



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



**RANCANG BANGUN *VISITOR MANAGEMENT*
SYSTEM (VMS) BERBASIS WEB DI PT TOYOTA
MOTOR MANUFACTURING INDONESIA**

SKRIPSI

KANIRA ERLIANA AZWA ZALEA

2107412041

**POLITEKNIK
NEGERI
JAKARTA**

**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA DAN KOMPUTER
POLITEKNIK NEGERI JAKARTA**

2025



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



RANCANG BANGUN *VISITOR MANAGEMENT SYSTEM* (VMS) BERBASIS WEB DI PT TOYOTA MOTOR MANUFACTURING INDONESIA

SKRIPSI

**Dibuat untuk Melengkapi Syarat-Syarat yang Diperlukan untuk
Memperoleh Diploma Empat Politeknik**

**KANIRA ERLIANA AZWA ZALEA
2107412041**

**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA DAN KOMPUTER
POLITEKNIK NEGERI JAKARTA**

2025



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

SURAT PERNYATAAN BEBAS PLAGIARISME

Saya yang bertanda tangan di bawah ini:

Nama : Kanira Erliana Azwa Zalea
NIM : 2107412041
Jurusan/Program Studi : T.Informatika dan Komputer / Teknik Informatika
Judul skripsi : Rancang Bangun *Visitor Management System* (VMS) Berbasis Web di PT Toyota Motor Manufacturing Indonesia

Menyatakan dengan sebenarnya bahwa skripsi ini benar-benar merupakan hasil karya saya sendiri, bebas dari peniruan terhadap karya dari orang lain. Kutipan pendapat dan tulisan orang lain ditunjuk sesuai dengan cara-cara penulisan karya ilmiah yang berlaku.

Apabila di kemudian hari terbukti atau dapat dibuktikan bahwa dalam skripsi ini terkandung ciri-ciri plagiat dan bentuk-bentuk peniruan lain yang dianggap melanggar peraturan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Depok, 22 Juli 2025

Yang membuat pernyataan



Kanira Erliana Azwa Zalea
2107412041



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

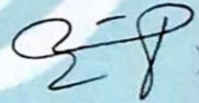
LEMBAR PENGESAHAN

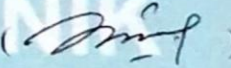
Skripsi diajukan oleh :

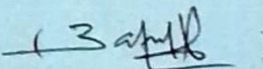
Nama : Kanira Erliana Azwa Zalea
NIM : 2107412041
Program Studi : Teknik Informatika
Judul Skripsi : Rancang Bangun *Visitor Management System* (VMS)
Berbasis Web di PT Toyota Motor Manufacturing
Indonesia

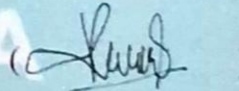
Telah diuji oleh tim penguji dalam Sidang Skripsi pada hari Senin, tanggal 23,
Bulan Juni, Tahun 2025 dan dinyatakan **LULUS**.

Disahkan oleh

Pembimbing I : Iklinta Ermis Ismail S.Kom., M.Kom. ()

Penguji I : Dr. Anita Hidayati, S.Kom., M.Kom. ()

Penguji II : Bambang Warsuta, S.Kom., M.T.I. ()

Penguji III : Dr. Indra Hermawan ()

Mengetahui:

Jurusan Teknik Informatika dan Komputer
Ketua



Dr. Anita Hidayati, S.Kom., M.Kom.
NIP. 197908032003122003



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

SURAT PERNYATAAN PERSETUJUAN PUBLIKASI SKRIPSI UNTUK KEPENTINGAN AKADEMIS

Sebagai sivitas akademik Politeknik Negeri Jakarta, saya bertanda tangan dibawah ini :

Nama : Kanira Erliana Azwa Zalea

NIM : 2107412041

Jurusan/Program Studi : T.Informatika dan Komputer / Teknik Informatika

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Politeknik Negeri Jakarta Hak Bebas Royalti Non-Eksklusif atas karya ilmiah saya yang berjudul :

Rancang Bangun *Visitor Management System* (VMS) Berbasis Web di PT Toyota Motor Manufacturing Indonesia

Beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Non-Eksklusif ini Politeknik Negeri Jakarta berhak menyimpan, mengalihmediakan/formatkan, mengelola dalam bentuk pangkalan data (database), merawat, dan mempublikasikan skripsi saya tanpa meminta izin dari saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Depok, 22 Juli 2025

Yang Menyatakan



Kanira Erliana Azwa Zalea

2107412041

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

KATA PENGANTAR

Puji Syukur saya panjatkan kepada Tuhan Yang Maha Esa, karena atas berkat dan rahmat-Nya, penulis dapat menyelesaikan laporan Skripsi ini. Penulisan laporan Skripsi ini dilakukan dalam rangka memenuhi salah satu syarat untuk mencapai gelar Diploma Empat Politeknik. Penulis menyadari bahwa, tanpa bantuan dan bimbingan dari berbagai pihak, dari masa perkuliahan hingga pada penyusunan laporan Skripsi, sangatlah sulit bagi penulis untuk menyelesaikan Laporan Skripsi ini. Oleh karena itu, penulis mengucapkan terima kasih kepada:

- a. Ibu Dr. Anita Hidayati, S.Kom., M.Kom., selaku Ketua Jurusan Teknik Informatika dan Komputer.
- b. Ibu Iklima Ermis Ismail S.Kom., M.Kom., selaku dosen pembimbing yang telah menyediakan waktu, tenaga, dan pikiran untuk mengarahkan penulis dalam penyusunan laporan Skripsi ini;
- c. Pihak PT Toyota Motor Manufacturing Indonesia yang telah banyak membantu dalam usaha memperoleh data yang penulis perlukan;
- d. Orang tua dan keluarga penulis yang telah memberikan bantuan dukungan moral dan material;
- e. Sahabat yang telah banyak membantu penulis dalam menyelesaikan laporan Skripsi ini.

Akhir kata, penulis berharap Tuhan Yang Maha Esa berkenan membalas segala kebaikan semua pihak yang telah membantu. Semoga laporan Skripsi ini membawa manfaat bagi pengembangan ilmu.

Depok, 22 Juli 2025



Kanira Erliana Azwa Zalea



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

**Rancang Bangun *Visitor Management System* (VMS) Berbasis Web
Menggunakan Metode *Waterfall* di PT Toyota Motor Manufacturing
Indonesia**

ABSTRAK

Dalam lingkup PT Toyota Motor Manufacturing Indonesia (TMMIN), pengelolaan kunjungan secara manual sering menjadi kendala, khususnya ketika melakukan registrasi yang memakan waktu cukup lama dan ketepatan dalam pencatatan data pengunjung. Oleh karena itu, penelitian ini bertujuan untuk membangun sistem *Visitor Management System* (VMS) berbasis web guna mendukung proses registrasi dan pengelolaan kunjungan di PT TMMIN. VMS menyediakan fitur utama seperti pencatatan data kunjungan, approval kunjungan, monitoring status check-in dan check-out secara real-time, visualisasi data kunjungan, serta daftar riwayat kunjungan. Terdapat 4 tipe user yang memiliki akses sistem ini yaitu visitor, host (karyawan PT TMMIN), admin, dan resepsionis/security di area Head Office PT TMMIN. Pendekatan penelitian mencakup metode kualitatif (observasi, wawancara, studi literatur) dan kuantitatif (Black-Box Testing, UAT). Sistem VMS dirancang menggunakan teknologi modern, yaitu ReactJS untuk antarmuka pengguna, NodeJS untuk pengelolaan logika, dan PostgreSQL sebagai basis data. Sistem VMS dikembangkan menggunakan metode *Waterfall* mulai dari analisis kebutuhan, perancangan, implementasi, hingga pengujian. Pengujian sistem dilakukan melalui dua metode pengujian, yaitu Black-Box Testing untuk memverifikasi fungsionalitas sistem dan User Acceptance Testing (UAT) yang dilakukan oleh pengguna akhir untuk mengukur penerimaan pengguna terhadap sistem. Hasil pengujian menunjukkan bahwa sistem VMS berhasil mencapai tingkat keberhasilan fungsional 100% pada 47 skenario Black-Box Testing. Selain itu, hasil UAT menunjukkan tingkat penerimaan pengguna terhadap sistem dengan persentase total 87,4%. Hasil observasi menunjukkan penurunan durasi proses registrasi sebesar 100% untuk planned visitor dan 56,32% untuk unplanned visitor.

Kata Kunci: *Visitor Management System, Waterfall, ReactJS, NodeJS, PostgreSQL.*



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

DAFTAR ISI

KATA PENGANTAR.....	iv
ABSTRAK	v
DAFTAR ISI.....	vi
DAFTAR GAMBAR.....	viii
DAFTAR TABEL	x
DAFTAR LAMPIRAN	xi
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang Masalah.....	1
1.2 Perumusan Masalah	3
1.3 Batasan Masalah.....	3
1.4 Tujuan dan Manfaat	4
1.4.1 Tujuan	4
1.4.2 Manfaat	4
1.5 Sistematika Penulisan	4
BAB II TINJAUAN PUSTAKA.....	6
2.1 <i>Visitor Management System</i>	6
2.2 <i>Waterfall</i>	6
2.3 Figma	8
2.4 ReactJS	8
2.5 Node.js	8
2.6 Express.js	9
2.7 Postgresql.....	9
2.8 Tailwind CSS	9
2.9 <i>User Acceptance Testing (UAT)</i>	9
2.10 Penelitian Terdahulu	10
BAB III METODE PENELITIAN	15
3.1. Rancangan Penelitian	15
3.2 Tahapan Penelitian	15
3.2.1 Identifikasi Masalah	16
3.2.2 Pengumpulan Data	16
3.2.3 Analisis Kebutuhan (<i>Requirements</i>)	19
3.2.4 Desain Sistem (<i>System Design</i>).....	19
3.2.5 Implementasi (<i>Implementation</i>)	20
3.2.6 Pengujian (<i>Testing</i>)	20
3.2.7 <i>Deployment</i>	20
3.2.8 Dokumentasi	20
3.3 Objek Penelitian	21
BAB IV HASIL DAN PEMBAHASAN	22
4.1 Analisis Kebutuhan	22
4.1.1 Kebutuhan Fungsional	22



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

4.1.2 Kebutuhan Non-Fungsional	25
4.2 Perancangan Sistem	26
4.2.1 <i>Entity Relationship Diagram</i> (ERD)	27
4.2.2 Business Process Flow	32
4.2.3 <i>Flowchart</i>	34
4.2.4 <i>Mockup</i>	39
4.3 Implementasi Sistem	46
4.3.1 Implementasi Autentikasi User	47
4.3.1.1 Login Pengguna Terdaftar	47
4.3.1.2 Implementasi Guest Access	50
4.3.2 Implementasi Dashboard	53
4.3.2.1 Implementasi Grafik & Tabel Pengunjung	53
4.3.2.2 Implementasi <i>Modal Scanner QR code</i>	58
4.3.3 Implementasi Registration	63
4.3.4 Implementasi <i>Approval</i>	70
4.3.5 Implementasi <i>Check-In/Check-Out</i>	76
4.3.6 Implementasi <i>Inquiry</i>	78
4.4 Pengujian	81
4.4.1 Deskripsi Pengujian	81
4.4.2 Prosedur Pengujian	82
4.4.3 Data Hasil Pengujian	85
4.4.3.1 Hasil <i>Black-Box Testing</i>	85
4.4.3.2 Hasil <i>User Acceptance Testing (UAT)</i>	95
4.4.4 Analisis Data / Evaluasi Pengujian	98
4.4.4.1 Analisis / Evaluasi <i>Black-Box Testing</i>	98
4.4.4.2 Analisis / Evaluasi <i>User Acceptance Testing (UAT)</i>	98
4.4.4.3 Analisis / Evaluasi <i>Before</i> dan <i>After</i> Implementasi Sistem VMS	104
4.4.4.4 Analisis / Evaluasi <i>Performance</i> Sistem VMS	107
BAB V KESIMPULAN DAN SARAN	110
5.1 Kesimpulan	110
5.2 Saran	111
DAFTAR RIWAYAT HIDUP	112
DAFTAR PUSTAKA	113
LAMPIRAN	115



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

DAFTAR GAMBAR

Gambar 2.1 Metode Waterfall.....	7
Gambar 3.1 Tahapan Penelitian	16
Gambar 4.1 <i>Entity Relationship Diagram</i> Sistem VMS	27
Gambar 4.2 Business Process Flow Sistem VMS.....	32
Gambar 4.3 Flowchart Sistem VMS Role Visitor	35
Gambar 4.4 Flowchart Sistem VMS Role Host	36
Gambar 4.5 Flowchart Sistem VMS Role Receptionist/Security	38
Gambar 4.6 Flowchart Sistem VMS Role Admin	39
Gambar 4.7 Mockup Halaman Login.....	40
Gambar 4.8 Mockup Halaman Dashboard.....	41
Gambar 4.9 Mockup Halaman Form Registrasi Individual	42
Gambar 4.10 Mockup Halaman Form Registrasi Grup	43
Gambar 4.11 Mockup Halaman Tabel Registrasi Kunjungan	44
Gambar 4.12 Mockup Halaman Approval	45
Gambar 4.13 Mockup Halaman Check-In/Out	45
Gambar 4.14 Mockup Halaman Inquiry	46
Gambar 4.15 Implementasi Halaman Login	47
Gambar 4.16 Sumber Kode Validasi Input Login	48
Gambar 4.17 Sumber Kode Post Login	48
Gambar 4.18 Sumber Kode User Login Berdasarkan Role	49
Gambar 4.19 Sumber Kode Token Guest	50
Gambar 4.20 Sumber Kode API Token	51
Gambar 4.21 Sumber Kode fungsi getGuestToken	52
Gambar 4.22 Sumber Kode Akses Halaman Guest	52
Gambar 4.23 Hasil Response API Token Guest	53
Gambar 4.24 Sumber Kode Fetch Visitor Dashboard	54
Gambar 4.25 Sumber Kode Lanjutan Fetch Visitor Dashboard	54
Gambar 4.26 Sumber Kode Bar Chart	55
Gambar 4.27 Sumber Kode Pie Chart.....	56
Gambar 4.28 Implementasi Tampilan Halaman Dashboard	57
Gambar 4.29 Sumber Kode Scanner QR	58
Gambar 4.30 Sumber Kode Validasi QR code	59
Gambar 4.31 Sumber Kode Tombol Check-In/Out	59
Gambar 4.32 Sumber Kode Modal Check-In/Out QR code	60
Gambar 4.33 Sumber Kode Fetch QR code	61
Gambar 4.34 Implementasi Tampilan Modal Check-In QR code	62
Gambar 4.35 Implementasi Tampilan Modal Check-Out QR code.....	63
Gambar 4.36 Sumber Kode Interface Proses Registrasi	63
Gambar 4.37 Sumber Kode Objek Form Registrasi	64



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Gambar 4.38 Sumber Kode Input Nama Host	65
Gambar 4.39 Sumber Kode Variabel Kondisi Role User	66
Gambar 4.40 Sumber Kode Submit Form Registrasi.....	67
Gambar 4.41 Sumber Kode Modal Message Registrasi	68
Gambar 4.42 Sumber Kode Status Request Kunjungan	68
Gambar 4.43 Sumber Kode Generate QR code	69
Gambar 4. 44 Implementasi Tampilan Halaman Registrasi Visitor oleh Host.....	70
Gambar 4.45 Sumber Kode Fetch Approval Visitor.....	70
Gambar 4.46 Sumber Kode Tombol Accept.....	71
Gambar 4.47 Sumber Kode Tombol Reject.....	72
Gambar 4.48 Sumber Kode Checkbox.....	73
Gambar 4.49 Sumber Kode Logika Checkbox Approval/Reject.....	74
Gambar 4.50 Sumber Kode Modal Approval	75
Gambar 4.51 Implementasi Halaman Approval.....	75
Gambar 4.52 Sumber Kode Fetch Visitor Check-In/Out.....	76
Gambar 4.53 Sumber Kode Validasi Check-In/Out QR Code	77
Gambar 4. 54 Implementasi Tampilan Halaman Check-In/Out	78
Gambar 4.55 Sumber Kode Fetch Visitor Inquiry	79
Gambar 4. 56 Sumber Kode Tampilan Body Tabel Inquiry.....	80
Gambar 4.57 Implementasi Tampilan Halaman Inquiry.....	81
Gambar 4.58 Perbandingan durasi registrasi planned visitor before dan after menggunakan sistem VMS	105
Gambar 4.59 Perbandingan durasi registrasi unplanned visitor before dan after menggunakan sistem VMS	106
Gambar 4.60 Database Daftar Registrasi Kunjungan	106
Gambar 4.61 Konfigurasi Skenario Apache JMeter	108
Gambar 4.62 Hasil Performance Test Sistem VMS	108
Gambar 4.63 Hasil Uptime Test Sistem VMS.....	109



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

DAFTAR TABEL

Tabel 1. Penelitian Terdahulu	10
Tabel 2. Instrumen Observasi Before Implementasi Sistem VMS	17
Tabel 3. Instrumen Observasi After Implementasi Sistem VMS.....	18
Tabel 4. Kebutuhan Fungsional Sistem VMS Berdasarkan Role Pengguna	22
Tabel 5. Fitur Sistem VMS Berdasarkan Role Pengguna	24
Tabel 6. Tabel Master Database VMS	28
Tabel 7. Tabel Transaksional Database VMS.....	30
Tabel 8. Legend Business Process Flow	33
Tabel 9. Prosedur Black-Box Testing	82
Tabel 10. Bobot Kriteria Penilaian Kuesioner	84
Tabel 11. Pernyataan Kuesioner User Acceptance Testing	84
Tabel 12. Skenario Black-Box Testing Autentikasi.....	86
Tabel 13. Skenario Black-Box Testing Dashboard.....	87
Tabel 14. Skenario Black-Box Testing Menu Registrasi.....	88
Tabel 15. Skenario Black-Box Testing Menu Approval.....	90
Tabel 16. Skenario Black-Box Testing Menu Check-In/Out.....	93
Tabel 17. Skenario Black-Box Testing Menu Inquiry	94
Tabel 18. Hasil Responden UAT	95
Tabel 19. Analisis Rata-Rata dan Persentase Hasil UAT	99
Tabel 20. Persentase Kriteria Penilaian UAT	101
Tabel 21. Hasil Persentase Reduce Time Durasi Total Proses Registrasi	107
Tabel 22. Hasil Persentase Reduce Time Durasi Registrasi	107



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

DAFTAR LAMPIRAN

Lampiran 1 Initial requirement dengan karyawan PT TMMIN.....	115
Lampiran 2 Pemaparan hasil analisis requirements dengan stakeholders PT TMMIN	117
Lampiran 3 User Acceptance Testing (UAT) PT TMMIN.....	118
Lampiran 4 Observasi Before Implementasi VMS di Head Office PT TMMIN	119
Lampiran 5 Observasi After Implementasi VMS di Head Office PT TMMIN ..	119
Lampiran 6 Responden Kuesioner UAT.....	120



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

BAB I PENDAHULUAN

1.1 Latar Belakang Masalah

Di era transformasi digital saat ini, PT Toyota Motor Manufacturing Indonesia terus mencari solusi inovatif untuk meningkatkan efisiensi operasional dan keamanan. Salah satu area yang menjadi perhatian adalah pengelolaan tamu di lingkungan perusahaan, di mana metode manual yang konvensional sering kali menimbulkan ketidakefisienan dan kesalahan. Metode tersebut tidak hanya memakan waktu tetapi juga rentan terhadap ketidakakuratan dalam pencatatan data tamu, sehingga berpotensi mengurangi tingkat keamanan dan transparansi operasional. Kebutuhan akan *Visitor Management System* (VMS) muncul dari meningkatnya permintaan untuk proses yang terintegrasi dalam mendaftarkan, memantau, dan mengelola tamu secara efektif. Dalam sektor bisnis, sistem informasi pengunjung berfungsi untuk merampingkan berbagai tugas seperti pemrosesan data, penyimpanan, dan pembuatan laporan, memberikan informasi yang cepat dan akurat (Prastomo A. et al, 2023). Dengan kemajuan teknologi, VMS menjadi kebutuhan penting, terutama pada industri dengan aktivitas kunjungan yang tinggi, karena mampu menggantikan proses manual yang sering kali tidak efisien dan memiliki potensi kesalahan.

PT Toyota Motor Manufacturing Indonesia (TMMIN) adalah perusahaan manufaktur otomotif berskala besar yang beroperasi di dua lokasi, yaitu Jakarta Utara dan Karawang terdiri atas Head Office, Karawang Plant, dan Sunter Plant. Dengan skala operasional yang melibatkan banyak pihak, seperti *outsourcer*, *project support*, VIP/VVIP *visitor*, hingga *vendor*, pengelolaan kunjungan yang sistematis menjadi sangat krusial. Berdasarkan wawancara dengan Bapak Endang Supriyadi selaku *Staff Visitor Handling Plant Management, GA, SHE Dept* dalam satu hari PT Toyota Motor Manufacturing Indonesia dapat menerima *visitor* hingga mencapai 200 individu dengan asumsi durasi waktu selama 21 menit dari awal

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

registrasi kunjungan hingga pihak internal/karyawan PT Toyota Motor Manufacturing Indonesia sebagai *host* menemui *visitor* terdapat pada Lampiran 1.

Sebagai perusahaan yang menerapkan standar tinggi dalam keamanan, PT Toyota Motor Manufacturing Indonesia membutuhkan sistem pengelolaan kunjungan yang mampu mendukung operasional secara terintegrasi dan *real-time*. Proses pengelolaan kunjungan di PT Toyota Motor Manufacturing Indonesia dilakukan secara manual dengan pencatatan berbasis buku pengunjung dan menyimpan data dalam Microsoft Excel, sehingga memunculkan beberapa permasalahan utama seperti ketidakefisienan dalam pencatatan data, keterlambatan dalam proses verifikasi, serta minimnya aksesibilitas terhadap informasi kunjungan secara *real-time*, khususnya status *check-in/check-out visitor*. Hal ini berpotensi menyebabkan hambatan operasional, risiko keamanan, dan kesulitan dalam pemantauan data historis pengunjung.

Penelitian mengenai “Sistem Informasi Kehadiran Siswa Berbasis SMS Gateway dengan QR Code” mengembangkan sistem pencatatan kehadiran siswa menggunakan QR Code sebagai *input* data kehadiran. Sistem ini dapat mempercepat proses absensi siswa melalui pemindaian kartu pelajar yang berisi QR Code, kemudian akan dikirimkan ke orang tua/wali siswa dalam bentuk SMS setelah kehadiran dipindai. Penelitian ini menghasilkan kesimpulan bahwa aplikasi ini dapat mempermudah pihak sekolah dalam memberikan informasi dan laporan absensi kepada orang tua siswa dengan cepat (Ngurah Mega Nata et al, 2023). Hal ini memungkinkan orang tua untuk ikut serta dalam mengawasi kehadiran anak mereka di sekolah.

Sebagai solusi, dalam penelitian ini *Visitor Management System* (VMS) berbasis web dibangun untuk mempersingkat proses registrasi, mengelola data pengunjung secara digital, dan memberi informasi secara *real-time* terkait kunjungan. Selain itu, sistem ini dirancang untuk mendukung proses *check-in/check-out* pengunjung menggunakan QR Code, memberikan akses sesuai peran pengguna untuk

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

pengelolaan data kunjungan, serta menyediakan pencatatan data historis pengunjung. Implementasi sistem ini diharapkan mempersingkat proses registrasi dan *check-in/out visitor*, memperkuat keamanan, dan mendukung digitalisasi menyeluruh dalam operasional PT Toyota Motor Manufacturing Indonesia.

1.2 Perumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan, rumusan masalah dari penelitian ini adalah “Bagaimana membangun *Visitor Management System* (VMS) berbasis web untuk proses registrasi kunjungan dan pengelolaan data pengunjung di PT Toyota Motor Manufacturing Indonesia?”.

1.3 Batasan Masalah

Batasan Masalah dari penelitian ini adalah sebagai berikut.

- Implementasi sistem VMS di Head Office PT Toyota Motor Manufacturing Indonesia (TMMIN).
- Penelitian ini hanya mencakup perancangan dan membangun sistem berbasis web untuk mengelola tamu secara digital, meliputi fitur registrasi, pelacakan status *check-in/check-out*, dan pencatatan data historis tamu. Sistem ini tidak mencakup pelacakan pergerakan tamu secara *real-time* di dalam area perusahaan.
- Sistem VMS dirancang untuk mendukung proses *check-in/check-out* menggunakan *QR code* sebagai metode utama validasi kunjungan. Penggunaan teknologi lain seperti biometrik tidak menjadi bagian dari penelitian ini.
- Penelitian ini menggunakan metode *Waterfall* sebagai pendekatan pengembangan sistem. Tidak mencakup proses *maintenance* karena penelitian ini berfokus untuk membangun sistem.

- e. Sistem ini dibangun menggunakan teknologi ReactJS untuk *frontend*, NodeJS untuk *backend*, dan PostgreSQL sebagai basis data.

1.4 Tujuan dan Manfaat

Berikut ini merupakan tujuan dan manfaat dari penelitian yang dibuat sebagai berikut.

1.4.1 Tujuan

Penelitian ini bertujuan untuk membangun *Visitor Management System* berbasis web untuk proses registrasi kunjungan dan pengelolaan data pengunjung di PT Toyota Motor Manufacturing Indonesia.

1.4.2 Manfaat

Penelitian ini memiliki beberapa manfaat kepada pengguna sebagai berikut.

1. Membantu *visitor* dalam proses registrasi kunjungan ke area *Head Office* PT Toyota Motor Manufacturing Indonesia.
2. Membantu karyawan PT Toyota Motor Manufacturing Indonesia menerima informasi dan memberikan persetujuan kunjungan *visitor*.
3. Menyediakan alat bagi *receptionist/security* untuk memantau informasi kunjungan *visitor* secara *real-time*, termasuk status *check-in/out* dan data historis kunjungan di area *Head Office* PT Toyota Motor Manufacturing Indonesia.
4. Membantu admin dalam mengelola data kunjungan di area *Head Office* PT Toyota Motor Manufacturing Indonesia.

1.5 Sistematika Penulisan

BAB I PENDAHULUAN

Bab ini menjelaskan tentang latar belakang masalah, rumusan masalah, batasan masalah, tujuan, manfaat, dan sistematika penulisan penelitian yang berjudul

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

“Rancang Bangun *Visitor Management System* (VMS) Berbasis Web di PT Toyota Motor Manufacturing Indonesia”.

BAB II TINJAUAN PUSTAKA

Bab ini menjelaskan teori dasar yang digunakan dalam penelitian seperti model yang digunakan sebagai pendukung penelitian dan beberapa referensi penelitian terdahulu yang relevan dengan penelitian ini.

BAB III METODE PENELITIAN

Bab ini menjelaskan tentang teknik pengumpulan dan analisis data dalam pengerjaan penelitian ini yang terdiri dari sub bab rancangan penelitian, tahapan penelitian, dan objek penelitian.

BAB IV HASIL DAN PEMBAHASAN

Bab ini menjelaskan tentang analisis kebutuhan fungsional dan non-fungsional, perancangan sistem yang terdiri atas *Business Process Flow* (BPF), *flowchart*, dan *Entity Relationship Diagram* (ERD), implementasi sistem VMS, dan pengujian sistem.

BAB V KESIMPULAN DAN SARAN

Bab ini merupakan bagian penutup yang menyajikan simpulan dari hasil penelitian serta rekomendasi yang dapat dijadikan acuan untuk penelitian selanjutnya atau pengembangan lebih lanjut.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

BAB II

TINJAUAN PUSTAKA

2.1 *Visitor Management System*

Visitor Management System (VMS) adalah sebuah sistem yang digunakan untuk mengatur dan memantau kehadiran pengunjung di suatu area atau lokasi tertentu, seperti gedung perkantoran, pusat perbelanjaan, institusi pendidikan, atau fasilitas publik lainnya. VMS dirancang untuk meningkatkan keamanan, efisiensi, dan pengalaman pengunjung (Prastomo A. et al, 2023). *Visitor Management System* adalah sistem yang digunakan untuk mengelola dan memantau aktivitas kunjungan pada suatu area atau bisnis. Dalam era modern, selain menggunakan teknologi *QR Code* adapun teknologi seperti biometrik dapat digunakan untuk meningkatkan sistem manajemen pengunjung di tempat-tempat umum dan bisnis dengan merekam informasi pengunjung secara akurat melalui pembaca sidik jari atau pemindaian wajah, untuk meningkatkan perlindungan dan keamanan bagi pengunjung maupun staff.

2.2 *Waterfall*

Model *Waterfall* juga dikenal sebagai model siklus urutan linier, menggambarkan proses pengembangan perangkat lunak dalam aliran sekuensial linier. Metode ini mengharuskan pengembangan dilaksanakan secara berurutan atau sekuensial, yang dimulai dari tahapan perencanaan konsep (*requirement analysis*), pemodelan sistem (*system design*), implementasi, pengujian (*testing*), dan pemeliharaan (*maintenance*) (Alif R et al, 2023).

Dalam model *waterfall* Gambar 2.1, tidak ada fase yang tumpang tindih dan setiap langkah harus diselesaikan sebelum beralih ke langkah berikutnya. Dengan demikian, tidak ada tahap dalam proses pengembangan yang dapat dimulai hingga tahap sebelumnya selesai.

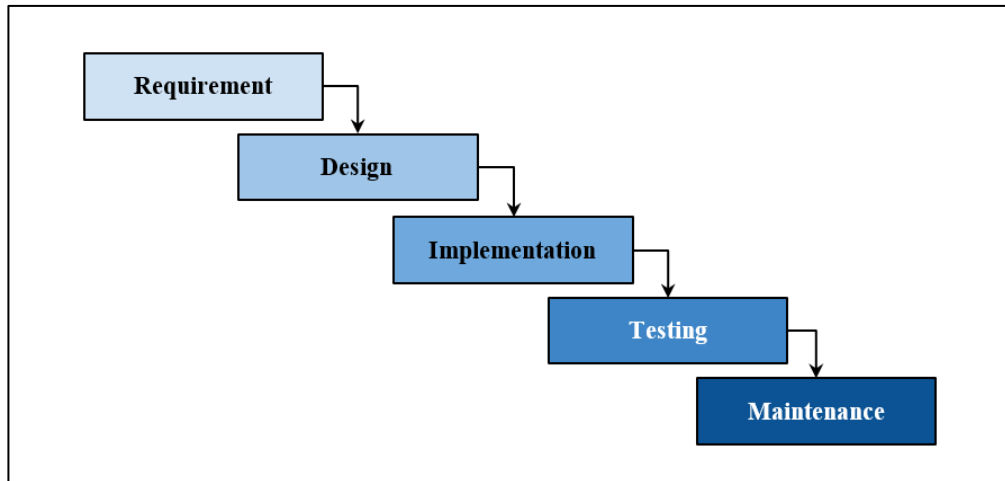
Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



Gambar 2.1 Metode *Waterfall*

Sumber : Alif, Tresya and Suharso, 2023

Berikut ini merupakan uraian tahapan metode *waterfall*.

a. Analisis Kebutuhan (*Requirement*)

Pada tahapan ini, analisis kebutuhan dilakukan dengan menganalisis kebutuhan dari segi pengguna untuk sistem yang akan dirancang. Informasi terkait analisis kebutuhan sistem yakni persyaratan mencakup batasan-batasan perancangan sistem dan spesifikasi kebutuhan sistem dapat dikumpulkan melalui wawancara dengan *stakeholders*.

b. Desain Sistem (*Design*)

Setelah dilakukan analisis kebutuhan, dituangkan dalam desain sistem meliputi struktur data, arsitektur perangkat lunak, dan *mockup interface* untuk selanjutnya dilakukan proses pengkodean.

c. Pengujian (*Implementation*)

Tahapan ini menerjemahkan desain sistem yang telah dirancang melalui proses pengkodean.

d. Pengujian (*Testing*)

Tahapan ini dilakukan untuk menguji sistem yang telah dibangun untuk dianalisis sesuai dengan kebutuhan pengguna.

e. Pemeliharaan (*Maintenance*)

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Tahapan akhir ini dilakukan apabila sistem mengalami *error* atau kerusakan yang tidak ditemukan sebelumnya agar menjaga sistem tetap berjalan dengan baik.

2.3 Figma

Figma adalah sebuah *platform* desain berbasis *cloud* yang digunakan untuk membuat antarmuka pengguna (UI/UX). Figma memungkinkan desainer untuk membuat *wireframe*, *prototype* interaktif, dan desain visual secara kolaboratif dalam *real-time*. Dengan akses berbasis web, Figma mempermudah desainer bekerja di berbagai perangkat tanpa perlu instalasi aplikasi tambahan. Dalam konteks proyek, Figma berperan penting dalam tahap perancangan antarmuka sebelum implementasi dilakukan. Figma dapat digunakan di sistem operasi windows, linux, ataupun mac dengan terhubung ke internet (Dhaniswara et al, 2023).

2.4 ReactJS

ReactJS adalah sebuah library JavaScript yang dikembangkan oleh Facebook untuk memfasilitasi pembuatan antarmuka pengguna yang interaktif. ReactJS memungkinkan pengembang dapat membangun sebuah komponen UI yang lebih interaktif, *stateful*, & *reusable* (Panjaitan J, Pakpahan A, 2021). ReactJS digunakan untuk menangani *view* layer pada aplikasi *single-page* dan aplikasi mobile (Clint I, Tony, 2021). Di dalam kaidah MVC (Model View Control), ReactJS bertanggung jawab hanya pada bagian *view* (Nursaid F F et al, 2020). Tujuan utamanya adalah untuk memungkinkan developer membuat UI yang cepat dan responsif dengan menggunakan konsep komponen dan Virtual DOM.

2.5 Node.js

Node.JS adalah *runtime environment* yang memungkinkan *programmer* menulis aplikasi *server-side* menggunakan JavaScript. Digunakan sebagai platform utama untuk pengembangan *backend*. Node.js memberikan kecepatan eksekusi tinggi dan memungkinkan pengembangan aplikasi yang *scalable* (Khatib Sulaiman J et al, 2023). Didasarkan pada V8 engine milik Google Chrome dan

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

memungkinkan *asynchronous input/output* operasi, yang membuatnya ideal untuk aplikasi *real-time* dan *streaming* media (Sauda S et al, 2022).

2.6 Express.js

ExpressJS merupakan salah satu *framework* NodeJS yang dirancang untuk memudahkan dan mempersingkat proses pembuatan aplikasi berbasis NodeJS dengan menggunakan pola desain yang fleksibel (Danar H et al, 2024). Hal ini dikarenakan *framework* ini minimalis dan fleksibel, serta dirancang untuk mempermudah pembuatan web dan API. *Framework* ini dibangun di dalam Node.js dan memiliki arsitektur yang ringan dan mudah dipahami, sehingga memungkinkan untuk mengembangkan web aplikasi dengan cepat menggunakan fitur bawaan, seperti *middleware*, *routing*, dan pengelolaan *HTTP response*.

