Bab ini membahas metodologi penelitian yang digunakan dalam pengembangan Sistem Informasi *Monitoring Smart Parking*, termasuk waktu dan tempat penelitian, perangkat keras dan perangkat lunak yang digunakan, serta tahapan penelitian yang dilakukan menggunakan metode *Rapid Application Development* (RAD).

BAB IV : HASIL DAN PEMBAHASAN

Bab ini berisi hasil dari pengembangan *front-end* Sistem Informasi *Monitoring Smart Parking* berbasis *website* menggunakan *framework* Next.js, serta pembahasan terkait implementasi sistem. Pembahasan mencakup desain *database*, hasil integrasi dengan perangkat IoT, hasil pengujian fitur, serta analisis sistem dalam memenuhi kebutuhan pengguna, khususnya untuk mendukung pengawasan dan keamanan parkir di Jurusan Teknik Geodesi Universitas Lampung.

BAB V : PENUTUP

Bab ini menyajikan kesimpulan dari penelitian yang telah dilakukan terkait pengembangan Sistem Informasi *Monitoring Smart Parking*, serta saran untuk pengembangan lebih lanjut agar sistem dapat meningkatkan kinerja, keamanan, dan manfaat bagi pengguna.

II. TINJAUAN PUSTAKA

2.1. *Website*

Website merupakan salah satu media yang digunakan oleh perusahaan, organisasi, maupun lembaga untuk menyampaikan informasi serta memperkenalkan produk dan layanan yang dimiliki. [6] dengan adanya *website*, pengguna dapat melakukan interaksi dengan pemilik sistem melalui fitur seperti formulir *online*, kolom komentar, hingga sistem manajemen konten. Selain itu, aplikasi berbasis *website* juga memudahkan pengguna atau karyawan dalam membuat serta berbagi dokumen, bekerja sama dalam sebuah proyek, dan mengakses file yang sama dari berbagai perangkat maupun lokasi yang berbeda. [7]

Website disusun menggunakan berbagai bahasa pemrograman seperti *Hypertext Markup Language* (HTML), *Cascading Style Sheets* (CSS), *JavaScript*, dan bahasa pemrograman lainnya yang berfungsi untuk membangun struktur, tampilan, serta interaktivitas halaman *website*. Konten yang terdapat pada sebuah *website* sangat beragam dan disesuaikan dengan tujuan pengembangannya. Beberapa *website* dibuat sebagai profil perusahaan atau institusi, sedangkan lainnya berfungsi untuk menyajikan data dalam bentuk grafik maupun informasi interaktif lainnya. [8] Salah satu bentuk penerapannya dapat dilihat pada sistem monitoring parkir berbasis *website*, di mana *website* berperan sebagai antarmuka utama untuk menampilkan informasi terkait kondisi parkir, jumlah kendaraan, serta ketersediaan area parkir secara *real-time*.

2.2. Sistem Informasi *Monitoring Parkir*

Sistem informasi adalah perpaduan antara teknologi, sumber daya manusia, dan prosedur kerja yang dirancang untuk melakukan pengumpulan, pengolahan, penyimpanan, serta penyebaran data menjadi informasi. Tujuan utama sistem ini adalah membantu proses pengambilan keputusan, meningkatkan koordinasi, mendukung analisis, serta memfasilitasi pengendalian dalam suatu organisasi. Melalui sistem informasi, data mentah dapat diubah menjadi informasi yang relevan dan bernilai bagi penggunanya. [9] Sistem informasi merupakan suatu kesatuan yang terintegrasi antara perangkat keras, perangkat lunak, dan sumber daya manusia yang berfungsi untuk mengolah data menjadi informasi yang bermanfaat bagi pengguna.

Dalam pengembangannya, sistem informasi terdiri dari beberapa komponen utama yang saling berkaitan, yaitu perangkat keras (*hardware*) sebagai media fisik pemrosesan data, perangkat lunak (*software*) yang berfungsi mengatur jalannya sistem, basis data (*database*) sebagai tempat penyimpanan dan pengelolaan data, prosedur (*procedure*) sebagai aturan dan langkah-langkah kerja sistem, serta sumber daya manusia (*peopleware*) sebagai pengguna dan pengelola sistem. Interaksi antar komponen tersebut menghasilkan suatu mekanisme kerja yang terstruktur dan efisien dalam mengelola informasi di lingkungan organisasi.

Melalui sistem informasi *monitoring smart parking* berbasis website, proses pengelolaan area parkir menjadi lebih efektif karena petugas dan pengguna dapat memperoleh informasi kapasitas parkir tanpa harus melakukan pemeriksaan manual. Dengan demikian, pengembangan sistem informasi *monitoring smart parking* berbasis website memiliki kontribusi yang signifikan dalam mendukung penerapan konsep smart parking sebagai bagian dari implementasi teknologi *Internet of Things* (IoT) di era digital saat ini.

2.3. Next.Js

Next.js merupakan sebuah *framework* berbasis React yang digunakan untuk membangun aplikasi *website full-stack*. *Framework* ini memanfaatkan komponen React dalam pembuatan antarmuka pengguna (*user interface*) serta menyediakan berbagai fitur dan optimasi tambahan yang mendukung pengembangan aplikasi secara efisien. Selain itu, Next.js secara otomatis melakukan konfigurasi terhadap alat bantu tingkat rendah seperti *bundler* dan *compiler*, sehingga pengembang dapat lebih berfokus pada proses pengembangan fitur dan penyelesaian produk tanpa perlu mengatur konfigurasi teknis yang kompleks. [10] Next.js merupakan *framework* React yang banyak digunakan dalam pengembangan aplikasi website dengan kemampuan *server-side rendering* (SSR).

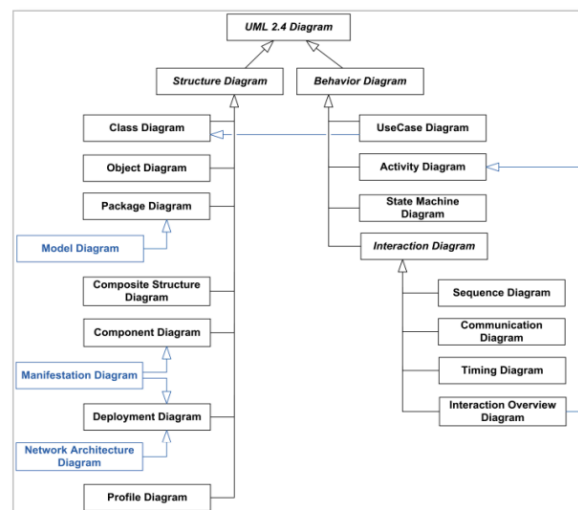
Framework ini dirancang untuk meningkatkan pengalaman pengguna, kinerja aplikasi, serta efisiensi proses pengembangan. Next.js menyediakan berbagai fitur penting yang tidak tersedia secara bawaan pada React, seperti *routing* otomatis, *server-side rendering*, dan *static site generation*. [11] Dukungan terhadap kedua metode tersebut menjadikan Next.js sebagai pilihan yang optimal untuk membangun aplikasi *website* dengan performa tinggi. *Framework* ini memiliki berbagai fitur unggulan seperti *automatic code splitting*, sistem *routing* berbasis berkas yang mudah digunakan, serta mekanisme optimasi kinerja yang terintegrasi. Selain itu, kemampuan Next.js untuk berintegrasi dengan berbagai alat dan pustaka pendukung menjadikannya solusi yang efektif bagi pengembang dalam menciptakan aplikasi *website* yang cepat, efisien, dan mudah dikembangkan. [12]

2.4. Tailwind

Tailwind CSS merupakan sebuah *utility first framework* yang dikembangkan oleh Adam Wathan dan pertama kali diperkenalkan pada akhir tahun 2017. *Framework* ini memperoleh popularitas yang cukup tinggi di kalangan pengembang, baik pada bidang

front-end maupun *back-end*. Kemudahan dalam penerapannya menjadikan Tailwind CSS banyak digunakan oleh pengembang *front-end* yang tidak memiliki keahlian mendalam dalam penulisan CSS konvensional. Tailwind CSS umumnya diintegrasikan bersama *framework* lain, seperti Laravel pada sisi *back-end*, maupun *framework* JavaScript modern seperti React dan Vue pada sisi *front-end*, sehingga menjadikannya pilihan yang fleksibel dalam pengembangan antarmuka aplikasi berbasis *website*. [13] Pendekatan *utility first* pada Tailwind CSS merupakan inovasi dalam pengembangan frontend yang memberikan fleksibilitas tinggi bagi pengembang untuk menyesuaikan desain sesuai kebutuhan proyek. Melalui penggunaan kelas-kelas fungsional, *framework* ini memungkinkan pembuatan antarmuka secara efisien tanpa bergantung pada aturan dari framework konvensional. Selain itu, Tailwind CSS berpengaruh terhadap skalabilitas, optimalisasi kinerja, dan konsistensi desain pada aplikasi *website*. Dengan fleksibilitas dan kemudahan kustomisasi yang dimilikinya, Tailwind CSS dinilai mampu meningkatkan efisiensi serta kualitas dalam proses perancangan dan pengembangan *website* modern. [14] Setiap *utility* pada Tailwind CSS memiliki tipe responsif yang memudahkan pengembang membangun antarmuka adaptif tanpa perlu menulis perintah CSS secara manual.

