

RANCANG BANGUN APLIKASI PEMESANAN ONLINE LAPANGAN BERBASIS WEB MENGGUNAKAN FRAMEWORK LARAVEL (STUDI KASUS : GOR EWANGGA KUNINGAN)

TUGAS AKHIR

Disusun sebagai salah satu syarat untuk kelulusan Program Strata 1,
Program Studi Teknik Informatika, Universitas Pasundan Bandung

oleh:

Mochammad Nizar Albaehaqi
Nrp. 20.304.0035



**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PASUNDAN BANDUNG
JULI 2026**

LEMBAR PENGESAHAN LAPORAN TUGAS AKHIR

Telah diujikan dan dipertahankan dalam Sidang Sarjana Program Studi Teknik Informatika Universitas Pasundan Bandung, pada hari dan tanggal sidang sesuai berita acara sidang, tugas akhir dari :

Nama : Mochammad Nizar Albaehaqi
Nrp : 20.304.0035

Dengan judul :

**“RANCANG BANGUN APLIKASI PEMESANAN ONLINE LAPANGAN BERBASIS
WEB MENGGUNAKAN FRAMEWORK LARAVEL”
(STUDI KASUS : GOR EWANGGA KUNINGAN)**

Bandung, 23 Juli 2026
Menyetujui,

Pembimbing Utama

A purple circular stamp of Universitas Pasundan is visible, featuring a central emblem and the text "FAKULTAS TEKNIK" and "UNIVERSITAS PASUNDAN". Overlaid on the stamp is a handwritten signature in black ink.

(Anggoro Ari Nurcahyo, S.T., M.Kom.)

LEMBAR PERNYATAAN KEASLIAN TUGAS AKHIR

Saya menyatakan dengan sesungguhnya bahwa :

1. Tugas akhir ini adalah benar-benar asli dan belum pernah diajukan untuk mendapatkan gelar akademik, baik di Universitas Pasundan Bandung maupun di Perguruan Tinggi lainnya.
2. Tugas akhir ini merupakan gagasan, rumusan dan penelitian saya sendiri, tanpa bantuan pihak lain kecuali arahan dari tim Dosen Pembimbing.
3. Dalam tugas akhir ini tidak terdapat karya atau pendapat orang lain, kecuali bagian-bagian tertentu dalam penulisan laporan Tugas Akhir yang saya kutip dari hasil karya orang lain telah dituliskan dalam sumbernya secara jelas sesuai dengan norma, kaidah, dan etika penulisan karya ilmiah, serta disebutkan dalam Daftar Pustaka pada tugas akhir ini.
4. Kakas, perangkat lunak, dan alat bantu kerja lainnya yang digunakan dalam penelitian ini sepenuhnya menjadi tanggung jawab saya, bukan tanggung jawab Universitas Pasundan Bandung.

Apabila di kemudian hari ditemukan seluruh atau sebagian laporan tugas akhir ini bukan hasil karya saya sendiri atau adanya plagiasi dalam bagian-bagian tertentu, saya bersedia menerima sanksi akademik, termasuk pencabutan gelar akademik yang saya sandang sesuai dengan norma yang berlaku di Universitas Pasundan, serta perundang-undangan lainnya.

Bandung, 23 Juli 2026

Yang membuat pernyataan,



(MOCHAMMAD NIZAR ALBAEHAQI)

NRP. 20.304.0035

ABSTRAK

Proses penyewaan lapangan di GOR Ewangga Kuningan masih dilakukan secara manual, di mana penyewa harus datang langsung ke lokasi untuk melakukan pemesanan. Kondisi ini menimbulkan sejumlah kendala, seperti keterbatasan akses informasi, kesulitan memperoleh jadwal lapangan secara akurat dan terkini, serta pelayanan yang kurang optimal ketika jumlah calon penyewa meningkat. Selain itu, tingginya jumlah pemesanan sering menyebabkan jadwal tumpang tindih karena belum adanya sistem pengelolaan jadwal yang transparan dan real-time.

Penelitian ini bertujuan merancang dan membangun aplikasi pemesanan lapangan berbasis web menggunakan framework Laravel sebagai solusi atas permasalahan tersebut. Sistem yang dikembangkan menyediakan fitur registrasi pengguna, pemesanan lapangan, pengecekan jadwal secara terkini, pembayaran digital, serta pengelolaan data oleh admin. Metode pengembangan yang digunakan adalah model Waterfall, yang meliputi tahap analisis kebutuhan, perancangan, implementasi, pengujian, dan pemeliharaan.

Hasil penelitian diharapkan dapat meningkatkan efisiensi proses pemesanan, memberikan transparansi informasi ketersediaan lapangan, serta mendukung pengelola GOR dalam mengoptimalkan pelayanan. Dengan adanya sistem ini, kualitas layanan GOR Ewangga Kuningan diharapkan meningkat seiring dengan perkembangan teknologi digital.

Kata kunci: Pemesanan Lapangan Online, GOR Ewangga Kuningan, Laravel, Web-Based Application, Waterfall.

ABSTRACT

The court rental process at GOR Ewangga Kuningan is still carried out manually, requiring renters to come directly to the venue to make reservations. This condition creates several issues, including limited access to information, difficulties in obtaining accurate and up-to-date schedules, and suboptimal service quality when the number of prospective renters increases. Moreover, the high demand often leads to overlapping schedules due to the absence of a real-time and transparent scheduling management system.

This study aims to design and develop a web-based court booking application using the Laravel framework as a solution to these problems. The system provides features such as user registration, online booking, real-time schedule checking, digital payment, and administrative data management. The development process adopts the Waterfall model, consisting of requirements analysis, system design, implementation, testing, and maintenance.

The results of this research are expected to improve the efficiency of the booking process, enhance transparency in court availability information, and support GOR administrators in optimizing service quality. The implementation of this system is anticipated to contribute to better service management at GOR Ewangga Kuningan in line with the advancement of digital technology.

Keywords: Online Court Booking, GOR Ewangga Kuningan, Laravel Framework, Web-Based Application, Waterfall Method.

KATA PENGANTAR

Ucapan dan rasa syukur penulis layangkan kepada hadirat Ilahi Robbi, yang telah berkenan menguatkan penulis untuk membuat Laporan Tugas Akhir dengan judul “Rancang Bangun Aplikasi Pemesanan Online Lapangan Berbasis Web Menggunakan Framework Laravel (Studi Kasus : GOR Ewangga Kuningan)”. Adapun penulisan laporan ini bertujuan untuk memenuhi salah satu syarat kelulusan Program Strata 1, di Program Studi Teknik Informatika Universitas Pasundan.

Penulis menyadari laporan ini dapat terwujud berkat bantuan dan dorongan dari berbagai pihak. Maka pada kesempatan ini penulis sampaikan terima kasih yang sebesar-besarnya atas segala bantuan yang penulis terima baik secara moril maupun materil, sehingga penulis dapat menyelesaikan laporan ini kepada :

1. Pembimbing yang terhormat, Bapak Anggoro Ari Nurcahyo, S.T., M.Kom. yang telah membimbing dalam melakukan penelitian dan memberikan motivasi serta dorongan hingga penelitian terselesaikan dengan lancar.
2. Penguji yang terhormat, Bapak Erik, S.T., M.Kom. dan Bapak Wanda Gusdya Purnama, S.T., M.T., yang telah memberikan arahan, masukan, serta kritik yang membangun sehingga laporan tugas akhir ini dapat terselesaikan dengan baik.
3. Kepada Orang Tua tersayang, dan keluarga yang selalu memberikan motivasi serta do'anya dalam pembuatan tugas akhir ini.
4. GOR Ewangga Kabupaten Kuningan, atas kerja sama dan data yang diberikan sebagai bagian dari studi kasus ini.
5. Seluruh civitas akademika Teknik Informatika di Universitas Pasundan Bandung, yang telah memberikan bekal ilmu selama penulis menimba ilmu.

Penulis menyadari bahwa tugas akhir/skripsi ini masih jauh dari sempurna. Oleh karena itu, penulis dengan senang hati menerima kritik dan saran yang membangun demi kesempurnaan tugas akhir/skripsi ini di masa mendatang.

Akhir kata, semoga tugas akhir/skripsi ini dapat memberikan manfaat, baik bagi penulis, civitas akademika, maupun perkembangan teknologi informasi, khususnya dalam mendukung kemajuan pengembangan website di Indonesia.

Bandung, 23 Juli 2026

Mochammad Nizar Albaehaqi

DAFTAR ISI

LEMBAR PENGESAHAN LAPORAN TUGAS AKHIR	i
LEMBAR PERNYATAAN KEASLIAN TUGAS AKHIR.....	ii
ABSTRAK	iii
ABSTRACT	iv
KATA PENGANTAR	v
DAFTAR ISI	vi
DAFTAR GAMBAR.....	viii
DAFTAR TABEL.....	xi
BAB 1 PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Identifikasi Masalah	2
1.3 Tujuan Tugas Akhir	2
1.4 Lingkup Tugas Akhir	2
1.5 Metodologi Tugas Akhir	3
1.5.1 Identifikasi Masalah	3
1.5.2 Pengumpulan Data.....	4
1.5.3 Pembangunan Perangkat Lunak.....	4
1.5.4 Kesimpulan dan Saran	4
1.6 Sistematika Penulisan Tugas Akhir.....	4
BAB 2 LANDASAN TEORI DAN PENELITIAN TERDAHULU	1
2.1 Teori Pendukung.....	1
2.1.1 Rancang Bangun.....	1
2.1.2 Aplikasi.....	1
2.1.3 Website	1
2.1.4 Framework Laravel.....	2
2.1.5 MySQL.....	2
2.1.6 Waterfall	3
2.2 Penelitian Terdahulu	4

2.3 Standar dan Kakas	5
BAB 3 SKEMA PENELITIAN	1
3.1 Alur Penyelesaian Tugas Akhir.....	1
3.2 Perumusan Masalah.....	4
3.2.1 Analisis Sebab Akibat.....	4
3.2.2 Analisis Solusi Masalah.....	5
3.3 Profil Penelitian.....	6
BAB 4 ANALISIS DAN PERANCANGAN	1
4.1 Analisis Sistem Yang Berjalan.....	1
4.1.1 Analisis Proses Bisnis.....	1
4.2 Analisis Kebutuhan	2
4.2.1. Analisis Kebutuhan Dasar	2
4.2.2 Pemodelan Berbasis Skenario.....	4
4.2.3 Pemodelan Berbasis Kelas.....	23
BAB 5 IMPLEMENTASI DAN PENGUJIAN	30
5.1 Coding.....	30
5.1.1. Struktur Folder	31
5.1.2. Migration.....	31
5.1.3 Seeder.....	34
5.1.4 Model	36
5.1.5 View	43
5.1.6 Controller	70
5.2 Testing.....	82
BAB 6 KESIMPULAN DAN SARAN	84
6.1 Kesimpulan	84
6.2 Saran	84
DAFTAR PUSTAKA	85
LAMPIRAN.....	86

DAFTAR GAMBAR

Gambar 1. 1 Metodologi Tugas Akhir	3
Gambar 2. 1 Model Waterfall	3
Gambar 3. 1 Analisis Sebab Akibat	4
Gambar 3. 2 Analisis Solusi Masalah	5
Gambar 3. 3 Gedung Gor Ewangga Kuningan	6
Gambar 4. 1 Alur Aktivitas Pencatatan Data Pemesanan.....	1
Gambar 4. 2 Use Case Diagram	5
Gambar 4. 3 Skenario Use Case Mengelola Data Pemesanan	7
Gambar 4. 4 Skenario Use Case Mengelola Data Penyewa	8
Gambar 4. 5 Skenario Use Case Mengelola Data Penyewa	9
Gambar 4. 6 Skenario Use Case Mengelola Data Penyewa	9
Gambar 4. 7 Skenario Use Case Mengelola Data Lapangan	10
Gambar 4. 8 Skenario Use Case Mengelola Data Lapangan	10
Gambar 4. 9 Skenario Use Case Registrasi	11
Gambar 4. 10 Skenario Use Case Registrasi	11
Gambar 4. 11 Skenario Use Case Pemesanan.....	12
Gambar 4. 12 Skenario Use Case Informasi Pemesanan	13
Gambar 4. 13 Skenario Use Case Informasi Pemesanan	14
Gambar 4. 14 Skenario Use Case Pembayaran.....	15
Gambar 4. 15 Skenario Use Case Pembayaran.....	15
Gambar 4. 16 Aliran Aktivitas Mengelola Data Pemesanan	16
Gambar 4. 17 Aliran Aktivitas Mengelola Data Penyewa	17
Gambar 4. 18 Aliran Aktivitas Mengelola Data Lapangan	18
Gambar 4. 19 Aliran Aktivitas Registrasi	19
Gambar 4. 20 Aliran Aktivitas Pemesanan	20
Gambar 4. 21 Aliran Aktivitas Informasi Pemesanan	21
Gambar 4. 22 Aliran Aktivitas Pembayaran	22
Gambar 4. 23 Diagram Kelas	25
Gambar 4. 24 Diagram Sequence Use Case UC-01 Mengelola Data Pemesanan	26
Gambar 4. 25 Diagram Sequence Use Case UC-02 Mengelola Data Penyewa	26
Gambar 4. 26 Diagram Sequence Use Case UC-03 Mengelola Data Lapangan	27
Gambar 4. 27 Diagram Sequence Use Case UC-04 Registrasi	27
Gambar 4. 28 Diagram Sequence Use Case UC-05 Pemesanan	28
Gambar 4. 29 Diagram Sequence Use Case UC-06 Informasi Pemesanan.....	28