2.7 Postgresql

PostgreSQL adalah salah satu sistem manajemen basis data relasional (Relational Database Management System, RDBMS) yang bersifat *open-source* dan berbasis SQL (*Structured Query Language*). PostgreSQL adalah perangkat lunak yang dipakai untuk berbagai aktivitas terkait pengolahan data, seperti menyimpan data, mengedit data, menghapus data, dan replikasi data dalam berbagai jenis aplikasi, mulai dari aplikasi web, analitik data, hingga sistem berbasis enterprise (Suryati P et al, 2024).

2.8 Tailwind CSS

Tailwind CSS adalah *framework* CSS yang digunakan untuk mendesain antarmuka pengguna (UI) secara cepat dan fleksibel. Tailwind CSS juga disebut sebagai *utility-first* karena menyediakan banyak *utility class* yang dapat digunakan untuk membuat custom desain tanpa harus menulis CSS (Azizi A et al, 2024).

2.9 User Acceptance Testing (UAT)

User Acceptance Testing (UAT) merupakan pengujian dimana pengguna akhir berinteraksi langsung dengan sistem untuk memastikan bahwa fungsionalitas yang ada dilakukan sesuai dengan kebutuhan/fungsinya. *Acceptance testing* merupakan

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

pengujian yang dilakukan oleh pengguna yang menggunakan teknik pengujian *Black-Box* untuk menguji sistem terhadap spesifikasinya (Wahyudi I et al, 2023).

2.10 Penelitian Terdahulu

Tabel 1 merupakan beberapa penelitian terdahulu yang relevan dengan implementasi metode *Waterfall* dalam membangun sistem berbasis web. Analisis ini bertujuan untuk memahami metode *Waterfall* dalam membangun sistem berbasis web.

Tabel 1. Penelitian Terdahulu

No	Judul	Penulis	Tahun	Tujuan	Kesimpulan
1	PENGEMBANGAN SISTEM INFORMASI PRESENSI MENGGUNAKAN METODE WATERFALL	Tri Wahyudi, Supriyanta, Husni Faqih	2021	Sistem informasi presensi yang dikembangkan dengan metode pengembangan software <i>Waterfall</i> yang bertujuan untuk memberikan solusi permasalahan presensi karyawan dan siswa di masa pandemi Covid-19.	Penggunaan teknik <i>Waterfall</i> Model sebagai metode pengembangan software untuk Sistem Informasi Presensi tepat sehingga menghasilkan aplikasi yang bermutu sesuai hasil dari Pengujian Beta yang menunjukkan nilai 96.5% pada rerata dari nilai persentase yang dicari (Y) dengan kesimpulan hasilnya adalah Sistem Informasi Presensi sudah user friendly, tata letak komponen yang baik, bermanfaat, dan mempermudah user dalam administrasi presensi.
2	UNIT TESTING DAN USER ACCEPTANCE TESTING PADA SISTEM	Jogi Abraham, Iklima Ermis Ismail,	2021	Dalam penelitian ini pengembangan sistem informasi menggunakan	Berdasarkan hasil pengembangan dari Sistem Informasi Pelayan

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

No	Judul	Penulis	Tahun	Tujuan	Kesimpulan
	INFORMASI PELAYANAN KATEGORIAL PELAYANAN ANAK	S.Kom., M.Kom		metode <i>Waterfall</i> dimana berfungsi sebagai mengatur dan memonitor semua data Pelayan Kategorial Pelayanan Anak (Pelkat PA) di GPIB Petra Bogor.	Kategorial Pelayanan Anak (Pelkat PA) bahwa sistem informasi memperoleh presentase keberhasilan aplikasi dengan presentase sebesar 97,06% untuk <i>black-box testing</i> dengan total 34 skenario dan dalam proses pengujian UAT menghasilkan presentase UAT dapat disimpulkan bahwa pertanyaan dengan kategori desain, efesiensi, dan fungsi mendapatkan total presentase yaitu 86%.
3	WATERFALL METHODE DALAM RANCANG BANGUN SISTEM INFORMASI POTENSI WISATA BERBASIS WEB	Chairul Rizal, Supiyandi, Barany Fachri, Muhammad Hasanuddin	2024	Penelitian ini bertujuan untuk merancang dan membangun sebuah Sistem Informasi Potensi Wisata Desa Kota Pari, yang berlokasi di Kecamatan Pantai Cermin, Kabupaten Serdang Bedagai, dengan pendekatan <i>Waterfall Method</i> .	Perancangan aplikasi administrasi desa berbasis web bertujuan untuk meningkatkan efisiensi, transparansi, dan partisipasi masyarakat dalam pengelolaan administrasi desa. Aplikasi ini memungkinkan warga untuk mengakses informasi penting dan layanan publik dengan mudah, serta memudahkan admin desa dalam mengelola data dan permohonan layanan secara <i>real-time</i> .

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta:

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

No	Judul	Penulis	Tahun	Tujuan	Kesimpulan
4	PENGEMBANGAN VISITOR MANAGEMENT SYSTEM SESUAI TUNTUTAN PERPUSTAKAAN PERGURUAN TINGGI DI ERA 4.0	Suroto, Pudji Rahaju, Diana Widyati, Humayyun Nabila Ramadhani	2023	Pengembangan sistem pencatatan pengunjung yang berbasis website dengan bantuan scan kartu digital yang ada di smartphone dan terintegrasi dengan database mahasiswa, dosen, tenaga kependidikan, dan catatan di layanan sirkulasi, serta memiliki catatan riwayat interaksi dengan ruang baca yang pernah dikunjungi ini untuk mengembangkannya <i>Visitor Management System</i> Unesa Library (VMSUL) yang sesuai dengan kebutuhan manajemen pengunjung seluruh ruang baca di Unesa. Metode yang digunakan adalah metode pengembangan dengan model <i>Waterfall</i> .	Aplikasi VMSUL dengan level user Admin Perpustakaan, Admin Ruang Baca yang dilengkapi dengan Kartu Perpustakaan Elektronik berbasis SSO yang dapat dibuka oleh semua warga Unesa melalui smartphone untuk presensi melalui scanner khusus yang tersedia di seluruh ruang baca selingkung Unesa, Riwayat kunjungan pemustaka, jumlah dan detail pengunjung yang tersedia di level user berhasil dikembangkan dan berfungsi secara optimal.

Berdasarkan hasil kajian terhadap empat penelitian terdahulu, metode *Waterfall* digunakan untuk mengembangkan sistem informasi berbasis web dengan tahapan pengembangan yang berurutan dan terdokumentasi secara jelas. Setiap penelitian menunjukkan bahwa metode ini mampu menghasilkan sistem yang berjalan sesuai fungsinya dan dapat digunakan oleh pengguna sesuai dengan kebutuhan.

Hak Cipta :**1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :**

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Penelitian “Pengembangan Sistem Informasi Presensi Menggunakan Metode *Waterfall*” menyatakan bahwa sistem presensi yang dikembangkan dengan metode *Waterfall* menghasilkan aplikasi dengan hasil pengujian beta sebesar 96,5% (Wahyudi T et al, 2021). Aplikasi tersebut telah dinyatakan dapat digunakan oleh pengguna, dengan tampilan yang sesuai dan navigasi yang mudah dipahami. Hal ini menunjukkan bahwa seluruh tahapan dalam metode *Waterfall* dapat dilaksanakan secara berurutan dan menghasilkan sistem yang dapat dijalankan dengan baik.

Pada penelitian “*Unit Testing dan User Acceptance Testing* pada Sistem Informasi Pelayan Kategorial Pelayanan Anak” menunjukkan sistem informasi Pelayan Kategorial Pelayanan Anak (SIPEKA PA) yang dikembangkan menggunakan metode *Waterfall* untuk GPIB Petra Bogor dalam mengatur dan memonitor data semua data Pelayan Kategorial Pelayanan Anak (Pelkat PA) mampu menyediakan sistem yang dibutuhkan secara akurat dan *real-time* sesuai kebutuhan pengguna dengan total persentase *Black-Box testing* sebesar 97,06% dan UAT sebesar 86% (Abraham J and Ismail, 2021).

Sementara itu, penelitian “*Waterfall Methode Dalam Rancang Bangun Sistem Informasi Potensi Wisata Berbasis Web*” menghasilkan sistem informasi potensi wisata berbasis web menggunakan metode *Waterfall* yang memungkinkan warga desa mengakses informasi wisata secara langsung melalui perangkat digital (Rizal C et al, 2024). Selain itu, administrator desa dapat memperbarui dan mengelola data potensi wisata tanpa perlu mencatat secara manual karena seluruh proses administrasi dilakukan melalui sistem dan tercatat dalam sistem.

Penelitian “Pengembangan *Visitor Management System* Sesuai Tuntutan Perpustakaan Perguruan Tinggi Di Era 4.0” menunjukkan bahwa *Visitor Management System Unesa Library* (VMSUL) berhasil dibangun menggunakan metode *Waterfall* dan digunakan oleh mahasiswa, dosen, dan tenaga kependidikan untuk mencatat kunjungan ke ruang baca dengan menggunakan *scan* digital dari *smartphone*. Sistem juga dapat mencatat data pengunjung serta riwayat interaksi dengan ruang baca dan dapat diakses oleh admin pusat maupun admin ruang baca.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Fitur-fitur tersebut telah diimplementasikan dan berfungsi sesuai kebutuhan institusi. Berdasarkan hasil pengujian *Black-Box testing* tidak ada kesalahan atau bug yang ditemukan atau dapat dikatakan tingkat keberhasilan pengujian sebesar 100% (Suroto et al, 2023)..

Dari keempat penelitian tersebut, dapat disimpulkan bahwa metode *Waterfall* mendukung pembangunan sistem informasi dengan hasil akhir yang dapat digunakan sesuai kebutuhan pengguna, melalui proses pembangunan yang terdokumentasi setiap tahapnya. Hal ini sesuai dengan pendekatan yang digunakan dalam *Visitor Management System* (VMS) berbasis web yang saat ini peneliti bangun.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

BAB III METODE PENELITIAN

3.1. Rancangan Penelitian

Pendekatan yang digunakan dalam penelitian ini adalah metode kuantitatif dan kualitatif. Metode kuantitatif adalah salah satu pendekatan sistematis yang bertujuan untuk mengukur, menganalisis, dan menginterpretasi fenomena melalui data numerik (Aida et al, 2025). Dalam penelitian ini, bertujuan untuk mengukur fungsionalitas sistem yang dibangun peneliti melalui pengujian *Black-Box Testing* dan *User Acceptance Testing* (UAT) menggunakan kuesioner. Selain itu, metode kualitatif dalam penelitian ini menggunakan teknik pengumpulan data melalui observasi ke area *Head Office* PT Toyota Motor Manufacturing Indonesia, wawancara dengan *stakeholders* dari PT Toyota Motor Manufacturing Indonesia, dan studi literatur yang membantu peneliti untuk memperoleh data yang lebih mendalam mengenai kebutuhan dan pengalaman pengguna terhadap sistem.

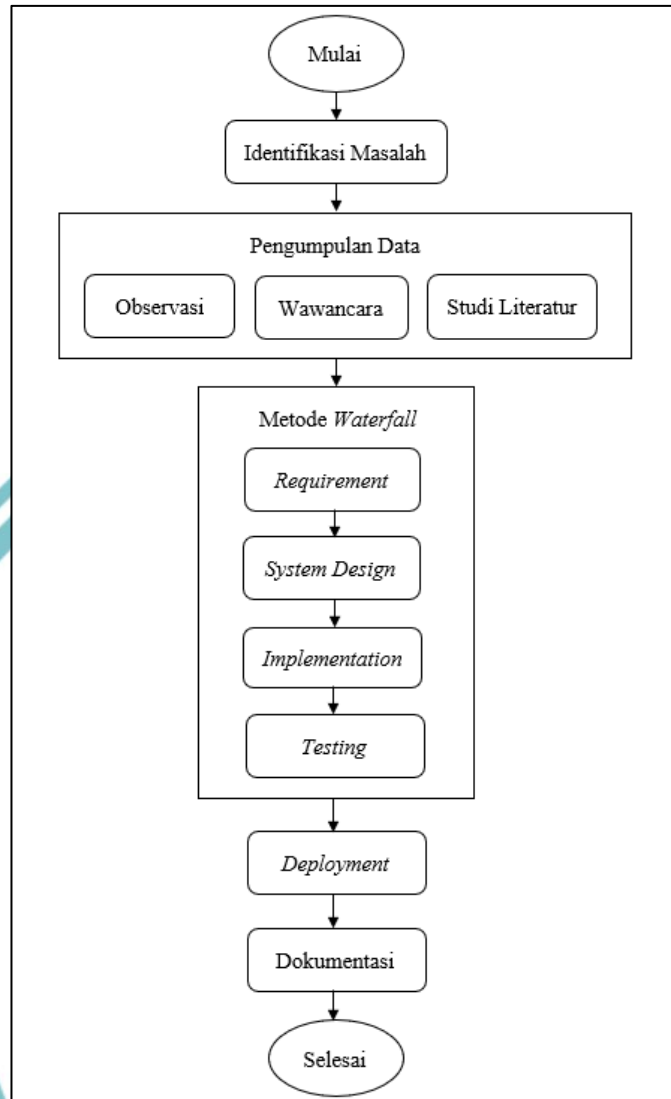
3.2 Tahapan Penelitian

Dalam penelitian ini dilakukan serangkaian tahapan yang dilakukan mulai dari tahapan awal hingga akhir untuk mencapai tujuan dari penelitian ini. Tahapan dalam penelitian ini berawal dari identifikasi masalah, pengumpulan data berupa wawancara, observasi, dan studi literatur, analisis kebutuhan (*requirements*), desain sistem (*design system*), implementasi, pengujian (*testing*), *deployment*, dan dokumentasi selama penelitian ini berlangsung. Gambar 3.1 menunjukkan gambaran besar tahapan penelitian.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



Gambar 3.1 Tahapan Penelitian

3.2.1 Identifikasi Masalah

Tahap awal penelitian diawali dengan mengidentifikasi permasalahan yang ada di lingkungan PT Toyota Motor Manufacturing Indonesia terkait sistem manajemen pengunjung saat ini. Sistem manual dalam manajemen pengunjung menjadi salah satu masalah dalam proses *check-in/check-out* dan mengidentifikasi pengunjung yang ada di lokasi.

3.2.2 Pengumpulan Data

Pada tahap ini, teknik wawancara dilakukan untuk menggali kebutuhan pengguna terhadap sistem bersama *stakeholder* dari PT Toyota Motor Manufacturing

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Indonesia dengan Bapak Endang Supriyadi selaku *Staff Visitor Handling Plant Management, GA, SHE Dept.*

Teknik observasi langsung ke *Head Office* PT Toyota Motor Manufacturing Indonesia digunakan untuk memahami proses registrasi secara manual yang sedang berjalan, termasuk kendala yang sering dihadapi dalam pengelolaan pengunjung. Selama observasi berlangsung peneliti mengamati pergantian *shift* resepsionis dilakukan sebanyak satu kali dan sebelum pergantian *shift*, resepsionis yang bertanggung jawab pada *shift* pertama perlu menyalin data kunjungan dalam buku tamu ke dalam Excel. Selain itu, observasi setelah implemmentasi sistem VMS dilakukan untuk mengevaluasi kinerja sistem terhadap pengguna. Tabel 2 merupakan instrumen observasi sebelum implementasi sistem VMS berdasarkan proses yang peneliti amati.

Tabel 2. Instrumen Observasi *Before* Implementasi Sistem VMS

No	Aspek yang Diamati	Deskripsi
1	Registrasi kunjungan	Proses registrasi kunjungan dilakukan setiap pengunjung diwajibkan mengisi data diri seperti nama, instansi, keperluan, dan waktu kunjungan secara tertulis.
2	Kondisi penerimaan kunjungan	Penerimaan pengunjung dilakukan oleh resepsionis yang berjaga sesuai <i>shift</i> .
3	Proses <i>Check-in</i>	Sebelum diizinkan melakukan <i>check-in</i> , pengunjung harus memastikan bahwa <i>host</i> atau karyawan internal yang akan ditemui telah <i>available</i> untuk janji temu. Proses konfirmasi dilakukan secara manual, melalui komunikasi langsung oleh resepsionis atau melalui telepon internal ke <i>host</i> yang bersangkutan.
4	Penyimpanan data kunjungan	Proses ini dilakukan untuk menyalin seluruh data pengunjung yang ditulis dalam buku tamu oleh

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

No	Aspek yang Diamati	Deskripsi
		resepsionis sebelum pergantian <i>shift</i> sebagai bagian dari pelaporan dan dokumentasi internal ke dalam file Excel.
5	Penggunaan perangkat pendukung	Dalam proses pencatatan ulang data dari buku tamu ke file digital, resepsionis menggunakan perangkat seperti komputer dan Excel.

Tabel 3 merupakan instrumen observasi yang dilakukan setelah diimplementasikan sistem VMS di PT Toyota Motor Manufacturing Indonesia

Tabel 3. Instrumen Observasi *After* Implementasi Sistem VMS

No	Aspek yang Diamati	Deskripsi
1	Registrasi kunjungan	Pengunjung mengisi form registrasi kunjungan melalui sistem VMS berbasis web dengan mencantumkan data pengunjung seperti nama, instansi, keperluan, area, dan waktu kunjungan.
2	Kondisi penerimaan kunjungan	Resepsionis membuka <i>dashboard</i> sistem untuk melihat daftar pengunjung yang telah terdaftar pada hari tersebut.
3	Proses <i>Check-in</i>	Pengunjung bertemu <i>host</i> dan menunjukkan <i>QR code</i> kepada resepsionis untuk pemindaian <i>QR</i> tersebut menggunakan <i>scanner</i> sebagai proses <i>check-in</i> secara digital.
4	Penyimpanan data kunjungan	Data hasil <i>check-in</i> dan <i>check-out</i> terekam secara otomatis dalam sistem dan tersimpan dalam <i>database</i> yang dapat diakses melalui fitur ekspor Excel atau <i>dashboard</i> sistem.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

No	Aspek yang Diamati	Deskripsi
5	Penggunaan perangkat pendukung	Resepsionis menggunakan komputer yang terhubung ke sistem VMS untuk membuka <i>dashboard</i> , memindai <i>QR code</i> menggunakan <i>scanner</i> , dan mengelola data kunjungan.

Studi literatur dilakukan untuk mendalami teori terkait pengembangan sistem berbasis *Waterfall*, metode *testing* seperti *User Acceptance Testing* (UAT), serta teknologi yang digunakan seperti ReactJS, NodeJS, dan PostgreSQL. Selain itu, studi literatur juga dilakukan untuk membandingkan metode yang sudah dilakukan dari penelitian-penelitian terdahulu yang dapat dijadikan landasan.

3.2.3 Analisis Kebutuhan (*Requirements*)

Tahapan ini untuk mengidentifikasi kebutuhan sistem sesuai hasil wawancara dan observasi terhadap kebutuhan fungsional dan non-fungsional terkait sistem manajemen kunjungan di area *Head Office* PT Toyota Motor Manufacturing Indonesia (TMMIN). Setelah mengetahui kebutuhan sistem, peneliti memiliki pemahaman tentang kebutuhan pengguna untuk setiap *function list* yang sudah didefinisikan, pemilihan metode pengembangan sistem perangkat lunak yang digunakan berbasis *Waterfall*, serta teknologi yang digunakan untuk membangun sistem yaitu ReactJS, NodeJS, dan PostgreSQL.

3.2.4 Desain Sistem (*System Design*)

Selanjutnya, tahapan ini akan membuat desain database secara menyeluruh menggunakan *Entity Relationship Diagram* (ERD). Tahap ini juga melibatkan pembuatan rancangan awal sistem dalam bentuk *Business Process Flow* (BPF) dan *flowchart* untuk menggambarkan alur kerja sistem yang peneliti rancang. Selain itu, desain antarmuka pengguna (UI/UX) dibuat untuk mencakup fitur-fitur dalam *Visitor Management System* yang dibangun.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

3.2.5 Implementasi (*Implementation*)

Tahapan ini mengimplementasikan pengembangan sistem sesuai dengan desain yang telah dirancang sebelumnya. Selama tahap ini akan bekerja pada modul yang telah diprioritaskan selama durasi yang ditentukan. Pembangunan sistem dilakukan menggunakan NodeJS dan ReactJS.

3.2.6 Pengujian (*Testing*)

Tahapan pengujian perangkat lunak dilakukan untuk memeriksa sistem berdasarkan *source code* yang telah dikembangkan menggunakan *Black-Box testing* dan *User Acceptance Testing* (UAT) sesuai dengan skenario yang telah disusun. UAT dilakukan dengan melibatkan pengguna akhir dalam lingkup *Head Office* PT Toyota Motor Manufacturing Indonesia untuk memastikan bahwa fungsionalitas sistem sesuai kebutuhan.

3.2.7 Deployment

Tahapan ini mencakup penyediaan sistem yang telah dibangun agar dapat diakses dan digunakan oleh pengguna akhir, yaitu *Visitor Management System*.

3.2.8 Dokumentasi

Dokumentasi mencakup seluruh hasil proses penelitian dan pembangunan *Visitor Management System*. Dokumen ini berisi laporan lengkap tentang seluruh tahapan pengembangan, termasuk dokumentasi teknis dan non-teknis, sehingga dapat digunakan sebagai referensi bagi pengembangan lebih lanjut di masa mendatang. Dokumentasi memastikan bahwa sistem yang telah dikembangkan dapat dipahami dan dikelola dengan baik oleh pihak-pihak yang berkepentingan. Beberapa dokumentasinya seperti Dokumentasi Kebutuhan Sistem, Dokumentasi *Design*, Dokumentasi *Development*, Dokumentasi *Testing*, Dokumentasi *Deployment*, serta penulisan laporan akhir hasil dari penelitian ini.



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

3.3 Objek Penelitian

Objek penelitian dalam proyek ini adalah *Visitor Management System* (VMS) yang dibangun untuk mempersingkat proses registrasi dan pengelolaan pengunjung di area *Head Office* PT Toyota Motor Manufacturing Indonesia.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

BAB IV HASIL DAN PEMBAHASAN

4.1 Analisis Kebutuhan

Analisis kebutuhan merupakan tahap awal dalam pengembangan sistem untuk memahami dan mendefinisikan apa saja yang diperlukan dari sistem yang akan dibangun. Proses ini dilakukan melalui wawancara dengan *stakeholder* dari PT Toyota Motor Manufacturing Indonesia dan observasi alur kerja di area *Head Office* PT Toyota Motor Manufacturing Indonesia. Tujuannya adalah untuk merancang fitur yang tepat guna mendukung proses registrasi dan pengelolaan kunjungan di area *Head Office* PT Toyota Motor Manufacturing Indonesia. Terdapat 2 analisis kebutuhan pada tahapan ini, yaitu kebutuhan fungsional dan non-fungsional sebagai berikut.

4.1.1 Kebutuhan Fungsional

Tabel 4 merupakan kebutuhan fungsional sistem berdasarkan 4 tipe *role* pengguna yang terdiri atas *Visitor*, *Host* (karyawan PT TMMIN), *Receptionist/Security*, dan Admin.

Tabel 4. Kebutuhan Fungsional Sistem VMS Berdasarkan *Role* Pengguna

<i>Role</i>	Kebutuhan Fungsional
<i>Visitor</i>	Melakukan registrasi/pendaftaran kunjungan.
	Menerima notifikasi melalui email terkait status approval kunjungan (<i>approve/reject</i>).
	Menerima <i>QR code</i> setelah mendapatkan <i>approval</i> .
<i>Host/User</i>	Melakukan <i>approval</i> atas reservasi kunjungan yang telah didaftarkan oleh <i>visitor</i> .
	Melakukan registrasi/pendaftaran kunjungan.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Role	Kebutuhan Fungsional
	Melakukan <i>edit</i> data kunjungan.
	Melakukan pembatalan registrasi atau kunjungan.
	Menerima notifikasi jika registrasi telah di- <i>approve</i> atau <i>reject</i> oleh <i>superior one level-up</i> .
	Melihat riwayat <i>visitor</i> yang pernah didaftarkan atau berkunjung.
Receptionist /Security	Melakukan registrasi/pendaftaran kunjungan.
	Melakukan edit data kunjungan.
	Melakukan pembatalan registrasi atau kunjungan.
	Melihat daftar status kunjungan dalam satu area.
	Melakukan <i>check-in/out</i> manual jika <i>visitor</i> terkendala <i>scanning</i> QR.
	Melihat data <i>real-time</i> dashboard jumlah <i>visitor</i> .
	Mengakses daftar riwayat kunjungan dalam satu area.
Admin	Melakukan registrasi/pendaftaran kunjungan dengan <i>bypass approval</i> .
	Melakukan edit data kunjungan.
	Melakukan pembatalan registrasi atau kunjungan.
	Memberikan <i>approval</i> atau <i>rejection</i> terhadap reservasi kunjungan.
	Melihat daftar status kunjungan dalam satu area.
	Melakukan <i>check-in/out</i> manual jika <i>visitor</i> tidak melakukan <i>QR</i>

Role	Kebutuhan Fungsional
	<i>scan.</i>
	Melihat data <i>real-time</i> dashboard jumlah <i>visitor</i> .
	Mengakses daftar riwayat kunjungan dalam satu area.

Tabel 5 merupakan fitur-fitur yang telah didefinisikan berdasarkan kebutuhan fungsional sistem masing-masing *role* pengguna.

Tabel 5. Fitur Sistem VMS Berdasarkan *Role* Pengguna

Fitur	Deskripsi	Role
<i>Login Page</i>	Halaman ini dapat diakses oleh seluruh <i>role</i> termasuk <i>visitor</i> dimana terdapat <i>link</i> yang menjadi akses bagi <i>visitor</i> untuk masuk ke halaman <i>form</i> registrasi kunjungan.	Seluruh Pengguna
<i>Registration Form Page</i>	Halaman ini menampilkan <i>form</i> registrasi yang digunakan pengunjung untuk mendaftar kunjungan ke area PT TMMIN tanpa perlu <i>login</i> .	<i>Visitor</i>
<i>Login Pengguna</i>	<i>Role Admin, Host, Security, dan Receptionist</i> dapat mengakses fitur sesuai <i>role</i> dengan memasukkan <i>username</i> dan <i>password</i> .	Admin, Host, Security, Receptionist
Dashboard	Halaman ini menampilkan visualisasi data kunjungan berupa grafik dan tabel, termasuk akses <i>form</i> registrasi, unduh data kunjungan harian, dan memfilter data kunjungan.	Admin, Security, Receptionist

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Fitur	Deskripsi	Role
Menu <i>Registration</i>	Halaman ini mengakses halaman <i>form</i> registrasi dan menampilkan list data tabel kunjungan yang terdaftar, termasuk menampilkan detail data kunjungan, mengubah data kunjungan, dan membatalkan kunjungan.	Admin, <i>Host</i> , <i>Security</i> , <i>Receptionist</i>
Menu <i>Approval</i>	Halaman ini digunakan untuk menampilkan list data kunjungan yang butuh persetujuan ataupun penolakan dari <i>host</i> .	Admin, <i>Host</i>
Menu <i>Check-In/Out</i>	Halaman ini menampilkan data kunjungan yang telah terjadwal setelah mendapat persetujuan digunakan untuk melakukan <i>check-in/out</i> secara manual jika terdapat kendala pada <i>QR Code</i> yang dikirimkan ke <i>visitor</i> .	Admin, <i>Security</i> , <i>Receptionist</i>
Menu <i>Inquiry</i>	Halaman ini menampilkan seluruh riwayat data kunjungan yang telah terdaftar.	Admin, <i>Host</i> , <i>Security</i> , <i>Receptionist</i>
Notifikasi <i>email</i>	<i>Host</i> ataupun <i>Visitor</i> mendapatkan pesan notifikasi melalui <i>email</i> secara berkala mulai dari <i>visitor</i> melakukan registrasi hingga mendapatkan persetujuan, termasuk <i>QR Code</i> untuk <i>check-in/out</i> .	<i>Visitor</i> , <i>Host</i>

4.1.2 Kebutuhan Non-Fungsional

Kebutuhan non-fungsional menjelaskan karakteristik sistem yang diperlukan untuk mendukung kinerja, keamanan, dan skalabilitas sistem.

1. Keamanan (*Security*)

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

- a. Sistem harus menggunakan autentikasi *username* dan *password* untuk mengakses web bagi karyawan PT TMMIN.
- b. Setiap pengguna hanya dapat mengakses sistem sesuai *role* (RBAC - *Role Based Access Control*).
2. Kinerja (*Performance*)
 - a. Sistem harus mampu menangani minimal 50 *request* secara bersamaan tanpa *lag*.
 - b. *Load time* halaman maksimal 3 detik pada koneksi standar.
3. Reliabilitas dan Ketersediaan (*Reliability & Availability*)

Sistem harus tersedia minimal 99.5% *uptime* selama penelitian ini berlangsung.
4. Kemudahan Penggunaan (*Usability*)
 - a. Sistem harus memiliki UI/UX yang intuitif dan dapat dipahami.
 - b. Setiap halaman harus memiliki navigasi yang jelas terhadap tindakan pengguna.
5. Skalabilitas (*Scalability*)
 - a. Selama penelitian sistem dikembangkan untuk mengakomodasi proses registrasi di *Head Office* PT Toyota Motor Manufacturing Indonesia.
 - b. Sistem dapat terintegrasi dengan perangkat tambahan seperti *scanner QR code*.
6. Portabilitas (*Portability*)

Sistem berbasis web dan dapat diakses melalui *browser* seperti Google Chrome, Safari, Mozilla Firefox, Microsoft Edge, dan lain sebagainya.

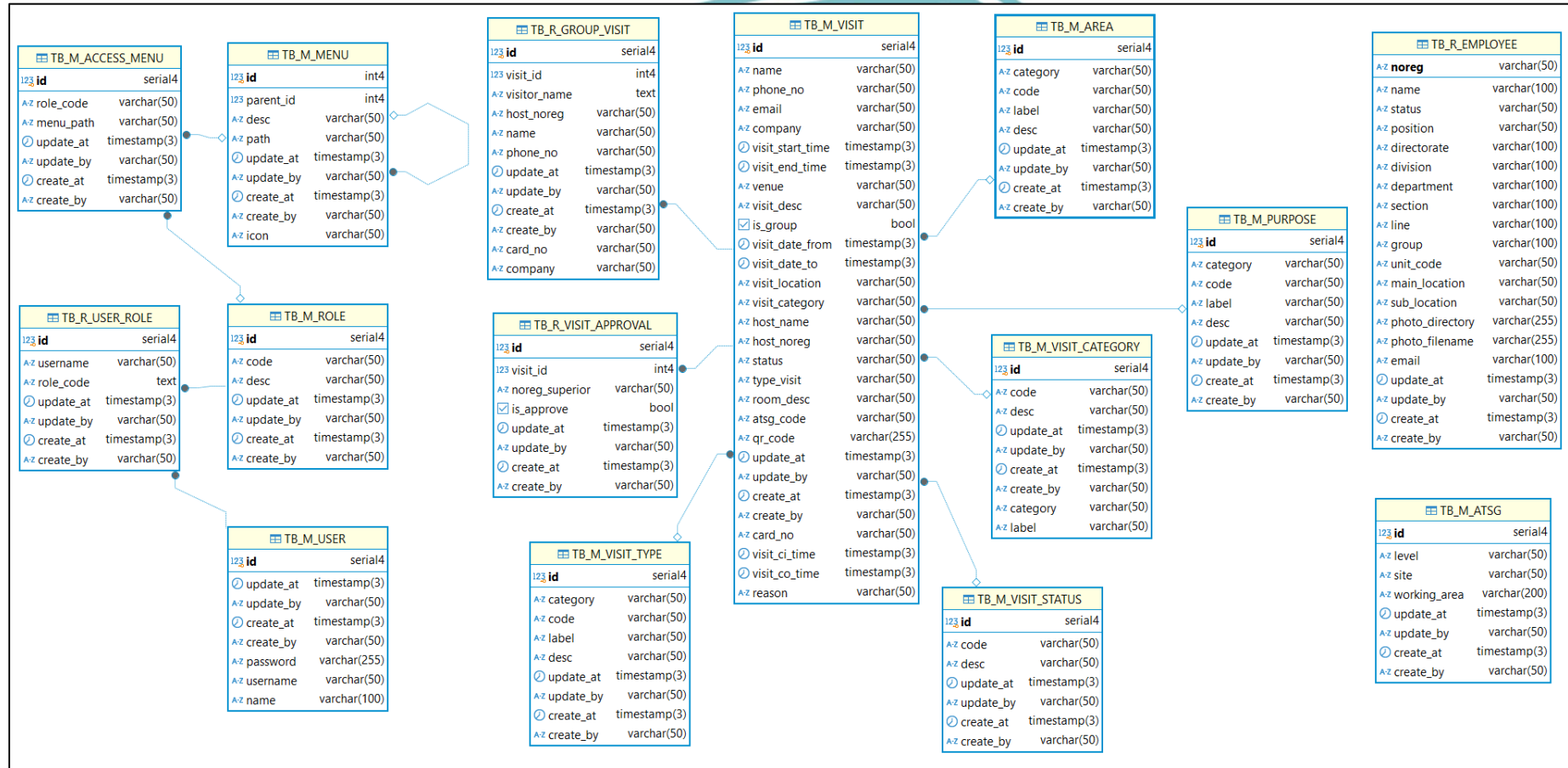
4.2 Perancangan Sistem

Perancangan sistem terdiri dari *Entity Relationship Diagram* (ERD), *Business Process Flow* (BPF), *flowchart*, dan *mockup*.

- Hak Cipta :**
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
 2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

4.2.1 Entity Relationship Diagram (ERD)



Gambar 4.1 Entity Relationship Diagram Sistem VMS

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

Gambar 4.1 menunjukkan *Entity Relationship Diagram* (ERD) dalam sistem VMS yang merepresentasikan struktur basis data yang digunakan untuk menyimpan seluruh informasi terkait kunjungan *visitor* di area perusahaan.

a. Tabel Master Data

Tabel 6 menunjukkan daftar tabel master yang terdapat dalam database sistem VMS.