2.5. Unified Model Language (UML)



Gambar 3. Struktur *Unified Model Language* [15]

Unified Modeling Language (UML) adalah bahasa spesifikasi standar yang digunakan untuk mendokumentasikan, menspesifikasikan, serta membangun perangkat lunak. UML merupakan metodologi yang umum dipakai dalam pengembangan sistem berorientasi objek dan berfungsi sebagai alat bantu dalam proses pengembangan sistem. [16] Beberapa jenis diagram yang digunakan dalam penelitian ini antara lain:

1) *Use Case* Diagram

Use case diagram digunakan untuk memodelkan perilaku (*behavior*) sistem informasi yang akan dikembangkan. Diagram ini membantu mengidentifikasi fungsi-fungsi yang tersedia dalam sistem serta menentukan siapa saja yang memiliki hak untuk menggunakan fungsi-fungsi tersebut. *Use case* diagram menggambarkan interaksi sederhana antara pengguna dengan sistem serta jenis interaksi yang terjadi. Diagram ini sering disebut sebagai “cetakan biru sistem” karena menyajikan representasi grafis paling sederhana mengenai apa yang harus dilakukan oleh sistem. *Use case* diagram menampilkan kebutuhan dasar dari sebuah sistem perangkat lunak.

2) *Activity* Diagram

Activity Diagram pada dasarnya merupakan bentuk visual dari alur kerja (*flowchart*) yang menampilkan rangkaian aktivitas secara bertahap yang dilakukan oleh aktor, termasuk adanya pilihan (*decision*) dan pengulangan (*iteration*). Diagram ini dapat dianggap serupa dengan *flowchart*, namun memiliki keunggulan dalam menggambarkan proses yang berlangsung secara bersamaan (*concurrency*), yang tidak dapat direpresentasikan dengan *flowchart* biasa. Selain itu, *activity diagram* juga menampilkan aspek dinamis dari suatu sistem, yaitu bagaimana proses dan aktivitas saling berinteraksi dalam alur kerja sistem tersebut.

3) *Class* Diagram

Class Diagram merupakan diagram dengan struktur *statis* yang menggambarkan bentuk dan susunan suatu sistem pada tingkat *classifier* seperti *class*, *interface*, dan komponen lainnya. Diagram ini menampilkan berbagai *classifier* yang terdapat dalam sistem, subsistem, atau komponen, beserta hubungan antar *classifier*, atribut yang

dimiliki, operasi yang dilakukan, serta batasan-batasan yang berlaku di dalam sistem tersebut[15]

2.6. *Rapid Application Development (RAD)*

Rapid Application Development (RAD) merupakan suatu model pengembangan perangkat lunak yang bersifat inkremental dengan fokus pada percepatan waktu pengembangan. Model ini merupakan bentuk modifikasi dari model sekuensial linear yang dirancang untuk menghasilkan sistem secara cepat melalui pemanfaatan komponen-komponen yang dapat digunakan kembali. Apabila kebutuhan sistem telah terdefinisi dengan jelas dan cakupan proyek tidak terlalu luas, metode RAD memungkinkan tim pengembang untuk membangun sistem yang dapat berfungsi secara optimal dalam waktu relatif singkat, yaitu sekitar 60 hingga 90 hari. Pendekatan ini umumnya diterapkan pada pengembangan sistem informasi dan terdiri dari beberapa tahapan proses utama.[17] Keunggulan utama dari metode RAD terletak pada kemampuannya mempercepat proses integrasi sistem serta mempersingkat waktu pengembangan aplikasi tanpa mengurangi kualitas hasil akhir. [18]

1) Tahap Perencanaan Kebutuhan (*Requirement Planning*)

Proses ini merupakan bagian penting dalam manajemen proyek yang berfokus pada identifikasi, perumusan, dan pengelolaan kebutuhan proyek agar tujuan yang telah ditetapkan dapat tercapai secara optimal. Tahapan ini memastikan bahwa setiap kebutuhan sistem atau aplikasi terdefinisi dengan jelas sejak awal pengembangan, sehingga meminimalkan risiko kesalahan dan perubahan besar di tahap selanjutnya. Selain itu, proses ini juga berfungsi untuk menyelaraskan pemahaman antara tim pengembang dan pengguna (*stakeholder*), memastikan bahwa hasil akhir proyek benar-benar sesuai dengan harapan dan kebutuhan yang diinginkan. [18]

2) Tahap Desain Pengguna (*User Design*)

Pada tahap ini dilakukan pengembangan model serta pembuatan prototipe sistem yang mencakup keseluruhan komponen, mulai dari input, proses, hingga output. Proses

perancangan dilakukan dengan memanfaatkan *Unified Modeling Language* (UML) sebagai alat bantu utama dalam menggambarkan struktur dan perilaku sistem secara terperinci. UML digunakan untuk memvisualisasikan rancangan sistem agar lebih mudah dipahami oleh pengembang maupun pihak terkait, sehingga setiap elemen sistem dapat terdefinisi dengan jelas sebelum tahap implementasi dilakukan. Tahap ini juga berperan penting dalam memastikan bahwa desain yang dibuat telah sesuai dengan kebutuhan pengguna dan tujuan pengembangan sistem.

3) Tahap Konstruksi (*Construction*)

Pada tahap ini dilakukan proses pengujian dan implementasi sistem berdasarkan prototipe yang telah dikembangkan sebelumnya. Penulis mulai mengonversi desain konseptual menjadi bentuk aplikasi yang fungsional melalui pengembangan kode program. Implementasi ini bertujuan untuk memastikan bahwa seluruh fitur dan alur sistem berjalan sesuai dengan rancangan yang telah dibuat. Selain itu, tahap ini juga melibatkan pengujian fungsionalitas guna memverifikasi bahwa setiap komponen sistem bekerja dengan benar dan memenuhi kebutuhan pengguna. Hasil dari tahap ini menjadi dasar untuk evaluasi lebih lanjut, sehingga sistem dapat disempurnakan sebelum digunakan secara penuh dalam lingkungan operasional.

4) Tahap Implementasi (*Cutover*)

Pada tahap akhir pengembangan, dilakukan serangkaian proses yang berkaitan dengan implementasi sistem secara menyeluruh, meliputi migrasi data, pengujian sistem, transisi dari sistem lama ke sistem baru, serta pelatihan bagi pengguna. Tahap ini bertujuan untuk memastikan bahwa sistem yang dikembangkan dapat beroperasi secara optimal di lingkungan sebenarnya dan mudah digunakan oleh pihak yang terlibat. Proses migrasi data dilakukan dengan hati-hati untuk menjamin integritas serta konsistensi data dari sistem sebelumnya. Sementara itu, pengujian sistem berfungsi untuk memastikan seluruh komponen telah berjalan sesuai spesifikasi dan kebutuhan pengguna.[19] Dibandingkan dengan metode pengembangan tradisional, metode *Rapid Application Development* (RAD) memungkinkan seluruh tahapan

implementasi ini dilakukan dengan lebih cepat dan efisien, tanpa mengorbankan kualitas sistem.

2.7. *Black Box Testing*

Black box testing merupakan metode pengujian perangkat lunak yang dilakukan tanpa memerlukan pengetahuan terhadap struktur internal sistem atau kode sumber aplikasi. Pengujian ini berfokus pada hasil keluaran (*output*) yang dihasilkan berdasarkan masukan (*input*) tertentu, tanpa menelusuri proses internal yang terjadi di dalam aplikasi. Dalam proses *black box testing*, pengujian dilakukan dengan memberikan berbagai variasi *input* untuk memastikan bahwa sistem berfungsi sesuai dengan kebutuhan dan harapan pengguna maupun pihak terkait. Pendekatan ini digunakan untuk menilai apakah aplikasi telah memenuhi spesifikasi fungsional yang telah ditetapkan secara menyeluruh. [20]



Gambar 4. Tahapan *Black Box Testing* [21]

Equivalence Partitioning merupakan salah satu teknik dalam *Black Box Testing* yang digunakan dengan cara membagi data uji menjadi beberapa kelompok atau partisi berdasarkan kesamaan karakteristik. Melalui teknik ini, proses pengujian menjadi lebih efisien karena cukup dilakukan satu kasus uji dari setiap partisi untuk mewakili keseluruhan kelompok data. Pendekatan ini sangat bermanfaat dalam pengujian aplikasi berbasis *website*, khususnya pada skenario yang melibatkan input pengguna seperti proses *login* dan registrasi. Dengan menerapkan *Equivalence Partitioning*,

penguji dapat menentukan batas validasi masukan (*input validation limits*) yang berpotensi menjadi sumber kerentanan keamanan maupun kesalahan fungsional pada sistem. [22]

Black box Testing dengan pendekatan *Equivalence Partitioning*. Tahapan pertama dimulai dengan pembuatan *test case*, yaitu proses merancang skenario uji berdasarkan kebutuhan fungsional sistem untuk memastikan setiap fitur diuji secara menyeluruh. Selanjutnya, dilakukan penentuan partisi setara (*Equivalence Partitions*), di mana data uji dibagi ke dalam beberapa kelompok dengan karakteristik serupa. Dari setiap partisi, cukup diambil satu contoh data untuk diuji sehingga pengujian menjadi lebih efisien tanpa harus menguji seluruh kemungkinan input. Tahap berikutnya adalah implementasi pengujian, yaitu menjalankan *test case* dengan memasukkan data uji ke dalam sistem untuk memverifikasi apakah hasil keluaran sudah sesuai dengan spesifikasi yang diharapkan. Terakhir, dilakukan analisis hasil pengujian guna mengevaluasi apakah sistem telah berfungsi dengan benar. Jika ditemukan ketidaksesuaian, maka dilakukan perbaikan dan pengujian ulang hingga sistem dinyatakan berjalan sesuai kebutuhan. Dengan demikian, Gambar 4. tersebut menggambarkan alur pengujian *Black box Testing* secara sistematis, mulai dari perencanaan hingga analisis hasil, guna memastikan kualitas dan keandalan perangkat lunak.