Gambar 4. 30 Diagram Sequence Use Case UC-07 Pembayaran	29
Gambar 5. 1 Migrasi users mendefinisikan tabel identitas pengguna.	32
Gambar 5. 2 Migrasi lapangans mendefinisikan tabel data lapangan olahraga.	32
Gambar 5. 3 Migrasi jadwal mendefinisikan tabel jam pemesanan.	33
Gambar 5. 4 Migrasi pemesanans mendefinisikan tabel pemesanan lapangan.	33
Gambar 5. 5 Migrasi pembayarans mendefinisikan tabel pembayaran pemesanan.	34
Gambar 5. 6 DatabaseSeeder mengisi data awal aplikasi.	34
Gambar 5. 7 JadwalSeeder mengisi tabel jadwal dengan jam otomatis.	35
Gambar 5. 8 UserSeeder mengisi tabel users dengan akun petugas dan penyewa.	35
Gambar 5. 9 Model Jadwal merepresentasikan tabel jadwal.	36
Gambar 5. 10 Model Lapangan merepresentasikan tabel lapangans.	36
Gambar 5. 11 Model Lapangan merepresentasikan tabel lapangans.	37
Gambar 5. 12 Model Lapangan merepresentasikan tabel lapangans.	37
Gambar 5. 13 Model Pembayaran merepresentasikan tabel pembayarans.	38
Gambar 5. 14 Model Pembayaran merepresentasikan tabel pembayarans.	39
Gambar 5. 15 Model Pemesanan merepresentasikan tabel pemesanans.	40
Gambar 5. 16 Model Pemesanan merepresentasikan tabel pemesanans.	41
Gambar 5. 17 Model Pemesanan merepresentasikan tabel pemesanans.	42
Gambar 5. 18 Model User merepresentasikan tabel users.	42
Gambar 5. 19 Halaman login untuk pengguna melalui username dan password.	43
Gambar 5. 20 Halaman login untuk pengguna melalui username dan password.	44
Gambar 5. 21 Halaman login untuk pengguna melalui username dan password.	44
Gambar 5. 22 Halaman register untuk pembuatan akun baru.	45
Gambar 5. 23 Halaman register untuk pembuatan akun baru.	45
Gambar 5. 24 Halaman register untuk pembuatan akun baru.	46
Gambar 5. 25 Halaman register untuk pembuatan akun baru.	46
Gambar 5. 26 Halaman register untuk pembuatan akun baru.	46
Gambar 5. 27 Halaman dashboard menampilkan halaman utama aplikasi.	47
Gambar 5. 28 Halaman dashboard menampilkan halaman utama aplikasi.	48
Gambar 5. 29 Halaman dashboard menampilkan halaman utama aplikasi.	48
Gambar 5. 30 Halaman dashboard menampilkan halaman utama aplikasi.	49
Gambar 5. 31 Halaman dashboard menampilkan halaman utama aplikasi.	50
Gambar 5. 32 Halaman jadwal menampilkan daftar pemesanan lapangan.	51
Gambar 5. 33 Halaman jadwal menampilkan daftar pemesanan lapangan.	52
Gambar 5. 34 Halaman jadwal menampilkan daftar pemesanan lapangan.	53
Gambar 5. 35 Halaman jadwal menampilkan daftar pemesanan lapangan.	53
Gambar 5. 36 Halaman lapangan menampilkan daftar lapangan olahraga.	54

Gambar 5. 37 Halaman lapangan menampilkan daftar lapangan olahraga.	55
Gambar 5. 38 Halaman lapangan menampilkan daftar lapangan olahraga.	56
Gambar 5. 39 Halaman lapangan menampilkan daftar lapangan olahraga.	57
Gambar 5. 40 Halaman lapangan menampilkan daftar lapangan olahraga.	57
Gambar 5. 41 Halaman laporan menampilkan daftar laporan pemesanan.	58
Gambar 5. 42 Halaman laporan menampilkan daftar laporan pemesanan.	59
Gambar 5. 43 Halaman laporan menampilkan daftar laporan pemesanan.	60
Gambar 5. 44 Halaman laporan menampilkan daftar laporan pemesanan.	61
Gambar 5. 45 Halaman pemesanan menampilkan pemesanan lapangan.	62
Gambar 5. 46 Halaman pemesanan menampilkan pemesanan lapangan.	63
Gambar 5. 47 Halaman pemesanan menampilkan pemesanan lapangan.	64
Gambar 5. 48 Halaman pemesanan menampilkan pemesanan lapangan.	65
Gambar 5. 49 Halaman pemesanan menampilkan pemesanan lapangan.	65
Gambar 5. 50 Halaman user menampilkan daftar akun pengguna beserta identitas.	66
Gambar 5. 51 Halaman user menampilkan daftar akun pengguna beserta identitas.	67
Gambar 5. 52 Halaman user menampilkan daftar akun pengguna beserta identitas.	68
Gambar 5. 53 Halaman user menampilkan daftar akun pengguna beserta identitas.	69
Gambar 5. 54 Halaman user menampilkan daftar akun pengguna beserta identitas.	69
Gambar 5. 55 AuthController mengatur autentikasi pengguna dalam aplikasi.	70
Gambar 5. 56 AuthController mengatur autentikasi pengguna dalam aplikasi.	71
Gambar 5. 57 Controller berisi kelas dasar aplikasi.	71
Gambar 5. 58 JadwalController mengatur pengelolaan jadwal pemesanan.	72
Gambar 5. 59 LapanganController mengatur pengelolaan data lapangan.	73
Gambar 5. 60 LapanganController mengatur pengelolaan data lapangan.	74
Gambar 5. 61 LapanganController mengatur pengelolaan data lapangan.	75
Gambar 5. 62 LapanganController mengatur pengelolaan data lapangan.	75
Gambar 5. 63 LaporanController mengatur pencetakan laporan pemesanan.	76
Gambar 5. 64 LaporanController mengatur pencetakan laporan pemesanan.	77
Gambar 5. 65 PemesananController mengatur proses pemesanan lapangan.	78
Gambar 5. 66 PemesananController mengatur proses pemesanan lapangan.	79
Gambar 5. 67 PemesananController mengatur proses pemesanan lapangan.	80
Gambar 5. 68 PemesananController mengatur proses pemesanan lapangan.	81
Gambar 5. 69 UserController mengatur pengelolaan data pengguna.	81

DAFTAR TABEL

Tabel 2. 1 Penelitian Terdahulu	4
Tabel 3. 1 Alur Penyelesaian Tugas Akhir	1
Tabel 3. 2 Analisis Sebab Akibat	4
Tabel 3. 3 Analisis Solusi Masalah	5
Tabel 4. 1 Pencatatan Data Pemesanan.....	1
Tabel 4. 2 Analisis Pengguna Sistem.....	2
Tabel 4. 3 Analisis Kebutuhan Fungsional	2
Tabel 4. 4 Analisis Kebutuhan Non Fungsional.....	3
Tabel 4. 5 Identifikasi User Requirement	4
Tabel 4. 6 Identifikasi Software Requirement	4
Tabel 4. 7 Use Case Diagram	5
Tabel 4. 8 Deskripsi Use Case	6
Tabel 4. 9 Deskripsi Aktor.....	6
Tabel 4. 10 Skenario Use Case Mengelola Data Pemesanan	7
Tabel 4. 11 Skenario Use Case Mengelola Data Penyewa	8
Tabel 4. 12 Skenario Use Case Mengelola Data Lapangan	10
Tabel 4. 13 Skenario Use Case Registrasi	11
Tabel 4. 14 Skenario Use Case Pemesanan	12
Tabel 4. 15 Skenario Use Case Informasi Pemesanan	13
Tabel 4. 16 Skenario Use Case Pembayaran.....	15
Tabel 4. 17 Aliran Aktivitas Mengelola Data Pemesanan.....	16
Tabel 4. 18 Aliran Aktivitas Mengelola Data Penyewa	17
Tabel 4. 19 Aliran Aktivitas Mengelola Data Lapangan	18
Tabel 4. 20 Aliran Aktivitas Registrasi	19
Tabel 4. 21 Aliran Aktivitas Pemesanan	20
Tabel 4. 22 Aliran Aktivitas Informasi Pemesanan.....	21
Tabel 4. 23 Aliran Aktivitas Pembayaran	22
Tabel 4. 24 Objek Data.....	23
Tabel 4. 25 Identifikasi Atribut Kelas.....	23
Tabel 4. 26 Diagram Kelas	25
Tabel 4. 27 Diagram Sequence Mengelola Data Pemesanan	26
Tabel 4. 28 Diagram Sequence Mengelola Data Penyewa.....	26
Tabel 4. 29 Diagram Sequence Mengelola Data Lapangan	27

Tabel 4. 30 Diagram Sequence Registrasi	27
Tabel 4. 31 Diagram Sequence Pemesanan	28
Tabel 4. 32 Diagram Sequence Informasi Pemesanan	28
Tabel 4. 33 Diagram Sequence Pembayaran.....	29
Tabel 5. 1 Spesifikasi Perangkat Keras.....	30
Tabel 5. 2 Perangkat lunak yang digunakan dalam pembangunan	30
Tabel 5. 3 Struktur Folder.....	31
Tabel 5. 4 Alpha testing menggunakan black box testing	82
Tabel 5. 5 Hasil Alpha testing.....	83

BAB 1

PENDAHULUAN

Bab ini memberikan gambaran umum pelaksanaan tugas akhir, meliputi latar belakang, identifikasi masalah, tujuan, lingkup, metodologi, serta sistematika penulisan.

1.1 Latar Belakang

Gelanggang Olahraga (GOR) Ewangga Kuningan merupakan fasilitas olahraga di Kabupaten Kuningan dengan dua jenis lapangan, futsal dan basket, beroperasi pukul 07.00-22.00. Berdasarkan observasi, jumlah pemesanan mencapai 20-40 orang per hari, terutama akhir pekan dan musim liburan. Saat ini, kebutuhan akan kemudahan akses dan efisiensi pengelolaan semakin meningkat. Manajemen fasilitas olahraga ialah proses memanfaatkan sarana dan prasarana dengan efektif dan efisien melalui kerjasama. Manajemen fasilitas olahraga memiliki tujuan yaitu untuk mengatur, mengawasi, pemeliharaan dan pengoperasian keuangan, fasilitas dan ramah lingkungan [RAH19]. Terdapat empat fungsi dari manajemen yang dikenal dengan sebutan “POAC” yaitu: planning (perencanaan), organizing (pengorganisasian), actuating (pengarahan) dan controlling (pengawasan) [RAH19].

Namun, pengelolaan lapangan GOR Ewangga saat ini belum sepenuhnya menerapkan prinsip manajemen tersebut. Penyewaan masih dilakukan dengan datang langsung ke lokasi untuk melakukan pemesanan dengan biaya Rp100.000 per jam. Proses ini memakan waktu, rawan kesalahan, sulit memantau data, dan kurang transparan. Pengelola juga mengalami kendala dalam melayani jumlah penyewa yang meningkat karena tidak adanya sistem terkini untuk menampilkan ketersediaan lapangan. Dalam era digital yang terus berkembang dengan pesat, teknologi informasi menjadi salah satu bagian terpenting dalam kehidupan sehari-hari. Hal ini bisa terjadi dikarenakan teknologi yang ada akan terus berkembang dengan cepat dan memberikan dampak yang sangat signifikan dalam berbagai aspek di kehidupan manusia [WIR19]. Melalui aplikasi pemesanan online GOR Ewangga Kuningan, penyewa dapat memeriksa ketersediaan lapangan, melakukan reservasi, dan membayar secara online. Sementara itu, pengelola GOR dapat lebih mudah mengatur jadwal, memantau pendapatan, dan meningkatkan layanan kepada pelanggan.

Untuk mengatasi kekurangan tersebut, penelitian ini mengembangkan sistem pemesanan online lapangan menggunakan laravel. Laravel adalah framework PHP bersifat open source dan berisi banyak modul dasar untuk mengoptimalkan kinerja PHP dalam pengembangan aplikasi web sedangkan PHP merupakan bahasa pemrograman yang dinamis. Laravel bertindak untuk membuat web lebih simpel, lebih aman, dan lebih cepat. [PUT23]. Penelitian ini berjudul “Rancang Bangun Aplikasi Pemesanan Online Lapangan Berbasis Web Menggunakan Framework Laravel (Studi Kasus: GOR Ewangga Kuningan)” dengan tujuan menciptakan website pemesanan yang mudah, efektif, efisien, serta diuji untuk memastikan keberhasilan implementasi di GOR Ewangga Kuningan.

1.2 Identifikasi Masalah

Berdasarkan hasil observasi dan analisis, terdapat beberapa permasalahan utama dalam sistem penyewaan lapangan GOR Ewangga Kuningan, yaitu:

1. Proses penyewaan masih dilakukan secara langsung di lokasi, sehingga memakan waktu dan menyulitkan penyewa yang memiliki keterbatasan jarak atau waktu.
2. Calon penyewa kesulitan memperoleh informasi ketersediaan lapangan secara jelas dan mudah diakses.
3. Pengelola GOR menghadapi kendala dalam melayani dan mengelola pemesanan, terutama saat jumlah penyewa meningkat.
4. Tingginya jumlah pemesanan sering menimbulkan jadwal tumpang tindih karena belum tersedia mekanisme pengelolaan jadwal yang transparan.

1.3 Tujuan Tugas Akhir

Tujuan dari penelitian ini dirumuskan untuk menjawab permasalahan yang ada dan menghasilkan solusi berupa aplikasi web pemesanan online lapangan. Adapun tujuan yang ingin dicapai adalah:

1. Menghasilkan aplikasi web pemesanan online lapangan GOR yang mempercepat proses pemesanan dan mudah digunakan.
2. Memberikan informasi jadwal lapangan yang jelas dan mudah diakses oleh calon penyewa.
3. Mendukung proses pengelolaan jadwal dan pencatatan pemesanan secara terstruktur.
4. Meningkatkan transparansi layanan melalui penyajian informasi ketersediaan lapangan yang terbuka dan mudah dipahami oleh penyewa.