Tabel 6. Tabel Master Database VMS

Nama Tabel	Deskripsi
TB_M_VISIT	Tabel master untuk menyimpan seluruh informasi terkait kunjungan <i>visitor</i> .
TB_M_VISIT_CATEGORY	Master untuk referensi data kategori kunjungan. Terdiri atas 2 (dua) kategori sebagai berikut. a. VIP adalah tamu/ <i>visitor</i> penting yang memerlukan prosedur khusus oleh <i>Security</i> dan tim <i>Building Utility</i> . Seperti Board of Executive afiliasi Toyota, Board of Directors afiliasi Toyota dan Supplier/Vendor, Pejabat Pemerintah. b. Non-VIP adalah tamu/ <i>visitor</i> yang tidak termasuk dalam kategori VIP.
TB_M_VISIT_TYPE	Master untuk referensi data tipe kunjungan. Terdiri atas 2 (dua) kategori tipe kunjungan sebagai berikut. a. Individual adalah tamu/ <i>visitor</i> yang terdiri dari 1 (satu) orang. b. <i>Group</i> adalah tamu/ <i>visitor</i> yang terdiri lebih dari 1 (satu) orang.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Nama Tabel	Deskripsi
TB_M_VISIT_STATUS	Master untuk referensi data status <i>visitor</i>. Terdiri atas 6 (enam) status kunjungan sebagai berikut. - CHECKIN menandakan <i>visitor</i> sudah memasuki area perusahaan. - CHECKOUT menandakan <i>visitor</i> sudah keluar dari area perusahaan. - RESERVED menandakan kunjungan <i>visitor</i> sudah terjadwalkan. - CANCEL menandakan kunjungan <i>visitor</i> dibatalkan. - WAITING APPROVAL menandakan kunjungan <i>visitor</i> memerlukan persetujuan <i>host</i>. - REJECTED menandakan kunjungan <i>visitor</i> ditolak.
TB_M_PURPOSE	Master untuk referensi data tujuan kunjungan. Terdiri atas 2 (dua) tujuan kunjungan sebagai berikut. <i>Meeting</i> adalah kunjungan yang bertujuan <i>offline meeting</i>. <i>General visit</i> adalah kunjungan yang bertujuan selain <i>meeting</i>, seperti kunjungan personal.
TB_M_AREA	Master untuk referensi data lokasi area kunjungan. Terdiri dari 5 (lima) area PT TMMIN, yaitu Head Office, Sunter 1, Sunter 2, Karawang 1&2, dan Karawang 3.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Nama Tabel	Deskripsi
TB_M_ATSG	Master untuk referensi data area kerja level ATSG berdasarkan area PT TMMIN.
TB_M_USER	Menyimpan data pengguna sistem seperti <i>host</i> , <i>security</i> , dan <i>admin</i> . <i>Field</i> yang disimpan meliputi <i>username</i> , <i>password</i> , dan <i>name</i> .
TB_M_ROLE	Master data yang mendefinisikan jenis <i>role</i> pengguna, yaitu <i>Guest</i> , <i>Admin</i> , <i>Host</i> , dan <i>Security/Receptionist</i> .
TB_M_MENU	Menentukan struktur dan hirarki menu dalam sistem melalui atribut <i>parent_id</i> .
TB_M_ACCESS_MENU	Menentukan hak akses pengguna terhadap menu berdasarkan <i>role_code</i> , sehingga sistem dapat mengontrol tampilan menu sesuai dengan peran pengguna.

b. Tabel Transaksional

Tabel 7 menunjukkan daftar tabel transaksional yang terdapat dalam database VMS berupa data-data hasil aktivitas (transaksi) yang dilakukan oleh pengguna atau sistem.

Tabel 7. Tabel Transaksional Database VMS

Nama Tabel	Deskripsi
TB_R_VISIT_APPROVAL	Menyimpan informasi terkait proses <i>approval</i> dari <i>host</i> terhadap <i>visitor</i> . <i>Field is_approve</i> menandakan status <i>approval</i> .
TB_R_GROUP_VISIT	Menyimpan detail <i>visitor</i> dalam kunjungan berkelompok (<i>group visit</i>)

TB_R_EMPLOYEE	Merupakan integrasi dari sistem HRIS yang menyimpan informasi lengkap tentang karyawan <i>internal</i> perusahaan yang berperan sebagai <i>host</i> dalam sistem. <i>Field</i> mencakup <i>noreg</i> , <i>email</i> , <i>position</i> , <i>department</i> , <i>division</i> , dan sebagainya.
---------------	---

c. Relasi Antar Tabel

- TB_M_VISIT terhubung dengan TB_R_VISIT_APPROVAL, TB_R_GROUP_VISIT, dan tabel-tabel master seperti *kategori*, *status*, *type*, *purpose*, dan *area*.
- TB_R_USER_ROLE menghubungkan antara pengguna dari tabel TB_M_USER dan peran/*role* dari tabel TB_M_ROLE.
- TB_M_USER terhubung dengan TB_R_USER_ROLE dan TB_M_ACCESS_MENU untuk pengaturan hak akses berdasarkan *role*.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

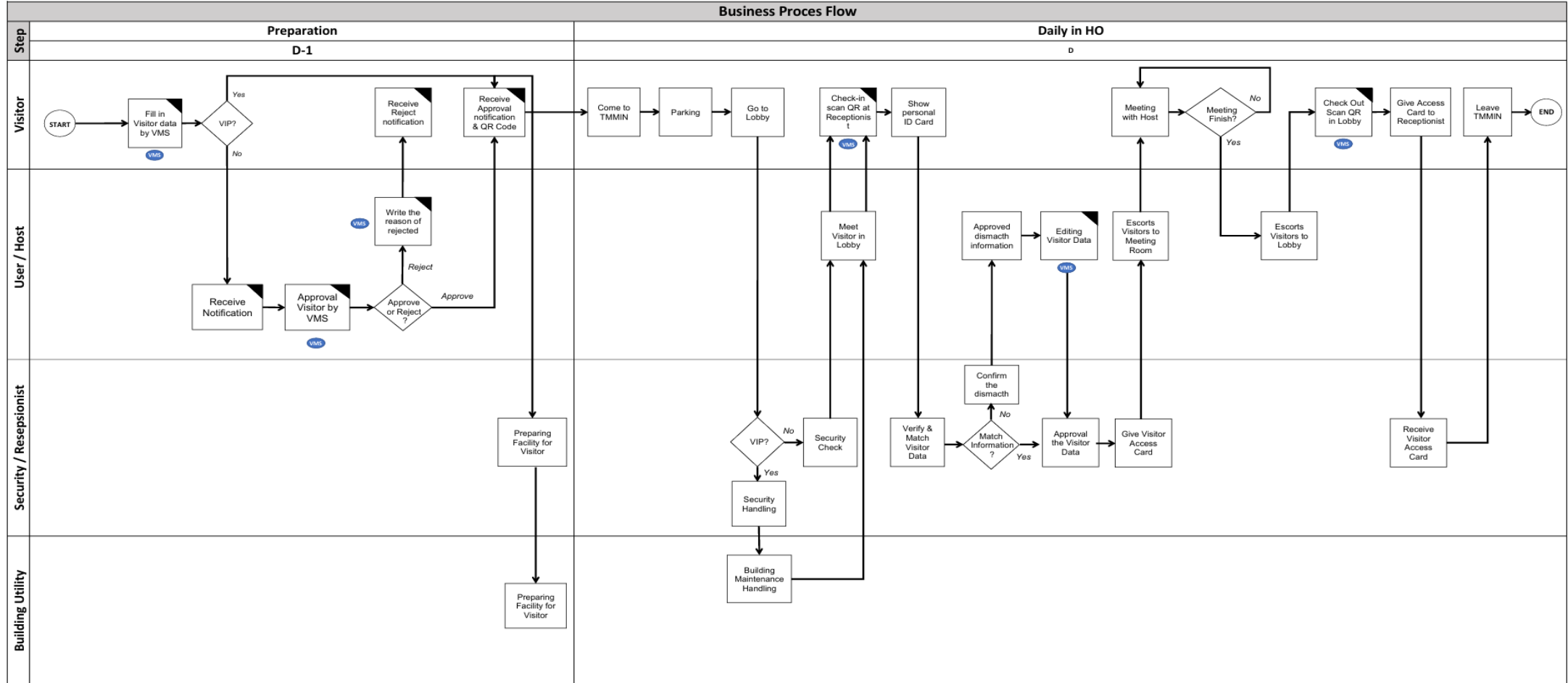
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

- Hak Cipta :**
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
 2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

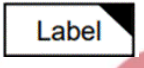
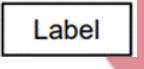
4.2.2 Business Process Flow



Gambar 4.2 Business Process Flow Sistem VMS

Gambar 4.2 merupakan alur proses bisnis *Visitor Management System* yang diadaptasi dari sistem operasional manajemen kunjungan di PT TMMIN. Tabel 8 menunjukkan simbol yang digunakan dalam pembuatan *Business Process Flow*.

Tabel 8. *Legend Business Process Flow*

Legend	Deskripsi
	Data Flow
	New system
	Current System
	VMS Apps

Business Process Flow pada sistem *Visitor Management System* yang dirancang terbagi ke dalam dua fase, yaitu *Preparation* (D-1) dan *Daily in HO* (D-day). Setiap proses digambarkan berdasarkan alur kegiatan masing-masing pihak yang terlibat, meliputi *Visitor*, *User/Host*, *Security/Receptionist*, serta *Building Utility*.

a. Tahap *Preparation* (D-1)

Tahapan ini merupakan proses pra-kunjungan yang dilakukan sebelum hari pelaksanaan kunjungan (H-1). Berikut ini adalah urutan proses pada tahap ini.

1. *Visitor* mengisi formulir kunjungan melalui sistem VMS tanpa perlu melakukan login.
2. Sistem akan mengirimkan notifikasi kepada *Host* untuk memberikan persetujuan terhadap permintaan kunjungan tersebut.
3. *Host* dapat memberikan persetujuan (*approve*) atau penolakan (*reject*). Jika ditolak, maka *Host* wajib mengisi alasan penolakan dan sistem akan mengirimkan notifikasi penolakan kepada *visitor*.
4. Jika permintaan disetujui, sistem akan mengirimkan notifikasi persetujuan kepada *visitor* beserta *QR Code* untuk keperluan *check-in* melalui *email*.

b. Tahap *Daily in HO* (D-day)

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Tahap ini mencakup aktivitas yang dilakukan pada hari kunjungan berlangsung, dimulai dari kedatangan *visitor* hingga meninggalkan area PT TMMIN. Prosesnya adalah sebagai berikut.

1. *Visitor* datang ke area PT TMMIN dan memarkirkan kendaraan di area parkir.
2. *Visitor* kemudian menuju *lobby* menemui *Host*.
3. Jika *visitor* merupakan kategori VIP, maka akan langsung diarahkan untuk penanganan khusus oleh *Security* dan *Building Utility*.
4. Jika bukan VIP, *visitor* akan melalui proses pengecekan keamanan oleh *Security* kemudian *check-in* dengan cara memindai *QR Code* yang telah dikirimkan sebelumnya.
5. *Visitor* menunjukkan kartu identitas pribadi kepada *Receptionist* sebagai bentuk verifikasi data.
6. Jika informasi yang ditampilkan sistem VMS tidak sesuai dengan kartu identitas, maka *Host* diharuskan mengedit data kunjungan pada sistem VMS.
7. Jika informasi yang ditampilkan sistem VMS sesuai dengan kartu identitas, maka *visitor* dapat melakukan *check-in* dan diberikan *visitor access card*.
8. *Visitor* kemudian diarahkan oleh *Host* untuk mengikuti agenda kunjungan.
9. Setelah kunjungan selesai, *visitor* diarahkan *Host* untuk kembali ke *lobby* dan melakukan proses *check-out* dengan memindai kembali *QR code* melalui sistem VMS.
10. *Visitor* menyerahkan kembali *visitor access card* kepada *Receptionist* dan meninggalkan area TMMIN.

4.2.3 Flowchart

Flowchart alur sistem berdasarkan masing-masing *role* terdiri dari *visitor*, *host*, *receptionist/security*, dan *admin*.

- a. Flowchart Sistem VMS *Role Visitor*

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

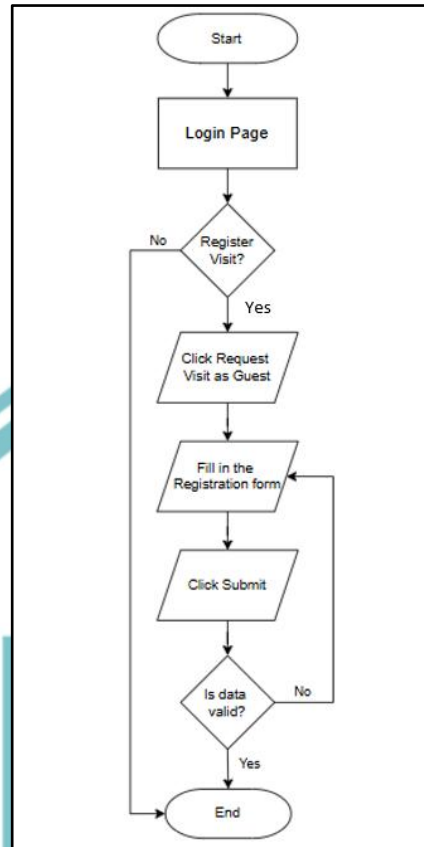
Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

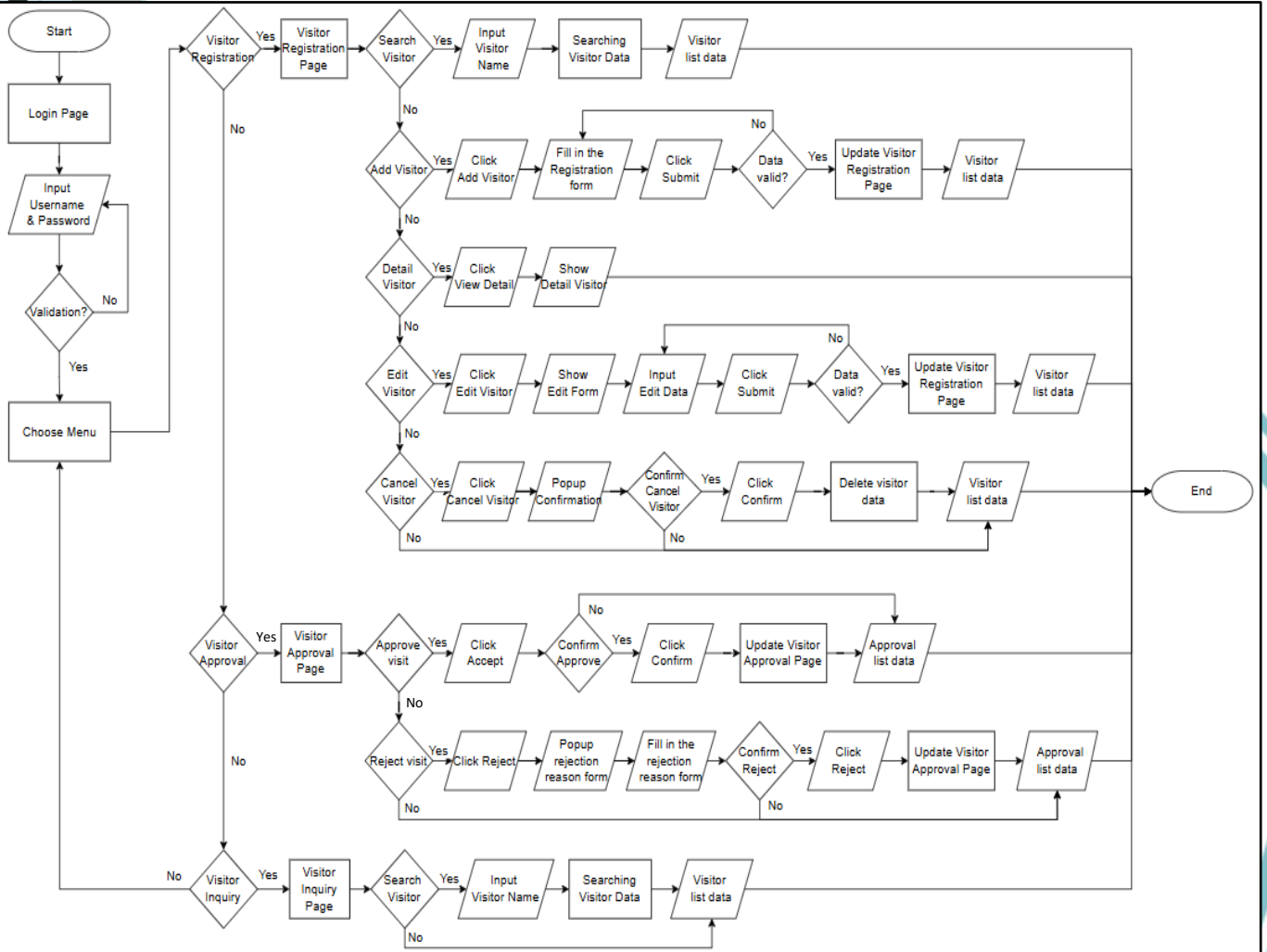


Gambar 4.3 Flowchart Sistem VMS Role Visitor

Flowchart pada Gambar 4.3 menunjukkan alur proses registrasi kunjungan yang dilakukan oleh *visitor* ketika menggunakan sistem *Visitor Management System* (VMS). Proses ini diawali *visitor* mengakses halaman *login*, *visitor* menekan tombol “*Request Visit as Guest*” untuk masuk ke halaman formulir registrasi tanpa perlu melakukan *login*. Setelah itu, *visitor* dapat mengisi formulir registrasi dan menekan tombol *submit* untuk mengirimkan data. Sistem kemudian akan memverifikasi apakah data yang diinput sudah valid. Apabila terdapat kesalahan atau data tidak lengkap, sistem akan mengarahkan *visitor* untuk kembali mengisi formulir dengan data yang benar. Jika seluruh data sudah valid, maka proses registrasi dinyatakan selesai. Alur ini dirancang untuk menyederhanakan proses registrasi kunjungan secara mandiri oleh pihak eksternal tanpa campur tangan langsung dari pihak perusahaan, serta memastikan bahwa data yang masuk ke sistem sudah sesuai sebelum diteruskan ke tahap berikutnya, yaitu proses *approval* dari pihak *host*.

b. Flowchart Sistem VMS *Role Host/User*

1. Di



Gambar 4.4 Flowchart Sistem VMS *Role Host*

Flowchart pada Gambar 4.4 merupakan alur proses dari sisi *host* atau karyawan PT Toyota Motor Manufacturing Indonesia dalam melakukan pengelolaan kunjungan melalui sistem *Visitor Management System* (VMS). Proses diawali dari halaman *login*, dimana *user* harus memasukkan *username* dan *password* terlebih dahulu. Apabila proses validasi berhasil, *user* dapat memilih menu utama yang terdiri dari tiga fungsi utama, yaitu: *Visitor Registration*, *Visitor Approval*, dan *Visitor Inquiry*.

Fungsi *Visitor Registration* bertujuan agar BOD dapat mendaftarkan tamu eksekutif yang dikategorikan sebagai VIP sehingga *visitor* tersebut tidak perlu mendaftar secara mandiri. Pada menu *Visitor Registration*, *host* dapat melakukan berbagai

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

aktivitas seperti mencari *visitor*, menambahkan *visitor* baru, melihat detail *visitor*, mengedit data *visitor*, hingga membatalkan kunjungan.

Pada menu *Visitor Approval*, *host* dapat memberikan persetujuan atau penolakan terhadap permintaan kunjungan. Jika *host* memilih untuk menyetujui kunjungan, maka sistem akan mengirimkan notifikasi kepada *visitor* melalui *email* yang sudah dicantumkan ketika registrasi. Sebaliknya, jika kunjungan ditolak, *host* perlu mengisi alasan penolakan terlebih dahulu melalui formulir khusus, kemudian melakukan konfirmasi penolakan.

Sementara itu, pada menu *Visitor Inquiry*, *host* dapat melakukan pencarian data *visitor* yang telah teregistrasi berdasarkan nama, kemudian sistem akan menampilkan daftar data sesuai dengan pencarian tersebut. Seluruh alur ini dirancang untuk memudahkan *host* dalam melakukan registrasi, *approval*, dan pelacakan data *visitor* secara efisien, sekaligus meminimalkan potensi kesalahan atau duplikasi data dengan adanya proses validasi yang ketat dalam sistem.

c. Flowchart Sistem *Receptionist/Security*

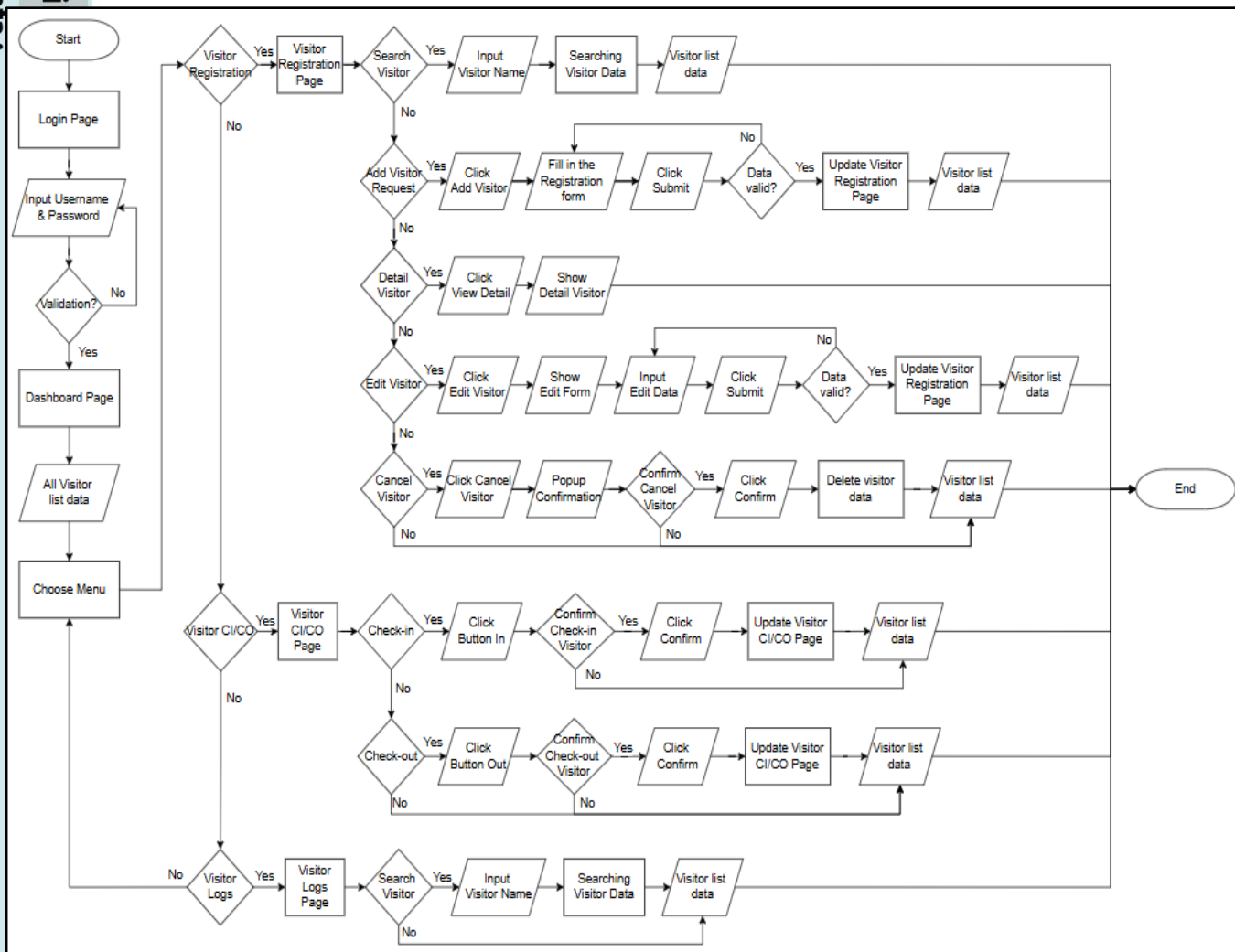
Gambar 4.5 menunjukkan *receptionist* atau *security* memulai menggunakan sistem dari halaman *login* dan melakukan validasi akun. Setelah berhasil *login*, *user* akan diarahkan ke halaman *Dashboard Page* yang menampilkan seluruh data *visitor* yang telah terdaftar di sistem. Fitur *Visitor CI/CO* digunakan oleh *receptionist/security* untuk melakukan proses *check-in* dan *check-out visitor* secara manual jika terdapat kendala ketika *scanning QR Code*. Pada fitur ini, *receptionist/security* dapat memilih *visitor* yang akan melakukan *check-in*, kemudian menekan tombol *Button In*.

Proses *check-out* memiliki alur serupa, dimana *receptionist/security* dapat menggunakan *Button Out* untuk mengonfirmasi aksi tersebut, dan sistem akan memperbarui status kunjungan *visitor* tersebut. Fitur ini berguna untuk memastikan bahwa seluruh aktivitas masuk dan keluar *visitor* tercatat secara *real-time*, terutama jika terjadi kendala pada proses otomatisasi atau jika sistem memerlukan *backup* data manual. Menu *Visitor Inquiry*, yang memungkinkan *receptionist/security* melakukan pencarian terhadap histori aktivitas *visitor* berdasarkan nama, serta

menampilkan daftar lengkap log kunjungan sebagai referensi atau keperluan pelaporan internal.

Hak Cipta :

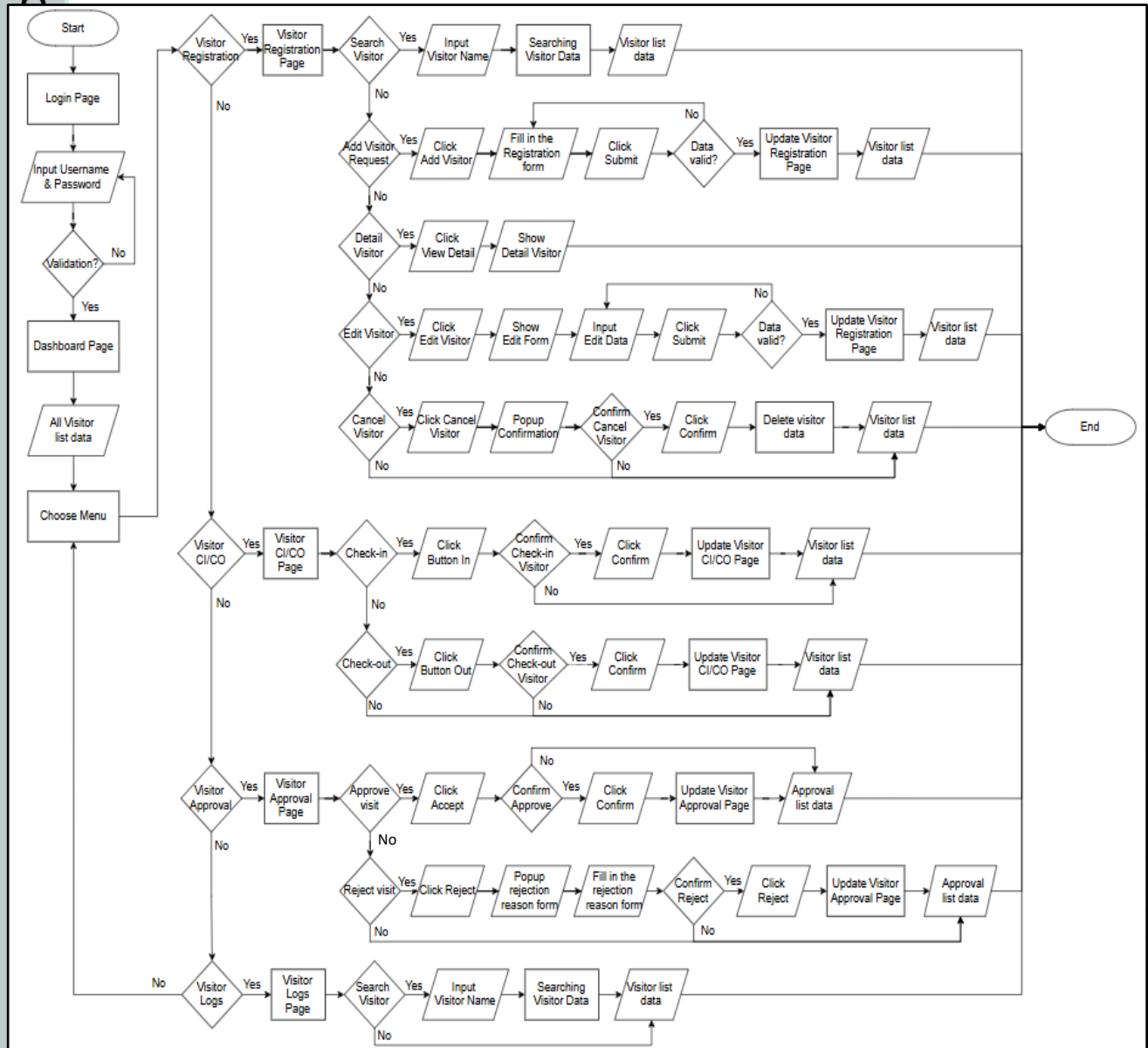
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



Gambar 4.5 Flowchart Sistem VMS Role Receptionist/Security

d. Flowchart Sistem Admin

Gambar 4.6 menunjukkan fungsi utama admin meliputi *Visitor Registration*, *Visitor Approval*, *Visitor CI/CO*, dan *Visitor Inquiry*. Admin dapat melakukan registrasi *visitor* baru, mengedit, melihat detail, atau membatalkan *request*. Pada menu *Approval*, admin berwenang untuk menerima atau menolak kunjungan *visitor*, dengan mengisi alasan penolakan bila diperlukan. Selain itu, admin juga dapat melakukan *Check-in/Check-out visitor* secara manual jika diperlukan. Admin dapat mengakses *Visitor Inquiry* untuk menelusuri histori kunjungan berdasarkan nama *visitor*, yang berguna untuk keperluan monitoring dan pelaporan.



Gambar 4.6 Flowchart SistemVMS Role Admin

4.2.4 Mockup

Tahap perancangan *mockup* ini menyajikan rancangan visual dari sistem yang dibangun. Seluruh rancangan antarmuka pengguna (*User Interface*) ini dibuat menggunakan *tools* Figma untuk menerjemahkan fungsionalitas dan memberikan gambaran yang jelas mengenai tampilan serta fungsionalitas sistem sebelum masuk ke tahap pengembangan.

a. Halaman Login

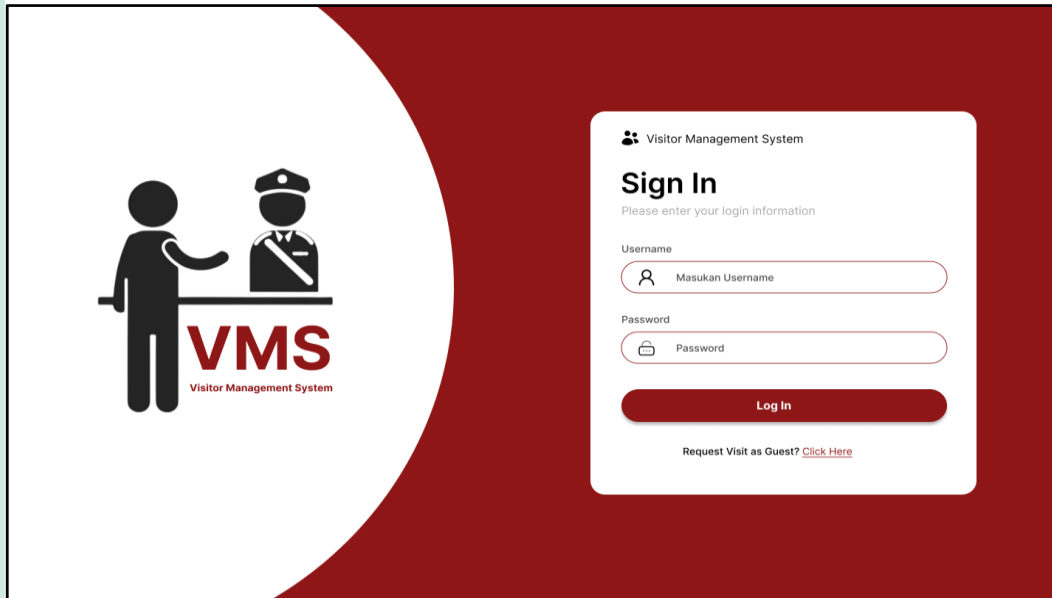
Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



Gambar 4.7 Mockup Halaman Login

Halaman pada Gambar 4.7 berfungsi sebagai lapisan keamanan pertama untuk memastikan hanya pengguna yang berwenang dapat mengakses sistem. Pengguna terdaftar seperti Admin, Karyawan (*Host*), dan Petugas Keamanan/Resepsionis (*Security/Receptionist*) harus memasukkan *username* dan *password* untuk mendapatkan akses ke *dashboard* dan fitur-fitur internal lainnya. Selain itu, terdapat tautan “*Request Visit as Guest*” sebagai gerbang *visitor* untuk memulai proses registrasi kunjungan tanpa harus memiliki akun.

b. Halaman Dashboard

Gambar 4.8 merupakan halaman *dashboard* sebagai pusat informasi yang memberikan visualisasi terkait data kunjungan. *Dashboard* ini dapat diakses hanya oleh *role* Admin dan *Security/Receptionist*. Halaman ini menampilkan data-data penting, seperti jumlah total *visitor* hari ini, *visitor* yang masih berada di dalam ataupun keluar area (*check-in/out*), statistik historis pengunjung dalam rentang 7 hari, serta rincian pengunjung berdasarkan kategori VIP dan Non-VIP. Selain itu, terdapat tabel yang menampilkan status *real-time* dari *visitor* yang telah melakukan *check-in* atau *check-out* pada hari H.

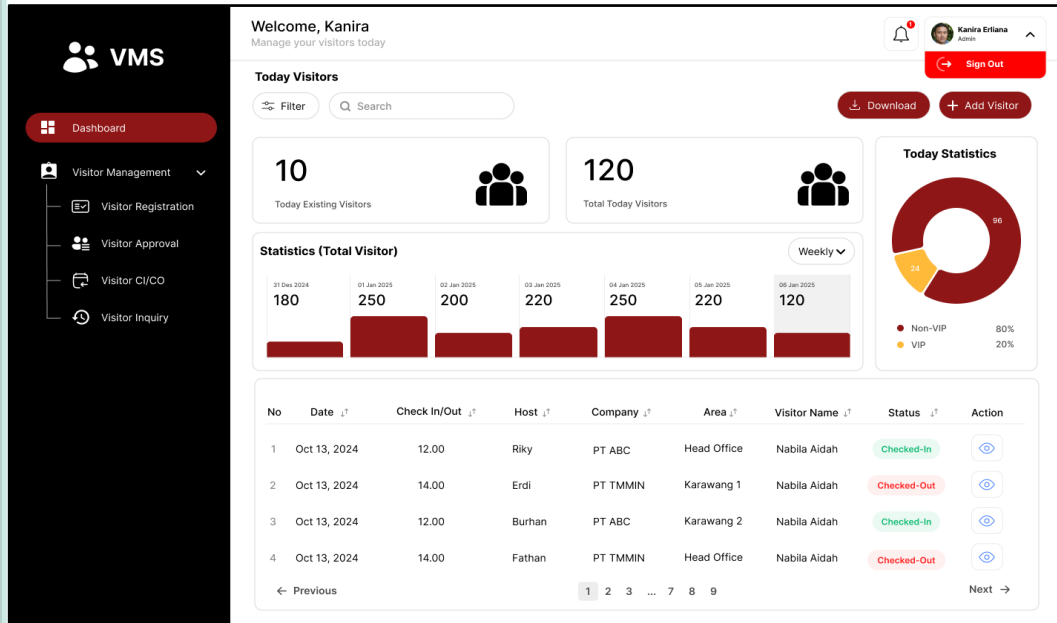
Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



Gambar 4.8 Mockup Halaman Dashboard

c. Halaman Registrasi (*Guest/Visitor*)

Halaman *form* registrasi di bawah ini diakses dan diisi oleh pengunjung atau *visitor*. Halaman ini dapat diakses melalui tautan “*Request Visit as Guest*” pada halaman Login. *Form* dirancang untuk mengakomodasi dua jenis kunjungan, yaitu individual dan grup. Gambar 4.9 merupakan tampilan utama *form* registrasi untuk *visitor* ketika tipe kunjungan “*Individual*”.