2.8. *System Usability Scale*

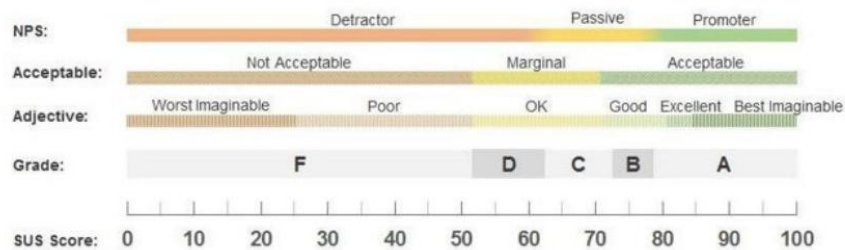
System Usability Scale (SUS) merupakan salah satu metode evaluasi kebergunaan sistem yang bersifat sederhana namun efektif. Metode ini menggunakan sepuluh butir pernyataan yang dirancang untuk memberikan gambaran menyeluruh mengenai tingkat *usability* suatu sistem. Instrumen SUS menggunakan skala *Likert*, di mana responden diminta menentukan tingkat persetujuan atau ketidaksetujuan terhadap setiap pernyataan, biasanya dalam rentang 5 hingga 7 poin. SUS dikenal sebagai alat ukur yang andal, hemat biaya, serta dapat digunakan untuk menilai *usability* secara

keseluruhan, baik pada produk perangkat lunak, aplikasi, maupun sistem interaktif lainnya. Metode ini banyak dipilih karena proses pengukurannya yang cepat, mudah dipahami, dan menghasilkan skor yang dapat dibandingkan secara standar. [23]

Tabel 1 Pertanyaan System Usability Scale [24]

No.	Pertanyaan
1.	Saya berpikir akan menggunakan sistem ini lagi.
2.	Saya merasa sistem ini rumit untuk digunakan.
3.	Saya merasa sistem ini mudah untuk digunakan.
4.	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.
5.	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.
6.	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada sistem ini.
7.	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat.
8.	Saya merasa sistem ini membingungkan.
9.	Saya merasa tidak ada hambatan dalam menggunakan sistem ini.
10.	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.

Data yang dikumpulkan melalui kuesioner selanjutnya diolah menggunakan metode *System Usability Scale* (SUS). Nilai yang diperoleh dari skala *Likert* akan dianalisis untuk mengukur tingkat kebergunaan aplikasi.



Gambar 5. Kategori Penilaian System Usability Scale [25]

Secara keseluruhan, penggunaan SUS memberikan metode evaluasi *usability* yang praktis dan terstruktur. Pendekatan ini membantu menciptakan komunikasi yang lebih mudah antara pengguna, pengembang, dan pemangku kepentingan, sekaligus menyediakan landasan yang jelas untuk melakukan perbaikan antarmuka dan peningkatan pengalaman pengguna. Dengan memasukkan hasil evaluasi SUS ke dalam proses pengembangan, sistem atau aplikasi yang dihasilkan diharapkan dapat lebih selaras dengan kebutuhan serta preferensi pengguna.

2.9. Penelitian Terkait

Tabel 2. Penelitian Terkait

No	Judul/ Peneliti (Tahun)	Framework/ Tools	Object Penelitian	Metode	Hasil
1.	Sistem Cerdas Reservasi dan Pemantauan Parkir pada Lokasi Kampus Berbasis Konsep <i>Internet of Things</i> Dodi Ichwana et al. (2018) [26]	Android Studio, Arduino IDE, PHP, MySQL	Sistem Cerdas Reservasi dan Pemantauan Parkir pada Lokasi Kampus	<i>black box testing</i>	Sistem dapat mendeteksi slot parkir kosong serta menyediakan pemantauan parkir di area kampus.
2.	Sistem Tempat Parkir Terintegrasi yang Dilengkapi dengan Aplikasi <i>Mobile</i> dan Mikrokontroller Kenji Andrean et al. (2020) [27]	Laravel	Tempat parkir di Indonesia khususnya kota Surabaya.	<i>Acceptance test</i>	Sistem ini memudahkan pemesanan dan pembayaran parkir yang menangani permasalahan penumpukan atau antrian parkir.

Tabel 2. Penelitian Terkait

No	Judul/ Peneliti (Tahun)	Framework/ Tools	Object Penelitian	Metode	Hasil
3.	Penerapan Sistem <i>Monitoring</i> Parkir Kendaraan Berbasis Android Pada Perguruan Tinggi Raharja Ary Budi Warsinto et al. (2017) [28]	Android Studio	Sistem parkir pada Perguruan Tinggi Raharja	SDLC	Mempermudah petugas parkir untuk menginput data kendaraan masuk dan keluar
4.	MoParking: Sistem <i>Monitoring</i> Parkiran Mobil Berbasis IoT Agrifaldy Rahmatillah et al. (2022) [29]	Android Studio	Sistem monitoring parkiran mobil berbasis IoT	<i>Agile Development</i> dan <i>BlackBox Testing</i>	Penerapan sistem IoT tersebut dapat mengoptimalkan tempat parkir dan efisiensi waktu parkir.
5.	Park-O: <i>A Smart Iot-Based Parking</i>	Next.js dan Tailwind	Sistem Manajemen Parkir Cerdas	<i>BlackBox Testing</i>	Sistem parkir berbasis IoT ini membuat

Tabel 2. Penelitian Terkait

No	Judul/ Peneliti (Tahun)	Framework/ Tools	Object Penelitian	Metode	Hasil
	<i>Management System With Real-Time Slot Allocation And Qr-Gate Automation</i> Kartik Radhesyam Yadav et al. (2025) [30]		Berbasis IoT dengan Alokasi Slot Secara <i>Real-Time</i> dan Otomasi Gerbang QR		proses parkir lebih cepat sehingga lebih unggul dari metode parkir konvensional.
6.	<i>A Decentralized Web Application for Parking Spot Booking and Listing: Improving Convenience and Efficiency in Urban Areas</i> Shigaonkar et al. (2023) [31]	HTML, CSS, JavaScript, Node.js	Sistem pemesanan parkir berbasis web terdesentralisasi	<i>Blockchain</i>	Mengatasi keterbatasan lahan parkir di kawasan padat dengan memudahkan pengemudi mencari dan memesan tempat parkir
7.	Penggunaan RFID Sistem	MIFARE <i>Reader</i>	Implementasi sistem parkir	RFID	meningkatkan efisiensi

Tabel 2. Penelitian Terkait

No	Judul/ Peneliti (Tahun)	Framework/ Tools	Object Penelitian	Metode	Hasil
	Informasi Parkir Berbasis Web Muhammad Abdullah (2024) [32]		dengan Mifare <i>Reader</i> di kampus X		manajemen parkir roda dua di lingkungan kampus.
8.	<i>Smart Parking System Based on Internet of Things</i> Mangnawi Poonam (2018) [33]	HTML, CSS, JavaScript, AJAX, PHP, Bootstrap, SQL	Tempat parkir yang masih menggunakan metode konvensional	<i>Internet of Things</i>	Menyediakan informasi ketersediaan slot parkir secara real-time sehingga dapat mengetahui jumlah yang tersedia
9.	<i>Smart Parking: Real-Time Parking Availability System</i> Sudheer Kumar C et al. (2025) [34]	MongoDB, Express.js, React.js, Node.js	<i>Real-Time Parking Availability System</i> untuk pengelolaan parkir di kawasan perkotaan.	<i>Agile Development</i> Dan <i>Design Thinking</i>	Berhasil meningkatkan efisiensi pengelolaan parkir dengan informasi ketersediaan slot secara real-time.