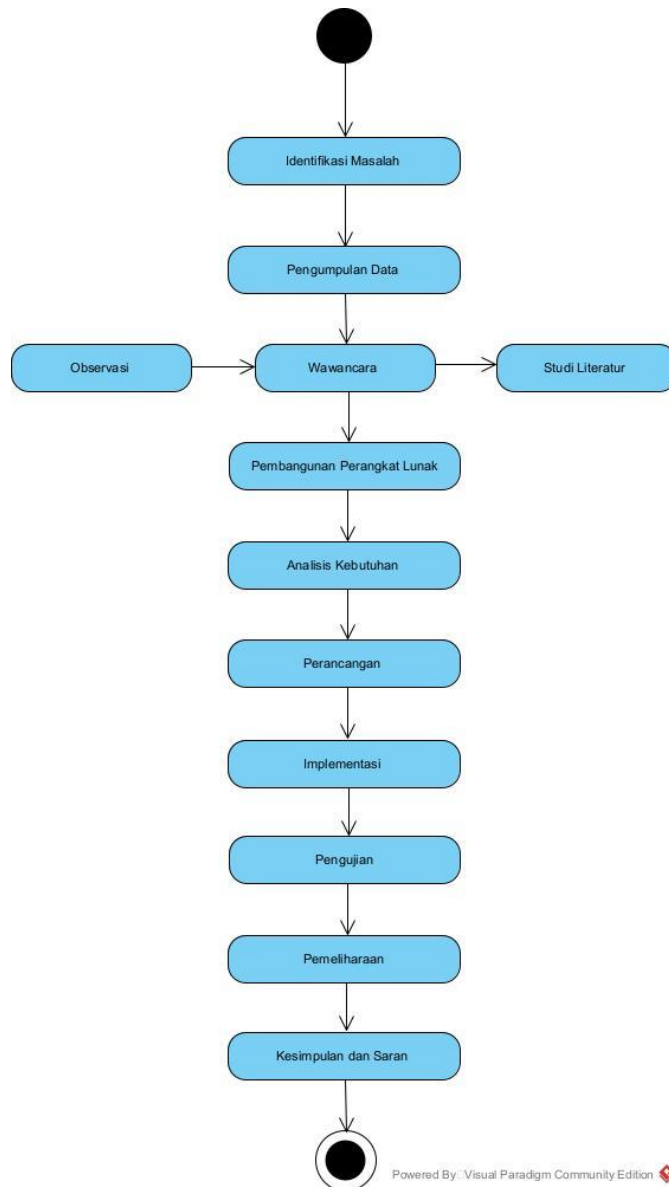
1.4 Lingkup Tugas Akhir

Agar penelitian tugas akhir ini dapat fokus pada tujuan yang ingin dicapai, maka lingkup dari tugas akhir ini adalah sebagai berikut :

1. Bahasa pemrograman yang digunakan dalam pembuatan aplikasi adalah PHP dan JavaScript dengan menggunakan Framework laravel.
2. Metode yang digunakan dalam membangun web ini yaitu metode waterfall.
3. Database yang digunakan yaitu menggunakan Mysql.
4. Aplikasi akan menyediakan fitur dan layanan terkait pemesanan lapangan secara terkini meliputi data penyewa lapangan dan informasi lapangan.
5. Aplikasi juga akan menyediakan fitur dan layanan terkait sistem pembayaran penyewaan lapangan dengan mengupload bukti pembayaran.

1.5 Metodologi Tugas Akhir

Metodologi tugas akhir yang digunakan dalam penelitian ini ditunjukkan pada Gambar 1.1, yang menggambarkan tahapan utama mulai dari identifikasi masalah hingga kesimpulan dan saran.



Gambar 1. 1 Metodologi Tugas Akhir

Metodologi tugas akhir yang digunakan dalam penelitian ini ditunjukkan pada Gambar 1.1. Diagram tersebut menggambarkan tahapan utama yang meliputi perancangan, pengujian, pemeliharaan, serta kesimpulan dan saran, yang menjadi dasar alur penelitian.

1.5.1 Identifikasi Masalah

Tahap identifikasi masalah merupakan langkah awal dalam suatu penelitian, di mana pada tahap ini dilakukan pengumpulan dan pengenalan permasalahan untuk memahami situasi yang sedang dihadapi, merumuskan permasalahan yang akan menjadi fokus penelitian, serta menganalisis kemungkinan solusi yang dapat diterapkan untuk mengatasinya.

1.5.2 Pengumpulan Data

Pengumpulan data dalam penelitian ini dilakukan melalui observasi, wawancara, dan studi literatur untuk memperoleh informasi akurat tentang kebutuhan sistem. Observasi langsung di GOR Ewangga Kuningan membantu memahami alur pemesanan manual, sementara wawancara dengan petugas dan calon penyewa mengidentifikasi kebutuhan fungsional seperti registrasi, pemesanan secara terkini, dan sistem pembayaran. Studi literatur digunakan untuk menganalisis framework Laravel, metode pengembangan sistem, dan referensi terkait. Data yang terkumpul dianalisis secara kualitatif untuk merancang solusi yang efisien, menjawab permasalahan nyata dalam proses pemesanan lapangan.

1.5.3 Pembangunan Perangkat Lunak

Metode Waterfall merupakan pendekatan pengembangan perangkat lunak yang bersifat linier dan sistematis, terdiri dari lima tahapan berurutan, analisis kebutuhan untuk mengidentifikasi spesifikasi sistem melalui observasi dan wawancara, perancangan sistem meliputi arsitektur, database, dan antarmuka pengguna, implementasi berupa pengkodean menggunakan framework Laravel dan MySQL, pengujian fungsionalitas sistem serta pemeliharaan untuk perbaikan bug dan penyesuaian kebutuhan. Metode ini dipilih dalam penelitian ini karena memberikan struktur jelas dengan dokumentasi terperinci, cocok untuk proyek dengan kebutuhan yang stabil seperti sistem pemesanan lapangan berbasis web untuk GOR Ewangga Kuningan. Keunggulan utamanya terletak pada kemudahan pengelolaan proyek dan kemampuan menghasilkan sistem yang sesuai dengan kebutuhan awal yang telah didefinisikan secara komprehensif.

1.5.4 Kesimpulan dan Saran

Pada tahap ini, dilakukan penarikan kesimpulan berdasarkan hasil dari proses penelitian dan pengembangan website yang telah dilaksanakan. Kesimpulan memuat penilaian terhadap efektivitas sistem dalam menyelesaikan permasalahan yang telah diidentifikasi sebelumnya. Di samping itu, disampaikan pula beberapa rekomendasi yang dapat dijadikan acuan untuk pengembangan lebih lanjut dalam penelitian atau implementasi di masa mendatang.

1.6 Sistematika Penulisan Tugas Akhir

Berikut dibawah ini merupakan sistematika penulisan yang akan digunakan dalam penyusunan tugas akhir ini :

BAB I : PENDAHULUAN

Pada bab ini akan berisikan tentang latar belakang, identifikasi masalah, tujuan tugas akhir, lingkup tugas akhir, dan sistematika penulisan tugas akhir.

BAB II : LANDASAN TEORI DAN PENELITIAN TERDAHULU

Pada bab ini akan berisikan tentang teori-teori dan konsep serta penelitian terdahulu yang dapat berhubungan dengan penelitian ini sehingga bisa membantu dalam menyelesaikan masalah yang ada.

BAB III : SKEMA PENELITIAN

Pada bab ini akan dibahas tentang skema penelitian yang akan dilakukan seperti rancangan penelitian, perumusan masalah, hingga kerangka berfikir teoritis.

BAB IV : ANALISIS DAN PERANCANGAN

Pada bab ini memaparkan populasi dan sampel, variable penelitian, jenis dan sumber data, metode analisis data, dan tahapan penelitian.

BAB V : IMPLEMENTASI DAN PENGUJIAN

Pada bab ini akan dibahas mengenai hasil dari penelitian kemudian dibuat pembahasannya.

BAB VI : KESIMPULAN DAN SARAN

Pada bab ini akan dibahas mengenai kesimpulan yang diperoleh dari hasil penelitian dan Analisa data yang telah dilakukan serta saran-saran yang dapat diterapkan dari hasil pengolahan data yang dapat menjadi masukan yang berguna kedepannya.

BAB 2

LANDASAN TEORI DAN PENELITIAN TERDAHULU

Bab ini menjelaskan kerangka pikir penulisan tugas akhir yang mencakup definisi, teori, dan konsep sebagai landasan penelitian, serta hasil-hasil kajian dari buku teks dan jurnal terkait.

2.1 Teori Pendukung

Sub bab ini berisi sejumlah teori atau konsep yang relevan digunakan dalam penelitian atau tugas akhir.

2.1.1 Rancang Bangun

“Rancang bangun merupakan membuat atau menciptakan suatu sistem maupun suatu aplikasi yang belum ada dalam suatu perusahaan atau instansi yang menjadi objek rancang bangun tersebut”. Rancang bangun merupakan suatu kegiatan dalam menerjemahkan hasil analisa menjadi bentuk satu perangkat lunak (software), kemudian membuat/ menciptakan suatu sistem atau sistem yang sudah ada di perbaiki supaya mendapat kinerja yang lebih maksimal [AHM19].

2.1.2 Aplikasi

Aplikasi adalah perangkat lunak komputer yang memanfaatkan kemampuan komputer langsung untuk melakukan suatu tugas yang diinginkan pengguna. Biasanya dibandingkan dengan peranti lunak sistem yang menyepadukan berbagai kemampuan komputer, tetapi tidak secara langsung menerapkan kemampuan tersebut untuk mengerjakan suatu tugas yang menguntungkan pengguna [AJI21].

2.1.3 Website

Website adalah platform digital yang dapat diakses melalui internet, yang berfungsi sebagai tempat di mana individu, bisnis, atau organisasi dapat hadir secara online. Website memungkinkan bisnis untuk membagikan informasi, menawarkan berbagai produk atau layanan, dan terhubung secara global. Seperti lokasi pada umumnya, website bekerja menggunakan sistem alamat yang akan memberitahukan lokasi tepatnya di internet sehingga Anda bisa mengaksesnya melalui web browser. Penjelasan mudahnya, alamat tersebut mirip seperti alamat yang Anda gunakan untuk menuju suatu tempat di aplikasi maps. Ketika Anda mengakses alamat situs, web browser akan menuju ke lokasi yang ditentukan dan mengambil file situs tersebut. Proses pengambilan informasi ini dilakukan oleh layanan web menggunakan teknologi seperti Hypertext Transfer Protocol (HTTP) dan File Transfer Protocol (FTP). Protokol ini pada dasarnya berfungsi untuk menentukan cara informasi dan file ditransmisikan melalui web. Browser kemudian menampilkan halaman tersebut di layar perangkat Anda menggunakan teknologi seperti HyperText Markup Language (HTML) dan Cascading Style Sheet (CSS). Teknologi ini menyusun dan menyajikan isi website kepada pengunjung dengan memberi tahu web browser tentang posisi penyajian informasi yang benar di layar. Meskipun membutuhkan bahasa pemrograman

atau coding agar website bisa ditampilkan di layar, untungnya Anda tidak perlu mempelajari bahasa pemrograman atau memiliki latar belakang di bidang IT untuk membuat website [CHA23].

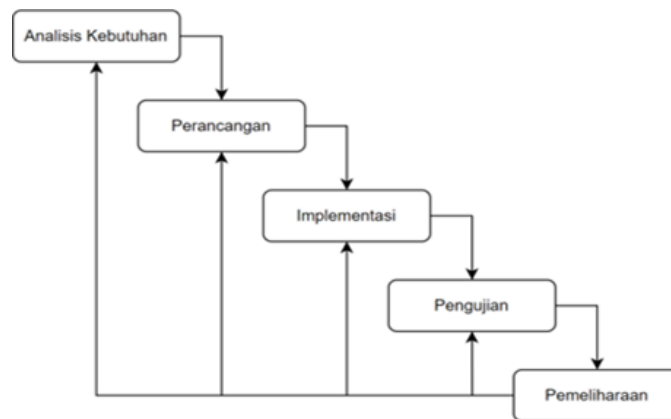
2.1.4 Framework Laravel

Laravel adalah kerangka kerja PHP yang membantu Anda mengembangkan situs web yang bebas digunakan. Dikembangkan oleh pengembang Amerika bernama Taylor Otwell, Laravel diluncurkan pada tahun 2011 dan telah mengalami pertumbuhan yang cukup eksponensial sejak saat itu. Pada tahun 2015, Laravel adalah framework yang paling banyak mendapatkan penghargaan dari Github. Sekarang framework ini menjadi salah satu yang paling populer di dunia, termasuk di Indonesia. Framework sendiri dapat diartikan sebagai sebuah framework yang berisi kumpulan kode-kode program yang dikumpulkan dan disusun rapi dalam folder yang terstruktur dengan baik agar mudah digunakan. Laravel adalah end-user centric, yang berarti berfokus pada kejelasan dan kesederhanaan baik dalam menulis maupun melihat, membangun aplikasi web yang berfungsi sebagaimana mestinya. Hal ini membuat pengembang dan perusahaan menggunakan framework ini untuk membangun segala sesuatu mulai dari proyek kecil hingga ukuran perusahaan tingkat lanjut [SHO22].

2.1.5 MySQL

MySQL merupakan Database Management System (DBMS) tools open source yang mendukung multiuser, multithreaded, populer, dan free. Sedangkan SQL adalah bahasa permintaan database tertentu yang digunakan untuk dapat membuat dan memanipulasi data di dalam database (Noviana, 2022). MySQL adalah database server yang gratis dengan lisensi GNU General Public License (GPL) sehingga dapat Anda pakai untuk keperluan pribadi atau komersil tanpa harus membayar lisensi yang ada. Seperti yang sudah disinggung di atas, MySQL masuk ke dalam jenis RDBMS (Relational Database Management System). Maka dari itu, istilah semacam baris, kolom, tabel, dipakai pada aplikasi database ini. Contohnya di dalam MySQL sebuah database terdapat satu atau beberapa tabel. SQL sendiri merupakan suatu bahasa yang dipakai di dalam pengambilan data pada relational database atau database yang terstruktur. Jadi MySQL adalah database management system yang menggunakan bahasa SQL sebagai bahasa penghubung antara perangkat lunak aplikasi dengan database server [NAT24].

2.1.6 Waterfall



Gambar 2. 1 Model Waterfall

Model Waterfall merupakan salah satu metodologi pengembangan perangkat lunak yang bersifat sekuensial. Model ini terdiri dari beberapa tahap yang harus diselesaikan sebelum melanjutkan ke tahap berikutnya [SOM11]. Berikut adalah penjelasan untuk setiap tahap dalam metodologi Waterfall:

1. Analisis Kebutuhan

Analisis kebutuhan merupakan tahap awal dalam pengembangan sistem. Tahap ini melibatkan pengumpulan informasi secara intensif untuk menspesifikasikan kebutuhan perangkat lunak agar dapat dipahami seperti apa yang dibutuhkan oleh pengguna [PRE10]. Proses ini melibatkan pengumpulan data melalui observasi, wawancara, dan studi literatur mengenai beberapa kebutuhan yang diperlukan untuk sistem.

2. Perancangan

Tahap perancangan berfokus pada empat atribut berbeda dari program: struktur data, arsitektur perangkat lunak, representasi antarmuka, dan detail prosedural [SOM11]. Dalam tahap ini, dilakukan penggambaran secara detail struktur dan fungsi sistem yang akan dibuat, diawali dengan pembuatan UML yang meliputi use case diagram, activity diagram, dan class diagram.

3. Implementasi

Implementasi adalah proses menerjemahkan desain ke dalam kode program. Pada tahap ini realnya pembuatan kode program dimulai [PRE10]. Proses implementasi ini melibatkan penerapan secara aktif desain yang telah dibuat, mentransformasikan konsep dan struktur menjadi kode pemrograman yang dapat dieksekusi menggunakan framework Laravel untuk pengkodean dan MySQL untuk manajemen database.