Form tersebut bertujuan mengumpulkan seluruh informasi yang diperlukan dari satu orang *visitor* untuk mengajukan permohonan kunjungan. Pada bagian *host detail*, *visitor* wajib mengisi nama karyawan internal yang akan ditemui (*Host*). Terdapat fitur pencarian untuk memudahkan menemukan nama *host* yang terdaftar menggunakan *dropdown*. Gambar 4.10 merupakan tampilan ketika pengunjung memilih tipe kunjungan “*Group*”. *Form* tersebut dirancang untuk memudahkan pendaftaran kunjungan yang melibatkan banyak orang sekaligus.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

DETAILS REQUEST VISIT

HOST DETAIL
 Host Name *

VISITOR DETAILS
 Type *
☒ Individual ☐ Group

Purpose Category

Full Name *

Company *

Email *

No Phone *

From To

In Out

TMMIN Area

ATSG

Gambar 4.9 Mockup Halaman Form Registrasi Individual

Alih-alih mengisi data untuk setiap orang, formulir pada Gambar 4.10 meminta data satu orang perwakilan atau *Person in Charge* (PIC) dari grup tersebut. Seperti yang tertera pada catatan, PIC akan menerima semua komunikasi utama, termasuk email konfirmasi dan QR code yang berlaku untuk grup.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

DETAILS REQUEST VISIT

HOST DETAIL

Host Name *

Search Host

VISITOR DETAILS

Type *

☐ Individual
☒ Group

Purpose
Category

PIC Full Name * (Group Representative that will received the QR Code)

Name

Company *

Company

Email *

mail@example.com

No Phone *

08xxxxxxxxxx

Total List of Visitor * (Whole Group, Excluding PIC's name)

1
Name
Company
Phone No

+ Add Visitor

or

Choose File
No file chosen

Download Template Here

From
To

Select date

Select date

In
Out

--:--

--:--

TMMIN Area

ATSG

Submit

Gambar 4.10 Mockup Halaman Form Registrasi Grup

Terdapat dua cara untuk mendaftarkan anggota grup.

- *Input Field Grup*: Menambahkan anggota satu per satu dengan mengisi nama, perusahaan, dan nomor telepon, lalu menekan tombol “Add Visitor” untuk menambah baris isian baru.

Jurusan Teknik Informatika dan Komputer – Politeknik Negeri Jakarta

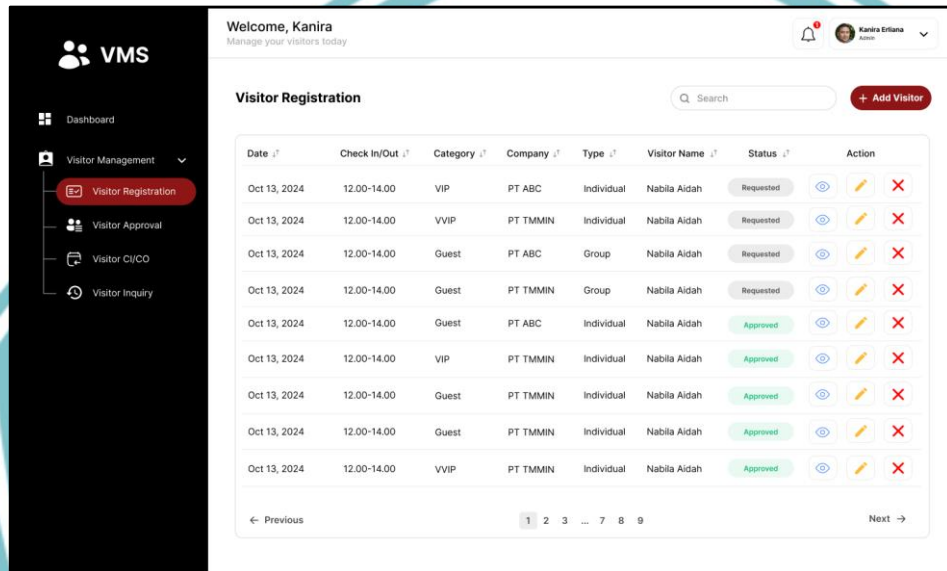
Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

- *Upload File*: Grup dengan jumlah anggota yang banyak, disediakan opsi untuk mengunggah daftar anggota secara massal. Pengguna dapat mengunduh *template* yang telah disediakan “*Download Template Here*”, mengisinya, lalu mengunggah kembali file tersebut.
- d. Halaman Registrasi (*Host, Admin, Security, Receptionist*)



Date	Check In/Out	Category	Company	Type	Visitor Name	Status	Action
Oct 13, 2024	12:00-14:00	VIP	PT ABC	Individual	Nabila Aidah	Requested	View Edit Delete
Oct 13, 2024	12:00-14:00	VVIP	PT TMMIN	Individual	Nabila Aidah	Requested	View Edit Delete
Oct 13, 2024	12:00-14:00	Guest	PT ABC	Group	Nabila Aidah	Requested	View Edit Delete
Oct 13, 2024	12:00-14:00	Guest	PT TMMIN	Group	Nabila Aidah	Requested	View Edit Delete
Oct 13, 2024	12:00-14:00	Guest	PT ABC	Individual	Nabila Aidah	Approved	View Edit Delete
Oct 13, 2024	12:00-14:00	VIP	PT TMMIN	Individual	Nabila Aidah	Approved	View Edit Delete
Oct 13, 2024	12:00-14:00	Guest	PT TMMIN	Individual	Nabila Aidah	Approved	View Edit Delete
Oct 13, 2024	12:00-14:00	Guest	PT TMMIN	Individual	Nabila Aidah	Approved	View Edit Delete
Oct 13, 2024	12:00-14:00	VVIP	PT TMMIN	Individual	Nabila Aidah	Approved	View Edit Delete

Gambar 4.11 Mockup Halaman Tabel Registrasi Kunjungan

Tampilan halaman Gambar 4.11 berfungsi sebagai pusat untuk mendaftarkan dan mengelola data pengunjung yang akan datang. Seluruh data kunjungan baru, baik yang didaftarkan oleh *admin*, *host*, maupun *security/receptionist*, akan tercatat. Halaman ini menampilkan detail seperti tanggal, waktu, perusahaan, nama pengunjung, serta status pendaftaran. Pengguna dapat menambah pengunjung baru, serta melihat, mengubah, atau menghapus data registrasi yang ada. Jika klik tombol “*Add Visitor*” maka tampilan *modal form* yang memiliki struktur *input field* seperti Gambar 4.9 dan Gambar 4.10.

e. Halaman *Approval*

Halaman pada Gambar 4.12 berfungsi sebagai “kotak masuk” untuk permintaan kunjungan yang memerlukan persetujuan. Di halaman ini, pengguna yang berwenang dapat melihat daftar permintaan kunjungan yang tertunda dan

melakukan aksi untuk menyetujui (*Approve*) atau menolak (*Reject*) kunjungan tersebut.

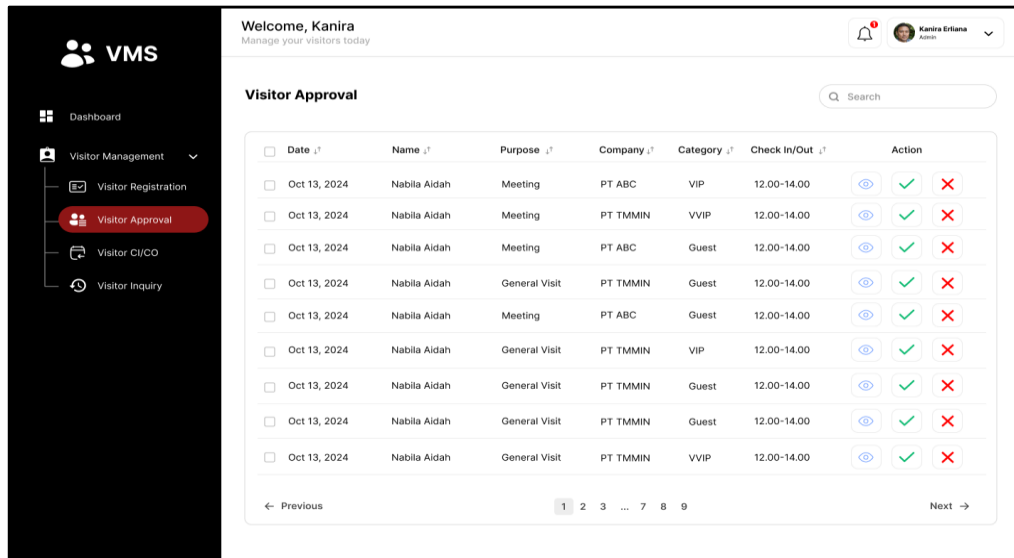
Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

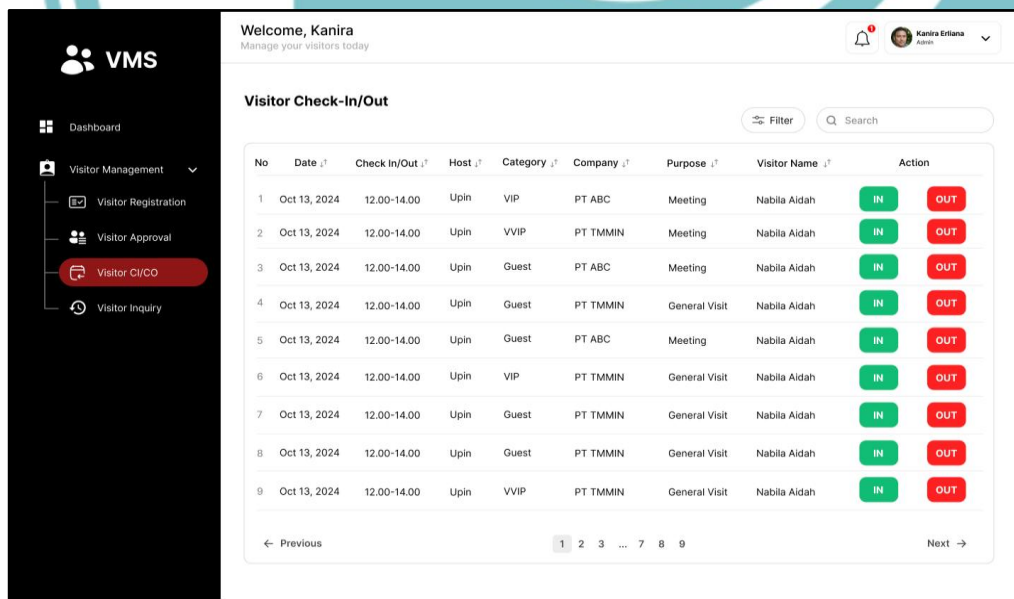
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



Gambar 4.12 Mockup Halaman Approval

f. Halaman Check-In/Out



Gambar 4.13 Mockup Halaman Check-In/Out

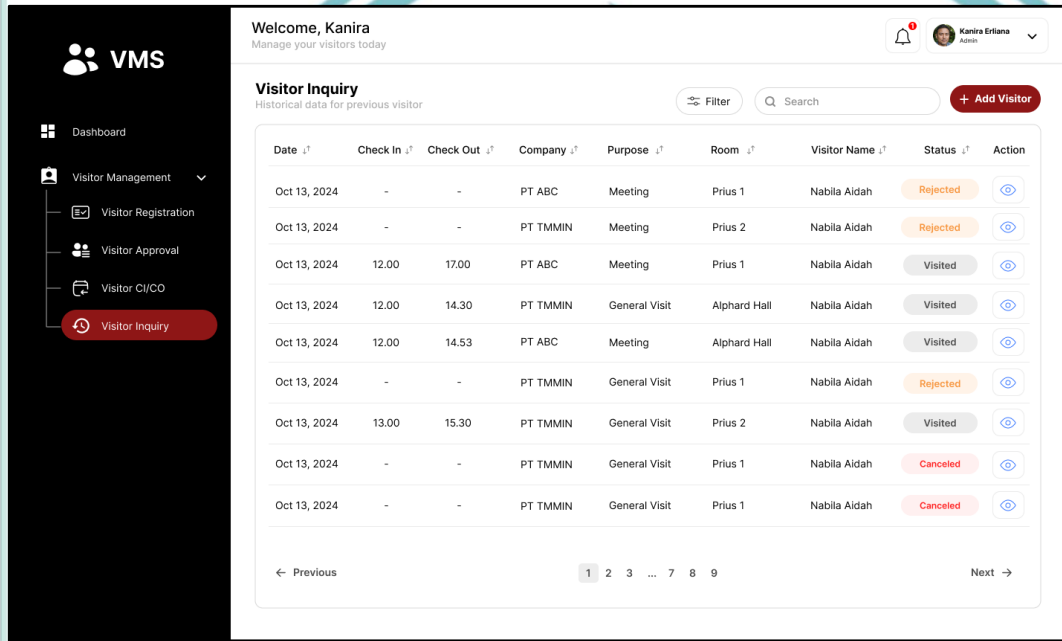
Gambar 4.13 menunjukkan halaman *check-in/out* yang dioperasikan oleh petugas di garda terdepan (*Security/Receptionist*) ketika pengunjung mengalami masalah *scanning QR code*. Halaman ini menampilkan daftar semua pengunjung yang telah disetujui (*Approve*) dan dijadwalkan untuk datang sesuai tanggal

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

kunjungan. Petugas dapat menekan tombol “IN” untuk mencatat waktu kedatangan (*check-in*). Sebaliknya, petugas dapat menekan tombol “OUT” untuk mencatat waktu pengunjung keluar area (*check-out*). Tombol “IN” akan otomatis *disable* jika pengunjung sudah melakukan *check-in*. Tombol “OUT” akan otomatis *disable* jika pengunjung belum melakukan *check-in*.

g. Halaman *Inquiry*



Date	Check In	Check Out	Company	Purpose	Room	Visitor Name	Status	Action
Oct 13, 2024	-	-	PT ABC	Meeting	Prius 1	Nabila Aidah	Rejected	
Oct 13, 2024	-	-	PT TMMIN	Meeting	Prius 2	Nabila Aidah	Rejected	
Oct 13, 2024	12.00	17.00	PT ABC	Meeting	Prius 1	Nabila Aidah	Visited	
Oct 13, 2024	12.00	14.30	PT TMMIN	General Visit	Alphard Hall	Nabila Aidah	Visited	
Oct 13, 2024	12.00	14.53	PT ABC	Meeting	Alphard Hall	Nabila Aidah	Visited	
Oct 13, 2024	-	-	PT TMMIN	General Visit	Prius 1	Nabila Aidah	Rejected	
Oct 13, 2024	13.00	15.30	PT TMMIN	General Visit	Prius 2	Nabila Aidah	Visited	
Oct 13, 2024	-	-	PT TMMIN	General Visit	Prius 1	Nabila Aidah	Canceled	
Oct 13, 2024	-	-	PT TMMIN	General Visit	Prius 1	Nabila Aidah	Canceled	

Gambar 4.14 Mockup Halaman *Inquiry*

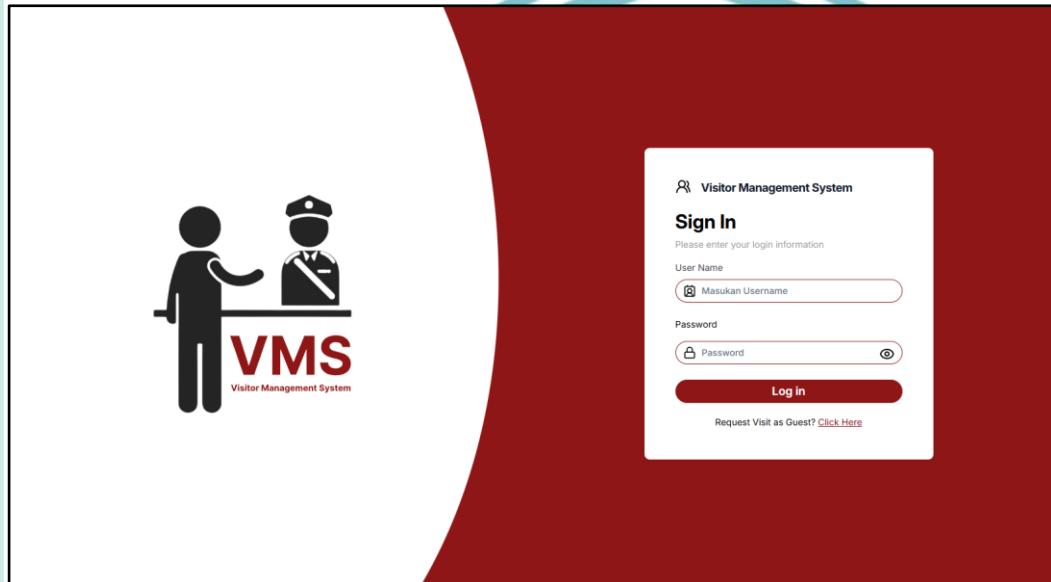
Tampilan halaman pada Gambar 4.14 berfungsi sebagai arsip atau pusat data historis dari seluruh pengunjung. Pengguna dapat mencari, menyaring (*filter*), dan melihat riwayat lengkap dari setiap kunjungan. Informasi yang ditampilkan mencakup detail kunjungan serta status akhir dari kunjungan tersebut, seperti “*Check-out*” (telah berkunjung), “*Rejected*” (ditolak), atau “*Canceled*” (dibatalkan).

4.3 Implementasi Sistem

Tahap awal pengembangan sistem difokuskan pada implementasi fitur autentikasi untuk masuk ke sistem. Proses implementasi untuk autentikasi pengguna dijelaskan sebagai berikut.

4.3.1 Implementasi Autentikasi User

Gambar 4.15 merupakan tampilan halaman dari *login*, pada halaman ini juga *guest* dapat mengakses halaman registrasi tanpa melakukan *login* dengan mengklik link *click here* di bawah *button Log In*.



Gambar 4.15 Implementasi Halaman *Login*

Berikut ini merupakan implementasi yang diterapkan untuk proses autentikasi antara pengguna terdaftar dan *visitor*.

4.3.1.1 Login Pengguna Terdaftar

Login dilakukan oleh karyawan, *receptionist*, dan *security* PT Toyota Motor Manufacturing Indonesia yang terdaftar di dalam *database*. Proses *login* diinisiasi dengan mendefinisikan *input username* dan *password* pengguna yang dijabarkan dalam Gambar 4.16.

Penggalan kode Gambar 4.16 dalam *form login* terdapat *loginInputSchema* yang dipanggil sebagai validasi data *input* menggunakan *Zod* untuk memastikan *username* diisi minimal 1 karakter dan *password* minimal 5 karakter. Kemudian *loginInputTypes* menghasilkan tipe data *input* berdasarkan hasil *z.infer* dari *loginInputSchema*. Dapat dikatakan *loginInputTypes* menentukan struktur dan tipe data *input* yang berasal dari *loginInputSchema*. Jika validasi data pada *loginInputSchema* berubah, maka tipe data *input* juga akan terupdate otomatis.

Hak Cipta:

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

```
import { z } from 'zod';

export const loginInputSchema = z.object({
  username: z.string().min(1, 'Required'),
  password: z.string().min(5, 'Required'),
});

export type loginInputTypes = z.infer<typeof loginInputSchema>;
```

Gambar 4.16 Sumber Kode Validasi *Input Login*

Pada penggalan kode Gambar 4.17 menunjukkan proses *postLogin*, fungsi *creatAsyncThunk* yang berguna untuk mengirim data pengguna dengan parameter *username* dan *password* berdasarkan *loginInputTypes*. Data pengguna tersebut dikirimkan melalui API ke *endpoint* *‘/auth/login’* untuk divalidasi, jika *login* berhasil (*isLogin* = *true*) maka berlanjut pengecekan *role* seperti gambar 4.9. Jika *login* gagal maka akan mengembalikan nilai *error*.

```
export const postLogin = createAsyncThunk(
  'auth/login',
  async ({ username, password }: loginInputTypes, { rejectWithValue }) => {
    try {
      const response: AuthResponse = await api.post(
        `${env.API_URL}/auth/login`,
        { username, password },
      );

      return response;
    } catch (error) {
      return rejectWithValue(error);
    }
  },
);
```

Gambar 4.17 Sumber Kode *Post Login*

Hak Cipta:

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

```
const [user] = useUser();
const navigate = useNavigate();

useEffect(() => {
  if (user.isLogin) {
    const checkRole = user.data?.roles.find(
      (roles: any) =>
        roles.role_code === 'HOST' ||
        roles.role_code === 'SECRETARY' ||
        roles.role_code === 'TAM_REGISTER' ||
        roles.role_code === 'TAM_APPROVE',
    );

    if (!checkRole) {
      navigate('/app', {
        replace: true,
      });
    } else {
      navigate(paths.app.visitor.register.getHref(), {
        replace: true,
      });
    }
  }
}, [user.data, navigate]);
```

Gambar 4.18 Sumber Kode *User Login Berdasarkan Role*

Penggalan kode pada Gambar 4.18 menunjukkan *useEffect* yang digunakan untuk menentukan *redirect* halaman berdasarkan *role user* yang teridentifikasi ketika *login*. Jika *user* sudah *login* (*user.isLogin* bernilai *true*), maka akan dilakukan pengecekan terhadap *role user*. Fungsi *find* digunakan untuk mencari *role user* dari empat *roles*, yaitu *HOST*, *SECRETARY*, *TAM_REGISTER*, atau *TAM_APPROVE*. Jika *role user* yang *login* bukan diantara empat *roles* tersebut, maka *user* akan diarahkan ke halaman dashboard “/app”. Namun, jika *role user* yang ditemukan merupakan salah satu dari empat *roles* tersebut, *user* akan diarahkan langsung ke halaman *Visitor Registration* (*paths.app.visitor.register.getHref()*). Proses navigasi menggunakan *{ replace: true }* untuk menggantikan entri di *history browser*, sehingga *user* tidak dapat kembali ke halaman sebelumnya menggunakan tombol “Back”.

4.3.1.2 Implementasi Guest Access

Berbeda dengan pengguna lain yang harus *login*, *visitor* (pengunjung) diberikan *role* khusus yaitu '*GUEST*'. *Role* ini memungkinkan *visitor* (*guest*) mendapatkan *token* JWT tanpa perlu login yang berfungsi sebagai kunci akses untuk membuka halaman *form* registrasi. Berikut ini adalah implementasi untuk proses akses bagi *visitor*.

a. Generate Token JWT untuk Guest

Di bawah ini merupakan penggalan kode fungsi untuk membentuk *token* JWT khusus *guest*.

```
async guestToken(): Promise<ServiceResponse<AuthResponse | null>> {
  try {
    const token = createToken({ id: 'guest' });

    const guestResponse: AuthResponse = {
      userId: 'guest',
      UserName: 'guest',
      Name: 'Guest User',
      UserImage: '',
      roles: [{ role_code: 'GUEST' }],
      token,
    };

    return ServiceResponse.success('Guest Token Generated', guestResponse);
  } catch (error) {
    return ServiceResponse.failure(
      'Failed to generate guest token',
      null,
      StatusCodes.INTERNAL_SERVER_ERROR,
    );
  }
}
```

Gambar 4.19 Sumber Kode *Token Guest*

Gambar 4.19 menampilkan fungsi *guestToken* yang bertugas menyediakan detail autentikasi untuk pengguna *guest* dan mengembalikan hasilnya yang terstruktur dalam format *ServiceResponse*. Dalam penggalan kode tersebut, *createToken* membuat sebuah *token* JWT yang berisi identitas '*guest*'. *guestResponse* berisi informasi untuk pengguna *guest* seperti *userId*, *UserName*, *name*, peran (*roles*) yang ditetapkan sebagai '*GUEST*', dan *token* JWT yang sudah terbentuk. Apabila

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

seluruh proses ini berjalan sukses, *guestToken* akan mengembalikan *response* melalui pemanggilan *ServiceResponse.success* yang berisi pesan konfirmasi “*Guest Token Generated*” beserta objek *guestResponse* yang telah disiapkan. Namun, jika terjadi kendala atau kesalahan selama proses pembuatan *token* maka blok penanganan kesalahan akan mengembalikan *response error* melalui *ServiceResponse.failure* dengan menyertakan pesan “*Failed to generate guest token*”.

b. *Endpoint Token JWT Guest*

Gambar 4.20 menampilkan proses mendefinisikan *endpoint API* yang berfungsi untuk menyediakan *token guest*.

```
authRegistry.registerPath({
  method: 'get',
  path: '/auth/guest-token',
  tags: ['Auth'],
  responses: {
    ...createApiResponse(AuthResponseSchema, 'Success', StatusCodes.OK),
  },
});

authRouter.get('/guest-token', authController.guestToken);
```

Gambar 4.20 Sumber Kode API *Token*

Pada Gambar 4.20 *authRegistry.registerPath()* mengawali proses pendaftaran *endpoint* dengan menetapkan bahwa *endpoint* ‘*/auth/guest-token*’ akan merespons permintaan dengan metode GET. Untuk membantu organisasi dalam dokumentasi, baris *tags: ['Auth']* mengelompokkan *endpoint* tersebut di bawah kategori ‘Auth’. Sehingga membuat dokumentasi API lebih mudah dinavigasi karena semua *endpoint* terkait autentikasi akan terkumpul dalam satu grup. Jika *success* maka *response* akan mengembalikan nilai berdasarkan *AuthResponseSchema* yang merupakan skema Zod yang menentukan struktur data dari body *response* dan kode status 200 (OK), menandakan bahwa permintaan telah berhasil diproses. Setelah mendefinisikan API, *authRouter* memberikan *response* ketika server menerima permintaan dengan metode GET ke *path* ‘*/guest-token*’. Kemudian *token JWT*

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

dikirimkan kepada pengguna yang berasal dari fungsi *guestToken* pada gambar 4.19.

c. fungsi *getGuestToken* pada link akses halaman *form*

```
export const getGuestToken = async () => {  
  const response: AuthResponse = await api.get('/auth/guest-token');  
  console.log('Guest token response:', response);  
  return response;  
};
```

Gambar 4.21 Sumber Kode fungsi *getGuestToken*

Pada penggalan kode Gambar 4.21 menampilkan fungsi *getGuestToken()* yang berfungsi mengambil *token* JWT untuk *guest* yang sudah disediakan di *endpoint* *‘/auth/guest-token’*.

```
const handleGuestAccess = async () => {  
  try {  
    const guest = await getGuestToken();  
  
    dispatch(  
      setState({  
        token: guest.token,  
        data: guest,  
      }),  
    );  
    setTimeout(() => {  
      navigate('/guest/registration');  
    }, 1000);  
  } catch (error) {  
    console.error('Failed to get guest token:', error);  
  }  
};
```

Gambar 4.22 Sumber Kode Akses Halaman *Guest*

Setelah menekan *link* “*click here*” pada halaman *login*, penggalan kode pada Gambar 4.22 menampilkan fungsi *handleGuestAccess* yang memanggil

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

getGuestToken() pada Gambar 4.21 agar dapat mengakses halaman *form* registrasi (*/guest/registration*) sehingga dapat melakukan *request* kunjungan. Jika gagal mendapatkan *token* JWT maka akan muncul *error*.

d. Hasil *Response API* untuk *Guest*

Gambar 4.23 merupakan hasil *response* yang disimpan sistem dalam bentuk objek sesuai dengan struktur *guestResponse* gambar 4.19 mencakup *token* JWT untuk *guest*.



Gambar 4.23 Hasil *Response API Token Guest*

4.3.2 Implementasi Dashboard

Menu *dashboard* memiliki fungsi utama untuk menampilkan seluruh informasi terkait kunjungan. Berikut ini implementasi proses di halaman *dashboard*.

4.3.2.1 Implementasi Grafik & Tabel Pengunjung

Fungsi *fetchVisitors* merupakan fungsi inti pada komponen *dashboard* yang bertanggung jawab untuk mengambil, memfilter, dan mempersiapkan seluruh data yang diperlukan untuk ditampilkan di *interface*, termasuk grafik dan tabel pengunjung.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

```
const fetchVisitors = async (
  page: number,
  search: string,
  startDate: string,
  endDate: string,
  area?: string | undefined,
) => {
  setIsLoading(true);
  setError(null);
  const today = moment();

  try {
    const skip = (page - 1) * itemsPerPage;
    const response = await api.get(
      `/visitor?skip=${skip}&take=${itemsPerPage}&search=${search}&dateTo=${endDate ?? moment(today).endOf('days').toISOString(false)}&dateFrom=${startDate ?? moment(today).startOf('days').toISOString(false)}${area ? '&area=${area}' : ''}&page=dashboard`,
    );

    const data = response.data;
    // ... proses selanjutnya ...
  }
}
```

Gambar 4.24 Sumber Kode *Fetch Visitor Dashboard*

Gambar 4.24 menunjukkan fungsi *fetchVisitors*, dimana mengatur *state isLoading* menjadi *true* untuk menampilkan indikator pemuatan di antarmuka dan *setError(null)* untuk membersihkan pesan *error* dari proses sebelumnya. Selanjutnya, sistem membuat sebuah permintaan data ke API menggunakan *api.get*. Dimana URL disusun berdasarkan parameter yang diterima, seperti *page* untuk paginasi, *search* untuk kata kunci pencarian, *startDate* dan *endDate* untuk rentang waktu, serta *area* untuk filter lokasi. Setelah permintaan berhasil, keseluruhan data yang diterima dari server disimpan dalam variabel *data* untuk diolah lebih lanjut.

```
// ...
const data = response.data;
setVisitors(data.visitor ?? []);

const nonRejectedVisitors = data.visitor.filter(
  (visitor: VisitorDetailsAll) =>
    visitor.status !== 'REJECTED' && visitor.status !== 'CANCEL',
);
setVisitorAll(nonRejectedVisitors);

setTotalPages(Math.ceil(nonRejectedVisitors.length / itemsPerPage));

setTotalPerday(data.countPerdays);
setTotalData(data.count);
// ... proses selanjutnya ...
```

Gambar 4.25 Sumber Kode Lanjutan *Fetch Visitor Dashboard*

Setelah data diterima, kemudian dipilah dan disiapkan untuk ditampilkan di tabel. Pada Gambar 4.25 menunjukkan data pengunjung (*data.visitor*) disaring menggunakan metode *filter()* untuk membuat sebuah *array* baru bernama *nonRejectedVisitors*. Proses ini bertujuan untuk menyingkirkan semua pengunjung yang statusnya “*REJECTED*” atau “*CANCEL*”. *Array* yang sudah bersih kemudian disimpan ke dalam *state visitorAll* dan akan menjadi sumber data utama untuk ditampilkan di tabel. Berdasarkan jumlah data di *nonRejectedVisitors*, sistem menghitung jumlah total halaman (*setTotalPages*) yang diperlukan untuk paginasi. Di samping itu, jumlah pengunjung hari ini (*data.countPerdays*) dan total pengunjung yang ada (*data.count*) langsung diambil dari respons API dan disimpan ke dalam masing-masing *state setTotalPerday* dan *setTotalData*.

```
// ...
// Weekly visitor data for BarChart
const weeklyData = data.countVisitorPerWeeks.map(
  (item: { visit_date_from: string; count: number }) => ({
    date: moment(item.visit_date_from).format('DD/MM/YYYY'),
    value: item.count,
  }),
);
setBarData(weeklyData);
// ... proses selanjutnya ...
```

Gambar 4.26 Sumber Kode *Bar Chart*

Penggalan kode pada Gambar 4.26 bertujuan untuk menyiapkan data statistik pengunjung mingguan yang ditampilkan pada grafik batang. Dimulai dengan mengambil data yang berasal dari *data.countVisitorPerWeeks*, yang merupakan sebuah *array* yang diterima dari API. Kemudian, fungsi *map()* dijalankan untuk melakukan transformasi pada setiap *item* dalam *array* yang berisi *visit_date_from* dan *count*. Sistem membuat sebuah objek baru dengan struktur yang berbeda. Properti *date* diisi *item.visit_date_from* dengan mengubah format menjadi DD/MM/YYYY. Sementara itu, properti *value* diisi dari nilai *item.count*. Hasil dari proses *map()* ini adalah sebuah *array* baru bernama *weeklyData* yang berisi objek-objek dengan format { *date*: '...', *value*: ... }. Kemudian, fungsi

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

setBarData(weeklyData) dipanggil untuk menyimpan data yang telah diformat ke dalam *state*, yang secara otomatis memicu pembaruan pada tampilan grafik batang di antarmuka pengguna.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

```
// ...
// Pie chart data for visitor categories
let categoryCounts: Record<string, number> = {};
if (visitCategoryData) {
  categoryCounts = objectSetter(
    categoryCounts,
    visitCategoryData?.map((item: any) => item.label),
  );

  data.countPerdaysCategory.forEach(
    (item: {
      _count: { visitCategory: number };
      visitCategory: string;
    }) => {
      if (categoryCounts[item.visitCategory] !== undefined) {
        categoryCounts[item.visitCategory] = item._count.visitCategory;
      }
    },
  );

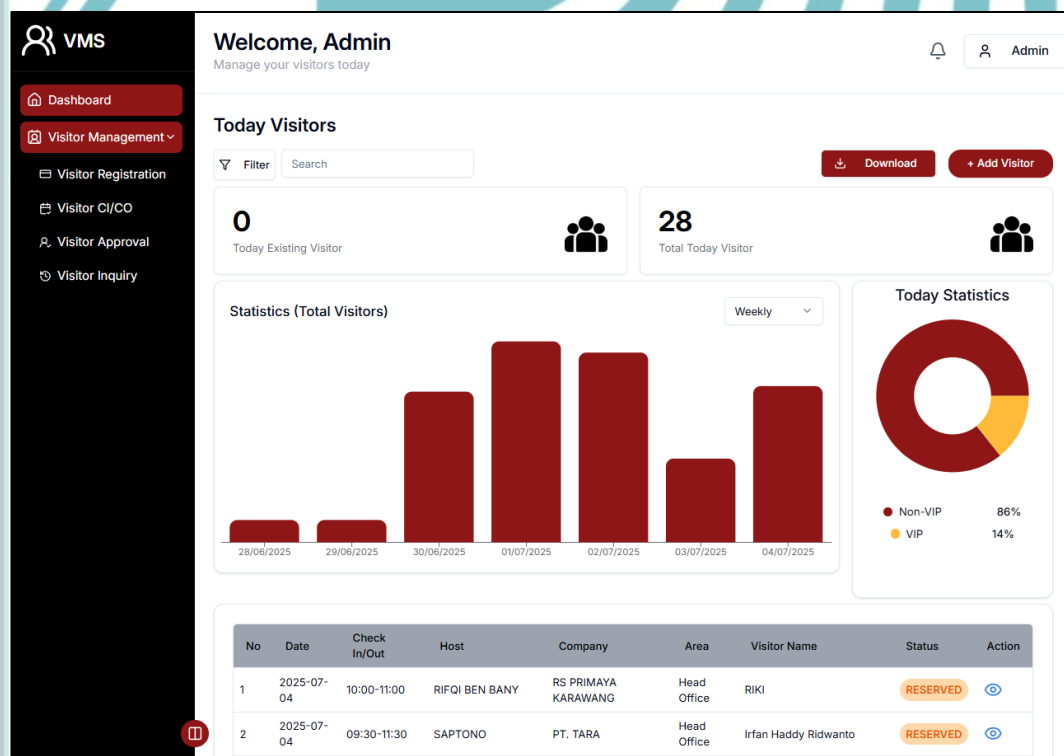
  setPieData(
    visitCategoryData?.map((item, index) => ({
      name: item.label,
      value: categoryCounts[item.label] || 0,
      color: pieData[index].color,
    })),
  );
}
// ... proses selanjutnya ...
```

Gambar 4.27 Sumber Kode *Pie Chart*

Penggalan kode Gambar 4.27 merupakan fungsi utama grafik lingkaran (*pie chart*) untuk menghitung pengunjung harian berdasarkan kategori. Proses hanya akan berjalan jika *visitCategoryData* (data master kategori seperti “VIP”, “NON-VIP”) sudah tersedia. Sebuah objek *categoryCounts* berfungsi sebagai ‘papan hitung’. Fungsi *objectSetter* digunakan untuk mengisi objek dengan nama-nama kategori sebagai *key* dan memberinya nilai awal 0 maka objek akan terbentuk seperti { ‘VIP’: 0, ‘Non-VIP’: 0 }. Kemudian sistem melakukan perulangan pada *data.countPerdaysCategory* (data riil dari API) menggunakan *forEach()*. Untuk

setiap item di dalamnya, sistem akan membarui nilai pada objek *categoryCounts* sesuai dengan jumlah kategori pengunjung tersebut.