Tabel 2. Penelitian Terkait

No	Judul/ Peneliti (Tahun)	Framework/ Tools	Object Penelitian	Metode	Hasil
10.	Perancangan Sistem Monitoring Ruang Kuliah Berbasis Nfc Dan Iot (Studi Kasus: Universitas Negeri Surabaya) Krisna Gilang Prakoso et al. (2024) [35]	Laravel	sistem <i>monitoring</i> ruangan kuliah berbasis Near Field Communication (NFC) dan Internet of Things (IoT)	<i>Rapid Application Development</i> (RAD), <i>Blackbox testing</i> , dan <i>White box testing</i>	Mampu mengelola kehadiran dan pemanfaatan ruang kuliah secara akurat dan efisien, sehingga dapat menjadi model pengelolaan ruang bagi perguruan tinggi lainnya.

Setiap penelitian pada bidang sistem parkir cerdas dan *monitoring* berbasis IoT maupun *website* menunjukkan keterkaitan yang kuat antara pemilihan teknologi, metode pengembangan, objek implementasi, serta hasil yang dicapai. *Framework* dan *tools* seperti Android Studio, Arduino IDE, Laravel, Next.js, Node.js, hingga platform berbasis IoT, RFID, dan NFC digunakan sesuai kebutuhan sistem dan karakteristik lingkungan penelitian. Misalnya, aplikasi reservasi parkir berbasis Android yang terintegrasi dengan sensor IoT memanfaatkan Arduino untuk mendeteksi ketersediaan slot parkir secara otomatis, sedangkan penelitian yang berfokus pada aplikasi *website real-time* memilih Next.js untuk menghasilkan pembaruan data yang cepat dan responsif.

Metode yang digunakan pada masing-masing penelitian meliputi *Black Box Testing*, *Acceptance Test*, SDLC, *Agile Development*, RAD, *Blockchain*, serta kombinasi pengujian *white-box* dan *black-box* diterapkan untuk memastikan kualitas sistem sesuai tujuan pengembangan. Metode pengujian seperti *Black Box Testing* berfungsi memvalidasi fitur inti, termasuk *login*, *logout*, CRUD, dan reservasi, sementara pendekatan pengembangan iteratif seperti *Agile* dan RAD memungkinkan penyempurnaan produk secara berkelanjutan. Di sisi lain, penggunaan *Blockchain*, RFID, dan NFC menunjukkan upaya peningkatan keamanan, akurasi identifikasi, serta otomatisasi proses manajemen aset dan ruang.

Objek penelitian yang beragam mulai dari area kampus, ruang kelas, kawasan perkotaan, hingga sistem parkir terdesentralisasi menentukan bentuk rancangan sistem serta pemilihan teknologi yang digunakan. Pada konteks perguruan tinggi, misalnya, Android dan RFID dipilih karena mendukung operasional petugas secara praktis, sedangkan sistem yang diterapkan di wilayah perkotaan menuntut integrasi IoT dan teknologi *web real-time* untuk menangani volume kendaraan yang lebih besar.

Secara keseluruhan, hasil penelitian menunjukkan bahwa pemilihan framework yang tepat, penerapan metode pengembangan yang sesuai, serta pengujian sistematis memberikan dampak signifikan terhadap efisiensi manajemen parkir, serta kualitas pengalaman pengguna.

III. METODOLOGI PENELITIAN

3.1 Waktu dan Tempat

Penelitian ini dilaksanakan pada:

1. Waktu Penelitian : Desember 2025-April 2026
2. Tempat Pelaksanaan : Laboratorium Komputer, Teknik Elektro Universitas Lampung

Tabel 3. Waktu Pelaksanaan Penelitian

No.	Aktivitas	Desember				Januari				Februari				Maret				April			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	Studi Literatur																				
2	Requirements Planning																				
	Planning (Wawancara, Identifikasi Kebutuhan)																				
	Analysis (Analisis Sistem, Kebutuhan Fungsionalitas)																				
3	User Design (Perancangan UML dan Arsitektur Database)																				
4	Construction																				
	Coding																				
	Testing (pengujian Performa, iterasi)																				

Tabel 3. Waktu Pelaksanaan Penelitian

No.	Aktivitas	Desember				Januari				Februari				Maret				April			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
5	Cutover (implementasi Sistem)																				
6	Penulisan Laporan																				

3.2 Tim Penelitian

Penelitian ini dilaksanakan oleh sebuah tim yang terdiri dari 5 anggota, di mana masing-masing memiliki peran yang berbeda, meliputi pengembang *front-end* dan pengembang *back-end*.

Tabel 4 Tim Penelitian

No	Nama	Role	Deskripsi Pekerjaan
1.	Fannisa Nurhaffifi	<i>Front-end</i>	1) Mengirim <i>request</i> ke <i>back-end</i> untuk memperoleh data yang dibutuhkan, seperti informasi slot parkir, status kendaraan. 2) Menerima <i>response</i> dari <i>database</i> lalu menampilkannya dalam bentuk yang mudah dipahami pengguna. 3) Berfungsi sebagai media visual dari sistem, misalnya data IoT, laporan parkir, hingga grafik akumulasi kendaraan.
2.	Alih Bangun Wicaksono	<i>Back-end</i>	1) Penghubung antara data IoT dengan <i>database</i> . <i>Back-end</i> menerima <i>request</i> dari perangkat

Tabel 4 Tim Penelitian

No	Nama	Role	Deskripsi Pekerjaan
			<p>IoT, lalu memproses data seperti status slot parkir.</p> <p>2) Mengelola <i>request</i> dan <i>response</i> antara <i>database</i> dan <i>front-end</i>. <i>Back-end</i> memastikan data yang dikirim ke <i>front-end</i> sudah melalui proses validasi dan pengolahan.</p> <p>3) Menyimpan, memperbarui, dan mengolah data di dalam <i>database</i>, termasuk data <i>real-time</i> dari IoT.</p>
3.	Tim IoT (3 orang)	IoT	<p>Tim penelitian IoT merancang dan memasang sensor <i>untuk</i> mendeteksi keberadaan kendaraan, menghubungkannya melalui perangkat IoT yang mengirimkan data ke <i>database</i>, sehingga sistem dapat menghitung kuota parkir dan menampilkan informasi ketersediaan lahan secara <i>real-time</i> melalui <i>front-end</i>.</p>

3.3 Alat dan Bahan Penelitian

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

Tabel 5 Alat dan Bahan

No	Perangkat	Spesifikasi	Kegunaan
1.	Laptop	Asus Vivobook K513EA, Windows 11, RAM 8GB	Perangkat utama untuk pengembangan sistem
2.	Wireframe	Figma	Digunakan untuk merancang tampilan antarmuka sebelum pengembangan.
3.	Diagram Tools	Draw.io	Digunakan untuk membuat <i>flowchart</i> , <i>use case</i> , dan <i>activity diagram</i> .
4.	<i>Framework Front-End</i>	Next.js Version 16	<i>Framework</i> Next.js yang mendukung SSR dan SSG
5.	<i>Framework CSS Tailwind</i>	Tailwind CSS V4	Digunakan untuk merancang tampilan yang responsif dan konsisten
6.	<i>Code Editor</i>	Visual Studio Code	Editor utama untuk menulis kode Next.js & Tailwind CSS
7.	<i>Version Control System</i>	Git dan Github	Mengelola versi kode dan kolaborasi
8.	Browser	Google Chrome	Melakukan uji coba tampilan dan responsivitas sistem di berbagai resolusi layar.

Tabel 5 Alat dan Bahan

No	Perangkat	Spesifikasi	Kegunaan
1.	GET	/api/admin/dashboard/summary	Mengambil ringkasan data beranda (jumlah slot, terisi, tersedia).
2.	GET	/api/admin/parkir	Mengambil daftar riwayat kendaraan masuk/keluar untuk tabel utama.
3.	GET	/api/admin/pengguna	Mengambil daftar seluruh mahasiswa yang terdaftar dalam sistem.
4.	DELETE	/api/admin/pengguna/[npm]	Menghapus data akun mahasiswa berdasarkan NPM tertentu.
5.	PUT	/api/admin/pengguna/status	Memperbarui status akun (Aktifkan/Verifikasi, batasi, atau Tolak).
6.	GET	/api/admin/export/csv	Menghasilkan file laporan berformat CSV (mendukung filter periode).
7.	GET	/api/admin/export/excel	Menghasilkan file laporan berformat Excel (mendukung filter periode).
8.	GET	/api/admin/export/pdf	Endpoint jembatan untuk mengunduh laporan PDF.
9.	POST	/api/admin/rfid/generate	Membuat/mendaftarkan kode kartu RFID baru untuk kendaraan.
10.	PUT	/api/admin/slot	Memperbarui jumlah total kapasitas (slot) parkir yang tersedia.