4. Pengujian

Pengujian fokus pada logika internal perangkat lunak dan fungsi eksternal. Pengujian dilakukan untuk menemukan kesalahan dan memastikan bahwa input yang didefinisikan akan menghasilkan hasil aktual yang sesuai dengan hasil yang dibutuhkan [SOM11]. Proses ini mencakup pengecekan secara menyeluruh terhadap berbagai aspek fungsionalitas, keamanan, dan kinerja sistem, serta melibatkan responden untuk mengisi kuesioner.

5. Pemeliharaan

Pemeliharaan mencakup koreksi kesalahan yang tidak ditemukan pada tahap-tahap sebelumnya, perbaikan atas implementasi unit sistem dan pengembangan pelayanan sistem, sementara persyaratan-persyaratan baru ditambahkan [PRE10]. Tahap ini melibatkan pemantauan dan pembaruan sistem secara berkelanjutan sesuai dengan kebutuhan yang berkembang.

2.2 Penelitian Terdahulu

Penelitian terdahulu yang relevan dengan topik ini disajikan pada Tabel 2.1, yang memuat daftar penelitian, judul, serta metode yang digunakan sebagai acuan dan perbandingan dalam penelitian ini.

Tabel 2. 1 Penelitian Terdahulu

No	Penelitian	Judul Penelitian	Metode Penelitian
1	Dewantara, Y. A., Primajaya, A., & Nurkifli, E. H. (2024)	RANCANG BANGUN WEBSITE PENYEWAAN LAPANGAN MENGGUNAKAN FRAMEWORK LARAVEL (STUDI KASUS: GOR JAYA ABADI).	Untuk memahami penggunaan website
2	Raga Kumia Pratama, Faisal Piliang (2019)	RANCANG BANGUN APLIKASI PENYEWAAN LAPANGAN FUTSAL BERBASIS WEB	Untuk memahami pembuatan website
3	Rachmat Hidayat, Arief Satriansyah, M. Sinta Nurhayati (2022)	Penggunaan Metode Waterfall untuk Rancangan Bangun Aplikasi Penyewaan Lapangan Olahraga	Untuk memahami pembuatan website
4	Nindi Sekar Ayu, Erliyan Redy Susanto, Muhaqiqin (2022)	RANCANG BANGUN WEBSITE SISTEM INFORMASI MANAJEMEN SEWA LAPANGAN FUTSAL STUDI KASUS DAMAI FUTSAL LAMPUNG	Untuk memahami pembuatan website

No	Penelitian	Judul Penelitian	Metode Penelitian
5	Andre Ardiman, Gerson Feoh, I Made Dwi Ardiada (2024)	Implementasi Framework Laravel pada Sistem Penyewaan Lapangan Futsal Menggunakan Metode Waterfall di Global Sport Futsal	Untuk memahami pembuatan website
6	Anharudin, Haggi Abdau Nasser (2020)	RANCANG BANGUN APLIKASI RESERVASI KAMAR HOTEL BERBASIS WEB	Untuk memahami pembuatan website

2.3 Standar dan Kakas

Alat yang digunakan dalam membuat aplikasi ini dibagi menjadi 2 yaitu hardware pendukung dan software yang digunakan adalah Processor Intel Core 5, RAM 8GB, SSD 477TB. Software dan tools yang digunakan Windows 11 sebagai sistem operasi laptop, Visual Studio Code sebagai code editor.

BAB 3 SKEMA PENELITIAN

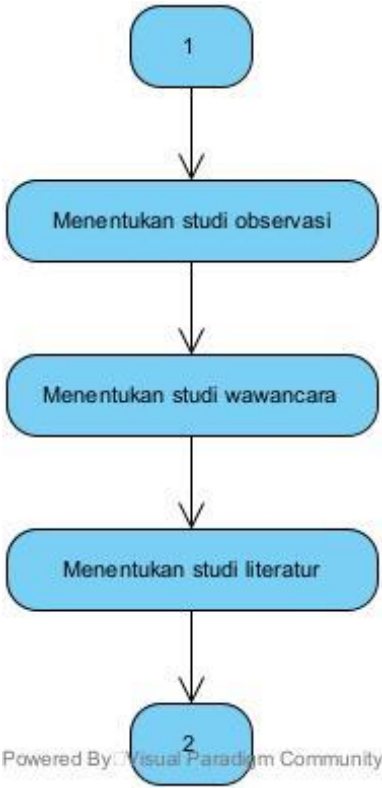
Bab ini berisi penjelasan alur penyelesaian tugas akhir, perumusan masalah, kerangka teoritis, serta profil tempat penelitian. Skema penelitian disajikan sebagai panduan awal bagi pembaca untuk memahami tujuan, ruang lingkup, dan landasan teoritis penelitian.

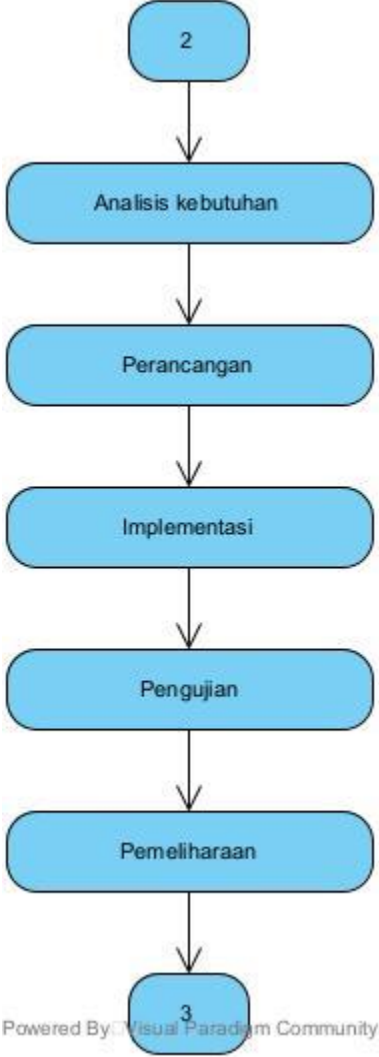
3.1 Alur Penyelesaian Tugas Akhir

Alur penyelesaian tugas akhir yang digunakan dalam penelitian ini disajikan pada Tabel 3.1, yang memuat tahapan penelitian beserta hasil yang diharapkan, langkah-langkah yang ditempuh, serta literatur dan referensi pendukung.

Tabel 3. 1 Alur Penyelesaian Tugas Akhir

Tahap dan Hasil	Langkah Penelitian	Literatur dan Referensi
Tahap 1: Menentukan topik tugas akhir, mengenali permasalahan yang akan diselesaikan, dan merumuskan tujuan serta Batasan lingkup tugas akhir. Hasil: Memilih topik yang tepat untuk tugas akhir, mengidentifikasi masalah yang perlu diselesaikan, serta merumuskan tujuan dan batasan lingkup penelitian merupakan langkah-langkah krusial dalam perencanaan dan pengembangan penelitian. Proses ini memerlukan analisis yang mendalam guna memastikan bahwa tugas akhir tidak hanya relevan, tetapi juga memberikan kontribusi yang bermakna dalam menyelesaikan permasalahan yang ada. Kontribusi:	 <pre> graph TD A([Menentukan topik tugas akhir]) --> B([Identifikasi masalah yang akan diselesaikan di tugas akhir]) B --> C([Menentukan tujuan dan lingkup tugas akhir]) C --> D([1]) </pre> Powered By: Visual Paradigm Community Edition	1. Latar Belakang 2. Identifikasi Masalah 3. Tujuan TA 4. Lingkup TA

Tahap dan Hasil	Langkah Penelitian	Literatur dan Referensi
Berguna dalam menentukan batasan tugas akhir, penerapan teori berperan sebagai alat penting untuk memahami konsep-konsep yang dibutuhkan dalam menyelesaikan tugas akhir.		
Tahap 2: Pengumpulan Data Hasil: Identifikasi kebutuhan untuk pengembangan website dapat dilakukan dengan mengumpulkan data melalui observasi, wawancara, dan studi literatur. Proses ini membantu memperoleh pemahaman yang komprehensif mengenai persyaratan yang harus dipenuhi dalam pembangunan website. Kontribusi: Berguna dalam mendukung proses membangun website pemesanan online lapangan berdasarkan kebutuhan pengguna. Pemahaman yang jelas tentang kebutuhan pengembangan website, sehingga dapat dirancang sesuai tujuan, meningkatkan fungsionalitas, dan mengurangi kesalahan dalam perancangan.	 <pre> graph TD 1((1)) --> A(Menentukan studi observasi) A --> B(Menentukan studi wawancara) B --> C(Menentukan studi literatur) C --> 2((2)) </pre>	1. Wawancara 2. Rancang Bangun[AHM19] 3. Website[CHA23]
Tahap 3 : Pembangunan Perangkat Lunak Hasil :		1. Aplikasi[AJI21] 2. Framework Laravel[SHO22] 3. MySQL[NAT24] 4. Waterfall[SOM11][PRE10]

Tahap dan Hasil	Langkah Penelitian	Literatur dan Referensi
Menghasilkan perangkat lunak yang mengikuti alur linier berurutan. Tahapan dimulai dari analisis kebutuhan untuk mendefinisikan spesifikasi perangkat lunak, dilanjutkan dengan perancangan sistem yang mencakup arsitektur, database, dan antarmuka. Hasil desain kemudian diimplementasikan melalui pengkodean hingga menghasilkan perangkat lunak awal. Selanjutnya, perangkat lunak diuji untuk memastikan semua fungsi berjalan sesuai kebutuhan. Setelah lulus pengujian, perangkat lunak diimplementasikan ke lingkungan produksi, dan tahap terakhir adalah pemeliharaan untuk memperbaiki bug atau menyesuaikan perubahan kebutuhan. Setiap tahap harus diselesaikan sepenuhnya sebelum melanjutkan ke tahap berikutnya. Kontribusi: Hasil dari proses ini adalah perangkat lunak yang terstruktur, sesuai spesifikasi, dan berkualitas, dengan pengembangan sistematis yang meminimalkan kesalahan serta mempermudah pemeliharaan.	 <pre> graph TD 2((2)) --> A[Analisis kebutuhan] A --> B[Perancangan] B --> C[Implementasi] C --> D[Pengujian] D --> E[Pemeliharaan] E --> 3((3)) </pre> Powered By: Visual Paradigm Community	

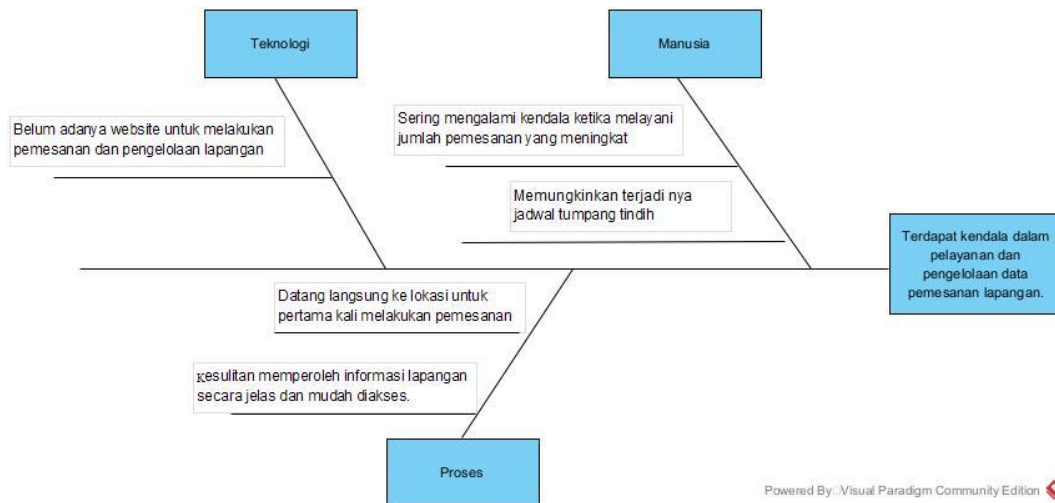
Tahap dan Hasil	Langkah Penelitian	Literatur dan Referensi
Tahap 4: Hasil : Kesimpulan dan Saran Kontribusi : Kesimpulan dan saran berdasarkan hasil apa yang telah dicapai dalam pengembangan perangkat lunak yang telah selesai.	 <pre> graph TD 3([3]) --> KS([Kesimpulan dan saran]) </pre>	

3.2 Perumusan Masalah

Bagian ini merumuskan permasalahan yang menjadi fokus penelitian, dengan tujuan memberikan gambaran jelas mengenai akar persoalan serta arah solusi yang akan dikembangkan.

3.2.1 Analisis Sebab Akibat

Analisis sebab-akibat ditunjukkan pada Gambar 3.1, yang memperlihatkan faktor utama seperti teknologi dan manusia sebagai penyebab permasalahan.



Gambar 3. 1 Analisis Sebab Akibat

Analisis sebab-akibat yang ditunjukkan pada Gambar 3.1 kemudian dijabarkan lebih rinci dalam Tabel 3.2, yang memuat daftar permasalahan beserta analisis penyebabnya.