Setelah perulangan selesai, *categoryCounts* akan berisi jumlah pengunjung *riil* untuk setiap kategori. Kemudian sistem melakukan *map()* pada *visitCategoryData*, dimana setiap kategori membuat objek baru dengan format yang dibutuhkan oleh komponen *PieChart*, yaitu *{ name, value, color }*. Properti *name* diisi dari nama kategori, *value* diambil dari *categoryCounts*, dan *color* diambil dari skema warna yang sudah didefinisikan. *Array* yang telah diformat kemudian disimpan ke dalam *state pieData* untuk ditampilkan sebagai grafik lingkaran (*pie chart*). Gambar 4.28 merupakan tampilan halaman *dashboard* yang menampilkan data pengunjung dalam bentuk grafik dan tabel pengunjung.



Gambar 4.28 Implementasi Tampilan Halaman *Dashboard*

Dalam menu *dashboard* selain menampilkan statistik data kunjungan, memiliki fungsi untuk menampilkan *modal* ketika melakukan *scan QR code* yang diimplementasikan dalam Sub bab 4.3.2.2.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

4.3.2.2 Implementasi *Modal Scanner QR code*

```
const isGroup = visitors?.is_group === true;

const isCheckIn =
  visitors?.status === 'RESERVED' || visitors?.status === 'CHECKOUT';

const isExpired = visitors?.visit_date_to
  ? moment().isAfter(moment.utc(visitors.visit_date_to).local(), 'day')
  : false;

const isStatusNull =
  visitors?.status === null ||
  visitors?.status === undefined ||
  visitors?.status === 'CANCEL' ||
  isExpired;
```

Gambar 4.29 Sumber Kode *Scanner QR*

Gambar 4.29 menunjukkan komponen *VisitorScannerModal*, terdapat beberapa logika utama yang menjadi penentu jalannya tampilan dan dalam menampilkan modal.

- isGroup* berfungsi memeriksa apakah kunjungan ini adalah untuk sebuah grup.
- isCheckIn* berfungsi menentukan apakah aksi yang harus dilakukan adalah “Check-In”. Ini berlaku jika status pengunjung masih “RESERVED” (belum datang) atau sudah “CHECKOUT”.
- isExpired* berfungsi untuk memeriksa apakah tanggal kunjungan sudah melewati hari ini (kedaluwarsa).
- isStatusNull* berfungsi menggabungkan beberapa kondisi tidak valid menjadi satu. Variabel ini akan bernilai *true* jika status pengunjung tidak ada, dibatalkan, atau sudah kedaluwarsa.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

```

if (isStatusNull) {
  return (
    <Dialog open={open} onOpenChange={onOpenChange}>
      <DialogContent className="p-0 sm:max-w-[520px] max-h-[90vh] overflow-y-auto">
        <DialogHeader className="flex-row items-center justify-center gap-2 bg-[#8E1616] p-6">
          <Contact2Icon className="size-8 gap-2 text-white" />
          <DialogTitle className="text-2xl text-white">
            Invalid Visitor QR
          </DialogTitle>
        </DialogHeader>
        <div className="space-y-6 p-6">
          <p className="text-center text-lg">
            This visitor has an invalid or unknown status. Please contact an
            administrator.
          </p>
          <div className="flex justify-center">
            <Button onClick={() => onOpenChange(false)}>Close</Button>
          </div>
        </div>
      </DialogContent>
    </Dialog>
  );
}

```

Gambar 4.30 Sumber Kode Validasi *QR code*

Penggalan kode Gambar 4.30 memiliki fungsi sebagai gerbang validasi pertama untuk menyaring QR yang tidak valid sejak awal, sebelum sistem mencoba memprosesnya lebih lanjut. *isStatusNull* memeriksa hasil dari variabel, jika bernilai *true* menandakan bahwa status pengunjung tidak ditemukan, sudah dibatalkan, atau tanggal kunjungannya telah kedaluwarsa. Kemudian penggalan kode gambar 4.30 menampilkan sebuah *modal* yang menampilkan pesan error dengan judul “*Invalid Visitor QR*” dan sebuah tombol “*Close*”.

```

if (isGroup) {
  return (
    <Dialog /* ... */>
      <DialogContent /* ... */>
        {/* ... Konten detail grup ... */}
        <div className="flex justify-center mt-6">
          <Button
            className={`w-40 ${isCheckIn ?
              'bg-green-500 hover:bg-green-600' : 'bg-red-500 hover:bg-red-600'}`}
            onClick={onAction}
          >
            {isCheckIn ? 'CHECK IN' : 'CHECK OUT'}
          </Button>
        </div>
      </DialogContent>
    </Dialog>
  );
}

```

Gambar 4.31 Sumber Kode Tombol *Check-In/Out*

Jika *QR code* lolos dari validasi pertama (*isNull* bernilai *false*), sistem akan melanjutkan ke pengecekan berikutnya: *if (isGroup)* pada Gambar 4.31. Jika kondisi ini *true*, maka *modal* yang ditampilkan akan disesuaikan untuk menangani grup. Pertama, untuk warna tombol, kode `{isCheckIn ? 'bg-green-500...' : 'bg-red-500...'}` akan memeriksa variabel *isCheckIn*. Jika bernilai *true*, tombol akan diberi warna hijau untuk aksi “*Check-In*”. Jika *false*, akan diberi warna merah untuk “*Check-Out*”. Kedua, untuk teks pada tombol, kode `{isCheckIn ? 'CHECK IN' : 'CHECK OUT'}` akan menampilkan tulisan yang sesuai.

```
return (
  <Dialog /* ... */>
    <DialogContent /* ... */>
      <DialogHeader /* ... */>
        <DialogTitle>
          Visitor {isCheckIn ? 'Check-In' : 'Check-Out'}
        </DialogTitle>
      </DialogHeader>
      <div className="space-y-6 p-6">
        <div className="flex justify-center">
          <div className={`rounded-full p-6 ${isCheckIn ? 'bg-green-100' : 'bg-red-100'}`}>
            {isCheckIn ? (
              <ArrowRight className="size-8 text-green-500" />
            ) : (
              <ArrowLeft className="size-8 text-red-500" />
            )}
          </div>
        </div>
      </div>
      {/* ... Konten detail individual ... */}
      <div className="flex justify-center space-x-4 ">
        {/* ... Tombol Cancel ... */}
        <Button
          className={`w-40 ${isCheckIn ? 'bg-green-500 ...' : 'bg-red-500 ...'}`}
          onClick={onAction}
        >
          {isCheckIn ? 'CHECK IN' : 'CHECK OUT'}
        </Button>
      </div>
    </DialogContent>
  </Dialog>
);
```

Gambar 4.32 Sumber Kode *Modal Check-In/Out QR code*

Gambar 4.32 merupakan penggalan kode jika sebuah *QR code* terbukti valid untuk kunjungan individual. Pertama, pada bagian judul “*DialogTitle*”, logika *Visitor {isCheckIn ? 'Check-In' : 'Check-Out'}* membuat judul berubah antara atau “*Visitor Check-Out*”. Jika *isCheckIn* bernilai *true*, maka judul akan berubah menjadi “*Visitor Check-In*”. Jika *isCheckIn* bernilai *false*, maka judul akan berubah menjadi

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

“*Visitor Check-Out*”. Seperti pada *modal* grup, tombol aksi utama menyesuaikan warna dan teksnya menjadi “*CHECK IN*” (hijau) atau “*CHECK OUT*” (merah).

Gambar 4.33 merupakan penggalan kode yang memicu komponen *VisitorScannerModal* menampilkan *modal* sesuai dengan data pengunjung yang didapatkan dari *QR code*.

```
const fetchQRCodeData = async (scannedCode: string) => {
  setIsLoading(true);
  setError(null);
  setVisitorsModal(undefined);

  try {
    const response = await api.get(
      `/master/decodeQRCode?code=${encodeURIComponent(scannedCode)}`,
    );
    const data = response.data;

    const decodedData = data;
    const visitorId = decodedData.visit_id;
    const visitorResponse = await api.get(`/visitor/${visitorId}`);
    const visitorData = visitorResponse.data;

    setVisitors(visitorData);
    setVisitorsModal(visitorData);

    // Check if the visitor has already checked out
  } catch (error) {
    console.error('Error fetching QR Code data:', error);
    setError('Failed to fetch QR Code data. Please try again later.');
```

Gambar 4.33 Sumber Kode *Fetch QR code*

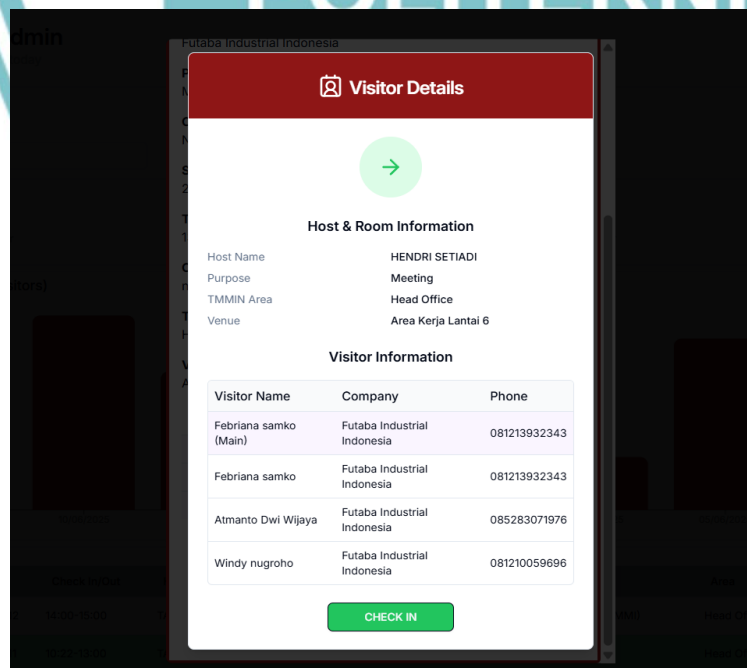
Gambar 4.33 menampilkan fungsi *fetchQRCodeData* yang dipanggil setelah sistem berhasil menangkap *QR code* dari pemindai (*scanner*). Tugas utamanya adalah mengambil detail lengkap pengunjung dari server berdasarkan *QR code* yang diberikan. *setIsLoading(true)* untuk menampilkan indikator pemuatan di antarmuka, *setError(null)* untuk menghapus pesan *error* sebelumnya, dan *setVisitorsModal(undefined)* untuk mengosongkan data *modal* dari pemindaian

- Hak Cipta :**
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
 2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

sebelumnya. Dalam blok *try-catch*, sistem terlebih dahulu melakukan permintaan API ke *endpoint* `‘/master/decodeQRCode’`. Permintaan ini mengirimkan *scannedCode* yang sudah diamankan dengan *encodeURIComponent* untuk mengubahnya menjadi *string* URL yang valid. Tujuan dari permintaan ini adalah untuk mendekode atau menerjemahkan *string QR code* menjadi sebuah identitas yang dikenali oleh sistem, yaitu *visitorId*.

Setelah *visitorId* berhasil didapatkan, sistem melakukan permintaan API ke *endpoint* `‘/visitor/{visitorId}’`. Permintaan ini menggunakan *visitorId* untuk mengambil data pengunjung yang lengkap dari database dan disimpan ke dalam *visitorData*. Setelah data berhasil diterima, data tersebut disimpan ke dalam *state visitorModal* menggunakan *setVisitorsModal(visitorData)*. Pembaruan *state* inilah yang akan memicu komponen *VisitorScannerModal* untuk tampil dengan menampilkan semua informasi pengunjung yang relevan. Blok *catch* berfungsi sebagai penanganan kesalahan; jika salah satu dari permintaan API gagal, sistem akan mencatat kesalahan teknis dan mengatur pesan kesalahan yang ramah pengguna.

Gambar 4.34 merupakan tampilan *modal* ketika *QR code check-in* pengunjung di *scan*.



Gambar 4.34 Implementasi Tampilan *Modal Check-In QR code*

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

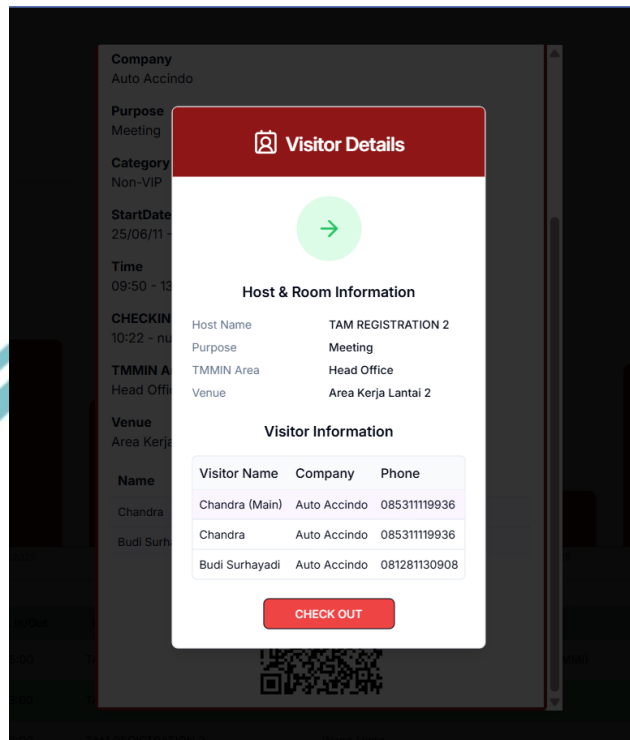
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Gambar 4.35 merupakan tampilan *modal* ketika *QR code check-out* pengunjung di *scan*.



Gambar 4.35 Implementasi Tampilan *Modal Check-Out QR code*

4.3.3 Implementasi Registration

Dalam tahapan implementasi registrasi, penggalan kode pada Gambar 4.36 *interface* berfungsi untuk mendefinisikan struktur dari objek diantaranya *GroupVisit*, *EmployeeData*, dan *HostName*.

```
interface GroupVisit {
    name: string;
    company: string;
    phone_no: string;
    idx: number;
}

interface EmployeeData {
    name: string;
    noreg: string;
    company: string;
}

interface HostName {
    data: EmployeeData[];
}
```

Gambar 4.36 Sumber Kode *Interface* Proses Registrasi

Dalam *interface GroupVisit* struktur objek harus berisi properti *name*, *company*, *phone_no*, dan *idx*. Sama halnya untuk *interface EmployeeData*. Sedangkan *interface HostName* merupakan *wrapper* atau pembungkus dari *array EmployeeData* yang akan digunakan dalam pencarian atau *dropdown host* melalui API.

Gambar 4.37 merupakan penggalan kode variabel *initGroupVisit* bertipe *array GroupVisit* yang memiliki *value* dari properti *name*, *company*, *phone_no*, dan *idx=0* sebagai index awal. Sedangkan variabel *initFormData* menyimpan *value* dari *form* ketika *visitor* melakukan registrasi. Jika *visitor* merupakan sebuah grup makan properti *members* diisi berdasarkan variabel *initGroupVisit*, berisi *array object* dengan *name*, *company*, dan *phone_no* dari anggota grup.

```
const initGroupVisit: GroupVisit[] = [
  {
    name: '',
    company: '',
    phone_no: '',
    idx: 0,
  },
];
const initFormData = {
  name: '',
  host_name: '',
  host_noreg: '',
  phone_no: '',
  email: '',
  company: '',
  visit_desc: '',
  visit_date_from: '',
  visit_date_to: '',
  visit_start_time: '',
  visit_end_time: '',
  type_visit: 'VT-001',
  visit_category: '',
  visit_location: '',
  is_group: false,
  venue: '',
  atsg_code: '0',
  members: initGroupVisit.map((member) => ({
    name: member.name,
    company: member.company,
    phone_no: member.phone_no,
  })),
};
```

Gambar 4.37 Sumber Kode Objek *Form* Registrasi

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Gambar 4.38 penggalan kode *state employees* bertipe *HostName* yaitu objek dengan properti *data* yang berupa array *EmployeeData* yang terlampir pada Gambar 4.36. Sedangkan *setEmployees* digunakan untuk menyimpan atau mengupdate nilai *employees*. Fungsi *handleHostnameChange* dipanggil saat isi *input field host_name* terisi dengan menerima parameter *value* yaitu *string*. Jika terdapat error pada *field host_name* fungsi *clearValidationError('host_name')* digunakan untuk menghapus pesan error validasi tersebut. *setFormData* berfungsi untuk menyimpan *value* ke dalam *formData* yang merupakan tempat penyimpanan seluruh *value* dari *form*.

```

const [employees, setEmployees] = useState<HostName>({ data: [] });
const [formData, setFormData] = useState<initFormData>();

const handleHostnameChange = async (value: string) => {
  clearValidationError('host_name');

  setFormData({
    ...formData,
    host_name: value,
    host_noreg: '',
  });

  if (value.length > 2) {
    try {
      const response = await api.get(
        `/master/employee?name=${value}&limit=10`,
      );

      if (response.data && response.data.length > 0) {
        setEmployees({ data: response.data });

        setValidationErrors((prev) => ({
          ...prev,
          host_name: '',
        }));
      } else {
        setEmployees({ data: [] });

        setValidationErrors((prev) => ({
          ...prev,
          host_name: 'Host tidak ditemukan. Pilih dari daftar.',
        }));
      }
    } catch (error) {
      console.error('Error fetching employees:', error);
      setEmployees({ data: [] });

      setValidationErrors((prev) => ({
        ...prev,
        host_name: 'Gagal mengambil data host.',
      }));
    }
  } else {
    setEmployees({ data: [] });

    setValidationErrors((prev) => ({
      ...prev,
      host_name: '',
    }));
  }
};

```

Gambar 4.38 Sumber Kode *Input Nama Host*

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Terdapat kondisi untuk mengecek *input user* harus lebih dari 2 karakter sehingga mencegah panggilan API terlalu cepat atau terlalu sering. Selanjutnya sistem mengirim GET request *‘/master/employee?name=\${value}&limit=10’* ke API untuk mencari data *employee* berdasarkan *name*. Jika data ditemukan, *setEmployees* akan menyimpan atau mengupdate data tersebut ke dalam *state employees* dan *setValidationErrors* akan menghapus *error* validasi. Namun, jika data tidak ditemukan maka *employees* akan kosong dan tampil *error* “Host tidak ditemukan. Pilih dari daftar”. Apabila request ke API gagal maka akan tampil *error* di konsol, *value employees* dikosongkan, dan *error* yang tampil pada *interface* “Gagal mengambil data host”. Jika kondisi *input user* tidak lebih dari 2 karakter *dropdown host* atau daftar nama *host* tidak akan muncul.

```
const isAdminOrSecurity =
  (user.data?.roles[0].role_code as string) === 'ADMIN' ||
  (user.data?.roles[0].role_code as string) === 'SECURITY' ||
  (user.data?.roles[0].role_code as string) === 'SUPER_ADMIN' ||
  (user.data?.roles[0].role_code as string) === 'RECEPTIONIST';
```

Gambar 4.39 Sumber Kode Variabel Kondisi Role User

Penggalan kode pada Gambar 4.39 menampilkan variabel *isAdminOrSecurity* digunakan untuk mengecek *role user* yang sedang *login* sebagai ADMIN, SECURITY, SUPER ADMIN, atau RECEPTIONIST.

Gambar 4.40 merupakan penggalan kode sebagai fungsi utama registrasi *visitor* yang diimplementasikan pada *button* “Submit” pada *form* registrasi.

Pada penggalan kode Gambar 4.40 menampilkan *handleVisitorRequest* yang memiliki fungsi *setIsLoading(true)* sebagai *loading state* untuk menampilkan UI. Dalam *try-catch* terdapat *let response* yang berfungsi menyimpan *response* dari server berdasarkan kondisi *isEdit*. Jika *isEdit* bernilai *false*, maka terindikasi sedang dilakukan registrasi *visitor* baru. Kemudian *response* mengirim POST request ke *endpoint* *‘/visitor/registry’* untuk menyimpan data *visitor* baru. Sedangkan jika *isEdit* bernilai *true*, maka terindikasi sedang melakukan perubahan data *visitor*.

Nilai perubahan data *visitor* tersebut dikirim melalui POST *request* API ke *endpoint* `‘/visitor/${data?.id}’`.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

```
const handleVisitorRequest = async () => {
  setIsLoading(true);
  try {
    let response;
    if (!isEdit) {
      response = await api.post('/visitor/registry', {
        ...formData,
        host_name: isAdminOrSecurity ? formData.host_name : user.data?.Name,
        host_noreg: isAdminOrSecurity
          ? formData.host_noreg
          : user.data?.userId,

        status: isAdminOrSecurity ? 'RESERVED' : 'APPROVE',
        visit_category: formData.visit_category || null,
        is_group: is_group,
        email: formData.email == '' ? undefined : formData.email,
        members: groupMembers.map((member) => ({
          name: member.name,
          company: member.company,
          phone_no: member.phone_no || null,
        })),
      });
    } else {
      response = await api.post(`/visitor/${data?.id}`, {
        ...formData,
        host_name: isAdminOrSecurity ? formData.host_name : user.data?.Name,
        host_noreg: isAdminOrSecurity
          ? formData.host_noreg
          : user.data?.userId,
        is_group: is_group,
        members: groupMembers.map((member) => ({
          name: member.name,
          company: member.company,
          phone_no: member.phone_no,
        })),
      });
    }
    setModalType(response?.status ? 'success' : 'error');
  }
}
```

Gambar 4.40 Sumber Kode *Submit Form* Registrasi

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

```

title={
  modalType === 'success'
    ? 'Registration Successful'
    : 'Registration Failed'
}
message={
  modalType === 'success'
    ? 'Visitor has been successfully registered.'
    : 'There was an error registering the visitor. Please try again.'
}
type={modalType}

```

Gambar 4.41 Sumber Kode *Modal Message* Registrasi

Modal berisi *message* sukses atau gagal registrasi akan muncul setelah *user* menekan tombol *submit* seperti penggalan kode Gambar 4.41. Jika *role user* merupakan salah satu dari variabel *isAdminOrSecurity* yang tercantum pada gambar 4.39, data *host_name* dan *host_noreg* perlu diisi manual dari *dropdown host* melalui *input field host_name*. Kemudian *status* menyimpan *value RESERVED* atau *APPROVE* berdasarkan *role user* yang di *handle* oleh penggalan kode *createVisitor* pada Gambar 4.42.

```

const tokenId = (token as { id: string })?.id;
const requester = await this.visitorRepository.checkRoleByUserId(tokenId);
const hostData = await this.visitorRepository.getEmployeeByNoreg(
  visitor.host_noreg || '',
);

let isAdmin = 0;
if (requester.includes('ADMIN') || requester.includes('SUPER_ADMIN')) {
  isAdmin = 1;
} else if (
  requester.includes('SECURITY') ||
  requester.includes('RECEPTIONIST')
) {
  isAdmin = 2;
} else if (requester.includes('TAM_REGISTER')) {
  visitor.status = 'APPROVE';
  isAdmin = 3;
} else if (
  requester.includes('TAM_APPROVE') ||
  requester.includes('SECRETARY')
) {
  visitor.status = 'RESERVED';
  isAdmin = 3;
}

if (isAdmin === 1) {
  visitor.status = 'RESERVED';
  noregSuperior = visitor.host_noreg;
  emailSuperior = hostData?.email;
} else if (isAdmin === 2) {
  visitor.status = 'APPROVE';
  noregSuperior = visitor.host_noreg;
  emailSuperior = hostData?.email;
}

```

Gambar 4.42 Sumber Kode *Status Request* Kunjungan

TokenId mengambil *id user* dari *token JWT* yang otomatis dibuat ketika *user* melakukan *login*. *Requester* untuk mengecek dan mengambil *role user* yang membuat permintaan berdasarkan *id user*. *HostData* mengambil data *host* berdasarkan *host_noreg* (nomor registrasi *host*). Berikutnya, *isAdmin* digunakan untuk mengkategorikan jenis *role user* pembuat permintaan yang diinisialisasi nilai awal 0. Jika *requester* adalah ADMIN dan SUPER_ADMIN, *isAdmin* bernilai 1 dan status *visitor* diatur menjadi RESERVED. Jika *requester* adalah SECURITY dan RECEPTIONIST, *isAdmin* bernilai 2 dan status *visitor* diatur menjadi APPROVE. Jika *requester* adalah TAM REGISTRATION, status *visitor* diatur menjadi APPROVE. Jika *requester* adalah TAM APPROVE dan SECRETARY, status *visitor* diatur menjadi RESERVED.

```
const generateQR = async (
  data: object | string,
  format: string = 'string',
): Promise<string> => {
  const qr = qrcode(0, 'L');

  if (typeof data === 'object') {
    qr.addData(JSON.stringify(data));
  } else {
    qr.addData(data);
  }

  qr.make();

  if (format === 'image') {
    const qrCodeImage = qr.createSvgTag();
    return qrCodeImage;
  }

  const qrCodeImage = qr.createDataURL();
  return qrCodeImage;
};
```

Gambar 4.43 Sumber Kode *Generate QR code*

Penggalan kode pada Gambar 4.43 berfungsi untuk menghasilkan *QR code* dengan format yang memiliki nilai *default string*. *QR code* hanya dapat menyimpan *string*, sehingga jika nilai *input* berupa *object* maka dikonversi ke *string* menggunakan *JSON.stringify*. Kemudian *qr.make()* akan membangun matriks *QR code* berdasarkan data yang telah ditambahkan. Jika format berupa *image* maka fungsi

Hak Cipta :

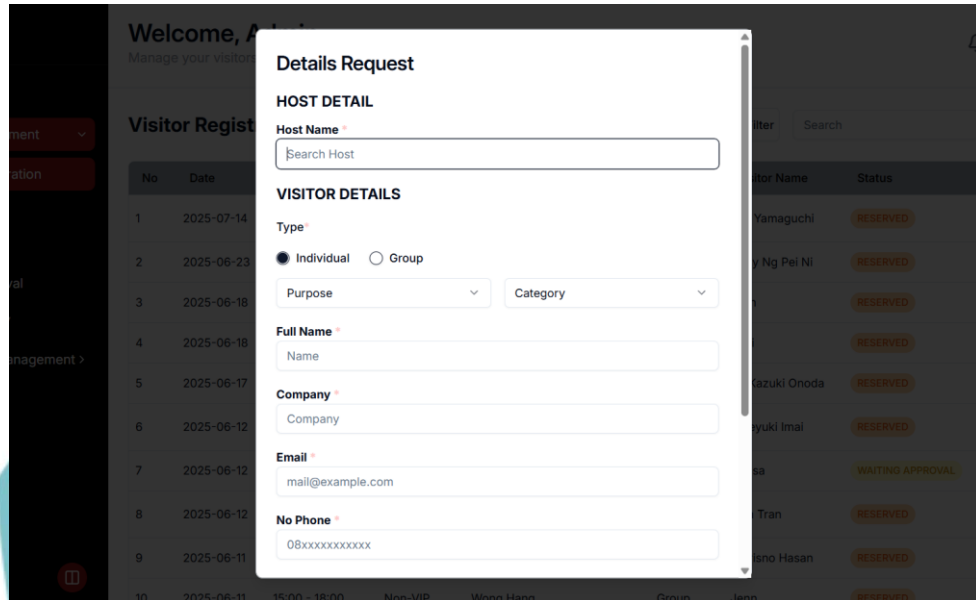
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengurukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

mengembalikan SVG yang dihasilkan dari *createSvgTag()*, sedangkan jika bukan *image* maka *QR code* akan dikembalikan dalam format *Data URL*. Gambar 4.44 merupakan tampilan halaman *registration* untuk mendaftarkan *visitor*.



Gambar 4. 44 Implementasi Tampilan Halaman Registrasi *Visitor* oleh *Host*

4.3.4 Implementasi *Approval*

```
const fetchVisitorsApproval = async (page: number) => {
  setIsLoading(true);
  setError(null);
  try {
    const skip = (page - 1) * itemsPerPage;
    const response = await api.get(
      `/visitor/approval/list?skip=${skip}&take=${itemsPerPage}&noreg=${user.data?.userId}`,
    );
    const data = response.data;
    setVisitors(response.data);
    setTotalPages(Math.ceil(data.count / itemsPerPage));
    if (visitors.length === 0 && currentPage > 1) {
      setCurrentPage((prev) => Math.max(prev - 1, 1));
    }
  } catch (error) {
  } finally {
    setIsLoading(false);
  }
};
```

Gambar 4.45 Sumber Kode *Fetch Approval Visitor*

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Penggalan kode pada Gambar 4.45 menunjukkan *fetchVisitorsApproval* dieksekusi setiap kali daftar *approval* dimuat atau diperbarui. Prosesnya diawali dengan mengatur *state isLoading* menjadi *true* untuk menampilkan indikator pemuatan UI. Kemudian, sistem melakukan permintaan data melalui API ke *endpoint* *‘/visitor/approval/list’*. Parameter *noreg=\${user.data?.userId}* mengirimkan nomor registrasi atau ID dari pengguna yang *login*. Hal ini memastikan bahwa API hanya akan mengembalikan daftar pengunjung yang menunggu persetujuan dari pengguna tersebut. Setelah data berhasil diterima, fungsi *setVisitors* akan menyimpan daftar pengunjung ke dalam *state*, dan *setTotalPages* akan menghitung jumlah total halaman yang dibutuhkan untuk paginasi berdasarkan total data (*data.count*). Terdapat juga sebuah kondisi *if (visitors.length === 0 && currentPage > 1)* jika pengguna berada di halaman terakhir dan menyetujui/menolak permintaan persetujuan kunjungan, sehingga halaman tersebut menjadi kosong. Sistem akan secara otomatis memindahkan pengguna ke halaman sebelumnya (*setCurrentPage((prev) => prev - 1)*) agar tidak terjebak di halaman yang kosong. Terakhir, blok *finally* akan selalu menyetel *isLoading* kembali ke *false* setelah semua proses selesai.

```
const handleAccept = async () => {
  if (!selectedVisitor) return;

  setIsLoading(true);
  try {
    const response = await api.post(`/visitor/approval/edit`, {
      visit_id: selectedVisitor.id,
      noreg_superior: user.data?.userId,
      status: 'RESERVED',
    });
    console.log('response', response);

    if (response.status === 200) {
      toast.success('Visitor approval sent successfully');
      fetchVisitorsApproval(currentPage);
    }
  } catch (error) {
    console.log(error);
  } finally {
    setIsLoading(false);
  }
};
```

Gambar 4.46 Sumber Kode Tombol *Accept*

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Gambar 4.46 menampilkan fungsi *handleAccept* dimana pertama-tama memastikan ada pengunjung yang dipilih (*selectedVisitor*). Kemudian, sistem mengirimkan permintaan POST ke *endpoint* *‘/visitor/approval/edit’* untuk memproses persetujuan. Data yang dikirim (*payload*) berisi *visit_id* dari pengunjung yang dipilih, *noreg_superior* (ID pengguna yang menyetujui), dan *status* yang diubah menjadi *“RESERVED”*. Jika permintaan berhasil, sebuah notifikasi sukses akan ditampilkan, dan fungsi *fetchVisitorsApproval* akan dipanggil kembali untuk memuat ulang daftar permintaan kunjungan, sehingga antarmuka pengguna akan diperbarui secara otomatis.

```
const handleReject = (reason: string) => {
  if (!selectedVisitor) return;
  confirmReject(reason);
  setIsRejectionModalOpen(false);
};

const confirmReject = async (reason: string) => {
  if (!selectedVisitor) return;

  setIsLoading(true);
  try {
    const response = await api.post(`/visitor/approval/edit`, {
      visit_id: selectedVisitor.id,
      noreg_superior: user.data?.userId,
      status: 'REJECTED',
      reason: reason,
    });

    if (response.status === 200) {
      setVisitors((prevVisitors) =>
        prevVisitors.map((visitor) =>
          visitor.id === selectedVisitor.id
            ? { ...visitor, status: 'REJECTED' }
            : visitor,
        ),
      );
      toast.success('Visitor rejected successfully');
      setIsConfirmationModalOpen(false);
      fetchVisitorsApproval(currentPage);
    }
  } catch (error) {
    console.error(error);
    toast.error('Failed to reject visitor');
  }
  setIsLoading(false);
};
```

Gambar 4.47 Sumber Kode Tombol *Reject*

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Kedua fungsi pada Gambar 4.47 bekerja sama untuk menolak kunjungan. *handleReject* dipicu oleh aksi pengguna ketika menekan tombol *reject*, yang kemudian memanggil *confirmReject* untuk eksekusi. *handleReject* yang menerima *reason* (alasan penolakan) dari pengguna, lalu memanggil *confirmReject* untuk meneruskan alasan tersebut. Fungsi *confirmReject* kemudian mengirimkan permintaan API ke *endpoint* `‘/visitor/approval/edit’` dan status diatur menjadi `“REJECTED”` serta menyertakan *reason*. Jika berhasil, *toast success* menampilkan pesan dan daftar pengunjung dimuat ulang.

```
const handleCheckboxChange = (visitorId: number | undefined) => {
  if (!visitorId) return;
  setSelectedVisitorIds((prev) => {
    if (prev.includes(visitorId)) {
      return prev.filter((id) => id !== visitorId);
    } else {
      return [...prev, visitorId];
    }
  });
};

const handleSelectAll = (checked: boolean) => {
  if (checked) {
    const allIds = visitors.map((visitor) => visitor.id as number);
    setSelectedVisitorIds(allIds);
  } else {
    setSelectedVisitorIds([]);
  }
};
```

Gambar 4.48 Sumber Kode *Checkbox*

Fungsi-fungsi pada Gambar 4.48 mengelola status item yang dipilih oleh pengguna di dalam tabel. *handleCheckboxChange* berfungsi mengatur kondisi ketika sebuah *checkbox* di samping nama pengunjung diklik. Jika ID pengunjung tersebut sudah ada dalam daftar yang dipilih, maka ID itu akan dihapus. Jika belum ada, maka akan ditambahkan. Ini menciptakan efek *toggle* (memilih/batal memilih). Sementara itu, *handleSelectAll* terhubung dengan *checkbox* utama di *header* tabel. Jika dicentang (*checked* adalah *true*), sistem akan memilih semua ID pengunjung yang ada di

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

halaman permintaan kunjungan atau dapat dikatakan seluruh *checkbox* otomatis tercentang. Jika tidak dicentang, sistem akan mengosongkan *checkbox*.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

```
const handleBatchApprove = async () => {
  if (selectedVisitorIds.length === 0) {
    toast.error('No visitors selected');
    return;
  }

  setIsLoading(true);
  try {
    const promises = selectedVisitorIds.map((id) =>
      api.post(`/visitor/approval/edit`, {
        visit_id: id,
        noreg_superior: user.data?.userId,
        status: 'RESERVED',
      })),
    );

    const results = await Promise.all(promises);

    const allSuccessful = results.every((res) => res.status === 200);

    if (allSuccessful) {
      toast.success(
        `${selectedVisitorIds.length} visitors approved successfully`,
      );
      setSelectedVisitorIds([]);
      fetchVisitorsApproval(currentPage);
    }
  } catch (error) {
    console.error(error);
    toast.error('Failed to approve some visitors');
  } finally {
    setIsLoading(false);
  }
};

const handleBatchReject = async (reason: string) => {
  //.. kode yang sama dengan handleBatchApprove
  try {
    const promises = selectedVisitorIds.map((id) =>
      api.post(`/visitor/approval/edit`, {
        visit_id: id,
        noreg_superior: user.data?.userId,
        status: 'REJECTED',
        reason: reason,
      })),
    );
  } finally {
    // .. kode yang sama dengan handleBatchApprove
  }
};
```

Gambar 4.49 Sumber Kode Logika *Checkbox Approval/Reject*

Penggalan kode pada Gambar 4.49 menunjukkan fungsi *handleBatchApprove*, untuk mengambil semua ID dari *selectedVisitorIds* dan menggunakan *.map()* untuk membuat sebuah array berisi data pengunjung ke dalam variabel *promises*.