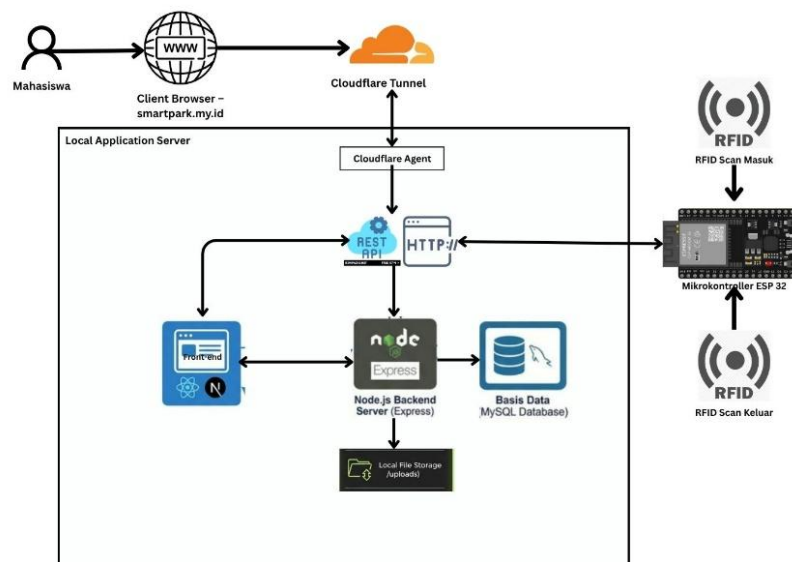
Tabel 5 Alat dan Bahan

No	Perangkat	Spesifikasi	Kegunaan
11.	PUT	/api/admin/kuota	Memperbarui jumlah kesempatan/kuota parkir untuk mahasiswa.
12.	POST	/api/auth/login	Verifikasi nama/NPM dan kata sandi untuk masuk ke dalam sistem.
13.	POST	/api/auth/logout	Menghapus sesi masuk dan keluar dari sistem.
14.	POST	/api/auth/register	Pendaftaran mahasiswa baru (input data diri dan <i>upload</i> STNK).
15.	POST	/api/auth/request-otp	Mengirimkan kode OTP ke email untuk verifikasi/lupa kata sandi.
16.	POST	/api/auth/verify-otp	Memvalidasi kode OTP yang dimasukkan oleh pengguna.
17.	POST	/api/auth/reset-password	Memperbarui kata sandi lama dengan kata sandi baru setelah verifikasi OTP.
18.	GET	/api/users/profile	Mengambil data detail profil mahasiswa yang sedang masuk.
19.	PUT	/api/users/change-password	Mengubah kata sandi akun mahasiswa dari profil.
20	GET	/api/users/riwayat/[npm]	Mengambil daftar sejarah parkir pribadi mahasiswa tertentu.

Tabel 5 Alat dan Bahan

No	Perangkat	Spesifikasi	Kegunaan
21.	GET	/api/statistik/kendaraan	Mengambil data mentah untuk kebutuhan grafik statistik kendaraan.
22.	GET	/api/statcard/parkir	Mengambil data ringkasan cepat untuk kartu statistik.

3.4 Arsitektur Sistem



Gambar 6 Arsitektur Sistem

Berdasarkan Gambar 6. Pada penelitian ini, bagian *front-end* sistem informasi *monitoring smart parking* dikembangkan menggunakan *framework* Next.js yang berbasis React. Next.js dipilih karena mendukung pengembangan aplikasi *website* modern dengan struktur yang terorganisir, performa yang baik, serta kemudahan dalam pengelolaan *routing* dan integrasi API. Dengan menggunakan Next.js, proses pengembangan antarmuka menjadi lebih sistematis karena telah tersedia mekanisme *routing* otomatis (*App Router*) dan dukungan komponen berbasis *client* maupun server. *Front-end* berfungsi sebagai antarmuka utama yang digunakan oleh pengguna untuk

berinteraksi dengan sistem. Melalui halaman beranda, pengguna dapat melihat informasi ketersediaan slot parkir, data kendaraan, riwayat parkir, serta melakukan proses pemesanan slot.

Sistem *Smart Parking* yang dikembangkan merupakan sistem terintegrasi yang menghubungkan perangkat IoT, *back-end*, *database*, dan *front-end* dalam satu arsitektur berbasis *website*. Proses dimulai ketika pengguna mengakses sistem melalui browser dengan domain *smartpark.my.id*, kemudian permintaan tersebut diteruskan melalui *Cloudflare Tunnel* sebagai penghubung antara jaringan publik dan *server* lokal. Selanjutnya, permintaan akan diproses oleh *server* aplikasi lokal yang menjalankan Next.js sebagai *front-end* sekaligus proxy layer, serta Node.js dengan *framework* Express sebagai backend yang menyediakan layanan REST API dan komunikasi *real time* menggunakan WebSocket. Pada sisi IoT, sistem memanfaatkan mikrokontroler ESP32 yang terintegrasi dengan RFID untuk melakukan proses identifikasi kendaraan. Ketika pengguna melakukan scan kartu RFID pada pintu masuk atau keluar, data hasil pemindaian akan dikirimkan oleh ESP32 ke back-end melalui protokol HTTP.

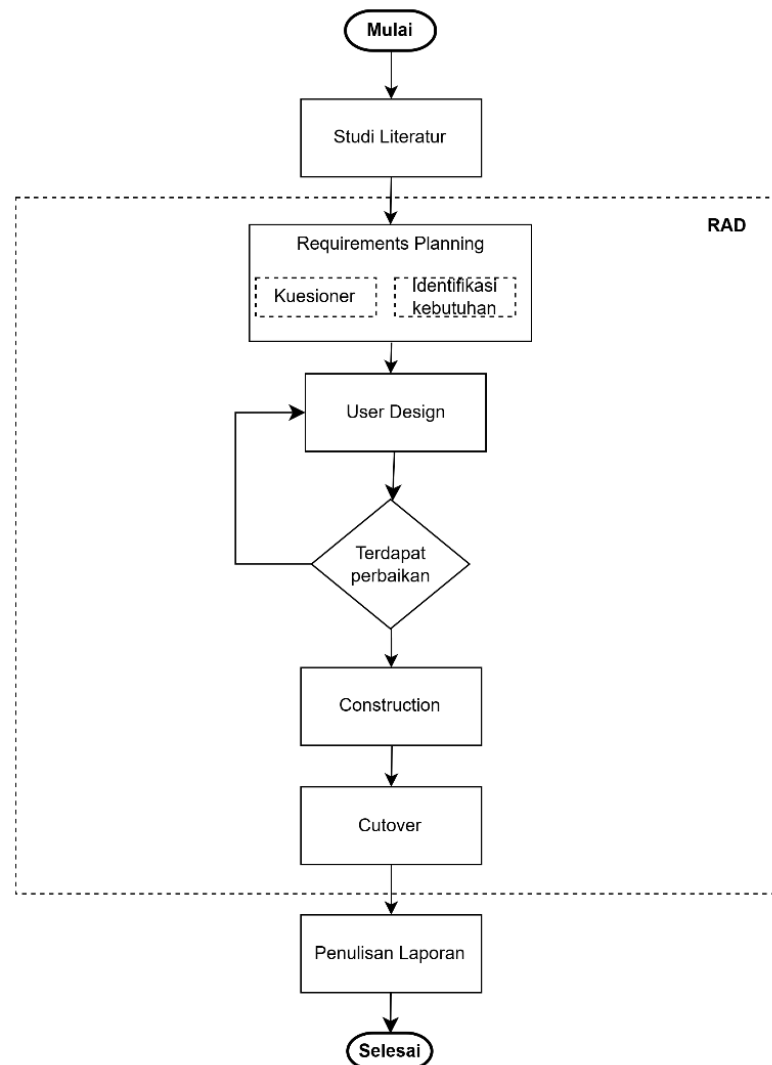
Secara umum, cara kerja *front-end* dimulai ketika pengguna mengakses aplikasi melalui *browser*, kemudian menampilkan antarmuka sesuai dengan rute yang diakses. Ketika pengguna melakukan suatu aksi, seperti masuk, melakukan pemesanan slot parkir, atau memperbarui data profil, *front-end* akan menangkap input tersebut melalui *event handler* yang terdapat pada komponen React. Data yang diinputkan kemudian dikirimkan ke *back-end* menggunakan metode HTTP melalui REST API. Proses komunikasi data dilakukan dengan mengirimkan *request* ke Next.js API *Routes* yang berfungsi sebagai perantara sebelum diteruskan ke server *back-end*. Setelah *back-end* memproses permintaan dan mengakses *database* MySQL, hasil respons akan dikirim kembali ke *front-end* dalam bentuk data JSON. Selain komunikasi berbasis REST API, *front-end* juga mengimplementasikan Socket.io *client* untuk menerima pembaruan data secara *real-time*.

Ketika terjadi perubahan status slot parkir atau hasil pemindaian kartu RFID pada sistem, backend kemudian melakukan proses validasi data, menentukan status kendaraan (masuk atau keluar), serta memperbarui informasi kuota parkir yang tersedia. Data tersebut selanjutnya disimpan ke dalam *database* MySQL sebagai basis data utama yang menyimpan informasi pengguna, log parkir, serta status slot parkir.

Setelah proses pengolahan data selesai, backend akan mengirimkan pembaruan informasi secara *real time* ke *front-end* menggunakan teknologi WebSocket. Hal ini memungkinkan tampilan pada sisi pengguna, seperti dashboard monitoring, dapat secara langsung memperbarui informasi jumlah slot parkir, status kendaraan, serta statistik penggunaan parkir tanpa perlu melakukan refresh halaman. Selain itu, *front-end* juga dapat melakukan permintaan data ke *back-end* melalui REST API untuk menampilkan informasi yang tersimpan di database. Dengan adanya integrasi antara alur komunikasi sistem dan alur data tersebut, sistem *Smart Parking* mampu memberikan layanan monitoring parkir secara real-time, akurat, dan terpusat. Arsitektur ini tidak hanya meningkatkan efisiensi pengelolaan parkir, tetapi juga memberikan kemudahan bagi pengguna dalam memperoleh informasi ketersediaan lahan parkir secara cepat dan responsif.