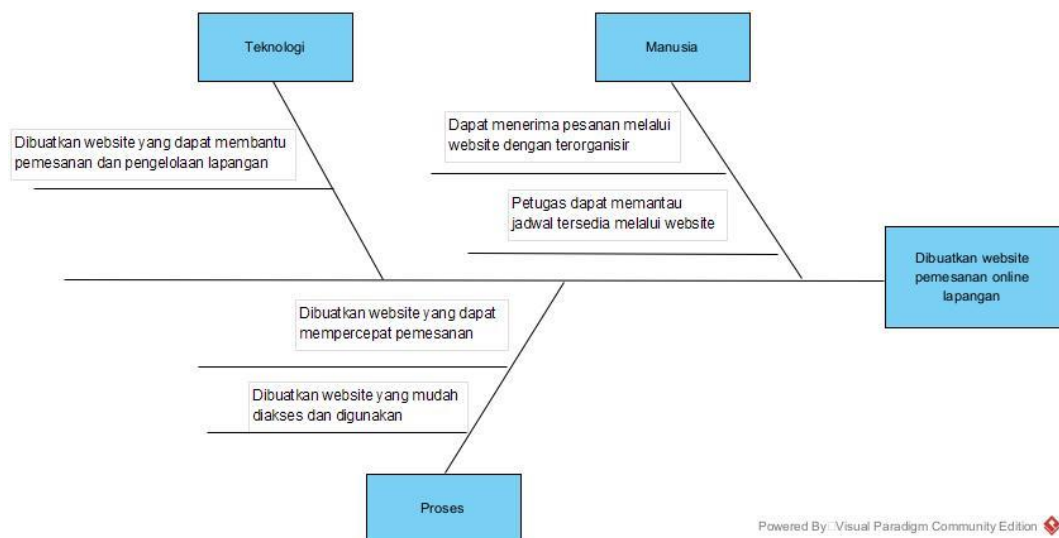
Tabel 3. 2 Analisis Sebab Akibat

No	Masalah	Analisis Sebab Akibat
1	Teknologi	Kendala utama dari sisi teknologi adalah belum tersedianya website khusus untuk pemesanan dan pengelolaan lapangan. Hal ini membuat seluruh

No	Masalah	Analisis Sebab Akibat
		proses masih dilakukan secara manual, sehingga menyulitkan pengguna dalam mengakses layanan dengan cepat dan praktis.
2	Proses	Dari sisi proses, penyewa harus datang langsung ke lokasi untuk melakukan pemesanan pertama kali. Cara ini memakan waktu dan mengurangi kenyamanan. Selain itu, informasi mengenai ketersediaan lapangan sulit diperoleh secara jelas dan mudah diakses, sehingga menimbulkan ketidakpastian bagi calon penyewa.
3	Manusia	Dari sisi manusia, petugas sering mengalami kesulitan ketika jumlah pemesanan meningkat. Kondisi ini menimbulkan beban kerja tambahan dan berpotensi menyebabkan jadwal tumpang tindih karena keterbatasan koordinasi dalam melayani banyak penyewa sekaligus.

3.2.2 Analisis Solusi Masalah

Analisis solusi masalah ditunjukkan pada Gambar 3.2, yang memperlihatkan alternatif penyelesaian melalui faktor teknologi dan manusia.



Gambar 3. 2 Analisis Solusi Masalah

Solusi atas permasalahan yang ditunjukkan pada Gambar 3.2 kemudian dijabarkan lebih detail dalam Tabel 3.3, yang memuat daftar masalah beserta analisis solusi dan keterkaitannya dengan sebab-akibat.

Tabel 3. 3 Analisis Solusi Masalah

No	Masalah	Analisis Solusi Masalah Sebab Akibat
1.	Teknologi	Solusi dari sisi teknologi adalah dengan dibuatnya website khusus yang dapat membantu proses pemesanan dan pengelolaan lapangan. Website ini berfungsi sebagai sarana digital yang menggantikan proses manual, sehingga pengguna dapat melakukan pemesanan secara lebih cepat dan terstruktur.

No	Masalah	Analisis Solusi Masalah Sebab Akibat
2.	Proses	Dari sisi proses, website yang dibangun mampu mempercepat alur pemesanan karena penyewa tidak perlu lagi datang langsung ke lokasi. Website juga dirancang agar mudah diakses dan digunakan, sehingga calon penyewa dapat memperoleh informasi ketersediaan lapangan dengan jelas dan melakukan pemesanan secara praktis.
3.	Manusia	Dari sisi manusia, petugas dapat menerima pesanan melalui website dengan lebih terorganisir. Selain itu, petugas juga dapat memantau jadwal ketersediaan lapangan secara langsung melalui sistem, sehingga mengurangi potensi terjadinya jadwal tumpang tindih dan memudahkan koordinasi pelayanan.

3.3 Profil Penelitian

GOR Ewangga, atau Gedung Olahraga Dipati Ewangga, adalah fasilitas olahraga yang terletak di Komplek GOR Ewangga, Kuningan, Kec. Kuningan, Kabupaten Kuningan, Jawa Barat. Gedung ini sering digunakan untuk berbagai kegiatan, termasuk pertandingan olahraga, acara budaya, pameran, dan kegiatan komunitas lainnya.



Gambar 3. 3 Gedung Gor Ewangga Kuningan

BAB 4 ANALISIS DAN PERANCANGAN

Bab ini menjelaskan analisis kebutuhan dan perancangan website, yang mencakup pendefinisian kebutuhan perangkat lunak, perancangan arsitektur, hingga antarmuka sistem yang akan dibangun.

4.1 Analisis Sistem Yang Berjalan

Proses penyewaan GOR masih dilakukan langsung di lokasi, sehingga memakan waktu, rawan kesalahan, dan kurang transparan. Pengelola juga kesulitan menghadapi lonjakan penyewa, membuat pelayanan tidak maksimal dan penjadwalan sering bermasalah karena belum ada sistem real-time.

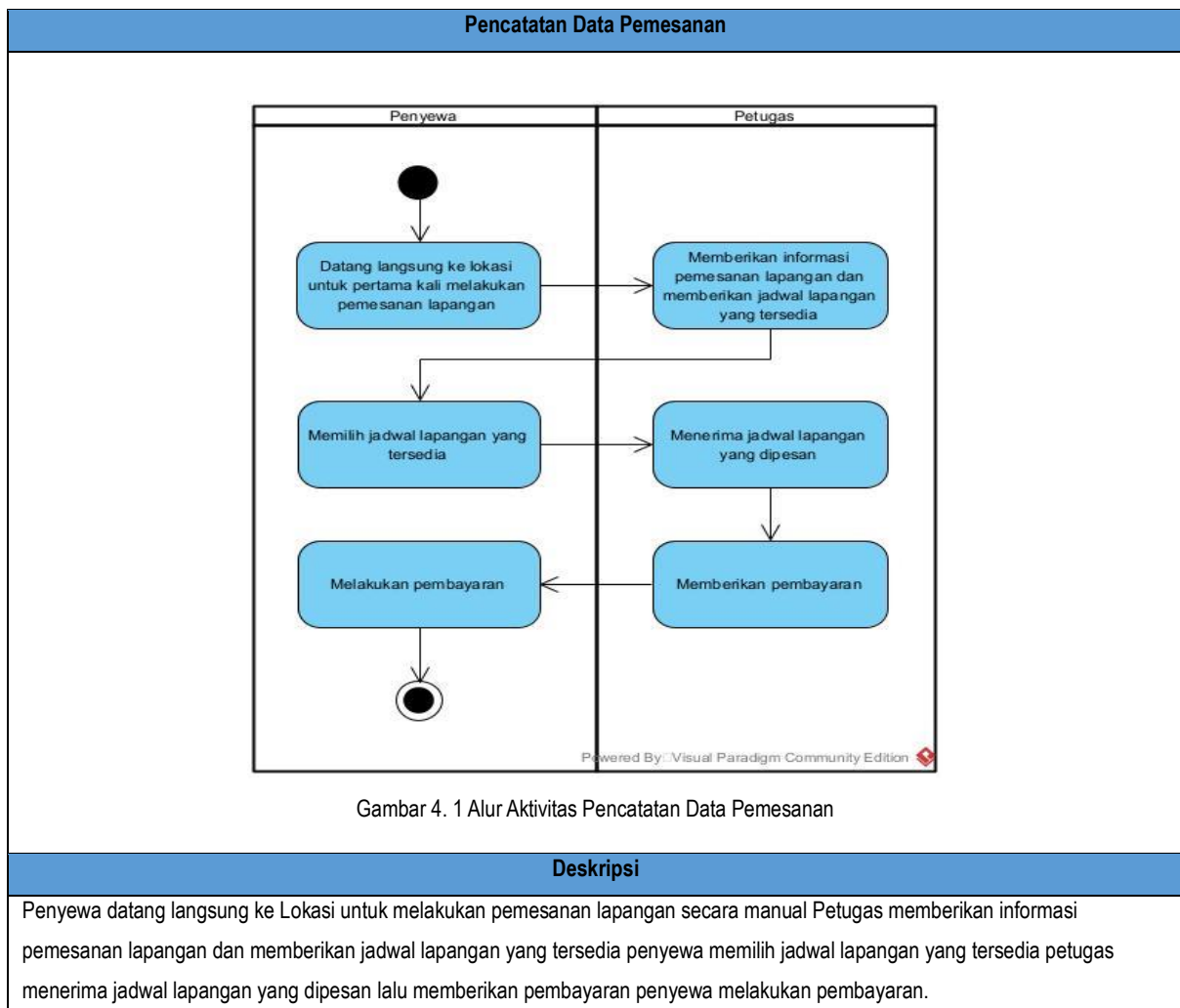
4.1.1 Analisis Proses Bisnis

Terdapat beberapa proses bisnis pada Gor Ewangga Kuningan diantaranya yaitu proses bisnis pencatatan data pemesanan lapangan.

4.1.1.1 Alur Aktivitas Pencatatan Data Pemesanan

Alur aktivitas pencatatan data pemesanan ditunjukkan pada Tabel 4.1, yang memuat tahapan pencatatan serta keterkaitannya dengan proses pengelolaan pemesanan.

Tabel 4. 1 Pencatatan Data Pemesanan



4.2 Analisis Kebutuhan

Analisis kebutuhan bertujuan untuk memahami kebutuhan dari sistem yang akan dibangun dan memahami spesifik, karakteristik, dan fitur yang akan dikembangkan pada perangkat lunak.

4.2.1. Analisis Kebutuhan Dasar

Pada tahap analisis kebutuhan maka dilakukan penentuan kebutuhan perangkat lunak yang akan dibangun serta hal-hal apa saja yang harus ada pada perangkat lunak.

4.2.1.1 Analisis Pengguna Sistem

Analisis pengguna sistem merupakan proses untuk mengidentifikasi pihak-pihak yang akan menggunakan perangkat lunak yang akan mempengaruhi fitur dari website yang akan dibangun. Daftar pengguna terdapat pada tabel 4.2.

Tabel 4. 2 Analisis Pengguna Sistem

No.	Pengguna	Deskripsi
1.	Penyewa	Melakukan registrasi, melakukan pengisian data diri yang sesuai, melihat informasi jadwal tersedia, melakukan pemesanan lapangan, dan melakukan pembayaran pemesanan yang dilakukan.
2.	Petugas	Merupakan orang yang mengelola pencatatan data pemesanan, Mengelola seluruh data yang terdapat pada system. Melakukan pengelolaan seluruh data pada sistem, mengelola akun, mengelola kegiatan pemesanan, pengecekan transaksi, dan pengelolaan ketersediaan lapangan.

4.2.1.2 Analisis Kebutuhan Fungsional

Analisis fungsional adalah tahap yang menghasilkan kebutuhan untuk perangkat lunak yang akan dikembangkan. Analisis fungsional dapat dilihat pada tabel 4.3.

Tabel 4. 3 Analisis Kebutuhan Fungsional

Kode	Kebutuhan Fungsional	Deskripsi
KF-001	Perangkat lunak harus dapat menyediakan akses ke halaman dashboard	bersifat umum yang didalamnya terdapat informasi terkait jadwal lapangan maupun informasi seputar tempat.
KF-002	Perangkat lunak harus dapat melakukan login.	dilakukan oleh seluruh pengguna agar dapat mengakses sistem sesuai dengan hak akses masing-masing user.
KF-003	Perangkat lunak harus dapat mengelola data pemesanan.	Perangkat lunak dapat mengelola data pemesanan, mulai dari menambah, melihat,

Kode	Kebutuhan Fungsional	Deskripsi
		mengedit, dan menghapus. dilakukan oleh penyewa dengan melakukan proses pemesanan dan melakukan transaksi.
KF-004	Perangkat lunak harus dapat melakukan registrasi.	dilakukan oleh penyewa baru untuk dapat melakukan pemesanan lapangan.
KF-005	Perangkat lunak harus dapat mengelola info pemesanan.	Bersifat umum yang didalam nya terdapat semua informasi pemesanan .
KF-006	Perangkat lunak harus dapat mengelola info lapangan.	bersifat umum yang didalam nya terdapat informasi lapangan, seperti harga, jenis lapang.
KF-007	Perangkat lunak harus dapat melakukan cara pemesanan.	bersifat umum yang didalamnya terdapat informasi cara pesan bagi pelanggan.
KF-008	Perangkat lunak harus dapat mengelola data penyewa.	Perangkat lunak dapat mengelola data pemesanan, mulai dari menambah, melihat, mengedit, dan menghapus. dilakukan oleh penyewa dengan melakukan proses pemesanan dan melakukan transaksi.

4.2.1.3 Analisis Kebutuhan Non Fungsional

Analisis Non Fungsional adalah tahap untuk menganalisis spesifikasi atau kebutuhan yang tidak berhubungan dengan fungsi pada perangkat lunak yang akan dikembangkan. Analisis Non Fungsional dapat dilihat pada tabel 4.4.

Tabel 4. 4 Analisis Kebutuhan Non Fungsional

Kode	Kebutuhan Non Fungsional	Deskripsi
KNF-001	Perangkat lunak harus mampu melindungi data pengguna dengan aman.	Perangkat lunak yang dibangun memiliki keamanan akses pengguna, untuk melindungi data pribadi dan informasi data pemesanan lapangan.
KNF-002	Perangkat lunak harus mampu menjaga ketersediaan layanan selama 24 jam sehari.	Perangkat lunak dapat diandalkan dan tersedia sepanjang waktu untuk memenuhi kebutuhan bisnis yang digunakan oleh penyewa dan petugas
KNF-003	Perangkat lunak harus dapat mudah dipahami.	Perangkat lunak memiliki tampilan yang mudah dipahami, mudah digunakan dan tidak membingungkan

4.2.1.4 Identifikasi User Requirement

Berdasarkan diagram aktivitas di atas, dapat dilihat dari pencatatan data pemesanan. Dengan demikian, didapatkan daftar user-requirement sebagai berikut:

Tabel 4. 5 Identifikasi User Requirement

Kode	User Requirement
UR-01	Penyewa dapat melakukan pemesanan lapangan melalui aplikasi..
UR-02	Penyewa dapat memilih jadwal pemesanan lapangan yang tersedia..
UR-03	Penyewa dapat melakukan pembayaran melalui aplikasi.

4.2.1.5 Identifikasi Software Requirement

Berdasarkan daftar user requirement tersebut, disusun daftar software requirement untuk memenuhi keinginan / requirement dari user tersebut:

Tabel 4. 6 Identifikasi Software Requirement

Kode	User Requirement	Kode	Software Requirement
UR-01	Penyewa dapat melakukan pemesanan lapangan melalui aplikasi.	SFR-01	Perangkat lunak harus dapat melakukan pemesanan lapangan yang tersedia.
UR-02	Penyewa dapat memilih jadwal pemesanan lapangan yang tersedia.	SFR-02	Perangkat lunak harus dapat menginput jadwal lapangan yang dipesan.
UR-03	Penyewa dapat melakukan pembayaran melalui aplikasi.	SFR-03	Perangkat lunak harus dapat melakukan transaksi pembayaran.