Kemudian, *Promise.all(promises)* dijalankan agar permintaan API tersebut secara bersamaan (*concurrently*), bukan satu per satu. Setelah permintaan selesai, jika berhasil sistem akan menampilkan *toast succes*, mengosongkan pilihan *checkbox*, dan memuat ulang data permintaan kunjungan. Logika yang sama persis diterapkan pada *handleBatchReject*, namun dengan status dan alasan penolakan yang sesuai.

```
<ConfirmationModal
  isOpen={isConfirmationModalOpen}
  onClose={() => setIsConfirmationModalOpen(false)}
  onConfirm={confirmationAction === "approve" ? handleAccept : () => setIsRejectionModalOpen(true)}
  title={confirmationAction === "approve" ? "Approve Visitor" : "Reject Visitor"}
  description="Are you sure?"
/>

<VisitorRejectionModal
  isOpen={isRejectionModalOpen}
  onClose={() => setIsRejectionModalOpen(false)}
  onConfirm={confirmReject}
/>

<VisitorDetailsModal
  visitor={selectedVisitorForModal}
  isOpen={showVisitorDetailModal}
  onClose={() => setShowVisitorDetailModal(false)}
/>
```

Gambar 4.50 Sumber Kode Modal *Approval*

Gambar 4.50 menunjukkan *ConfirmationModal* digunakan sebelum aksi *approve* atau *reject* dilakukan. Jika pengguna mengonfirmasi *reject*, maka *modal VisitorRejectionModal* akan muncul untuk menginput alasan penolakan. Selain itu, terdapat *VisitorDetailsModal* yang menampilkan informasi detail pengunjung saat tombol "Detail" ditekan. Gambar 4.51 merupakan hasil implementasi halaman *approval*.

Welcome, Admin
Manage your visitors today

Visitor Approval

Approve Selected (3) Reject Selected (3)

☐	No	Date	Check In/Out	Host Name	Company	Purpose	Visitor Name	Category	Action
☐	1	2025-06-12	14:00-15:00	ARIF NURUDIN	Meldoh	Meeting	Ghesa	Non-VIP	Detail ☒ ☐
☐	2	2025-06-05	15:00-16:00	DANANG WAHYU KRATON	Mushashi	Meeting	Anastasya	Non-VIP	Detail ☒ ☐
☒	3	2025-06-04	08:30-09:00	DENY SURYA YUDHA	PT Sinergi Samudra S	Meeting	Bahtiar Oemar	Non-VIP	Detail ☒ ☐
☐	4	2025-06-03	10:15-14:00	ARIES FREDI DINATA	PT. Kojima Auto Technology Indonesia	Meeting	Mrs. Sabila	Non-VIP	Detail ☒ ☐
☒	5	2025-05-19	13:00-14:30	JOKO LELONO	TMMIN	Meeting	JOKO LELONO	Non-VIP	Detail ☒ ☐
☒	6	2025-05-17	08:00-21:00	YOSEPH ADHITYA PAMBUDHI	Next Leader	Meeting	Christine Handayani	Non-VIP	Detail ☒ ☐

< Previous Page 1 Next >

Gambar 4.51 Implementasi Halaman *Approval*

4.3.5 Implementasi *Check-In/Check-Out*

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

```
const fetchVisitors = async (
  search: string,
  startDate: string,
  endDate: string,
  area?: string | undefined,
) => {
  setIsLoading(true);
  setError(null);
  try {
    const response = await api.get(
      `/visitor?page=cico&dateFrom=${startDate}&dateTo=${endDate}&search=${search}${area ? `&area=${area}` : ''}`,
    );
    const data = response.data.visitor || [];

    // Set all visitors from API response
    setVisitors(data);
    if (filteredVisitors.length === 0 && currentPage > 1) {
      setCurrentPage((prev) => Math.max(prev - 1, 1));
    }
  } catch (error) {
    console.error('Error fetching visitors:', error);
    setError('Failed to fetch visitors. Please try again later.');
```

Gambar 4.52 Sumber Kode *Fetch Visitor Check-In/Out*

Gambar 4.52 menunjukkan fungsi *fetchVisitors* pada halaman *Check-In/Check-Out* dirancang secara khusus untuk mengambil daftar pengunjung yang sudah terjadwal. Ketika dieksekusi, fungsi menginisialisasi kondisi antarmuka dengan menjalankan *setIsLoading(true)* untuk menampilkan indikator pemuatan dan *setError(null)* untuk membersihkan pesan kesalahan yang mungkin ada dari proses sebelumnya. Selanjutnya, sistem menjalankan sebuah permintaan data ke server melalui pemanggilan *await api.get()*. Alamat URL untuk permintaan disusun dengan menyertakan parameter *page=cico* yang menginstruksikan server agar hanya mengirimkan pengunjung yang statusnya relevan untuk proses *check-in/check-out*. *startDate*, *endDate*, dan *area* akan diambil sekaligus untuk diolah di sisi klien.

Setelah respons dari server berhasil diterima, daftar pengunjung diekstrak dari *response.data.visitor*.

Kemudian data disimpan ke dalam *state* utama komponen melalui pemanggilan *setVisitors(data)*. Kondisi *if (filteredVisitors.length === 0 && currentPage > 1)* akan memeriksa halaman yang sedang ditampilkan menjadi kosong setelah suatu aksi. Jika ya, *setCurrentPage* akan dipanggil untuk secara otomatis mengarahkan pengguna ke halaman sebelumnya. Seluruh proses ini dibungkus dalam blok *try* untuk menangani potensi kegagalan, di mana blok *catch* akan mengatur pesan kesalahan melalui *setError()*. Blok *finally* memastikan bahwa *setIsLoading(false)* akan selalu dieksekusi untuk menyembunyikan indikator pemuatan setelah seluruh operasi, baik berhasil maupun gagal.

```
const confirmCheckInOut = async () => {
  if (!selectedVisitor || !confirmationAction) return;

  try {
    await api.post(`/visitor/validation?process=${confirmationAction}`, {
      code: selectedVisitor.qr_code,
    });

    // Refresh visitor list after action
    fetchVisitors(search, startDate, endDate, areaFilter);
  } catch (error) {
    console.error('Error validating visitor:', error);
  } finally {
    setIsConfirmationModalOpen(false);
  }
};
```

Gambar 4.53 Sumber Kode Validasi Check-In/Out QR Code

Gambar 4.53 menampilkan fungsi *confirmCheckInOut* terdapat *if (!selectedVisitor || !confirmationAction) return;*. Baris ini berfungsi sebagai mekanisme pengaman untuk memastikan fungsi tidak akan berjalan jika tidak ada pengunjung yang dipilih (*selectedVisitor*) atau jika jenis aksi (*confirmationAction*, contohnya *'CHECKIN'* atau *'CHECKOUT'*) belum ditentukan. Apabila kondisi tersebut terpenuhi, fungsi akan mengirimkan sebuah permintaan POST ke *endpoint* *‘/visitor/validation’* melalui *api.post*. Di dalam URL permintaan, disisipkan sebuah parameter query

Hak Cipta:

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

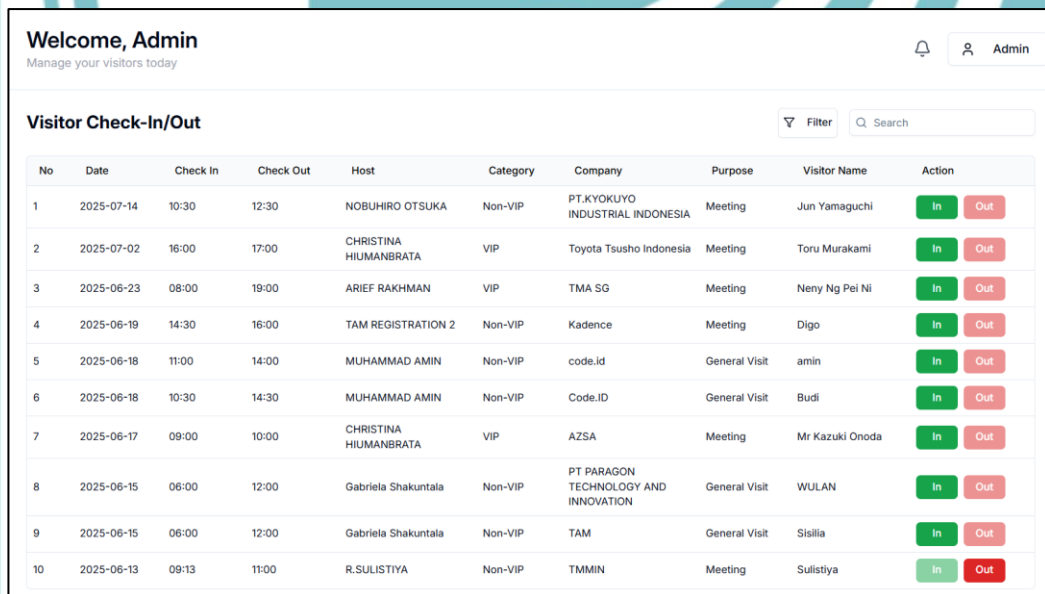
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

$process=\$ \{confirmationAction\}$ yang secara dinamis menentukan jenis proses yang diminta ke server.

Selain itu, permintaan tersebut juga membawa sebuah *payload* atau badan data yang berisi $\{code:selectedVisitor.qr_code\}$, di mana *qr_code* dari pengunjung terpilih digunakan oleh server untuk mengidentifikasi rekaman data yang akan divalidasi. Setelah permintaan berhasil dieksekusi, fungsi *fetchVisitors* segera dipanggil kembali. Proses ini untuk memuat ulang data di tabel, sehingga perubahan status pengunjung (misalnya dari "Reserved" menjadi "Check-In") dapat langsung terlihat di antarmuka pengguna. Seluruh proses ini dibungkus dalam blok *try-catch* untuk menangani potensi kesalahan, dan blok *finally* memastikan bahwa jendela *modal* konfirmasi (*setIsConfirmationModalOpen(false)*) akan selalu ditutup, baik prosesnya berhasil maupun gagal. Gambar 4.54 merupakan tampilan halaman *check-in/out*.



No	Date	Check In	Check Out	Host	Category	Company	Purpose	Visitor Name	Action
1	2025-07-14	10:30	12:30	NOBUHIRO OTSUKA	Non-VIP	PT.KYOKUYO INDUSTRIAL INDONESIA	Meeting	Jun Yamaguchi	In Out
2	2025-07-02	16:00	17:00	CHRISTINA HIUMANBRATA	VIP	Toyota Tsusho Indonesia	Meeting	Toru Murakami	In Out
3	2025-06-23	08:00	19:00	ARIEF RAKHMAN	VIP	TMA SG	Meeting	Neny Ng Pei Ni	In Out
4	2025-06-19	14:30	16:00	TAM REGISTRATION 2	Non-VIP	Kadence	Meeting	Digo	In Out
5	2025-06-18	11:00	14:00	MUHAMMAD AMIN	Non-VIP	code.id	General Visit	amin	In Out
6	2025-06-18	10:30	14:30	MUHAMMAD AMIN	Non-VIP	Code.ID	General Visit	Budi	In Out
7	2025-06-17	09:00	10:00	CHRISTINA HIUMANBRATA	VIP	AZSA	Meeting	Mr Kazuki Onoda	In Out
8	2025-06-15	06:00	12:00	Gabriela Shakuntala	Non-VIP	PT PARAGON TECHNOLOGY AND INNOVATION	General Visit	WULAN	In Out
9	2025-06-15	06:00	12:00	Gabriela Shakuntala	Non-VIP	TAM	General Visit	Sisilia	In Out
10	2025-06-13	09:13	11:00	R.SULISTIYA	Non-VIP	TMMIN	Meeting	Sulistiya	In Out

Gambar 4. 54 Implementasi Tampilan Halaman *Check-In/Out*

4.3.6 Implementasi *Inquiry*

Tahap berikutnya menambahkan fitur lanjutan, termasuk visualisasi data historis, yaitu *inquiry*. *Inquiry* digunakan untuk melihat riwayat kunjungan yang telah dilakukan oleh *user*.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

```
const fetchVisitors = async (
  page: number,
  search = '',
  startDate = '',
  endDate = '',
  area = '',
) => {
  try {
    const skip = (page - 1) * itemsPerPage;
    const response = await api.get(
      `/visitor?skip=${skip}&take=${itemsPerPage}&search=${search}&dateTo=${endDate}&dateFrom=${startDate}${user.data?.userId.toLowerCase().includes('admin') ? '' : '&noreg=${user.data?.userId}`} ${area} ?&area=${area}` : ''&page=inquiry`,
    );

    const data = response.data;
    const filteredVisitors = data.visitor.filter(
      (visitor: any) =>
        visitor.name.toLowerCase().includes(search.toLowerCase()) ||
        visitor.email.toLowerCase().includes(search.toLowerCase()) ||
        visitor.phone_no.toLowerCase().includes(search.toLowerCase()),
    );

    setVisitors(filteredVisitors);
    setTotalPages(Math.ceil(data.count / itemsPerPage));
  } catch (error) {
    console.error('Error fetching visitors:', error);
  }
};
```

Gambar 4.55 Sumber Kode *Fetch Visitor Inquiry*

Fungsi pada Gambar 4.55 menunjukkan *fetchVisitors* yang bertugas untuk mengambil data riwayat pengunjung dari server dengan menerapkan filter. Prosesnya diawali dengan pengiriman permintaan GET ke *endpoint* API *‘/visitor’*. Alamat URL permintaan ini disusun dengan menyertakan parameter untuk paginasi (*skip* dan *take*), pencarian (*search*), rentang tanggal (*dateFrom*, *dateTo*), dan *area*. Terdapat kondisional pada URL *`\${user.data?.userId.toLowerCase().includes('admin')} ?`* :*&noreg=\${user.data?.userId}`*. Logika ini memeriksa ID pengguna yang sedang masuk mengandung kata *‘admin’*. Jika ya, maka tidak ada parameter tambahan yang dikirim, dimana pengguna tersebut seorang admin dapat melihat seluruh riwayat pengunjung tanpa batasan.

Sebaliknya, jika pengguna bukan admin, maka parameter *&noreg* akan ditambahkan ke URL dengan menyertakan ID pengguna tersebut. Hal ini

membatasi hasil data yang dikembalikan oleh server hanya untuk riwayat pengunjung yang terkait dengan pengguna sesuai noreg. Setelah data diterima dari server, terdapat langkah pemfilteran tambahan yang dilakukan di sisi klien melalui fungsi `data.visitor.filter()`. Fungsi ini menyaring kembali data yang diterima berdasarkan kata kunci pencarian pada kolom nama, *email*, dan nomor telepon untuk memastikan hasil yang ditampilkan di antarmuka pengguna relevan. Hasil akhir dari pemfilteran ini kemudian disimpan ke dalam *state visitors* melalui `setVisitors(filteredVisitors)`, sementara total halaman untuk paginasi dihitung melalui `setTotalPages` dengan menggunakan `data.count` yang berasal langsung dari server untuk menjaga akurasi.

Untuk memudahkan fungsionalitas *inquiry*, dibuatlah tabel sebagai pendukung tampilan pada halaman ini. Berikut merupakan penggalan kode yang digunakan dalam halaman *inquiry*.

```
<TableBody>
  {visitors.map((visitor, index) => (
    <TableRow key={index} className="hover:bg-gray-50 transition-colors duration-200">
      {[
        visitor.date,
        visitor.checkInOut,
        visitor.hostName,
        visitor.company,
        visitor.purpose,
        visitor.name,
      ].map((value, i) => (
        <TableCell key={i} className="px-4 py-2 break-words">{value}</TableCell>
      ))}
    </TableRow>
  ))}
</TableBody>
```

Gambar 4. 56 Sumber Kode Tampilan *Body* Tabel *Inquiry*

Penggalan kode pada gambar 4.56 menunjukkan kode untuk membentuk *body* dari tabel *inquiry* yang akan menampilkan data dari *user*. Dalam kode tersebut, digunakan `.map()` untuk melakukan iterasi terhadap setiap *row* dari data *visitor*. Gambar 4.57 menunjukkan tampilan halaman *inquiry*.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Welcome, Admin
Manage your visitors today

Visitor Inquiry

Filter Search + Add Visitor

No	Date	Check In	Check Out	Host	Company	Purpose	Visitor Name	Status	Action
11	2025-06-15	06:00	12:00	Gabriela Shakuntala	PT PARAGON TECHNOLOGY AND INNOVATION	General Visit	WULAN	RESERVED	
12	2025-06-15	06:00	12:00	Gabriela Shakuntala	TAM	General Visit	Sisilia	RESERVED	
13	2025-06-13	09:13	11:00	R.SULISTIYA	TMMIN	Meeting	Sulistiya	CHECKIN	
14	2025-06-13	10:30	11:30	ARIF BANGUN SANTOSO	PT. Hilex	Meeting	Ody Septian	RESERVED	
15	2025-06-12	16:00	18:21	WAHYU PUGUH BAKHTIAR	Denso ID	Meeting	Dhania	CHECKOUT	
16	2025-06-12	14:08	16:20	NURFAEZI	Agung Trans	General Visit	Pandapotan	CHECKOUT	
17	2025-06-12	14:35	16:30	TAM REGISTRATION 2	TAM	Meeting	Amanda Ratri Yasmin	CHECKOUT	
18	2025-06-12	15:44	15:44	TAM REGISTRATION 2	Kadence	Meeting	Digo	CHECKOUT	
19	2025-06-12	09:52	10:47	VINSENSIUS BRILYAN SANTOSA	DAK	Meeting	Salsabila Thanesya Putri	CHECKOUT	
20	2025-06-12	10:51	12:24	VINSENSIUS BRILYAN SANTOSA	DAK	Meeting	Salsabila Thanesya Putri	CHECKOUT	

Gambar 4.57 Implementasi Tampilan Halaman Inquiry

4.4 Pengujian

Dalam tahapan ini merupakan pengujian yang dilakukan setelah pengembangan sistem.

4.4.1 Deskripsi Pengujian

Pengujian menggunakan metode *Black-Box Testing* yang dilakukan dalam penelitian memiliki tujuan untuk mengamati hasil dari *input* dan *output* yang dihasilkan oleh sistem perangkat lunak (*software*) tanpa perlu mengetahui struktur kode dari sistem tersebut. Kemudian, *User Acceptance Testing* dilakukan bersama *end user* yang akan diberikan akses secara langsung menggunakan sistem yang telah dirancang untuk mengetahui kinerja dari sistem tersebut. Tahap akhir pengambilan data hasil pengujian dilakukan menggunakan kuesioner berisi pernyataan seputar sistem yang telah diuji. Terakhir, untuk melihat perbandingan durasi waktu proses registrasi sebelum dan sesudah implementasi sistem VMS dilakukan analisis berdasarkan data durasi waktu yang tercatat dalam sistem dengan durasi waktu proses registrasi manual di area *Head Office* PT Toyota Motor Manufacturing Indonesia.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

4.4.2 Prosedur Pengujian

Prosedur pengujian yang dilakukan berdasarkan metode *Black-Box Testing* dan *User Acceptance Testing*. Sebelum dimulai pengujian sistem perlu dirancang sebuah rencana pengujian sistem. Tabel 9 merupakan prosedur pengujian *Visitor Management System* untuk menguji fungsionalitas sistem, yaitu seluruh fitur berdasarkan *role* pengguna.

Tabel 9. Prosedur *Black-Box Testing*

Item Uji	Kode Uji	Butir Uji	Tester
Autentikasi	AU1	Menampilkan halaman <i>login</i>	<i>Host, Admin, Security, Receptionist</i>
	AU2	Melakukan <i>login</i>	
	AU3	Melakukan <i>logout</i>	
Dashboard	DB1	Menampilkan data grafik dan tabel kunjungan di hari H	<i>Admin, Security, Receptionist</i>
	DB2	Menampilkan modal detail data kunjungan	
	DB3	Mencari data tabel kunjungan di hari H berdasarkan nama pengunjung	
	DB4	Menampilkan filter data kunjungan dalam rentang tanggal	
	DB5	Mengunduh data tabel kunjungan di hari H	
	DB6	Menampilkan modal form registrasi	
	DB7	Menampilkan modal informasi <i>check-in/out</i> hasil <i>scan QR code</i>	
Menu <i>Registration</i>	RG1	Menampilkan list data tabel kunjungan yang didaftarkan	<i>Visitor, Host, Admin, Security,</i>
	RG2	Mencari data kunjungan berdasarkan nama pengunjung	

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Item Uji	Kode Uji	Butir Uji	Tester
	RG3	Mendaftarkan kunjungan baru	Receptionist
	RG4	Menampilkan modal detail data kunjungan	
	RG5	Mengubah detail data kunjungan yang terdaftar	
	RG6	Membatalkan kunjungan yang terdaftar	
Menu <i>Approval</i>	APR1	Menampilkan list data kunjungan yang butuh persetujuan	Host, Admin
	APR2	Mencari data kunjungan berdasarkan nama pengunjung	
	APR3	Menampilkan modal detail data kunjungan	
	APR4	Menyetujui kunjungan	
	APR5	Menyetujui kunjungan menggunakan <i>checkbox</i>	
	APR6	Menolak kunjungan	
	APR7	Menolak kunjungan menggunakan <i>checkbox</i>	
Menu <i>Check-In/Out</i>	CC1	Menampilkan list data kunjungan yang telah disetujui (<i>approve</i>)	Admin, Security, Receptionist
	CC2	Mencari data kunjungan berdasarkan nama pengunjung	
	CC3	Melakukan <i>Check-In/Out</i> secara manual	
Menu <i>Inquiry</i>	INQ1	Menampilkan seluruh riwayat data kunjungan	Host, Admin, Security, Receptionist
	INQ2	Menampilkan filter riwayat data kunjungan dalam rentang waktu yang dipilih	

Item Uji	Kode Uji	Butir Uji	Tester
	INQ3	Mencari riwayat data kunjungan berdasarkan nama pengunjung	
	INQ4	Menampilkan modal detail riwayat data kunjungan	

Pengujian UAT dilakukan menggunakan skenario pada Tabel 7. Waktu pelaksanaan pengujian dalam rentang tanggal 13 Mei 2025 hingga 10 Juni 2025. Pernyataan diberikan dalam bentuk kuesioner dengan target responden mewakili masing-masing pengguna, yaitu *Visitor*, *Admin*, *Host*, *Secretary*, *Receptionist*, dan *Security* PT Toyota Motor Manufacturing Indonesia untuk mengetahui kepuasan dalam menggunakan dan tampilan sistem.

Tabel 10. Bobot Kriteria Penilaian Kuesioner

Bobot	Jawaban Kuesioner
1	Sangat Tidak Setuju (STS)
2	Tidak Setuju (TS)
3	Ragu-ragu (RR)
4	Setuju (S)
5	Sangat Setuju (SS)

Sumber : Abraham and Ismail, 2021

Tabel 10 merupakan bobot penilaian untuk setiap pernyataan yang diajukan dalam kuesioner pada Tabel 11.

Tabel 11. Pernyataan Kuesioner *User Acceptance Testing*

No	Pernyataan
1	Tampilan antarmuka pengguna (UI) pada sistem VMS dapat dipahami.
2	Sistem VMS dapat menampilkan data atau informasi yang saya

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

No	Pernyataan
	butuhkan.
3	Petunjuk atau informasi yang disediakan dalam sistem VMS jelas.
4	Menu yang tersedia telah sesuai dengan kebutuhan saya.
5	Fitur dan halaman yang saya akses berfungsi dengan baik.
6	Sistem VMS memberikan respons yang sesuai setelah saya melakukan suatu tindakan.
7	Sistem VMS merespons dengan cepat saat digunakan.
8	Sistem VMS berjalan stabil dan tidak mengalami gangguan saat digunakan.
9	Sistem VMS dapat mempersingkat proses registrasi <i>visitor</i> .
10	Formulir registrasi dapat dimengerti dan diisi oleh pengguna.
11	Sistem VMS berhasil mencatat dan menyimpan data yang saya input dengan benar.
12	Sistem VMS dapat mengirimkan notifikasi berisi informasi kunjungan yang jelas.

4.4.3 Data Hasil Pengujian

4.4.3.1 Hasil *Black-Box Testing*

Pengujian *Black-Box* dilaksanakan untuk memvalidasi fungsionalitas sistem dari perspektif pengguna akhir. Setiap skenario uji difokuskan pada kesesuaian fungsionalitas sistem antara kondisi diharapkan (*Expected Result*) dengan kondisi sesungguhnya (*Actual Result*). Tabel 12 s.d. 17 merupakan skenario dan hasil pengujian dari setiap menu yang tersedia dalam sistem.

Tabel 12. Skenario *Black-Box Testing* Autentikasi

Kode	Skenario	<i>Expected Result</i>	<i>Actual Result</i>	Status
Menampilkan Halaman <i>Login</i>				
AU1	Memasukkan URL halaman login VMS melalui <i>web browser</i>	Sistem menampilkan halaman <i>login</i> VMS dengan <i>form input username</i> dan <i>password</i> , serta link akses sebagai <i>guest (visitor)</i>	Tampil halaman <i>login</i> VMS	Sesuai
Melakukan <i>Login</i>				
AU2.1	Memasukkan <i>username</i> dan <i>password</i> yang valid	Sistem menampilkan <i>toast message</i> “Berhasil Login” dan mengarahkan ke halaman sesuai <i>role</i>	Tampil <i>toast message</i> “Berhasil Login” dan halaman pengguna	Sesuai
AU2.2	Memasukkan <i>username</i> dan <i>password</i> yang tidak valid	Sistem menampilkan <i>toast message</i> “Incorrect Username or Password”	Tampil <i>toast message</i> “Incorrect Username or Password”	Sesuai
Melakukan <i>logout</i>				
AU3.1	Klik <i>icon user</i>	Sistem menampilkan tombol “ <i>Sign Out</i> ”	Tampil tombol “ <i>Sign Out</i> ”	Sesuai
AU3.2	Klik tombol “ <i>Sign Out</i> ”	Sistem mengalihkan ke halaman <i>login</i>	Tampil halaman <i>login</i>	Sesuai

Hak Cipta :

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Tabel 13. Skenario *Black-Box Testing* Dashboard

Kode	Skenario	<i>Expected Result</i>	<i>Actual Result</i>	Status
Menampilkan data grafik dan tabel kunjungan di hari H				
DB1	Setelah <i>login</i> , klik menu halaman “ <i>Dashboard</i> ” pada <i>sidebar menu</i>	Sistem menampilkan grafik dan tabel data kunjungan hari ini	Grafik dan tabel kunjungan ditampilkan sesuai data hari H	Sesuai
Menampilkan modal detail data kunjungan				
DB2	Memilih dan klik tombol <i>icon eye</i> “ <i>view detail</i> ” data kunjungan pada tabel kunjungan <i>dashboard</i>	Sistem menampilkan modal detail data kunjungan	Tampil modal detail data kunjungan	Sesuai
Mencari data tabel kunjungan di hari H berdasarkan nama pengunjung				
DB3	Klik <i>input</i> di halaman <i>dashboard</i> dan ketik nama pengunjung	Sistem memfilter dan menampilkan isi tabel kunjungan berdasarkan nama pengunjung	Tampil isi tabel sesuai nama pengunjung yang dicari	Sesuai
Menampilkan filter data kunjungan dalam rentang tanggal				
DB4	Klik tombol dan memilih rentang tanggal	Sistem menampilkan data kunjungan sesuai rentang tanggal	Tampil data kunjungan sesuai filter tanggal	Sesuai
Mengunduh data tabel kunjungan di hari H				
DB5	Klik tombol “ <i>Download</i> ” pada halaman <i>dashboard</i>	Sistem menghasilkan file unduhan berformat Excel/CSV	File berhasil diunduh sesuai data hari H	Sesuai
Menampilkan modal <i>form</i> registrasi				

Hak Cipta :

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Kode	Skenario	Expected Result	Actual Result	Status
DB6	Klik tombol “Add Visitor” pada halaman dashboard	Sistem menampilkan modal form registrasi pengunjung	Tampil modal form registrasi pengunjung	Sesuai
Menampilkan modal informasi check-in/out hasil scan QR code				
DB7.1	Scan QR code kunjungan yang telah dikirimkan sistem ke email pengunjung	Sistem menampilkan modal check-in	Tampil modal check-in	Sesuai
DB7.2	Scan QR code kunjungan yang telah melakukan check-in	Sistem menampilkan modal check-out	Tampil modal check-out	Sesuai
DB7.3	Scan QR code kunjungan yang telah melakukan check-out	Sistem menampilkan modal “Invalid Visitor QR”	Tampil modal “Invalid Visitor QR”	Sesuai

Tabel 14. Skenario Black-Box Testing Menu Registrasi

Kode	Skenario	Expected Result	Actual Result	Status
Menampilkan list data tabel kunjungan yang didaftarkan				
RG1	Setelah login, klik menu halaman “Visitor Registration” pada sidebar menu	Sistem menampilkan list data kunjungan terdaftar	Tampil data kunjungan yang terdaftar	Sesuai
Mencari data kunjungan berdasarkan nama pengunjung				
RG2	Klik input “Search” di halaman visitor registration dan	Sistem memfilter tabel dan menampilkan data kunjungan	Tampil isi tabel sesuai nama pengunjung yang dicari	Sesuai

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Kode	Skenario	Expected Result	Actual Result	Status
	ketik nama pengunjung	berdasarkan nama pengunjung yang dicari		
Mendaftarkan kunjungan baru				
RG3.1	Klik tombol “Add Visitor” pada halaman <i>visitor registration</i>	Sistem menampilkan modal <i>form</i> registrasi pengunjung	Tampil modal <i>form</i> registrasi pengunjung	Sesuai
RG3.2	Mengisi <i>form</i> registrasi dengan detail kunjungan seperti nama pengunjung, tujuan, email pengunjung, dll dan klik <i>submit</i>	Sistem menyimpan dan menampilkan data kunjungan baru pada list tabel	Tampil data kunjungan baru pada list tabel	Sesuai
RG3.3	Tidak mengisi seluruh <i>input field</i>	Sistem menampilkan <i>alert</i> seluruh <i>input field</i> harus diisi	Tampil <i>alert</i> seluruh <i>input field</i> harus diisi	Sesuai
Menampilkan modal detail data kunjungan yang terdaftar				
RG4	Memilih dan klik tombol <i>icon eye</i> “view detail” data kunjungan pada tabel <i>visitor registration</i>	Sistem menampilkan modal detail data kunjungan	Tampil modal detail data kunjungan	Sesuai
Mengubah detail data kunjungan yang terdaftar				
RG5.1	Memilih dan klik tombol <i>icon pen</i> “edit visitor” pada tabel kunjungan di <i>visitor registration</i>	Sistem menampilkan modal ubah data kunjungan	Tampil modal ubah data kunjungan	Sesuai
RG5.2	Mengubah isi data kunjungan yang	Sistem menyimpan dan menampilkan	Tampil data kunjungan yang	Sesuai

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Kode	Skenario	Expected Result	Actual Result	Status
	dipilih dan klik <i>submit</i>	data kunjungan yang diperbarui	diperbarui pada tabel	
Membatalkan kunjungan yang terdaftar				
RG6.1	Memilih dan klik tombol <i>icon X "cancel visitor"</i> pada tabel kunjungan di <i>visitor registration</i>	Sistem menampilkan modal konfirmasi hapus data kunjungan	Tampil modal konfirmasi hapus data kunjungan	Sesuai
RG6.2	Klik "Delete" untuk konfirmasi hapus data kunjungan	Sistem mengubah status data menjadi "CANCEL" dan tampil tabel registrasi tanpa data kunjungan yang telah dibatalkan	Status berubah menjadi "CANCEL" dan tampil tabel registrasi tanpa data kunjungan yang telah dibatalkan	Sesuai

Tabel 15. Skenario *Black-Box Testing* Menu *Approval*

Kode	Skenario	Expected Result	Actual Result	Status
Menampilkan list data kunjungan yang butuh persetujuan				
APR1	Setelah <i>login</i> , klik menu halaman <i>"Visitor Approval"</i> pada <i>sidebar menu</i>	Sistem menampilkan list data kunjungan yang menunggu persetujuan	Tampil data kunjungan yang menunggu persetujuan	Sesuai
Mencari data kunjungan berdasarkan nama pengunjung				
APR2	Klik <i>input "Search"</i> di halaman <i>visitor approval</i> dan ketik nama pengunjung	Sistem memfilter tabel dan menampilkan data kunjungan berdasarkan nama pengunjung yang dicari	Tampil isi tabel sesuai nama pengunjung yang dicari	Sesuai

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Kode	Skenario	Expected Result	Actual Result	Status
Menampilkan modal detail data kunjungan				
APR3	Memilih dan klik tombol <i>icon eye</i> “ <i>view detail</i> ” data kunjungan pada tabel <i>visitor approval</i>	Sistem menampilkan modal detail data kunjungan	Tampil modal detail data kunjungan	Sesuai
Menyetujui kunjungan				
APR4	Memilih dan klik tombol <i>icon checklist</i>	Sistem mengubah status menjadi “ <i>RESERVED</i> ”, tampil perubahan tabel list data kunjungan dan mengirim notifikasi ke email pengunjung	Status menjadi “ <i>RESERVED</i> ”, tampil perubahan tabel list data kunjungan dan mengirim notifikasi ke email pengunjung	Sesuai
Menyetujui kunjungan melalui modal detail				
APR4.1	Memilih dan klik tombol <i>icon eye</i> “ <i>view detail</i> ” data kunjungan pada tabel <i>visitor approval</i>	Sistem menampilkan modal detail data kunjungan	Tampil modal detail data kunjungan	Sesuai
APR4.2	Klik tombol “ <i>Accept</i> ”	Sistem mengubah status menjadi “ <i>RESERVED</i> ”, tampil perubahan tabel list data kunjungan, dan mengirim notifikasi ke email pengunjung	Status menjadi “ <i>RESERVED</i> ”, tampil perubahan tabel list data kunjungan, dan mengirim notifikasi ke email pengunjung	Sesuai
Menyetujui kunjungan menggunakan <i>checkbox</i>				