3.5 Tahapan Penelitian

Penelitian ini menggunakan metode *Rapid Application Development* (RAD) sebagai pendekatan pengembangan sistem. Metode RAD dipilih karena mampu mempercepat proses pembuatan aplikasi melalui tahapan yang bersifat iteratif dan terstruktur. Adapun proses pengembangan sistem sebagai berikut:



Gambar 7. Tahapan Penelitian

3.5.1 *Requirements Planning*

Tahap awal penelitian ini adalah *Requirements Planning* (Perencanaan Kebutuhan), yang berfokus pada proses pengumpulan data serta penentuan kebutuhan sistem. Pada tahap ini dilakukan identifikasi tujuan pengembangan, analisis permasalahan yang ada, serta penentuan kebutuhan pengguna. Proses pengumpulan data dilakukan melalui beberapa metode berikut:

3.5.1.1 Kuesioner

Tahap *Requirements Planning* dilakukan pengumpulan informasi awal untuk memahami permasalahan dan kebutuhan sistem yang akan dikembangkan. Kegiatan utama pada tahap ini meliputi kuesioner dengan pihak terkait, seperti pengelola dan pengguna sistem, guna memperoleh gambaran proses yang berjalan serta kendala yang dihadapi. Selain itu, dilakukan identifikasi kebutuhan pengguna untuk menentukan fungsi-fungsi utama yang harus tersedia dalam sistem, sehingga ruang lingkup pengembangan dapat ditetapkan secara jelas.

3.5.1.2 Identifikasi Kebutuhan

Tahap identifikasi kebutuhan bertujuan untuk menganalisis sistem yang sedang berjalan dan merumuskan kebutuhan sistem secara lebih rinci. Pada tahap ini dilakukan analisis terhadap alur kerja sistem eksisting untuk mengidentifikasi kelemahan dan peluang perbaikan. Selanjutnya, ditentukan kebutuhan fungsional sistem yang akan dikembangkan, meliputi fitur-fitur yang dibutuhkan pengguna agar sistem dapat berjalan sesuai dengan tujuan yang telah ditetapkan. Hasil dari tahap ini menjadi dasar dalam proses perancangan sistem pada tahap berikutnya.

Dalam perancangan sistem informasi *monitoring smart parking* ini, terdapat dua kelompok pengguna utama yang menjadi acuan, yaitu Admin dan Mahasiswa. Kedua kelompok ini memiliki peran, kebutuhan, serta permasalahan yang berbeda sehingga perlu dipertimbangkan dalam pengembangan sistem. Admin merupakan pihak yang bertanggung jawab dalam pengelolaan dan pengawasan sistem parkir secara keseluruhan. Admin umumnya berada pada rentang usia 30 hingga 50 tahun dan memiliki tugas utama mengelola data keluar dan masuk kendaraan, data mahasiswa, serta memastikan ketertiban area parkir. Permasalahan yang sering dihadapi oleh admin adalah keterbatasan dalam memantau kondisi parkir secara langsung, terutama terkait jumlah slot parkir yang masih tersedia. Oleh karena itu, admin membutuhkan sebuah sistem yang menyediakan beranda parkir yang informatif, mudah dipahami, serta dilengkapi dengan laporan penggunaan parkir yang dapat digunakan sebagai bahan

evaluasi. Selain itu, sistem yang stabil dan mudah digunakan menjadi kebutuhan utama agar proses pengelolaan dapat berjalan secara efektif.

Sementara itu, Mahasiswa sebagai pengguna parkir merupakan mahasiswa Jurusan Teknik Geodesi minimal semester tiga dengan rentang usia antara 18 hingga 25 tahun. Mahasiswa menggunakan sistem parkir untuk mendukung aktivitas perkuliahan sehari-hari. Permasalahan yang sering dialami adalah kondisi parkir yang tidak teratur pada saat jam sibuk, sehingga waktu banyak terbuang untuk mencari slot parkir yang kosong. Oleh karena itu, mahasiswa membutuhkan informasi ketersediaan parkir yang jelas dan dapat diakses dengan mudah sebelum memasuki area parkir. Tampilan website yang sederhana, responsif, serta penyajian data parkir secara *real-time* menjadi hal penting agar sistem dapat membantu mahasiswa dalam merencanakan aktivitas parkir dengan lebih efisien.

3.5.2 User Design

Setelah kebutuhan pengguna berhasil diidentifikasi, tahap berikutnya adalah melakukan perancangan sistem guna memvisualisasikan alur kerja, struktur, dan tampilan antarmuka yang akan dikembangkan. Pada tahap ini, dibuat beberapa model perancangan antara lain *use case* diagram, *activity* diagram, serta *wireframe* antarmuka. Seluruh rancangan tersebut berfungsi sebagai representasi awal dari sistem informasi *monitoring smart parking* sebelum masuk ke tahap implementasi.

3.5.2.1 Use Case Diagram

Perancangan *use case* diagram dilakukan untuk memetakan interaksi antara aktor dan sistem yang akan dibangun. Diagram ini memberikan gambaran menyeluruh mengenai alur proses serta fungsi utama yang dapat dijalankan oleh pengguna. Tahap awal perancangan dimulai dengan mengidentifikasi aktor yang terlibat, sehingga peran masing-masing aktor serta relasinya terhadap sistem dapat dipahami dengan jelas.

3.5.2.2 Activity Diagram

Perancangan *activity diagram* dilakukan untuk menggambarkan alur aktivitas dan proses kerja dalam sistem yang akan dibangun. Diagram ini memberikan gambaran urutan aktivitas yang terjadi, mulai dari proses awal hingga akhir, serta menunjukkan keputusan dan kemungkinan alur yang dapat terjadi dalam sistem. Perancangan *activity diagram* diawali dengan mengidentifikasi aktivitas utama yang dilakukan oleh pengguna dan sistem, sehingga alur proses dapat dipahami secara jelas dan sistematis. Melalui *activity diagram*, hubungan antar aktivitas serta aliran proses dalam sistem dapat divisualisasikan dengan baik.

3.5.2.3 Wireframe

Wireframe digunakan sebagai gambaran awal sistem sebelum tahap pengembangan dilakukan. *Wireframe* berfungsi untuk menunjukkan tata letak antarmuka, alur navigasi, dan interaksi pengguna secara umum, sehingga memudahkan pemahaman terhadap struktur dan fungsi sistem. Selain itu, *wireframe* membantu meminimalkan kesalahan desain dengan memungkinkan evaluasi dan penyesuaian sejak tahap awal..

3.5.3 Construction

3.5.3.1 Pengembangan Sistem

Pada tahap pengembangan sistem dilakukan dengan menerjemahkan desain yang telah dirancang ke dalam kode program. Sistem dikembangkan dalam bentuk aplikasi berbasis *website* menggunakan JavaScript dengan *framework* Next.js. Proses ini mencakup pembangunan komponen antarmuka, pengintegrasian data melalui API, serta penyesuaian fitur agar sesuai dengan kebutuhan yang telah ditentukan. Fokus utama dalam tahap ini adalah membangun antarmuka *front-end* yang interaktif, responsif, dan dinamis sesuai dengan rancangan pada tahap sebelumnya. Sementara itu, pada tahap *design*, tahapan ini mencakup pembuatan *wireframe*, serta prototipe antarmuka yang kemudian divalidasi melalui umpan balik langsung dari pengguna.

Pendekatan ini memungkinkan penyesuaian desain secara iteratif berdasarkan hasil evaluasi pengguna, sehingga menghasilkan rancangan sistem yang tidak hanya fungsional tetapi juga intuitif, mudah digunakan, dan memberikan pengalaman penggunaan yang optimal.

3.5.3.2 Testing

Tahap *Testing* dilakukan untuk memastikan sistem berjalan sesuai kebutuhan fungsional dan mudah digunakan oleh pengguna. Pada penelitian ini, proses pengujian dilakukan dengan dua pendekatan, yaitu *Blackbox Testing* dan *System Usability Scale (SUS)*. Pada *Blackbox Testing*, pengujian difokuskan pada fungsi-fungsi utama sistem tanpa melihat struktur internal kode. Setiap fitur diuji berdasarkan input yang diberikan dan output yang dihasilkan, untuk memastikan bahwa sistem bekerja sesuai dengan spesifikasi yang telah ditetapkan. Pengujian ini meliputi pemeriksaan validasi data, alur proses pada setiap modul, serta respons sistem terhadap kondisi normal maupun kesalahan input. Hasil pengujian *Blackbox* memberikan gambaran apakah setiap fungsi sistem telah berjalan dengan benar dan bebas dari error sebelum digunakan oleh pengguna akhir.