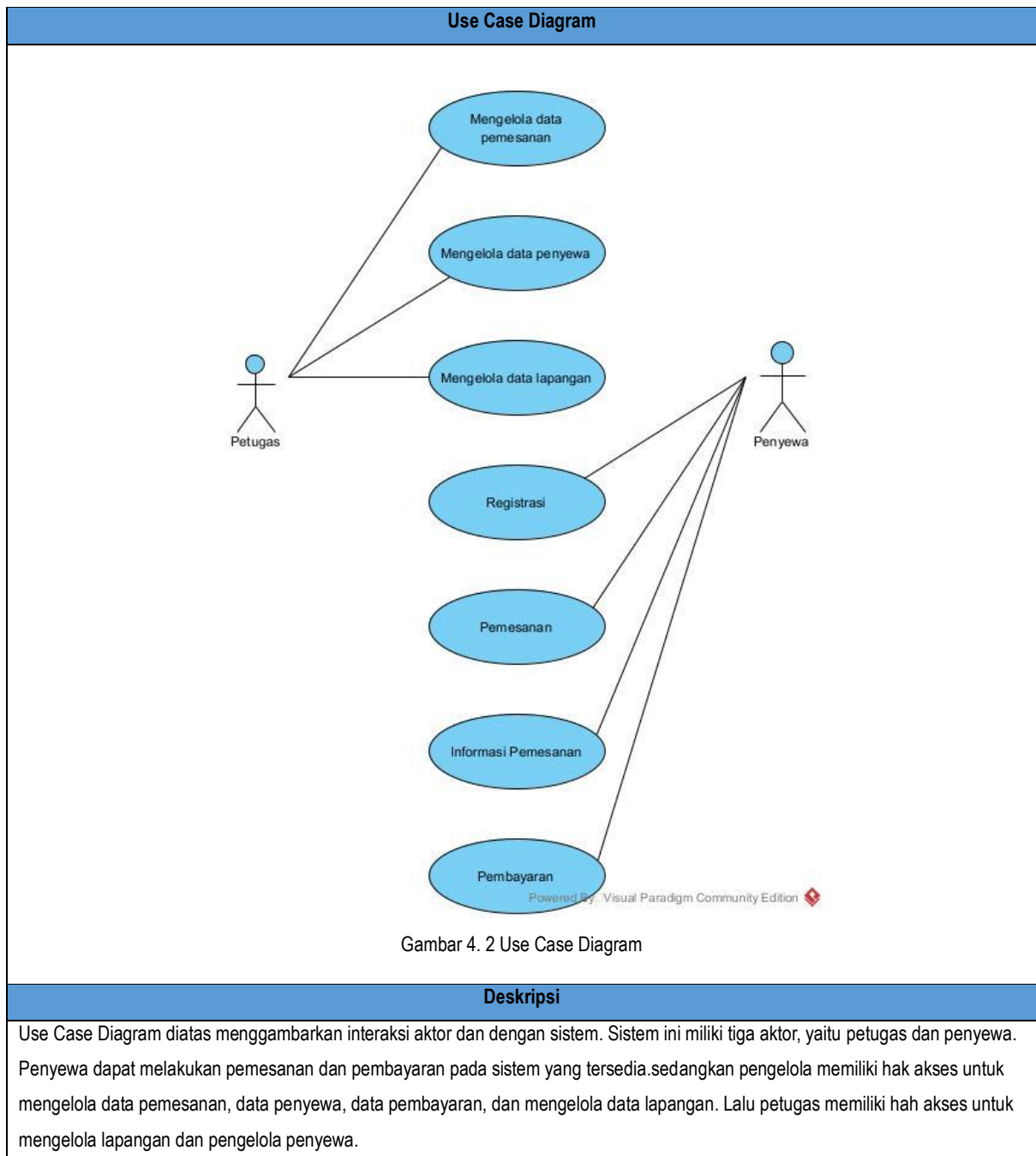
4.2.2 Pemodelan Berbasis Skenario

Pemodelan berbasis skenario digunakan untuk menggambarkan interaksi antara aktor dengan sistem. Pada pemodelan berbasis scenario terdapat beberapa hal diantaranya adalah diagram use case, skenario use case dan aliran aktivitas yang akan dijelaskan pada poin-poin berikut ini.

4.2.2.1 Diagram Use Case

Pemodelan berbasis scenario digunakan untuk merepresentasikan interaksi antar aktor dengan sistem yang akan dikembangkan. Pada umumnya dimulai dengan pembuatan use case, dan diagram aktivitas atau bisa disebut swimlane. Berikut merupakan beberapa use case pada perangkat lunak yang akan dibangun.

Tabel 4. 7 Use Case Diagram



4.2.2.2 Deskripsi Use Case

Deskripsi masing-masing use case ditunjukkan pada Tabel 4.8, yang memuat kode, nama, serta uraian singkat mengenai fungsi dan alur setiap use case.

Tabel 4. 8 Deskripsi Use Case

No.	Kode	Nama Use Case	Deskripsi
1.	UC-001	Mengelola Data Pemesanan	Use Case menampilkan informasi Perangkat lunak dapat mengelola data pemesanan, mulai dari menambah, melihat, mengedit, dan menghapus. dilakukan oleh pelanggan dengan melakukan proses pemesanan dan melakukan transaksi.
2.	UC-002	Mengelola Data Penyewa	Use Case mengelola data penyewaan mulai dari menambah, melihat, mengedit dan menghapus.
3.	UC-003	Mengelola Data Lapangan	Use Case mengelola data bersifat umum yang didalam nya terdapat informasi lapangan, seperti harga, jenis lapang.
4.	UC-004	Registrasi	Use Case dilakukan oleh user / penyewa baru untuk dapat melakukan pemesanan lapangan.
5.	UC-005	Pemesanan	Use Case Perangkat lunak dapat mengelola data pemesanan, mulai dari menambah, melihat, mengedit, dan menghapus. dilakukan oleh penyewa dengan melakukan proses pemesanan dan melakukan transaksi.
6.	UC-006	Informasi Pemesanan	Use Case Bersifat umum yang didalam nya terdapat semua informasi pemesanan .
7.	UC-007	Pembayaran	Use Case perangkat lunak dapat melakukan pembayaran.

4.2.2.3 Deskripsi Aktor

Deskripsi aktor yang terlibat dalam sistem ditunjukkan pada Tabel 4.9, yang memuat kode aktor, jenis pengguna, serta uraian peran masing-masing.

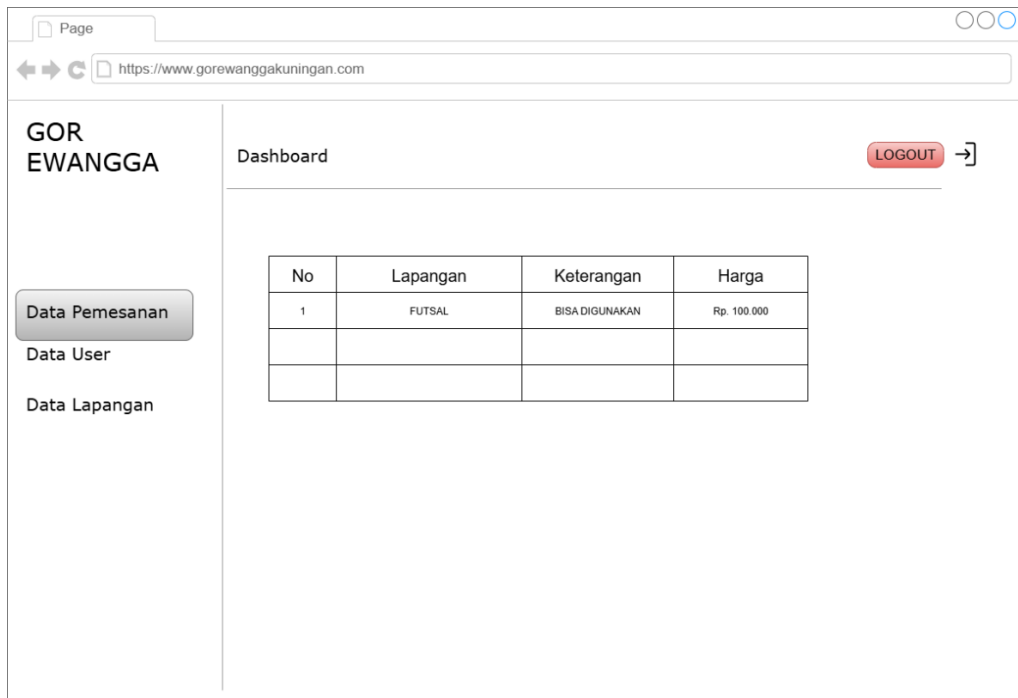
Tabel 4. 9 Deskripsi Aktor

No.	Kode Aktor	Pengguna Sistem	Deskripsi
1.	AC-001	Penyewa	Melakukan registrasi, melakukan pengisian data diri yang sesuai., melihat informasi jadwal tersedia, melakukan pemesanan lapangan, dan melakukan pembayaran pemesanan yang dilakukan.
2.	AC-002	Petugas	Merupakan orang yang mengelola pencatatan data pemesanan, Mengelola seluruh data yang terdapat pada system. Melakukan pengelolaan seluruh data pada sistem, mengelola akun, mengelola kegiatan pemesanan, pengecekan transaksi, pengelolaan ketersediaan lapangan.

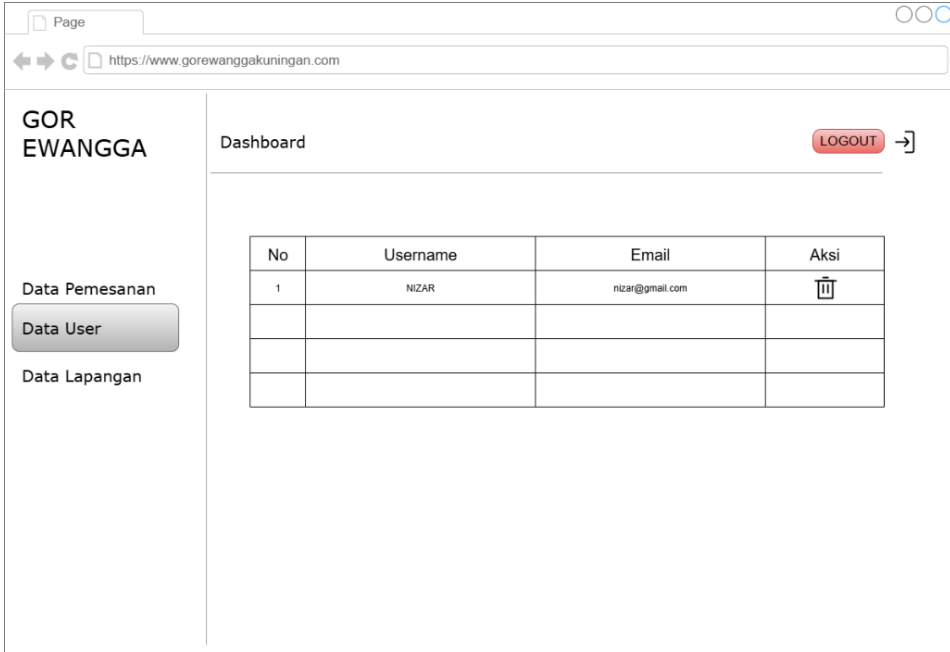
4.2.2.4 Skenario Use Case

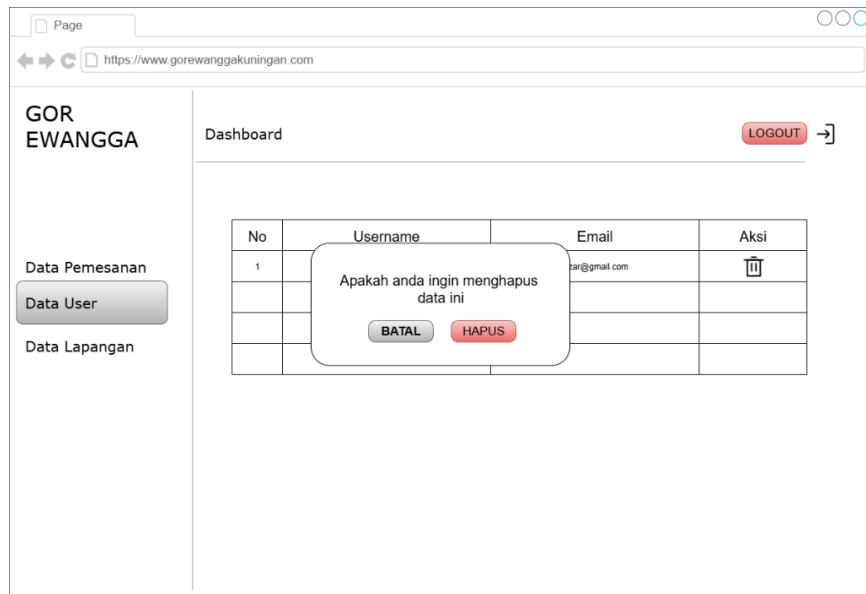
Skenario use case yang berkaitan dengan pengelolaan data pemesanan ditunjukkan pada Tabel 4.10, yang memuat identifikasi, alur interaksi, serta deskripsi proses yang terjadi dalam sistem.