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Kode	Skenario	Expected Result	Actual Result	Status
APR5.1	Centang beberapa atau seluruh <i>checkbox</i> pada tepi tabel list data kunjungan dan	Sistem menampilkan tombol “ <i>Approved Selected</i> ”	Tampil tombol “ <i>Approved Selected</i> ”	Sesuai
APR5.2	Klik tombol “ <i>Approved Selected</i> ”	Sistem mengubah status kunjungan menjadi “ <i>RESERVED</i> ” dan mengirim notifikasi ke email pengunjung	Status kunjungan menjadi “ <i>RESERVED</i> ” dan mengirim notifikasi ke email pengunjung	Sesuai
Menolak kunjungan				
APR6	Memilih dan klik tombol <i>icon X</i> dan isi alasan penolakan pada modal <i>rejection</i> yang tampil	Sistem mengubah status kunjungan menjadi “ <i>REJECTED</i> ” dan mengirim notifikasi ke email pengunjung	Status kunjungan menjadi “ <i>REJECTED</i> ” dan mengirim notifikasi ke email pengunjung	Sesuai
Menolak kunjungan melalui modal detail				
APR7.1	Memilih dan klik tombol <i>icon eye</i> “ <i>view detail</i> ” data kunjungan pada tabel <i>visitor approval</i>	Sistem menampilkan modal detail data kunjungan	Tampil modal detail data kunjungan	Sesuai
APR7.2	Klik tombol “ <i>Reject</i> ”	Sistem menampilkan modal <i>input rejection</i> untuk mengisi alasan	Tampil modal <i>input rejection</i> untuk mengisi alasan	Sesuai
APR7.3	Isi modal <i>rejection</i> dan klik tombol “ <i>Reject</i> ”	Sistem mengubah status menjadi “ <i>REJECTED</i> ” dan mengirim notifikasi ke email	Status menjadi “ <i>RESERVED</i> ” dan mengirim notifikasi ke email	Sesuai

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Kode	Skenario	Expected Result	Actual Result	Status
		pengunjung	pengunjung	
Menolak kunjungan menggunakan <i>checkbox</i>				
APR8.1	Centang beberapa atau seluruh <i>checkbox</i> pada tepi tabel list data kunjungan dan	Sistem menampilkan tombol “ <i>Reject Selected</i> ”	Tampil tombol “ <i>Reject Selected</i> ”	Sesuai
APR8.2	Klik tombol “ <i>Reject Selected</i> ”	Sistem menampilkan modal <i>input rejection</i> untuk mengisi alasan	Tampil modal <i>input rejection</i> untuk mengisi alasan	Sesuai
APR8.3	Isi modal <i>rejection</i> dan klik tombol “ <i>Reject</i> ”	Sistem mengubah status menjadi “ <i>REJECTED</i> ” dan mengirim notifikasi ke email pengunjung	Status menjadi “ <i>REJECTED</i> ” dan mengirim notifikasi ke email pengunjung	Sesuai

Tabel 16. Skenario *Black-Box Testing* Menu *Check-In/Out*

Kode	Skenario	Expected Result	Actual Result	Status
Menampilkan list data kunjungan yang telah disetujui (<i>approve</i>)				
CC1	Setelah <i>login</i> , klik menu halaman “ <i>Visitor CICO</i> ” pada <i>sidebar menu</i>	Sistem menampilkan list data kunjungan yang telah terjadwal	Tampil data kunjungan yang telah terjadwal	Sesuai
Mencari data kunjungan berdasarkan nama pengunjung				
CC2	Klik <i>input</i> “ <i>Search</i> ” di halaman <i>visitor cico</i> dan ketik nama pengunjung	Sistem memfilter tabel dan menampilkan data kunjungan berdasarkan nama	Tampil isi tabel sesuai nama pengunjung yang dicari	Sesuai

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Kode	Skenario	Expected Result	Actual Result	Status
		pengunjung yang dicari		
Melakukan <i>Check-In/Out</i> secara manual				
CC3.1	Memilih dan klik tombol “IN” pada tabel data kunjungan	Sistem mengubah status kunjungan menjadi “CHECK IN” dan tombol “OUT” aktif	Status kunjungan berubah menjadi “CHECK IN” dan tombol “OUT” aktif	Sesuai
CC3.2	Memilih dan klik tombol “OUT” pada tabel data kunjungan	Sistem mengubah status kunjungan menjadi “CHECK OUT”	Status kunjungan berubah menjadi “CHECK OUT”	Sesuai

Tabel 17. Skenario *Black-Box Testing* Menu *Inquiry*

Kode	Skenario	Expected Result	Actual Result	Status
Menampilkan seluruh riwayat data kunjungan				
INQ1	Setelah <i>login</i> , klik menu halaman “ <i>Visitor Inquiry</i> ” pada <i>sidebar menu</i>	Sistem menampilkan list riwayat data kunjungan	Tampil list riwayat data kunjungan	Sesuai
Menampilkan filter riwayat data kunjungan dalam rentang tanggal				
INQ2	Klik tombol “ <i>Filter</i> ” dan memilih rentang tanggal	Sistem menampilkan riwayat data kunjungan sesuai rentang waktu	Tampil riwayat data kunjungan sesuai filter tanggal	Sesuai
Mencari riwayat data kunjungan berdasarkan nama pengunjung				
INQ3	Klik <i>input</i> “ <i>Search</i> ” di halaman <i>visitor inquiry</i> dan ketik	Sistem memfilter tabel dan menampilkan riwayat data	Tampil isi tabel riwayat kunjungan sesuai nama	Sesuai

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Kode	Skenario	Expected Result	Actual Result	Status
	nama pengunjung	kunjungan berdasarkan nama pengunjung yang dicari	pengunjung yang dicari	
Menampilkan modal detail riwayat data kunjungan				
INQ4	Memilih dan klik tombol <i>icon eye</i> “view detail” data kunjungan pada tabel <i>visitor inquiry</i>	Sistem menampilkan modal detail riwayat kunjungan	Tampil modal detail riwayat data kunjungan	Sesuai

4.4.3.2 Hasil User Acceptance Testing (UAT)

Tabel 18 menunjukkan hasil dari responden kuesioner UAT yang telah disebar di PT Toyota Motor Manufacturing Indonesia. UAT melibatkan sebanyak 30 responden, terdiri atas 10 *Visitor*, 2 *Admin*, 9 *Host*, 5 *Board of Directors (BoD)* *Secretary*, 3 *Receptionist*, dan 1 *Security*.

Tabel 18. Hasil Responden UAT

No	Pernyataan	Nilai Responden					Jumlah
		STS	TS	RR	S	SS	
		1	2	3	4	5	
1	Tampilan antarmuka pengguna (UI) pada sistem VMS dapat dipahami				7	23	30
2	Sistem VMS dapat menampilkan data atau informasi yang		1		17	12	30

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

No	Pernyataan	Nilai Responden					Jumlah
		STS	TS	RR	S	SS	
		1	2	3	4	5	
	saya butuhkan						
3	Petunjuk atau informasi yang disediakan dalam sistem VMS jelas			2	20	8	30
4	Menu yang tersedia dalam sistem VMS telah sesuai dengan kebutuhan saya				17	13	30
5	Fitur dan halaman sistem VMS yang saya akses berfungsi dengan baik		2	2	12	14	30
6	Sistem VMS memberikan respons yang sesuai setelah saya melakukan suatu tindakan				14	16	30
7	Sistem VMS merespons dengan cepat saat digunakan (contoh : submit form registrasi, berpindah antar			4	19	7	30

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

No	Pernyataan	Nilai Responden					Jumlah
		STS	TS	RR	S	SS	
		1	2	3	4	5	
	halaman, dsb)						
8	Sistem VMS berjalan stabil dan tidak mengalami <i>error</i> saat digunakan		4	2	14	10	30
9	Sistem VMS dapat mempersingkat proses registrasi <i>visitor</i>			1	21	8	30
10	Sistem VMS memiliki formulir registrasi yang dapat dimengerti dan diisi oleh pengguna			1	14	15	30
11	Sistem VMS berhasil mencatat dan menyimpan data yang saya <i>input</i> dengan benar				15	15	30
12	Sistem VMS dapat mengirimkan notifikasi melalui email berisi informasi kunjungan			1	10	19	30

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

No	Pernyataan	Nilai Responden					Jumlah
		STS	TS	RR	S	SS	
		1	2	3	4	5	
	yang jelas						

4.4.4 Analisis Data / Evaluasi Pengujian

4.4.4.1 Analisis / Evaluasi *Black-Box Testing*

Berdasarkan hasil pengujian *Black-Box Testing*, sebanyak 47 skenario uji telah dieksekusi mencakup seluruh fungsionalitas utama sistem. Selanjutnya perhitungan persentase keberhasilan sistem menggunakan *Black-Box Testing* dapat dihitung menggunakan persamaan rumus 1.

$$\text{Persentase keberhasilan} = \frac{\text{Jumlah skenario berhasil}}{\text{Total skenario}} \times 100\% \quad (1)$$

(Abraham and Ismail, 2021)

Sehingga perhitungan persentase keberhasilan pengujian sebagai berikut.

$$\frac{47}{47} \times 100\% = 100\%$$

Hasil perhitungan persentase keberhasilan di atas menunjukkan pengujian skenario untuk sistem VMS mendapatkan nilai evaluasi 100%. Sehingga dapat disimpulkan kondisi sesungguhnya (*Actual Result*) pada sistem sesuai dengan yang diharapkan (*Expected Result*) dan sistem berjalan dengan baik sesuai skenario yang telah dieksekusi.

4.4.4.2 Analisis / Evaluasi *User Acceptance Testing (UAT)*

Berdasarkan hasil pengujian UAT pada Tabel 18 melibatkan sebanyak 30 responden. Analisis data UAT dari kuesioner dilakukan dengan menghitung bobot

pada setiap pilihan jawaban. Persamaan rumus 2 merupakan perhitungan nilai rata-rata dari hasil pengujian UAT.

$$\text{Nilai Rata-rata} = \frac{\text{Total bobot nilai responden}}{\text{Total responden}} \quad (2)$$

(Abraham and Ismail, 2021)

Total bobot nilai responden untuk setiap pernyataan dihitung dengan mengalikan jumlah responden dan bobot nilai dari jawaban yang dipilih, terlampir pada Tabel 10. Setelah mengetahui nilai rata-rata dari pengujian UAT dan bobot maximum merupakan bobot terbesar dari pilihan jawaban kuesioner, persamaan rumus 3 merupakan rumus untuk memperoleh nilai persentase hasil evaluasi pernyataan UAT.

$$\text{Persentase} = \frac{\text{Nilai Rata-rata}}{\text{Bobot Maximum}} \times 100\% \quad (3)$$

(Abraham and Ismail, 2021)

Tabel 19 menunjukkan hasil nilai rata-rata dan persentase dari masing-masing pernyataan kuesioner.

Tabel 19. Analisis Rata-Rata dan Persentase Hasil UAT

No	Pernyataan	Nilai Rata-rata	Persentase
1	Tampilan antarmuka pengguna (UI) pada sistem VMS dapat dipahami	$\frac{(7 \times 4) + (23 \times 5)}{30} = \frac{143}{30} = 4,77$	$\frac{4,77}{5} \times 100\% = 95,3\%$
2	Sistem VMS dapat menampilkan data atau informasi yang saya butuhkan	$\frac{(1 \times 2) + (17 \times 4) + (12 \times 5)}{30} = \frac{130}{30} = 4,33$	$\frac{4,33}{5} \times 100\% = 86,7\%$
3	Petunjuk atau informasi yang disediakan dalam	$\frac{(2 \times 3) + (20 \times 4) + (8 \times 5)}{30}$	$\frac{4,20}{5} \times 100\% = 84\%$

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

No	Pernyataan	Nilai Rata-rata	Persentase
	sistem VMS jelas	$= \frac{126}{30} = 4,20$	
4	Menu yang tersedia dalam sistem VMS telah sesuai dengan kebutuhan saya	$\frac{(17 \times 4) + (13 \times 5)}{30}$ $= \frac{133}{30} = 4,43$	$\frac{4,43}{5} \times 100\%$ $= 88,7\%$
5	Fitur dan halaman sistem VMS yang saya akses berfungsi dengan baik	$\frac{(2 \times 2) + (2 \times 3) + (12 \times 4) + (14 \times 5)}{30}$ $= \frac{128}{30} = 4,26$	$\frac{4,26}{5} \times 100\%$ $= 85,3\%$
6	Sistem VMS memberikan respons yang sesuai setelah saya melakukan suatu tindakan	$\frac{(14 \times 4) + (16 \times 5)}{30}$ $= \frac{136}{30} = 4,53$	$\frac{4,53}{5} \times 100\%$ $= 90,7\%$
7	Sistem VMS merespons dengan cepat saat digunakan (contoh : <i>submit form</i> registrasi, berpindah antar halaman, dsb)	$\frac{(4 \times 3) + (19 \times 4) + (7 \times 5)}{30}$ $= \frac{123}{30} = 4,1$	$\frac{4,1}{5} \times 100\%$ $= 82\%$
8	Sistem VMS berjalan stabil dan tidak mengalami <i>error</i> saat digunakan	$\frac{(4 \times 2) + (2 \times 3) + (14 \times 4) + (10 \times 5)}{30}$ $= \frac{120}{30} = 4$	$\frac{4}{5} \times 100\%$ $= 80\%$
9	Sistem VMS dapat mempersingkat proses registrasi <i>visitor</i>	$\frac{(1 \times 3) + (21 \times 4) + (8 \times 5)}{30}$ $= \frac{127}{30} = 4,23$	$\frac{4,23}{5} \times 100\%$ $= 84,7\%$
10	Sistem VMS memiliki formulir registrasi yang dapat dimengerti	$\frac{(1 \times 3) + (14 \times 4) + (15 \times 5)}{30}$ $= \frac{134}{30} = 4,47$	$\frac{4,47}{5} \times 100\%$ $= 89,3\%$

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

No	Pernyataan	Nilai Rata-rata	Persentase
	dan diisi oleh pengguna		
11	Sistem VMS berhasil mencatat dan menyimpan data yang saya <i>input</i> dengan benar	$\frac{(15 \times 4) + (15 \times 5)}{30} = \frac{135}{30} = 4,5$	$\frac{4,5}{5} \times 100\% = 90\%$
12	Sistem VMS dapat mengirimkan notifikasi melalui email berisi informasi kunjungan yang jelas	$\frac{(1 \times 3) + (10 \times 4) + (19 \times 5)}{30} = \frac{138}{30} = 4,6$	$\frac{4,6}{5} \times 100\% = 92\%$

Setelah perhitungan pengujian UAT, Tabel 20 merupakan parameter persentase untuk perolehan nilai evaluasi pengujian UAT dari setiap pernyataan kuesioner.

Tabel 20. Persentase Kriteria Penilaian UAT

Persentase	Kriteria
0% - 20%	Sangat Kurang Baik
21% - 40%	Kurang Baik
41% - 60%	Cukup Baik
61% - 80%	Baik
81% - 100%	Sangat Baik

Sumber : Abraham and Ismail, 2021

a. Analisis pernyataan pertama

Berdasarkan tabel 18 dengan perolehan persentase sebesar 95,3%, respons terhadap kemudahan pemahaman antarmuka pengguna (UI) tergolong ke dalam kategori Sangat Baik. Nilai rata-rata tertinggi (4,77 dari 5) menunjukkan bahwa mayoritas pengguna merasa sangat setuju bahwa desain sistem VMS dapat dipahami.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

b. Analisis pernyataan kedua

Pernyataan ini mendapatkan rata-rata sebesar 4,33 dan persentase 86,7%, yang tergolong dalam kategori Sangat Baik. Hal ini menunjukkan bahwa sistem VMS telah berhasil memenuhi kebutuhan fungsionalnya dalam menampilkan data atau informasi yang dibutuhkan mayoritas pengguna.

c. Analisis pernyataan ketiga

Pernyataan ini mendapatkan rata-rata sebesar 4,20 dan persentase 84%, yang tergolong dalam kategori Sangat Baik. Hal ini menunjukkan bahwa mayoritas pengguna memahami petunjuk atau informasi yang disediakan dalam sistem VMS.

d. Analisis pernyataan keempat

Pernyataan ini mendapatkan rata-rata sebesar 4,43 dan persentase 88,7%, yang tergolong dalam kategori Sangat Baik. Hal ini menunjukkan bahwa menu yang tersedia dalam sistem VMS telah sesuai dengan kebutuhan mayoritas pengguna.

e. Analisis pernyataan kelima

Pernyataan ini mendapatkan rata-rata sebesar 4,27 dan persentase 85,3%, yang tergolong dalam kategori Sangat Baik. Hal ini menunjukkan mayoritas pengguna setuju bahwa fitur dan halaman dalam sistem VMS berfungsi dengan baik.

f. Analisis pernyataan keenam

Pernyataan ini mendapatkan rata-rata sebesar 4,53 dan persentase 90,7%, yang tergolong dalam kategori Sangat Baik. Hal ini menunjukkan mayoritas pengguna sangat setuju bahwa sistem VMS memberikan respons yang sesuai dengan perintah pengguna.

g. Analisis pernyataan ketujuh

Pernyataan ini mendapatkan rata-rata sebesar 4,1 dan persentase 82%, yang tergolong dalam kategori Sangat Baik. Hal ini menunjukkan mayoritas pengguna setuju bahwa sistem VMS merespons dengan cepat saat digunakan.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

h. Analisis pernyataan kedelapan

Pernyataan ini mendapatkan rata-rata sebesar 4 dan persentase 80%, yang tergolong dalam kategori Baik. Diantara pernyataan lain, pernyataan ini memiliki persentase terkecil dalam pengujian UAT. Setelah peneliti menganalisis berdasarkan rentang waktu pengisian kuesioner UAT terdapat perbedaan signifikan antara fase awal UAT tanggal 13 Mei s.d 22 Mei pengguna mengalami sistem *error* ketika sedang digunakan. Sedangkan setelah adanya *bug fixing* pada *error*, hasil respons kuesioner dalam rentang tanggal 29 Mei s.d. 10 Juni menunjukkan tidak ada pengguna yang mengalami kendala atau *error*.

i. Analisis pernyataan kesembilan

Pernyataan ini mendapatkan rata-rata sebesar 4,23 dan persentase 84,7%, yang tergolong dalam kategori Sangat Baik. Hal ini menunjukkan mayoritas pengguna setuju bahwa sistem VMS dapat mempersingkat proses registrasi *visitor*.

j. Analisis pernyataan kesepuluh

Pernyataan ini mendapatkan rata-rata sebesar 4,47 dan persentase 89,3%, yang tergolong dalam kategori Sangat Baik. Hal ini menunjukkan bahwa mayoritas pengguna mengerti dan dapat mengisi formulir registrasi pada sistem VMS.

k. Analisis pernyataan kesebelas

Pernyataan ini mendapatkan rata-rata sebesar 4,5 dan persentase 90%, yang tergolong dalam kategori Sangat Baik. Hal ini menunjukkan mayoritas pengguna setuju bahwa sistem VMS berhasil mencatat dan menyimpan data dengan benar.

l. Analisis pernyataan keduabelas

Pernyataan ini mendapatkan rata-rata sebesar 4,6 dan persentase 92%, yang tergolong dalam kategori Sangat Baik. Hal ini menunjukkan mayoritas pengguna setuju bahwa notifikasi yang dikirimkan sistem VMS melalui email berisi informasi kunjungan yang jelas.

m. Analisis persentase total

Persentase total merupakan akumulasi seluruh persentase 12 pernyataan kuesioner dibagi dengan total pernyataan kuesioner. Sehingga perolehan persentase total sebesar $\frac{1048,1}{12} = 87,4\%$. Dapat disimpulkan bahwa sistem VMS telah memenuhi kebutuhan pengguna serta berfungsi dengan sangat baik dan dapat diterima oleh pengguna akhir.

4.4.4.3 Analisis / Evaluasi *Before* dan *After* Implementasi Sistem VMS

Analisis ini dilakukan untuk menunjukkan perbandingan durasi waktu proses registrasi yang dilakukan secara manual (*before*) dengan proses registrasi menggunakan sistem VMS (*after*) terhadap 2 (dua) kondisi pengunjung, yaitu *planned visitor* (kondisi ideal) dan *unplanned visitor* (kondisi tidak ideal).

a. *Planned visitor*

- Manual: Tamu/*visitor* yang memiliki janji temu dengan *host* dan melakukan registrasi *on-site* di resepsionis.
- Sistem VMS: Tamu/*visitor* yang memiliki janji temu dengan *host* dan sudah memiliki *QR Code* untuk *check-in* di *Head Office* PT Toyota Motor Manufacturing Indonesia.

b. *Unplanned visitor*

- Manual: Tamu/*visitor* yang memiliki janji temu dengan *host* dan melakukan registrasi *on-site* di resepsionis.
- Sistem VMS: Tamu/*visitor* yang tidak memiliki janji temu dengan *host* dan melakukan registrasi *on-site* di resepsionis menggunakan sistem VMS.

Observasi proses registrasi manual ke *Head Office* PT Toyota Motor Manufacturing Indonesia dilakukan pada tanggal 21 Februari 2025, tercatat total durasi *planned visitor* selama 17 menit 19 detik dan *unplanned visitor* berdurasi 26 menit 01 detik dari melakukan registrasi, *host* datang menemui *visitor*, hingga *visitor* melakukan *check-in*. *Planned visitor* melakukan registrasi kunjungan dengan durasi selama 7 menit 29 detik dan *unplanned visitor* selama 8 menit 06

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak mengizinkan kepenggunaan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengutip dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

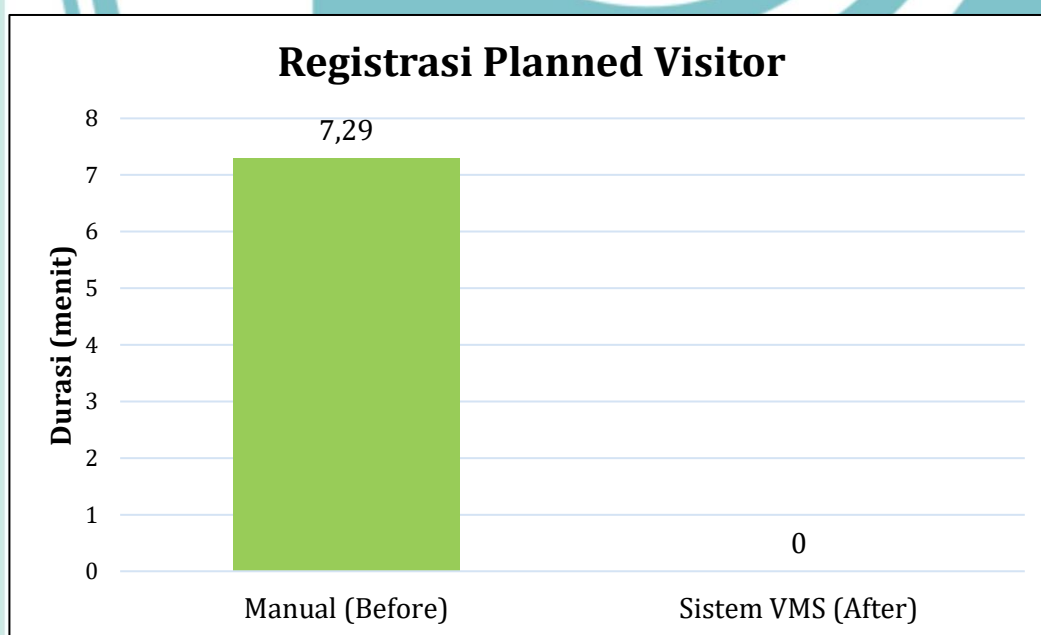
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

detik. Dapat disimpulkan selama kurun waktu tersebut terdaftar satu orang *visitor*.

Selanjutnya observasi setelah pengujian implementasi sistem VMS di *Head Office* PT Toyota Motor Manufacturing Indonesia pada tanggal 24 Juni 2025, tercatat total durasi waktu bagi *planned visitor* selama 9 menit 15 detik dari *host* datang menemui *visitor* hingga *visitor* melakukan *check-in* dan *unplanned visitor* berdurasi 14 menit 22 detik dari melakukan registrasi, *host* datang menemui *visitor*, hingga *visitor* melakukan *check-in*. Tercatat durasi waktu *unplanned visitor* mendaftarkan kunjungan di resepsionis menggunakan sistem VMS selama 3 menit 12 detik. *Planned visitor* yang sudah memiliki *QR Code* tidak melakukan registrasi dan hanya perlu melakukan *check-in*.

Gambar 4.58 merupakan grafik perbandingan data hasil observasi *before* dan *after* implementasi sistem VMS terhadap kondisi pengunjung *planned visitor* (kondisi ideal).



Gambar 4.58 Perbandingan durasi registrasi *planned visitor* *before* dan *after* menggunakan sistem VMS

Gambar 4.59 merupakan grafik perbandingan data hasil observasi *before* dan *after* implementasi sistem VMS terhadap kondisi pengunjung *unplanned visitor* (kondisi tidak ideal).

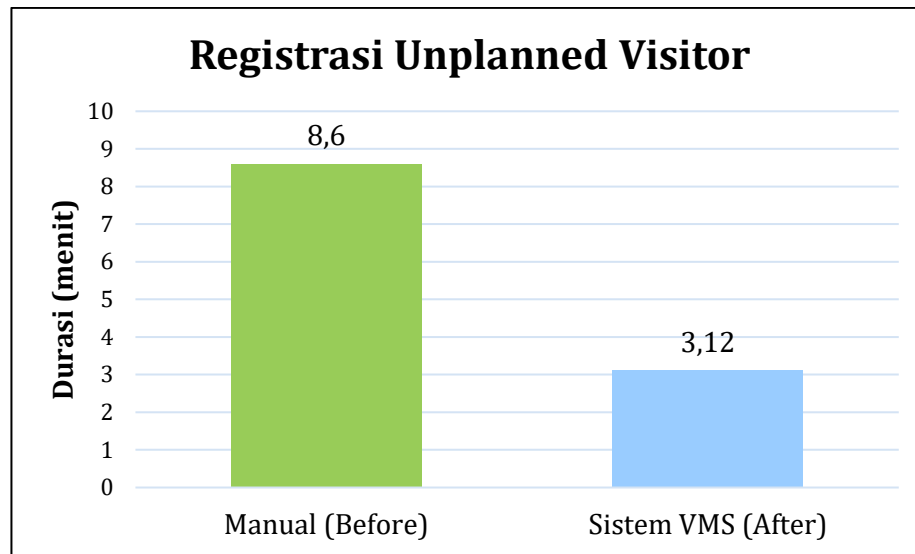
Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



Gambar 4.59 Perbandingan durasi registrasi *unplanned visitor before* dan *after* menggunakan sistem VMS

Selain itu, berdasarkan hasil analisis data kunjungan yang tercatat dalam database TB_M_VISIT pada Gambar 4.60 menampilkan dalam jangka waktu 10:33 s.d. 10:39 sistem mencatat sebanyak 7 (tujuh) kunjungan yang terdaftar. Dapat disimpulkan untuk kondisi pengunjung *planned visitor*, sistem VMS mampu menerima 7 (tujuh) registrasi kunjungan dalam kurun waktu 6 menit dengan rata-rata durasi ~0.86 menit/*visitor*.

TB_M_VISIT				Enter a SQL expression to filter results (use Ctrl+Space)	
Grid		Az qr_code		Az	create_at
43		bFy6GF4RZjCf4XwLO0+ZQ40eL1NQpBLYUL1WFf	ULL	[NU	2025-06-25 01:12:56.293
44		LRtUfZE7UxymtQJk4Yq4Xvxy+ZxQCdQai/4xPC2	ULL	[NU	2025-06-24 13:39:35.392
45		eSHyuz7ILB5xPcP26o/qoHfAfQgGuZZIDmtrbGjH	ULL	[NU	2025-06-24 10:43:12.332
46		5QcFKI3KmVy/dT4Ze4PFNPfK0oce9bA+s2mGLk	ULL	[NU	2025-06-24 10:39:00.651
47		/Xg364CBn3YhI5hCds4z08yyF4i28aYkG5X3eQtTa	ULL	[NU	2025-06-24 10:38:11.727
48		nQDfQ3IgTkxjZ6BQJ6NQoFezuCHFefz+7V5YrAo	ULL	[NU	2025-06-24 10:37:11.748
49		Yjo3k6FIntMLXRvWNUswqgZ2LEAEH38fidl7f1Jjn	ULL	[NU	2025-06-24 10:36:21.277
50		oRdZcHHaKosCd5iUZ6qU660iulEIM54GBa08a2u	ULL	[NU	2025-06-24 10:35:31.804
51		glUtVD00evPB4eeppu6yYRo0so77RTsSRyVFaMw	ULL	[NU	2025-06-24 10:34:39.987
52		yfv55RnyCHG8JCGhUneoGrdh3tseUzMsxBmNvO	ULL	[NU	2025-06-24 10:33:09.555
53		87PSmzDhjjzQFSNVcXZJAsCktzXMs5ZPeTzr6wMg	ULL	[NU	2025-06-24 10:01:14.031

Gambar 4.60 Database Daftar Registrasi Kunjungan

Tabel 21 merupakan hasil kesimpulan persentase setelah implementasi sistem VMS terhitung dari awal registrasi hingga *check-in* berdasarkan 2 (dua) kondisi, yaitu *planned visitor* dan *unplanned visitor* menggunakan persamaan 4.

$$\text{Persentase Reduce Time} = \frac{\text{Waktu Before} - \text{Waktu After}}{\text{Waktu Before}} \times 100\% \quad (4)$$

Tabel 21. Hasil Persentase *Reduce Time* Durasi Total Proses Registrasi

Kondisi Pengunjung	Durasi Before Manual	Durasi After Sistem VMS	Pengurangan waktu	Persentase Reduce Time
<i>Planned Visitor</i>	17 menit 19 detik	9 menit 15 detik	8 menit 04 detik	46,77%
<i>Unplanned Visitor</i>	26 menit 01 detik	14 menit 22 detik	11 menit 39 detik	43,79%

Tabel 22 menunjukkan hasil persentase setelah implementasi sistem VMS saat registrasi berdasarkan 2 (dua) kondisi, yaitu *planned visitor* dan *unplanned visitor* menggunakan persamaan 4.

Tabel 22. Hasil Persentase *Reduce Time* Durasi Registrasi

Kondisi Pengunjung	Durasi Before Manual	Durasi After Sistem VMS	Pengurangan waktu	Persentase Reduce Time
<i>Planned Visitor</i>	7 menit 29 detik	0 (tidak registrasi ulang)	7 menit 29 detik	100%
<i>Unplanned Visitor</i>	8 menit 06 detik	3 menit 12 detik	4 menit 54 detik	56,32%

4.4.4.4 Analisis / Evaluasi *Performance* Sistem VMS

Analisis *performance* sistem VMS menggunakan *tools* Apache JMeter. Gambar 4.61 menunjukkan konfigurasi yang digunakan dalam Apache JMeter untuk mengatur skenario simulasi beban (*load testing*) terhadap sistem VMS.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

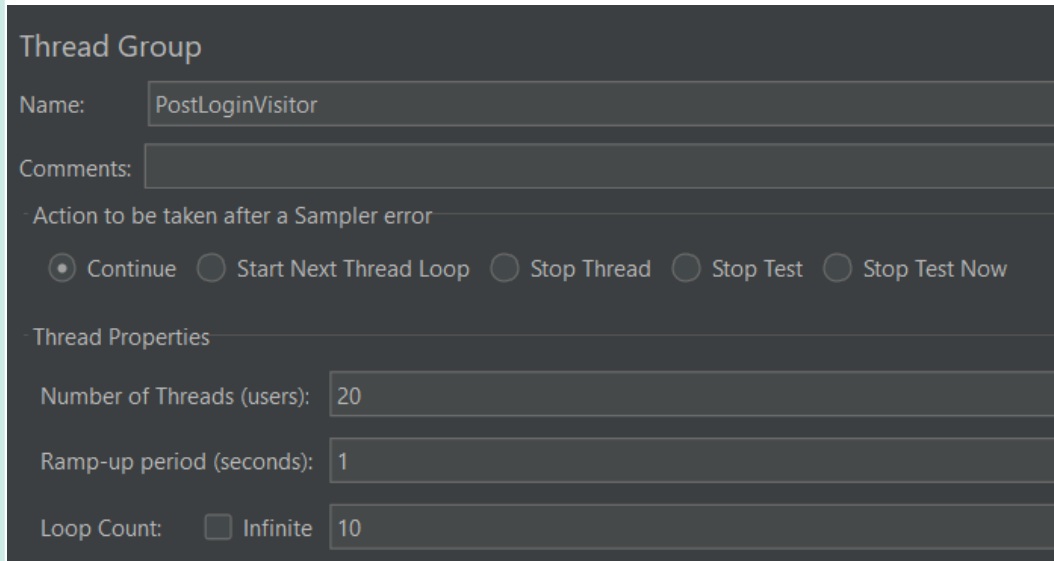
Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

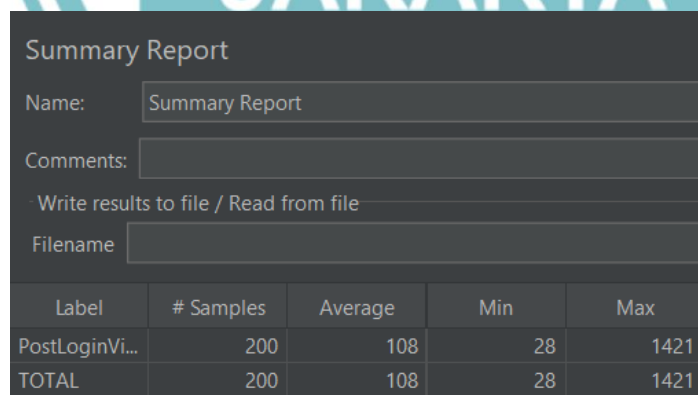


The screenshot shows the configuration for a Thread Group in Apache JMeter. The Name is set to 'PostLoginVisitor'. The Action to be taken after a Sampler error is set to 'Continue'. The Thread Properties section shows 'Number of Threads (users)' set to 20, 'Ramp-up period (seconds)' set to 1, and 'Loop Count' set to 10 (with the 'Infinite' checkbox unchecked).

Gambar 4.61 Konfigurasi Skenario Apache JMeter

Number of Threads (users) adalah jumlah *virtual users* atau *thread* yang akan dikirimkan secara paralel, dalam pengujian *performance* ini 20 pengguna akan disimulasikan. *Ramp-up Period (seconds)* adalah waktu (dalam detik) yang dibutuhkan untuk memulai seluruh *thread*. Dengan nilai 1 detik, maka seluruh 20 *thread* akan dimulai dalam 1 detik secara bersamaan, tanpa jeda antar *thread*.