3.5.4 Cutover

3.5.4.1 Implementasi Sistem

Tahap *Cutover* merupakan tahap akhir dalam metode RAD dan dilakukan setelah sistem dinyatakan stabil pada tahap integrasi dan pengujian. Pada tahap ini, sistem informasi *monitoring smart parking* yang dibangun dengan Next.js mulai dipindahkan dari lingkungan pengembangan ke lingkungan penggunaan. Proses ini meliputi *build* aplikasi, konfigurasi *hosting*, pengaturan *domain*, serta penyesuaian *environment variable* agar aplikasi dapat terhubung dengan *back-end* dan perangkat IoT. Selain itu, pada tahap ini juga dilakukan proses *deployment* atau implementasi sistem ke server produksi. Setelah aplikasi berhasil dijalankan pada lingkungan produksi, dilakukan pemeriksaan awal untuk memastikan tidak terdapat *error*, tampilan antarmuka sesuai

dengan rancangan, dan seluruh fitur dapat berfungsi sebagaimana mestinya. Implementasi sistem juga mencakup pengaktifan fitur-fitur utama yang sebelumnya telah melalui proses pengujian pada lingkungan pengembangan, seperti pemanggilan data ketersediaan parkir, pembaruan informasi sensor secara *real-time*, serta mekanisme komunikasi antara aplikasi *front-end* dengan backend. Setelah seluruh komponen dipastikan berjalan dengan baik, sistem mulai diperkenalkan kepada pengguna, disertai dengan proses *monitoring* awal untuk menjamin kestabilan aplikasi ketika digunakan dalam kondisi operasional.

3.5.4.2 Analisis dan Pembahasan

Tahap analisis dan pembahasan merupakan tahap evaluatif yang dilakukan setelah proses iterasi menghasilkan sistem yang stabil dan siap dinyatakan final. Pada tahap ini, dilakukan penilaian menyeluruh terhadap sistem berdasarkan hasil pengujian dan perbaikan yang telah dilakukan pada siklus Analisis berfokus pada pencapaian sistem terhadap kebutuhan yang telah dirumuskan pada tahap perencanaan, baik dari sisi fungsionalitas, performa, maupun aspek *usability*. Pembahasan pada tahap ini juga mencakup identifikasi kelebihan dan keterbatasan sistem. Kelebihan dapat berupa kesesuaian fungsi, kecepatan akses, tampilan antarmuka yang responsif, atau integrasi yang baik dengan perangkat IoT. Sementara keterbatasan dapat meliputi fitur yang belum optimal, ketergantungan pada koneksi jaringan, atau kebutuhan pengembangan lebih lanjut. Bagian ini juga dapat memuat refleksi atas efektivitas metode RAD dalam proses pengembangan, misalnya terkait kecepatan iterasi, keterlibatan pengguna, atau efisiensi perbaikan sistem. Hasil analisis dan pembahasan pada tahap ini menjadi dasar untuk menarik kesimpulan penelitian, memberikan rekomendasi pengembangan lanjutan, serta menentukan tingkat keberhasilan sistem yang telah dibangun.

V. KESIMPULAN DAN SARAN

5.1 Kesimpulan

- 1) Pengembangan frontend Sistem Monitoring Smart Parking berhasil dilakukan dengan menggunakan framework Next.js yang mendukung pembuatan antarmuka pengguna yang responsif, terstruktur, dan mudah digunakan. Implementasi ini memungkinkan sistem menyesuaikan tampilan dan akses berdasarkan peran pengguna, yaitu Admin dan Mahasiswa, sehingga setiap pengguna memperoleh fitur dan informasi sesuai dengan kebutuhannya. Dengan arsitektur yang terorganisir dan integrasi *real-time*, Next.js mampu mendukung proses *monitoring* dan pengelolaan parkir secara efektif dan efisien.
- 2) Penerapan metode Rapid Application Development (RAD) dalam pengembangan sistem dilakukan selama dua bulan melalui dua kali iterasi. Iterasi pertama berfokus pada pembangunan fungsi dasar sesuai kebutuhan fungsional, sedangkan iterasi kedua pada penyempurnaan fitur dan integrasi sistem yaitu menambahkan fitur *scan* kartu RFID untuk *role* mahasiswa dan menambahkan fitur pengaturan kesempatan parkir dan pengaturan slot parkir untuk *role* Admin. Pendekatan iteratif ini memberikan pengalaman pengembangan yang lebih adaptif karena setiap tahap memungkinkan evaluasi dan perbaikan berdasarkan umpan balik pengguna. Dengan demikian, RAD efektif dalam menghasilkan sistem yang lebih sesuai kebutuhan dalam waktu yang relatif singkat. Selain itu, keterlibatan pengguna pada tahap awal melalui kuesioner dan observasi membantu memastikan bahwa sistem yang dibangun sesuai dengan kebutuhan.

- 3) Sistem Smart Parking telah berfungsi dengan baik berdasarkan hasil Black Box Testing dengan tingkat keberhasilan 100% pada seluruh fitur. Selain itu, hasil pengujian System Usability Scale (SUS) memperoleh skor rata-rata 80,25 yang termasuk kategori *Excellent*, sehingga menunjukkan bahwa sistem mudah digunakan dan diterima dengan baik oleh pengguna. Pengujian ini dilakukan dengan target pengguna dari Jurusan Teknik Geodesi, sehingga hasil yang diperoleh merepresentasikan tingkat penerimaan sistem pada lingkungan pengguna yang sesuai dengan implementasi di lapangan.

5.2 Saran

- 1) Sistem informasi *monitoring smart parking* dapat dikembangkan lebih lanjut dengan menambahkan fitur notifikasi otomatis, seperti pemberitahuan sisa kuota parkir atau notifikasi ketika kendaraan berhasil masuk dan keluar area parkir.
- 2) Pengembangan versi aplikasi *mobile* dapat menjadi alternatif agar pengguna dapat mengakses informasi parkir dengan lebih praktis melalui *smartphone*.
- 3) Sistem Smart Parking yang dikembangkan dalam penelitian ini dirancang untuk dapat diterapkan secara langsung pada area parkir Jurusan Teknik Geodesi Fakultas Teknik Universitas Lampung. Namun, pada tahap penelitian ini implementasi nyata belum dapat direalisasikan karena keterbatasan waktu serta kebutuhan koordinasi dengan pihak birokrasi kampus terkait kesiapan infrastruktur pendukung. Oleh sebab itu, penelitian selanjutnya disarankan untuk melakukan implementasi sistem secara langsung agar dapat diuji dalam kondisi operasional yang sesungguhnya.

DAFTAR PUSTAKA

- [1] N. Khairunnisa *Et Al.*, “Pemerataan Pembangunan Infrastruktur Lahan Parkir Di Universitas Pembangunan Nasional ‘Veteran’ Jakarta Menurut Mahasiswa Fakultas Hukum Upnvj 2020-2022,” *Jurnal Ilmiah Wahana Pendidikan*, Vol. 10, No. 11, Pp. 508–514, 2024, Doi: 10.5281/Zenodo.12771312.
- [2] D. I. Putra, R. Aisuwarya, S. Ardopa, And I. Purnama, “Sistem Cerdas Reservasi Dan Pemantauan Parkir Pada Lokasi Kampus Berbasis Konsep Internet Of Things,” *Jurnal Teknologi Dan Sistem Komputer*, Vol. 6, No. 2, Pp. 57–63, Mar. 2018, Doi: 10.14710/Itsiskom.6.2.2018.57-63.
- [3] A. N. Abbood, A. R. I. Ahmed, And H. K. K. Ajam, “Evaluation Of Parking Demand And Future Requirement In The Urban Area,” *Civil Engineering Journal (Iran)*, Vol. 7, No. 11, Pp. 1898–1908, Nov. 2021, Doi: 10.28991/Cej-2021-03091767.
- [4] A. M. Putri *Et Al.*, “Kontribusi Mahasiswa Universitas Negeri Semarang Dalam Mengurangi Polusi Kendaraan Pribadi Menuju Kampus Yang Lebih Ramah Lingkungan,” 2025.
[Online]. Available: [Http://Jurnalilmiah.Org/Journal/Index.Php/Majemuk](http://Jurnalilmiah.Org/Journal/Index.Php/Majemuk)
- [5] D. Hernikawati, “Perbandingan Solusi Parkir Konvensional Dengan Smart Parking The Comparison Of Conventional Parking Solutions With Smart Parking,” *Majalah Semi Ilmiah Populer Komunikasi Massa*, Vol. Hal: 118-130, Dec. 2021.
- [6] I. Gede *Et Al.*, “Evaluasi Dan Redesign Website Menggunakan System Usability Scale Dan Automated Software Testing,” *Jurnal Sains Dan Teknologi*, Vol. 11, Pp. 18–28, 2022, Doi: 10.23887/Jst-Undiksha.V11i1.