Tabel 4. 10 Skenario Use Case Mengelola Data Pemesanan

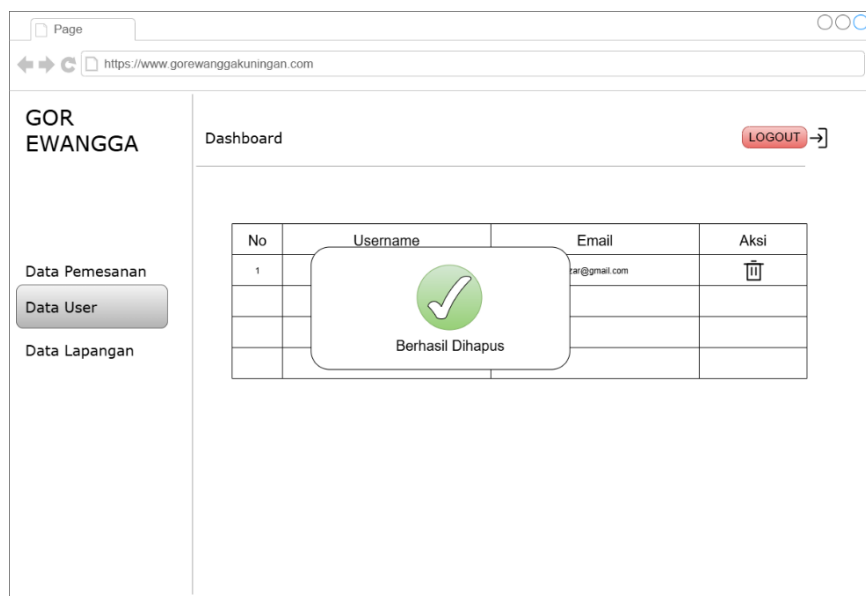
Identifikasi																	
Kode	UC-001																
Nama	Mengelola data pemesanan																
Tujuan	Petugas dapat mengelola dan meninformasikan pemesanan lapangan																
Deskripsi	Petugas dapat mengelola dan menginformasikan pemesanan lapangan mulai dari jenis lapangan, keterangan lapangan dan harga sewa lapangan.																
Aktor	Petugas																
Skenario Utama																	
Kondisi awal	Aktor berada di halaman dashboard admin																
Reaksi Aktor	Reaksi Sistem																
 The screenshot shows a web browser window with the URL https://www.gorewanggakuningan.com. The page title is 'GOR EWANGGA'. On the left, there is a sidebar with a 'Data Pemesanan' button and links for 'Data User' and 'Data Lapangan'. The main content area is titled 'Dashboard' and contains a table with the following data: <table><thead><tr><th>No</th><th>Lapangan</th><th>Keterangan</th><th>Harga</th></tr></thead><tbody><tr><td>1</td><td>FUTSAL</td><td>BISA DIGUNAKAN</td><td>Rp. 100.000</td></tr><tr><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td></tr></tbody></table> Below the screenshot is the caption: Gambar 4. 3 Skenario Use Case Mengelola Data Pemesanan		No	Lapangan	Keterangan	Harga	1	FUTSAL	BISA DIGUNAKAN	Rp. 100.000								
No	Lapangan	Keterangan	Harga														
1	FUTSAL	BISA DIGUNAKAN	Rp. 100.000														
1. Petugas membuka halaman data pemesanan	2. Menampilkan halaman data pemesanan																
Kondisi Akhir	Menampilkan halaman data pemesanan lapangan																

Tabel 4. 11 Skenario Use Case Mengelola Data Penyewa

Identifikasi	
Kode	UC-002
Nama	Mengelola data penyewa
Tujuan	Petugas dapat mengelola data penyewa
Deskripsi	Petugas dapat mengelola data penyewa dengan menghapus data user
Aktor	Petugas
Skenario Utama	
Kondisi awal	Aktor berada di halaman data user
Reaksi Aktor	Reaksi Sistem
 Gambar 4. 4 Skenario Use Case Mengelola Data Penyewa	
1. Admin menghapus data user	2. Menghapus data user
Kondisi Akhir	Jika data user sudah di hapus maka akan menampilkan data berhasil di hapus
Skenario Alternatif	
Kondisi Awal	



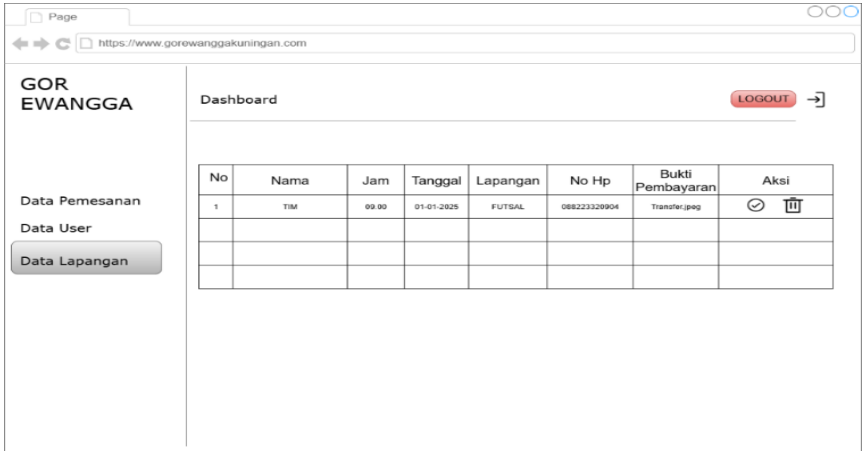
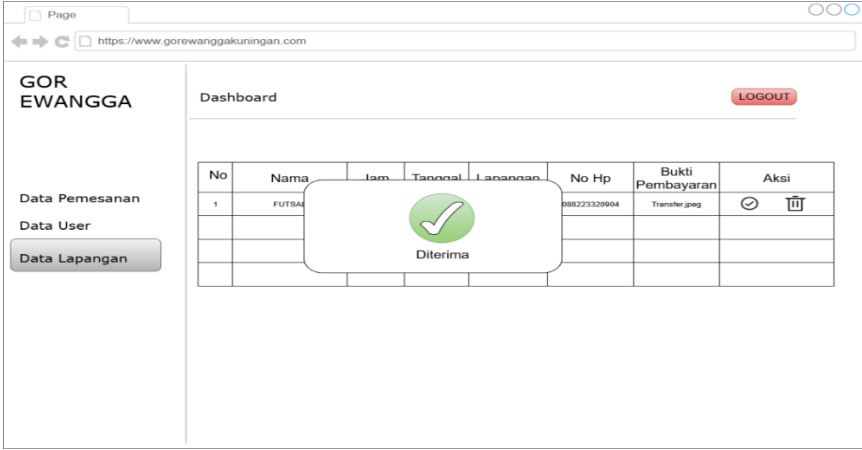
Gambar 4. 5 Skenario Use Case Mengelola Data Penyewa



Gambar 4. 6 Skenario Use Case Mengelola Data Penyewa

Aksi Aktor	Reaksi Sistem
1. Admin menghapus data user	2. Menghapus data user
Kondisi Akhir	Data user berhasil terhapus

Tabel 4. 12 Skenario Use Case Mengelola Data Lapangan

Identifikasi			
Kode	UC-003		
Nama	Mengelola data lapangan		
Tujuan	Petugas dapat mengelola data pemesanan lapangan		
Deskripsi	Petugas dapat mengelola dan mengatur data pemesanan lapangan yang sudah di pesan dengan memvalidasi pemesanan atau membatalkan pemesanan		
Aktor	Petugas		
Skenario Utama			
Kondisi awal	Aktor berada di halaman data lapangan		
Reaksi Aktor	Reaksi Sistem		
			
Gambar 4. 7 Skenario Use Case Mengelola Data Lapangan			
			
Gambar 4. 8 Skenario Use Case Mengelola Data Lapangan			
1. Admin menerima atau membatalkan data pemesanan	2. Menerima atau membatalkan pemesanan		
<table> <tr> <th>Kondisi Akhir</th><th>Pesanan berhasil di daftarkan</th></tr> </table>		Kondisi Akhir	Pesanan berhasil di daftarkan
Kondisi Akhir	Pesanan berhasil di daftarkan		

Tabel 4. 13 Skenario Use Case Registrasi

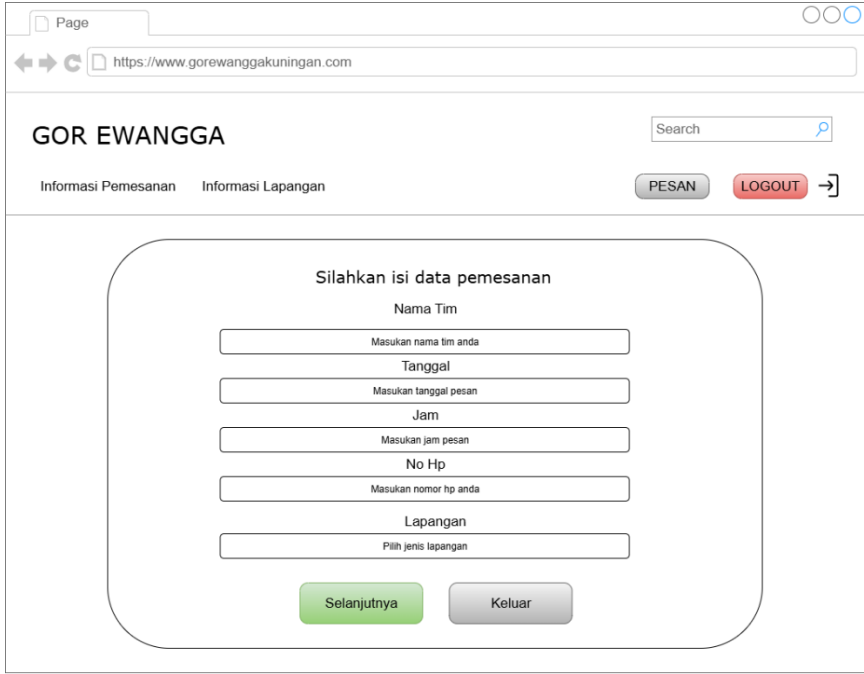
Identifikasi	
Kode	UC-004
Nama	Registrasi
Tujuan	Penyewa melakukan <i>register</i> dan memiliki akun di aplikasi
Deskripsi	Penyewa melakukan registrasi dengan mengisi username, email dan password
Aktor	Penyewa
Skenario Utama	
Kondisi awal	Aktor berada di halaman registrasi
Reaksi Aktor	Reaksi Sistem

Gambar 4. 9 Skenario Use Case Registrasi

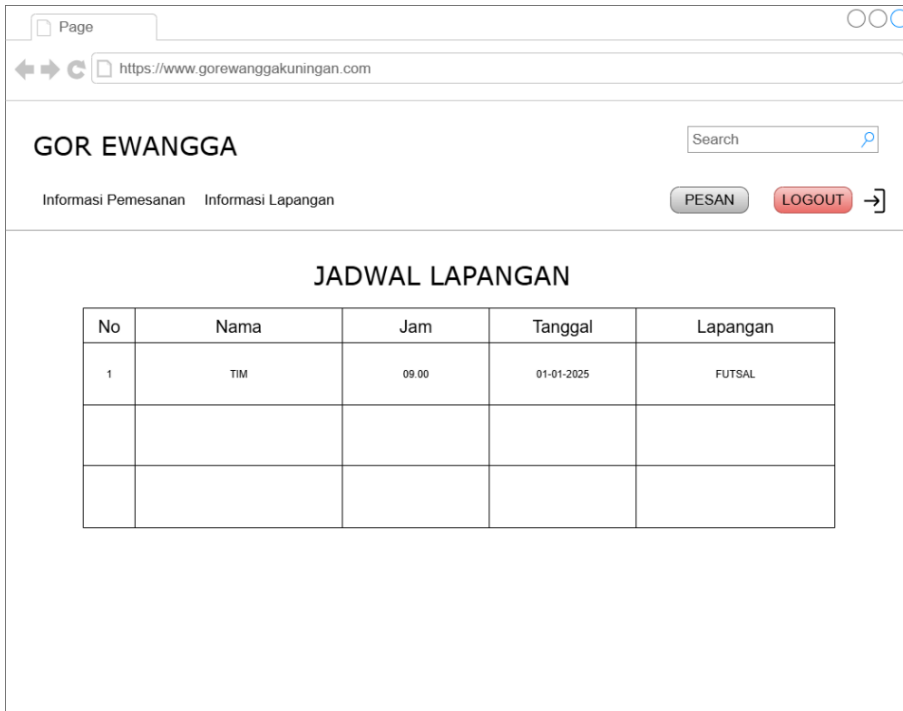
Gambar 4. 10 Skenario Use Case Registrasi

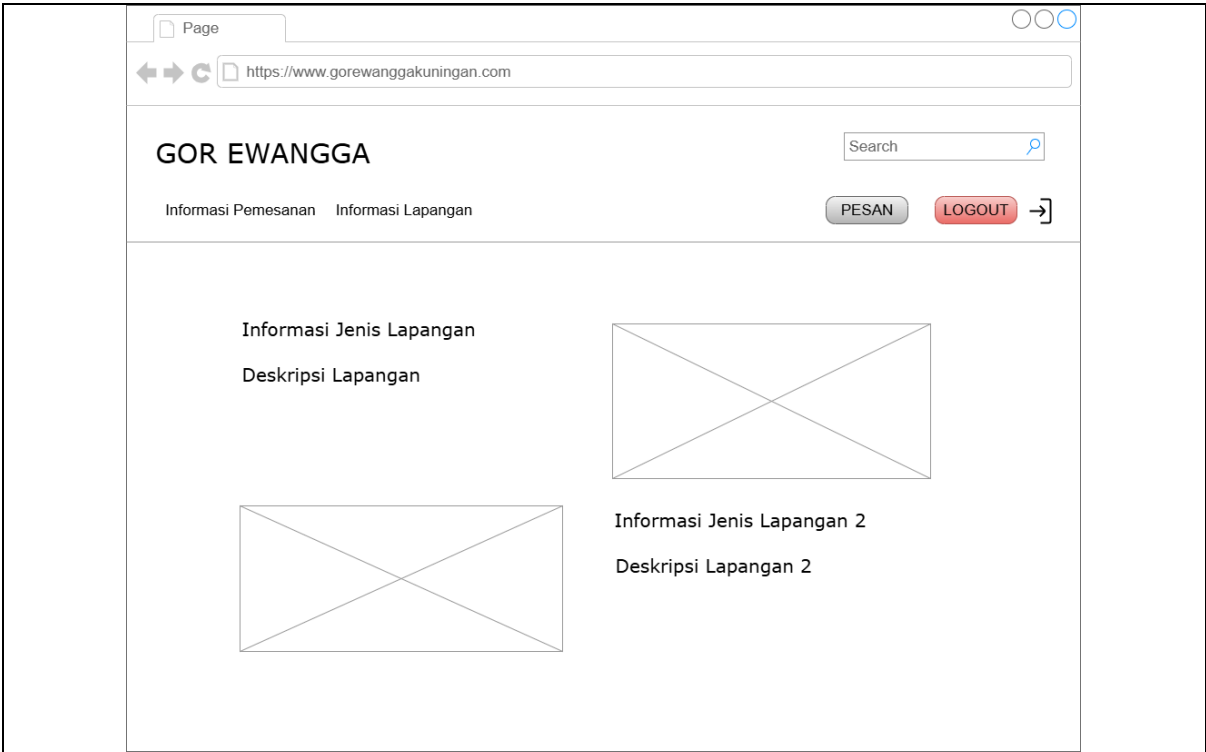
1. user memilih <i>register</i>	2. menampilkan <i>form register</i>
3. Menginput Email aktif	4. Input nama, <i>username</i> , <i>email</i> aktif dan <i>password</i>
5. Menyimpan data	6. Validasi Email
7. Membaca Email	8. Simpan
	9. Masuk ke halaman <i>dashboard</i>
Kondisi Akhir	Jika aktor telah masuk maka <i>username</i> dan <i>password</i> yang di input digunakan untuk login. Jika untuk validasi Email berfungsi untuk mengecek Email yang digunakan benar pengguna yang mendaftar