Loop Count dimana setiap *thread (user)* akan menjalankan skenario sebanyak 10 kali. Karena tidak mencentang "*Infinite*", maka *loop* akan berhenti setelah 10 kali pengulangan. Sehingga total *request (sampler execution)* dari 20 *threads* dengan 10 *loop per thread* adalah sebanyak 200 *request*. Hasil pengujian *performance* sistem VMS ditunjukkan pada Gambar 4.62 terhadap 200 *request*.



The screenshot shows the Summary Report window in Apache JMeter. It displays the Name 'Summary Report' and a table with performance metrics for the 'PostLoginVi...' thread group and the total.

Label	# Samples	Average	Min	Max
PostLoginVi...	200	108	28	1421
TOTAL	200	108	28	1421

Gambar 4.62 Hasil *Performance Test* Sistem VMS

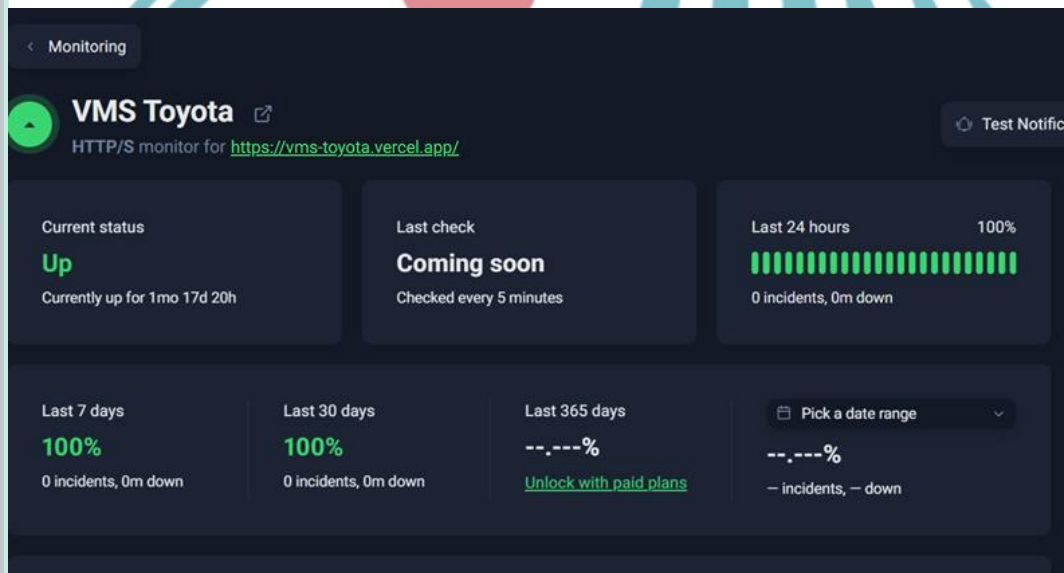


Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Gambar 4.62 merupakan hasil pengujian *performance* sistem VMS dimana menunjukkan hasil *load time* halaman rata-rata (*Average*) 0.108 detik pada koneksi standar terhadap 200 *request*. Dengan nilai *load time* tercepat (Min) sebesar 0.028 detik dan terlama (Max) 1.4 detik. Dapat disimpulkan hasil *load time* tersebut telah memenuhi kriteria sistem dimana maksimal *load time* selama 3 detik.

Selain itu, dilakukan analisis terhadap *uptime* sistem selama penelitian ini berlangsung menggunakan *tools* UptimeRobot. Gambar 4.63 menunjukkan sistem tidak mengalami *down* pada server selama dalam kurun waktu 1 (satu) bulan 17 hari.



Gambar 4.63 Hasil *Uptime Test* Sistem VMS

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil penelitian dan pengujian terhadap sistem *Visitor Management System* (VMS) yang dibangun untuk PT Toyota Motor Manufacturing Indonesia, dapat disimpulkan bahwa sistem telah memenuhi seluruh kebutuhan sistem, dibuktikan oleh hasil *Black-Box Testing* terhadap 47 skenario pengujian dengan tingkat keberhasilan 100%. Hasil *User Acceptance Test* (UAT) menunjukkan bahwa sistem diterima dengan baik oleh pengguna, dengan nilai total sebesar 87,4% dalam kategori interpretasi sangat baik.

Sistem VMS terbukti mempercepat proses registrasi kunjungan. Berdasarkan hasil observasi sebelum implementasi sistem, total waktu yang dibutuhkan untuk memproses kunjungan *planned visitor* adalah 17 menit 19 detik, sedangkan setelah sistem diimplementasikan, durasi tersebut menurun menjadi 9 menit 15 detik, menunjukkan pengurangan waktu sebesar 8 menit 04 detik atau setara 46,77%. Untuk *unplanned visitor*, durasi sebelum implementasi tercatat sebesar 26 menit 01 detik dan berkurang menjadi 14 menit 22 detik setelah implementasi, dengan selisih waktu 11 menit 39 detik atau setara 43,79%. Sedangkan waktu yang dibutuhkan untuk proses registrasi menurun dari 7 menit 29 detik menjadi 0 (nol) menit untuk *planned visitor* karena tidak perlu melakukan registrasi ulang, serta dari 8 menit 06 detik menjadi 3 menit 12 detik untuk *unplanned visitor*, dengan pengurangan waktu sebesar 4 menit 54 detik atau 56,32%.

Selain itu, sistem mencatat bahwa dalam waktu 6 menit, terdapat 7 *planned visitor* yang berhasil terdaftar, menunjukkan peningkatan proses pencatatan data kunjungan dibandingkan proses manual sebelumnya. Proses pengisian formulir dan penggunaan antarmuka sistem yang tercatat dalam UAT masing-masing memperoleh persentase 89,3% dan 95,3%, yang menunjukkan bahwa proses tersebut dapat dilakukan dengan lebih cepat oleh pengguna. Dari sisi pengelolaan

data, sistem VMS dapat mencatat data lebih akurat dibandingkan proses registrasi sebelumnya. Hal ini diperkuat oleh 90% responden UAT bahwa penyimpanan data dilakukan dengan benar dan akurat. Berdasarkan keseluruhan hasil tersebut, sistem VMS dinyatakan dapat mempersingkat proses registrasi dan mendukung pengelolaan data kunjungan secara digital.

5.2 Saran

Berdasarkan kesimpulan yang telah diuraikan, terdapat beberapa saran yang dapat menjadi pertimbangan penelitian selanjutnya untuk pengembangan sistem VMS di masa depan maupun implementasi di PT Toyota Motor Manufacturing Indonesia. Saran-saran ini bertujuan untuk memaksimalkan manfaat dan keberlanjutan sistem.

1. Untuk lebih mempercepat proses entri data dan meningkatkan akurasi *input* pendaftaran, sistem dapat dilengkapi dengan teknologi OCR (*Optical Character Recognition*) untuk memindai kartu identitas seperti KTP, SIM, atau paspor. Sehingga mengurangi kesalahan pengetikan (*human error*).
2. Mengintegrasikan sistem VMS dengan sistem keamanan seperti *flap gate barrier*. Dimana kartu akses pengunjung dapat secara otomatis aktif saat *check-in* dan nonaktif saat *check-out*, sehingga meningkatkan keamanan secara *real-time*.
3. Sistem VMS disediakan fitur pembatalan yang dapat dilakukan oleh *visitor* untuk mengakomodasi perubahan rencana mendadak.
4. Fitur penjadwalan untuk melihat *availability host* sehingga memungkinkan *visitor* atau *host* untuk menentukan tanggal dan jam kunjungan lebih fleksibel sesuai dengan agenda yang tersedia.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

DAFTAR RIWAYAT HIDUP



Lahir di Jakarta, 27 Agustus 2003. Lulus dari SDN Negeri 10 Jakarta pada tahun 2015, SMP Negeri 166 Jakarta pada tahun 2018, dan SMA Negeri 46 Jakarta pada tahun 2021. Saat ini sedang menempuh Program Studi Diploma Empat Teknik Informatika di Politeknik Negeri Jakarta. Memiliki ketertarikan pada bidang Pengembangan Web dan Data Science.

**POLITEKNIK
NEGERI
JAKARTA**



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

DAFTAR PUSTAKA

- Abraham, J., & Ismail, I. E. (2021). *Unit Testing dan User Acceptance Testing pada Sistem Informasi Pelayan Kategorial Pelayanan Anak*. Repository Politeknik Negeri Jakarta.
- Aida, Hermina, D., & Norlaila. (2025). JENIS DATA PENELITIAN KUANTITATIF (Korelasional, Komparatif, Dan Eksperimen). *Jurnal Ilmiah Keislaman Dan Kemasyarakatan*, 10(1).
- Alif Ramadhan, J., Tresya Haniva, D., & Suharso, A. (2023). Systematic Literature Review Penggunaan Metodologi Pengembangan Sistem Informasi Waterfall, Agile, dan Hybrid. In *Journal Information Engineering and Educational Technology* (Vol. 07).
- Azizi, A., Naufal, M., Ghazali, A., Khair, O. S., Tsabit, Z., & Kusnandar, S. (2024). Pengembangan Dashboard Admin Bukupedia. *Jurnal Teknik Indonesia*, 3(1). <https://doi.org/10.58860/jti.v3i1>
- Clint, I. R., & Tony. (2021). *Rancang Bangun Aplikasi Pembelajaran ReactJS Menggunakan Metode Gamifikasi Berbasis Android*. <https://kc.umn.ac.id/id/eprint/15836>
- Danar Hadi Bachtiar, Paniran Paniran, & I Made Budi Suksmadana. (2024). Perancangan Back-end Api pada Aplikasi Mobile Fruityfit Menggunakan Framework Express JS. *Mars : Jurnal Teknik Mesin, Industri, Elektro Dan Ilmu Komputer*, 2(3), 107–117. <https://doi.org/10.61132/mars.v2i3.138>
- Dhaniswara, A. A., Hasniatasya, 1, Miranda, A., Diana3, D., Afifah, Z., Inez, R., Febrina5, H., & Alit, R. (2023). PERANCANGAN ULANG DESAIN WEBSITE RESPONSIVE PT. PPI DENGAN MENGGUNAKAN FIGMA. *Jurnal Ilmu Teknik*, 1(2), 69–78.
- Khatib Sulaiman, J., Syifa Qaisa, R., Putri, A., Maghfirah, H., & Artikel Abstrak, I. (2023). Perancangan Aplikasi Todo menggunakan Node.js dan REST API. *Indonesian Journal of Computer Science Attribution*, 12(6), 2023–4132.
- Ngurah Mega Nata, G., Wayan Wiraguna, I., Putu Ramayasa, I., Studi Manajemen Informatika, P., Studi Sistem Informasi, P., & STIKOM BALI Jln Raya Puputan No, I. (2023). SISTEM INFORMASI KEHADIRAN SISWA BERBASIS SMS GATEWAY DENGAN QR CODE. *Jurnal Manajemen Informatika & Sistem Informasi (MISI)*, 6(1). <https://doi.org/10.36595/misi.v5i2>
- Nursaid, F. F., Hendra Brata, A., & Kharisma, A. P. (2020). Pengembangan Sistem Informasi Pengelolaan Persediaan Barang Dengan ReactJS Dan React Native Menggunakan Prototype (Studi Kasus : Toko Uda Fajri). *Jurnal*



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Pengembangan Teknologi Informasi Dan Ilmu Komputer, 4(1), 46–55.
<http://j-ptiik.ub.ac.id>

Panjaitan, J., & Pakpahan, A. F. (2021). Perancangan Sistem E-Reporting Menggunakan ReactJS dan Firebase. *Jurnal Teknik Informatika Dan Sistem Informasi*, 7(1). <https://doi.org/10.28932/jutisi.v7i1.3098>

Prastomo, A., & Syuhardi, Y. I. (2023). DESIGN OF VISITOR MANAGEMENT SYSTEM APPLICATION IN THE INTERLINK LOBBY OF SEJAHTERA DATA CENTER. *Journal of Information System, Informatics and Computing*, 7(1), 139. <https://doi.org/10.52362/jisicom.v7i1.1101>

Purnomo, S., & Alijoyo, F. A. (2024). Sistem Peminjaman Barang Menggunakan QR Code Berbasis Aplikasi Android. *Jurnal Teknologi Dan Sistem Informasi Bisnis*, 6(2), 322–328. <https://doi.org/10.47233/jteksis.v6i2.1350>

Rizal, C., Fachri, B., & Hasanuddin, M. (2024). WATERFALL METHODE DALAM RANCANG BANGUN SISTEM INFORMASI POTENSI WISATA BERBASIS WEB. In *Journal of Science and Social Research* (Issue 4). <http://jurnal.goretanpena.com/index.php/JSSR>

Sauda, S., & Barokah, M. (2022). PENERAPAN NODEJS DAN POSTGRESQL SEBAGAI BACKEND PADA APLIKASI ECOMMERCE LOCALLA. *INFOTECH*, 8. <https://doi.org/10.31949/infotech.v8i2.2944>

Suroto, Rahaju, P., Widyati, D., & Ramadhani, H. N. (n.d.). *Development_of_a_Visitor_Management_Syst.*

Suryati, P., Susanto, A., & Sudarmanto. (2024). SISTEM DATABASE TERDISTRIBUSI UNTUK PENINGKATAN KE TERSEDIAAN DATAPADA APLIKASI REKAM MEDIS. *Journal of Innovation Research and Knowledge*, 3. <https://bajangjournal.com/index.php/JIRK/article/view/7375/5828>

Wahyudi, I., & Alameka, F. (2023). ANALISIS BLACKBOX TESTING DAN USER ACCEPTANCE TESTING TERHADAP SISTEM INFORMASI SOLUSIMEDSOSKU. *Jurnal Teknosains Kodepena* |, 04, 1–9.

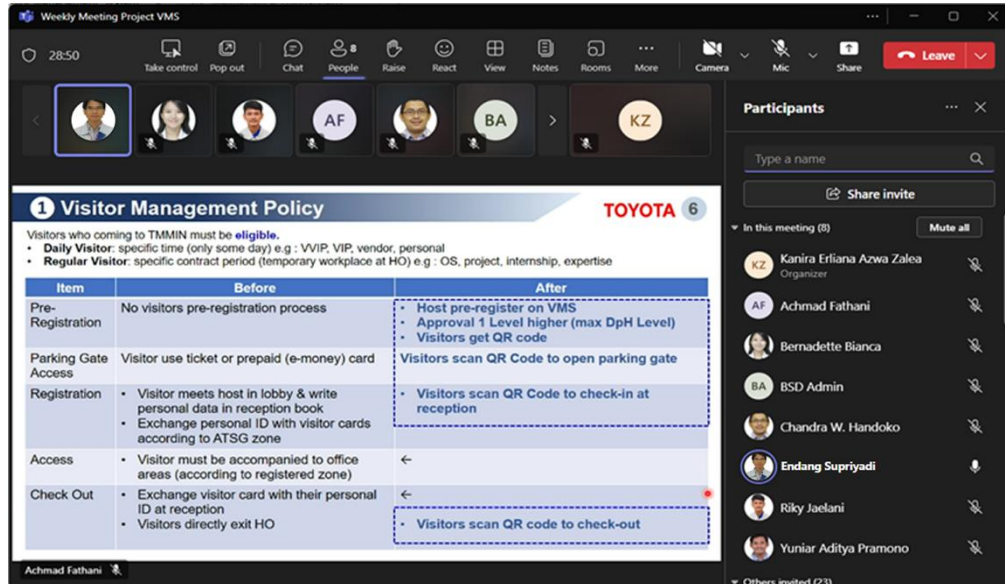
Wahyudi, T., & Faqih, H. (2021). Pengembangan Sistem Informasi Presensi Menggunakan Metode Waterfall. *Indonesian Journal on Software Engineering (IJSE)*, 7(2), 120–129. <http://ejournal.bsi.ac.id/ejurnal/index.php/ijse120>

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

LAMPIRAN

Lampiran 1 *Initial requirement* dengan karyawan PT TMMIN



Gambar di atas merupakan dokumentasi wawancara yang diselenggarakan pada tanggal 09 Januari 2025 via daring bersama *stakeholders* dari PT TMMIN, salah satunya Bapak Endang Supriyadi selaku *Staff Visitor Handling Plant Management, GA, SHE Dept* untuk membahas *requirements* dari sistem VMS yang akan dibangun.

Hari, Tanggal	Kamis, 09 Januari 2025
Tempat	Microsoft Teams (<i>Head Office & Karawang Plant PT TMMIN</i>)
Poin Penting	Proses registrasi tamu yang masih dilakukan secara manual, baik di area <i>Head Office</i> maupun kawasan Plant, belum berjalan secara optimal. Belum tersedianya sistem digital yang mampu menyelesaikan <i>pain point</i> yang ditemukan, terlebih karena adanya perbedaan perlakuan (<i>special treatment</i>) pada tiap area. Meskipun demikian, <i>Head Office</i> sebagai pusat perusahaan dinilai memiliki potensi yang kuat untuk dijadikan <i>stepping stone</i>

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

dalam merealisasikan dan mengadopsi sistem *Visitor Management System* (VMS) secara bertahap.

- *Pain point* pertama, registrasi masih dilakukan melalui buku tamu fisik, yang menyebabkan keterbatasan informasi, seperti seringkali tidak tercatat siapa *host* dari pengunjung dan perusahaan tidak tahu kapan waktu masuk dan keluar pengunjung. Divisi internal seperti *General Affairs Division* (GAD) dan *Plant Administration Division* (PAD) memiliki alur dan perlakuan penerimaan tamu yang berbeda karena kepala divisi yang berbeda pula. Hal ini menyebabkan koordinasi kunjungan antar area hanya mengandalkan komunikasi informal seperti *chat* WhatsApp antar *staff*, yang tidak terdokumentasi secara resmi dan sistematis.
- Dalam waktu satu kali penerimaan tamu yang individu rata-rata ketika registrasi memakan waktu selama 21-25 menit. Sejauh ini regulasi perusahaan belum menetapkan kuota pengunjung dalam satu hari hanya bergantung pada tempat terutama *working area/ruang meeting* yang masih *available*.
- *Pain point* kedua, Di level operasional, *staff* General Reception Officer (GRO) masih melakukan rekap data kunjungan secara manual sebelum pergantian *shift* pagi dan siang. Hal ini rentan menyebabkan kesalahan input data.
- *Pain point* lainnya, masih sering ditemukan kasus pengunjung datang ke area perusahaan tanpa adanya janji temu sebelumnya. Hal ini menyulitkan proses penyaringan, pencatatan, dan penerimaan tamu,

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

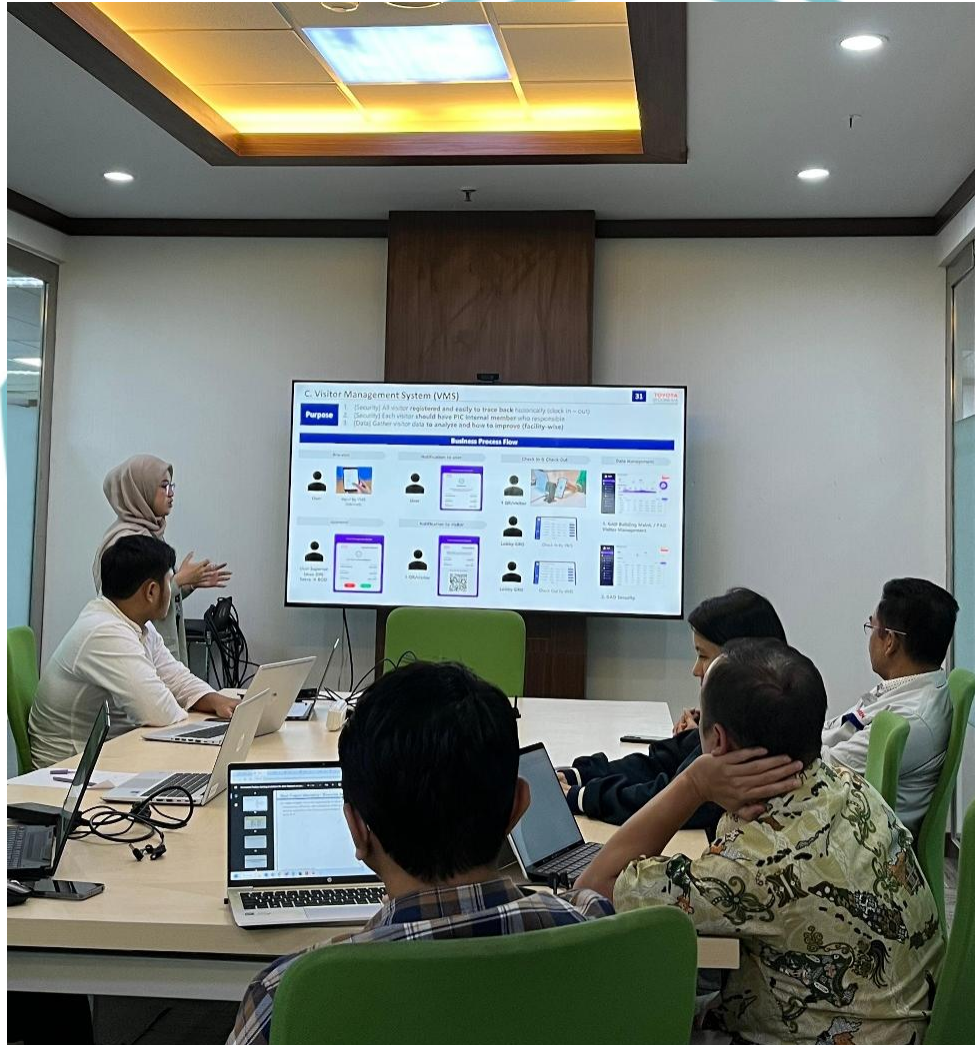
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

serta berpotensi menimbulkan gangguan terhadap ketertiban dan keamanan lingkungan perusahaan.

Lampiran 2 Pemaparan hasil analisis *requirements* dengan *stakeholders* PT TMMIN



Pada tanggal 16 Januari 2025 kegiatan pemaparan hasil analisis kebutuhan sistem VMS dan fitur yang telah didefinisikan sesuai dengan kebutuhan perusahaan. Kegiatan ini dihadiri juga oleh Bapak Yandri Pardomuan selaku *Division Head* dan Direktur *Administration, Corporate Planning & Legal Office, & Product Business Management*.

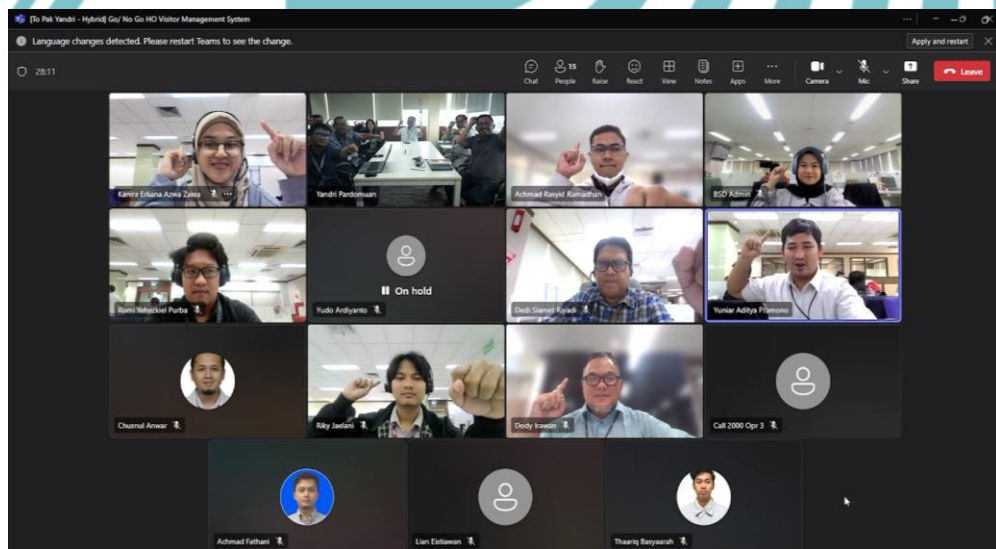
Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Lampiran 3 User Acceptance Testing (UAT) PT TMMIN

TOYOTA INDONESIA Operation Readiness: FGD, User Test, & Go Live

Activity	Participant	Item to be Check	Problem/Kadai	Countermeasure	Eval
FGD: HO User voice grasping 18 Feb/25	1. Operation : - GRO - Security 2. User : Rep. Div - GA TMMIN - HR/GA TAM - Union	1. Dashboard 2. Registration Visitor	- Visitor graph data is not real time - Visitor data files cannot be downloaded - Not integrated VMS vs TAM HR Portal (Protected)	- Synchronize time zone server and client - Upgrade function for download to excel file - Create a special TAM user account for each floor (2nd, 4th, 5th Floor)	O O O
User Test: Related parties 21 Feb/25	1. Operation : - GRO - Security 2. User : Rep. Div - GA TMMIN - HR/GA TAM	3. Approval Visitor	- TAM user can't input user name - Registration results don't appear in register menu - Incomplete visitor data can still be submitted - UI on smartphone is less responsive - Visitor group list input cannot scroll - SH can still approve it himself - Time difference : input data VMS and approval notification	- Input TAM employee data to VMS database - Upgrade function for register menu - Improve visitor data logic - Resize conditions for mobile phone - Upgrade function for visitor group list - Improve for next level approval - Synchronize time zone server and client	O O O O O O O
800 Secretary voice 24 Feb/25	1. User : - All 800 Secretary	4. Check-in / Check-Out Visitor 5. Inquiry 6. Logs Visitor	- Not all group visitor data is displayed yet - Group visitor can not partial check-in - Visitor data is not recorded	- Upgrade for partial check-in - Synchronize time zone server and client - Synchronize time zone server and client	O O O O
	Machine	1. Tab for Receptionist /Security 2. Keyboard 3. QR Scanner 4. Ms. Edge / Chrome 5. Server	USB input can't be multiconnection QR scanner and changing (only 1 input) N/A N/A N/A N/A	Adding a USB connector (more than 1 input) N/A N/A N/A N/A	O O O O O
	Man	User knowledge	Not yet socialization to all user	Preparing socialization for HO in W3-W4 March	O
	Material	Data	N/A	N/A	O



Setelah dilakukan UAT dengan pengguna PT TMMIN, agenda *Go-Live* dilakukan sebagai inisiasi penggunaan sistem VMS di area *Head Office* PT TMMIN bersama dengan *stakeholders* dari PT TMMIN. Pelaksanaan kegiatan ini dilakukan *hybrid online & offline*.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Lampiran 4 Observasi *Before* Implementasi VMS di Head Office PT TMMIN



Observasi dilakukan di area *lobby* PT Toyota Motor Manufacturing Indonesia Jakarta Utara pada tanggal 21 Februari 2025 untuk mengamati kondisi proses registrasi secara manual sebelum diberlakukannya sistem VMS sebagai sarana registrasi. Tujuan kegiatan ini untuk menganalisis masalah utama proses registrasi kunjungan yang dilakukan oleh *visitor*.

Lampiran 5 Observasi *After* Implementasi VMS di Head Office PT TMMIN



Observasi dilakukan di area *lobby* PT Toyota Motor Manufacturing Indonesia Jakarta Utara pada tanggal 24 Juni 2025 untuk mengamati kondisi proses registrasi secara digital setelah diberlakukannya sistem VMS sebagai sarana registrasi. Tujuan kegiatan ini dilakukan untuk mengevaluasi hasil kinerja sistem dalam proses registrasi yang dilakukan oleh *visitor*.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

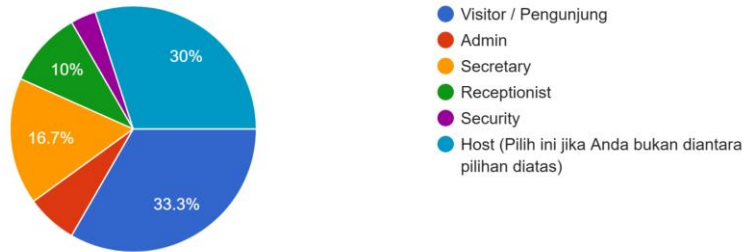
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

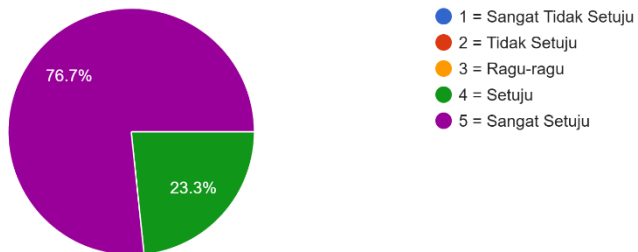
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Lampiran 6 Responden Kuesioner UAT

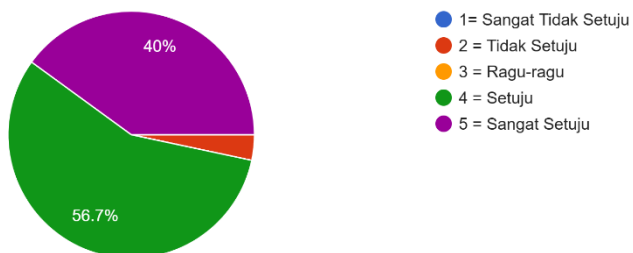
Tester (Isi sesuai dengan posisi/role Anda)
30 responses



Tampilan antarmuka pengguna (UI) pada sistem VMS dapat dipahami
30 responses



Sistem VMS dapat menampilkan data atau informasi yang saya butuhkan
30 responses



Hak Cipta :

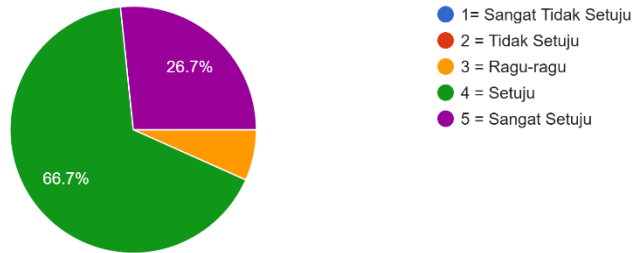
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

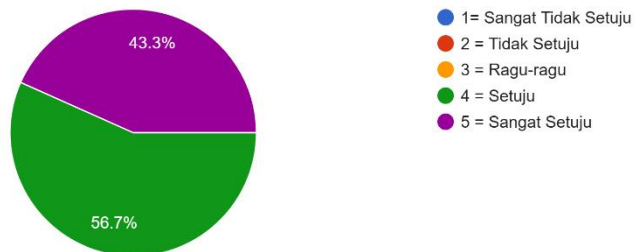
Petunjuk atau informasi yang disediakan dalam sistem VMS jelas

30 responses



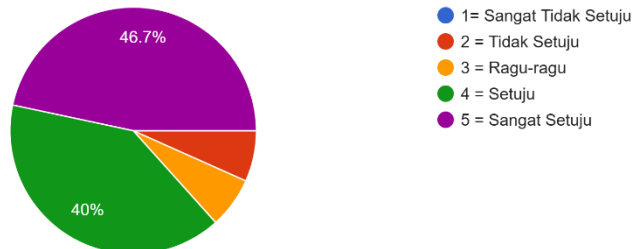
Menu yang tersedia dalam sistem VMS telah sesuai dengan kebutuhan saya

30 responses



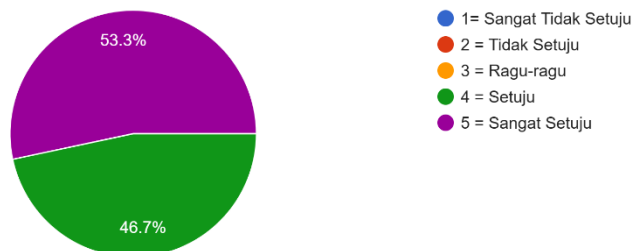
Fitur dan halaman sistem VMS yang saya akses berfungsi dengan baik

30 responses



Sistem VMS memberikan respons yang sesuai setelah saya melakukan suatu tindakan

30 responses



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

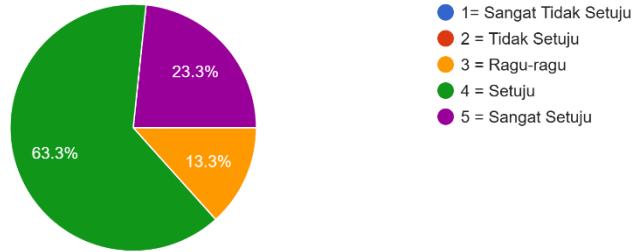
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

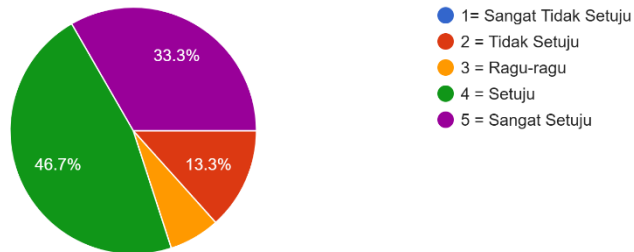
Sistem VMS merespons dengan cepat saat digunakan (contoh : submit form registrasi, berpindah antar halaman, dsb)

30 responses



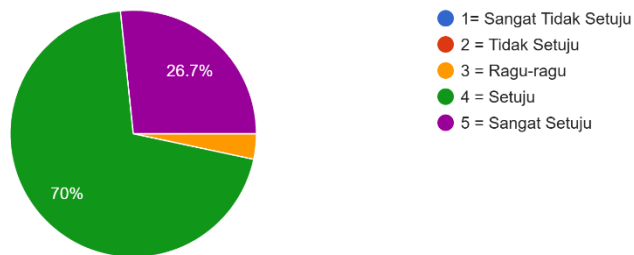
Sistem VMS berjalan stabil dan tidak mengalami error saat digunakan

30 responses



Sistem VMS dapat mempersingkat proses registrasi visitor

30 responses



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

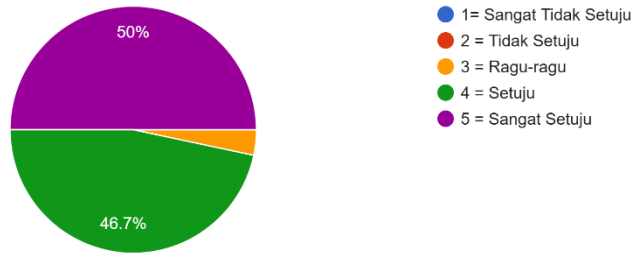
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

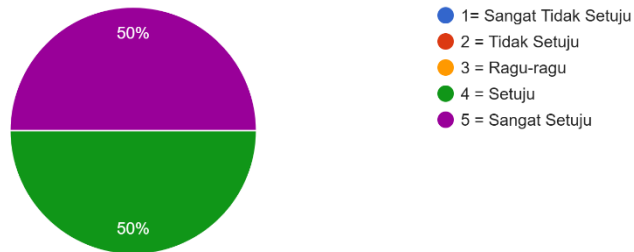
Sistem VMS memiliki formulir registrasi yang dapat dimengerti dan diisi oleh pengguna

30 responses



Sistem VMS berhasil mencatat dan menyimpan data yang saya input dengan benar

30 responses



Sistem VMS dapat mengirimkan notifikasi melalui email berisi informasi kunjungan yang jelas

30 responses