- [7] O. M. Suryawinata, *Buku Ajar Mata Kuliah Pengembangan Aplikasi Berbasis Web Diterbitkan Oleh Umsida Press*. 2019.
- [8] A. Basyar Tsani *Et Al.*, “Pembuatan Web Dashboard Interaktif Untuk Analisis Data Penjualan Vending Machine,” *Jurnal Informatika Dan Sains Teknologi*, Vol. 3, No. 2, Pp. 51–61, 2025, Doi: 10.62951/Modem.V3i2.406.
- [9] Santoso *Et Al.*, *Buku Ajar Pengantar Sistem Informasi*, 1st Ed. Bekasi: Pt. Sonpedia Publishing Indonesia, 2025.
- [10] “Next.Js Docs | Next.Js.” Accessed: Nov. 09, 2025. [Online]. Available: <https://nextjs.org/docs>
- [11] V. Koteswara, R. Ballamudi, And H. Desamsetti, “Getting Started Modern Web Development With Next.Js: An Indispensable React Framework,” Vol. 1, No. 1, 2021, [Online]. Available: <https://www.researchgate.net/publication/374291707>
- [12] Harrison Oke Ekpobimi, “Building High-Performance Web Applications With Nextjs,” *Computer Science & It Research Journal*, Vol. 5, No. 8, Pp. 1963–1977, Aug. 2024, Doi: 10.51594/Csitj.V5i8.1459.
- [13] M. Nauval Azhar, *Tailwind Css Sebuah Framework Css Untuk Developer Yang Memiliki Phobia Terhadap Css*, 1st Ed. Malang: Cv. Multimedia Edukasi, 2020.
- [14] N. Dixit, V. Shrivastava, A. Pandey, And E. R. Sharma, “Revolutionizing Web Design With Tailwind Css: A Comprehensive Exploration,” *International Journal Of Research Publication And Reviews*, No. 5, 2024, [Online]. Available: www.ijrpr.com
- [15] I. Akil, *Referensi Dan Panduan Uml 2.0 Singkat Tepat Jelas*. 2021. [Online]. Available: <https://www.researchgate.net/publication/354328679>
- [16] D. Bantuan, S. Setiaji, F. Akbar, A. Abdillah, And J. Fachrizal, “Implementasi Model Unified Modelling Language(Uml) Pada Perancangan Sistem Informasi Data Kependudukan Dan Bantuan Sosial,” *Jurnal Informatika Teknologi Dan Sains*, Vol. 6, No. 3, Pp. 549–58, Jul. 2024, Accessed: Dec. 03, 2025. [Online]. Available: <https://www.jurnal.uts.ac.id/jinteks>
- [17] R. S. Pressman, *Software Engineering*, 9th Ed. Thomas Casson, 2019.

- [18] I. Saputri, R. A. Putri, And H. Santoso, “Design And Build An E-Commerce Application For Building Materials Using The Web-Based Cross Selling Method,” *International Journal Of Information System & Technology*, Vol. 8, No. 158, Pp. 280–293, 2025, Accessed: Nov. 25, 2025. [Online]. Available: <https://ijistech.org/ijistech/index.php/ijistech/article/view/374>
- [19] Y. Ruhayat And R. Nurfalah, “Implementasi Metode Rapid Application Development Dan Pengujian Katalon Pada Aplikasi Laporan Barang Rusak,” *Inti Nusa Mandiri*, Vol. 19, No. 2, Pp. 154–163, Feb. 2025, Doi: 10.33480/Inti.V19i2.6113.
- [20] Dini Nurul Azizah *Et Al.*, “Analysis And Testing Of The Combox Web Application System Using Black Box Testing With The Equivalence Partitioning Method,” *International Journal Of Electrical Engineering, Mathematics And Computer Science*, Vol. 1, No. 4, Pp. 37–43, Nov. 2024, Doi: 10.62951/Ijeemcs.V1i4.118.
- [21] A. R. Apriliandra And I. Nuryasin, “Pengujian Blackbox Pada Website Sistem Pemesanan Travel Online Gemilang Travel Berbasis Teknik Equivalence Partitions,” *Jurnal Teknologi Sistem Informasi Dan Aplikasi*, Vol. 7, No. 2, Pp. 859–867, Apr. 2024, Doi: 10.32493/Jtsi.V7i2.39049.
- [22] J. Nadhifah *Et Al.*, “Black Box Testing On The Wingpos Website Using The Equivalence Partitioning Technique,” *International Journal Of Information Engineering And Science*, Vol. 1, No. 4, Pp. 81–88, Nov. 2024, Doi: 10.62951/Ijies.V1i4.128.
- [23] I. Salamah, “Evaluasi Usability Website Polsri Dengan Menggunakan System Usability Scale,” *Jurnal Nasional Pendidikan Teknik Informatika*, Vol. 8, No. 3, Dec. 2019.
- [24] A. Lupita Dyayu And H. Yani, “Evaluasi Usability Aplikasi Pedulilindungi Menggunakan Metode Usability Testing Dan System Usability Scale (Sus),” *Jurnal Manajemen Teknologi Dan Sistem Informasi (Jms)*, Vol. 3, No. 1, 2023, [Online]. Available: <http://ejournal.unama.ac.id/index.php/jms>

- [25] D. Indrawan And S. Lestanti, “System Usability Scale Aplikasi Grab Dengan Evaluasi Metode Design Thinking Grab Application Usability Scale System Using Design Thinking Method Evaluation,” *Journal Of Information System And Application Development*, Vol. 3, Pp. 76–88, 2025, Doi: 10.26905/Jisad.V3i2.16043.
- [26] D. I. Putra, R. Aisuwarya, S. Ardopa, And I. Purnama, “Sistem Cerdas Reservasi Dan Pemantauan Parkir Pada Lokasi Kampus Berbasis Konsep Internet Of Things,” *Jurnal Teknologi Dan Sistem Komputer*, Vol. 6, No. 2, Pp. 57–63, Mar. 2018, Doi: 10.14710/Jtsiskom.6.2.2018.57-63.
- [27] K. Andrean, H. Armanto, And C. Pickerling, “Sistem Tempat Parkir Terintegrasi Yang Dilengkapi Dengan Aplikasi Mobile Dan Mikrokontroller,” *Journal Of Information System, Graphics, Hospitality And Technology*, Vol. 7, No. 1, Jun. 2020, Doi: <https://doi.org/10.37823/Insight.V7i1>.
- [28] A. Budi Warsito, M. Aspuri, C. Wulandari, D. Stmik Raharja, And M. Stmik Raharja, “Penerapan Sistem Monitoring Parkir Kendaraan Berbasis Android Pada Perguruan Tinggi Raharja,” *Technomedia Journal (Tmj)*, Vol. 2, No. 1, 2017.
- [29] A. Rahmatillah, J. Informatika, I. Vitra, P. J. Informatika, K. Dwi, And I. J. Informatika, “Moparking: Sistem Monitoring Parkiran Mobil Berbasis Iot,” *Journal Uii Automata*, No. 2, 2022, Accessed: Nov. 27, 2025. [Online]. Available: <https://journal.uii.ac.id/automata/>
- [30] K. Radheshyam Yadav, R. Rajesh Shah, R. Amar Yemul, P. Shivasharan Malge, And A. Vijaykumar Mophare, “Park-O: A Smart Iot-Based Parking Management System With Real-Time Slot Allocation And Qr-Gate Automation,” *International Research Journal Of Modernization In Engineering Technology And Science*, Vol. 7, No. 5, May 2025, Doi: 10.56726/Irjmets75573.
- [31] S. Shigaonkar, A. Vetel, P. Yeolekar, And S. Walke, “A Decentralized Web Application For Parking Spot Booking And Listing: Improving Convenience And Efficiency In Urban Areas,” *Web Engineering & Technology*, Vol. 10, No. 1, 2023, Doi: 10.37591/Jowet.

- [32] M. Z. F. Abdullah, “Penggunaan Rfid Sistem Informasi Parkir Berbasis Web,” *Nuansa Informatika*, Vol. 18, No. 1, Pp. 1858–3911, Jan. 2024, Doi: <https://doi.org/10.25134/nuansa>.
- [33] P. Mangwani, “Smart Parking System Based On Internet Of Things,” *International Journal Of Applied Engineering Research*, Vol. 13, No. 12, Pp. 10281–10285, 2018, Accessed: Nov. 27, 2025. [Online]. Available: https://r.search.yahoo.com/_ylt=Awrjfkfdochpp5umy3fxnyoa;_ylu=Y29sbwnnctecg9zazeednrpzamec2vja3ny/Rv=2/Re=1765453278/Ro=10/Ru=https%3a%2f%2fwww.ripublication.com%2fijaer18%2fijaerv13n12_24.pdf/Rk=2/Rs=Kubkh1xaautnmtceagtjvuozbq8-
- [34] Sudheer Kumar C, Bhanuchand C, Ganesh G, Ezaaz Basha U, And Meenakshi B, “Smart Parking: Real-Time Parking Availability System,” *International Journal Of Engineering Technology And Management Sciences*, Vol. 9, No. 2, 2025, Doi: 10.46647/Ijetms.2025.V09i02.013.
- [35] K. Gilang Prakoso And B. Sisepahputra, “Perancangan Sistem Monitoring Ruangan Kuliah Berbasis Nfc Dan Iot (Studi Kasus: Universitas Negeri Surabaya),” *Journal Of Emerging Information Systems And Business Intelligence*, Vol. 5, No. 2, 2024, Doi: 10.26740/Jeisbi.