Tabel 4. 14 Skenario Use Case Pemesanan

Identifikasi	
Kode	UC-005
Nama	Pemesanan
Tujuan	Penyewa dapat melakukan pemesanan lapangan
Deskripsi	Penyewa melakukan pemesanan dengan mengisi data pemesanan berupa nama tim, tanggal pemesanan, jam pemesanan, nomor handphone yang dapat dihubungi dan jenis lapangan
Aktor	Penyewa
Skenario Utama	
Kondisi awal	Aktor telah masuk ke halaman utama pemesanan
Reaksi Aktor	Reaksi Sistem
	
Gambar 4. 11 Skenario Use Case Pemesanan	
1. Aktor memilih menu Pesan	2. Menampilkan halaman pesan
3. Aktor memesan sesuai tanggal dan waktu	4. Menyimpan pesanan
5. Menampilkan pembayaran	6. Aktor input pembayaran
7. simpan	8. pemesanan selesai
Kondisi Akhir	Aktor dapat melakukan pemesanan lapangan dengan memilih waktu dan tanggal, setelah itu melakukan pembayaran dengan mengupload bukti pembayaran

Tabel 4. 15 Skenario Use Case Informasi Pemesanan

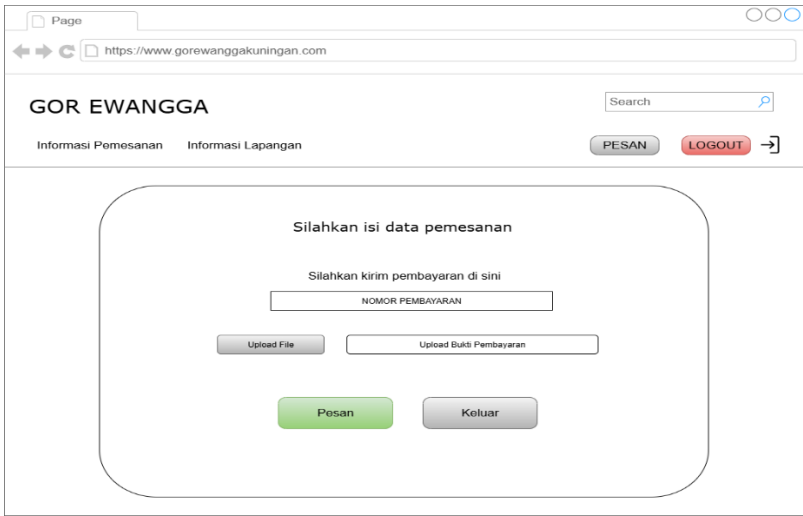
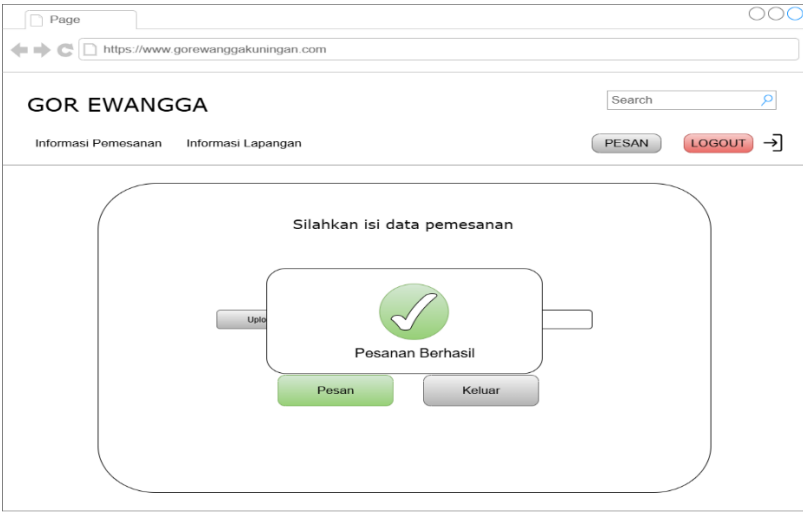
Identifikasi	
Kode	UC-006
Nama	Informasi Pemesanan
Tujuan	Penyewa dapat melihat informasi pemesanan
Deskripsi	Penyewa dapat melihat jadwal lapangan yang tersedia secara terkini
Aktor	Penyewa
Skenario Utama	
Kondisi awal	Aktor telah masuk ke halaman informasi pemesanan
Reaksi Aktor	Reaksi Sistem
 Gambar 4. 12 Skenario Use Case Informasi Pemesanan	
1. Aktor memilih menu info pemesanan	2. Menampilkan halaman info pemesanan
Kondisi Akhir	Aktor dapat melihat info pemesanan yang sudah terisi secara terkini
Skenario Alternatif	
Kondisi Awal	



Gambar 4. 13 Skenario Use Case Informasi Pemesanan

Aksi Aktor		Reaksi Sistem	
1.	Memilih informasi lapangan	2.	Menampilkan informasi lapangan
Kondisi Akhir		Aktor dapat melihat informasi jenis lapangan	

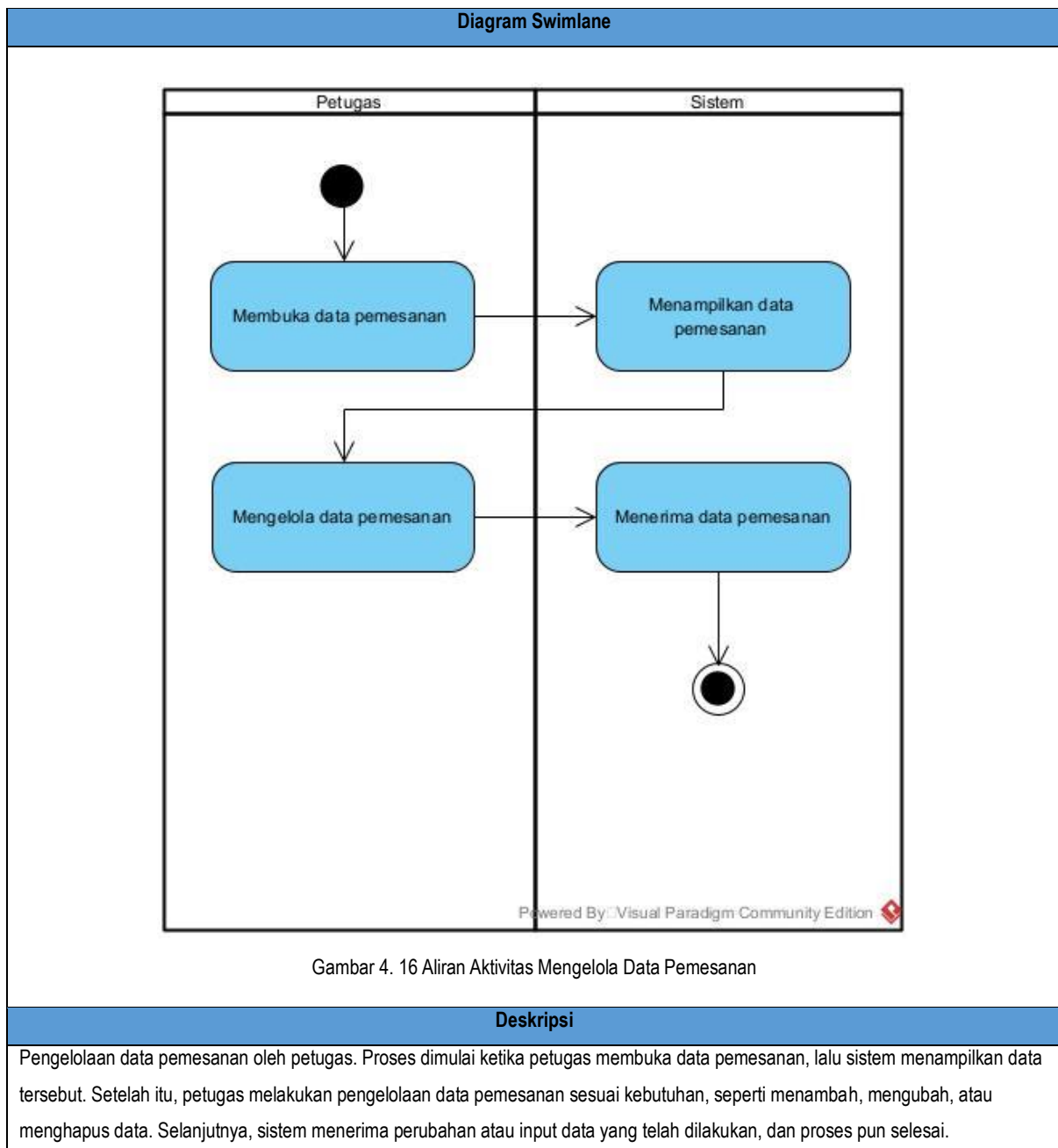
Tabel 4. 16 Skenario Use Case Pembayaran

Identifikasi	
Kode	UC-007
Nama	Pembayaran
Tujuan	Penyewa dapat melakukan pembayaran
Deskripsi	Penyewa dapat melakukan pembayaran dengan mengupload bukti pembayaran
Aktor	Penyewa
Skenario Utama	
Kondisi awal	Aktor telah masuk ke halaman pembayaran
Reaksi Aktor	Reaksi Sistem
	
Gambar 4. 14 Skenario Use Case Pembayaran	
	
Gambar 4. 15 Skenario Use Case Pembayaran	
1. Aktor mengupload bukti pembayaran	2. Menyimpan bukti pembayaran
Kondisi Akhir	
Aktor dapat melakukan pembayaran dengan mengupload bukti pembayaran dan pesanan akan berhasil di simpan	

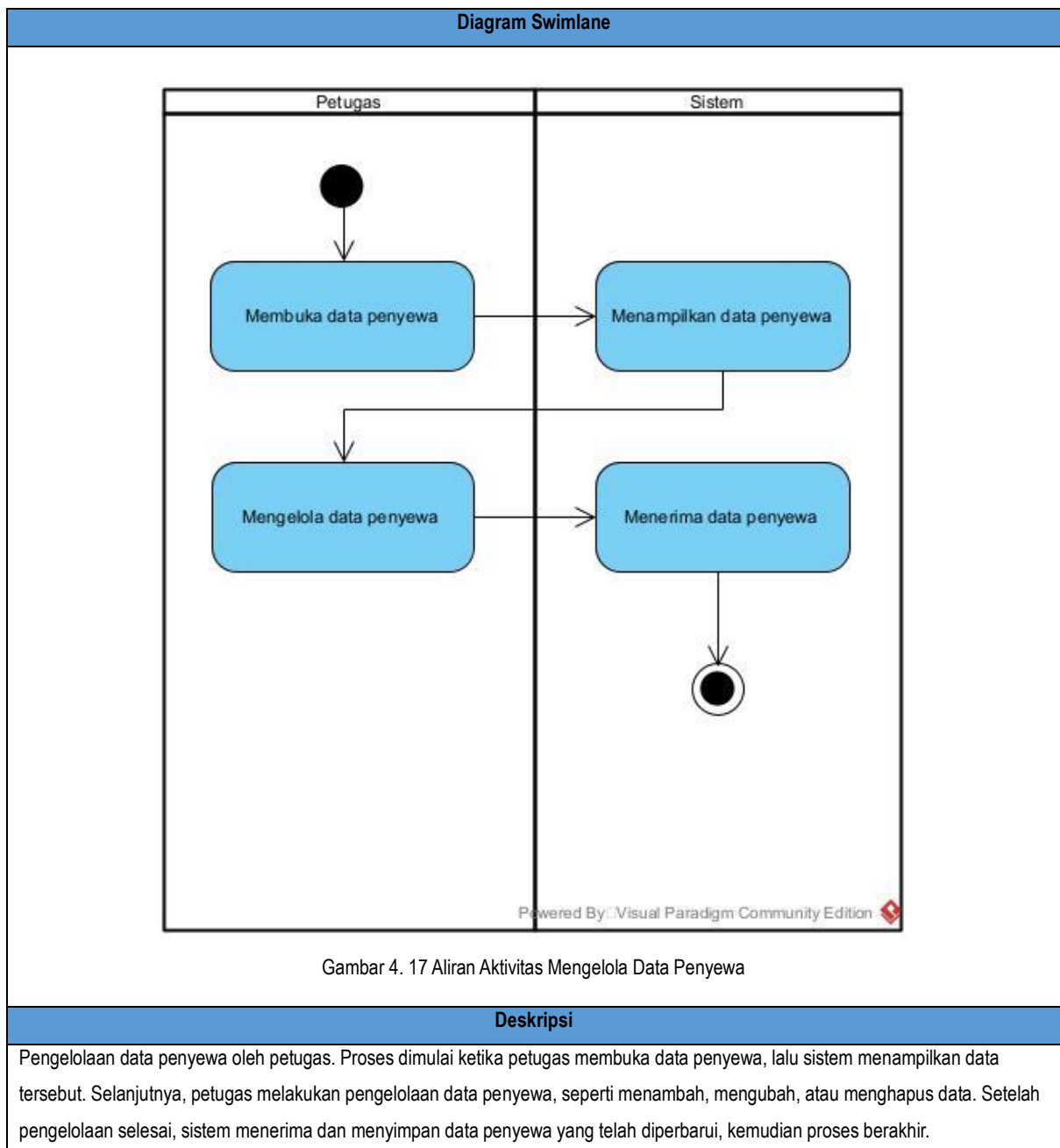
4.2.2.5 Aliran Aktivitas

Aliran aktivitas digunakan untuk mendefinisikan aliran tampilan dari sistem yang akan dibangun. Diagram aktivitas swimlane mengelompokkan aktivitas menjadi kolom swimlane yang berisi aktivitas yang masuk ke dalam katagori yang diwakili oleh swimlane. Aliran aktivitas dapat dilihat pada beberapa gambar berikut.

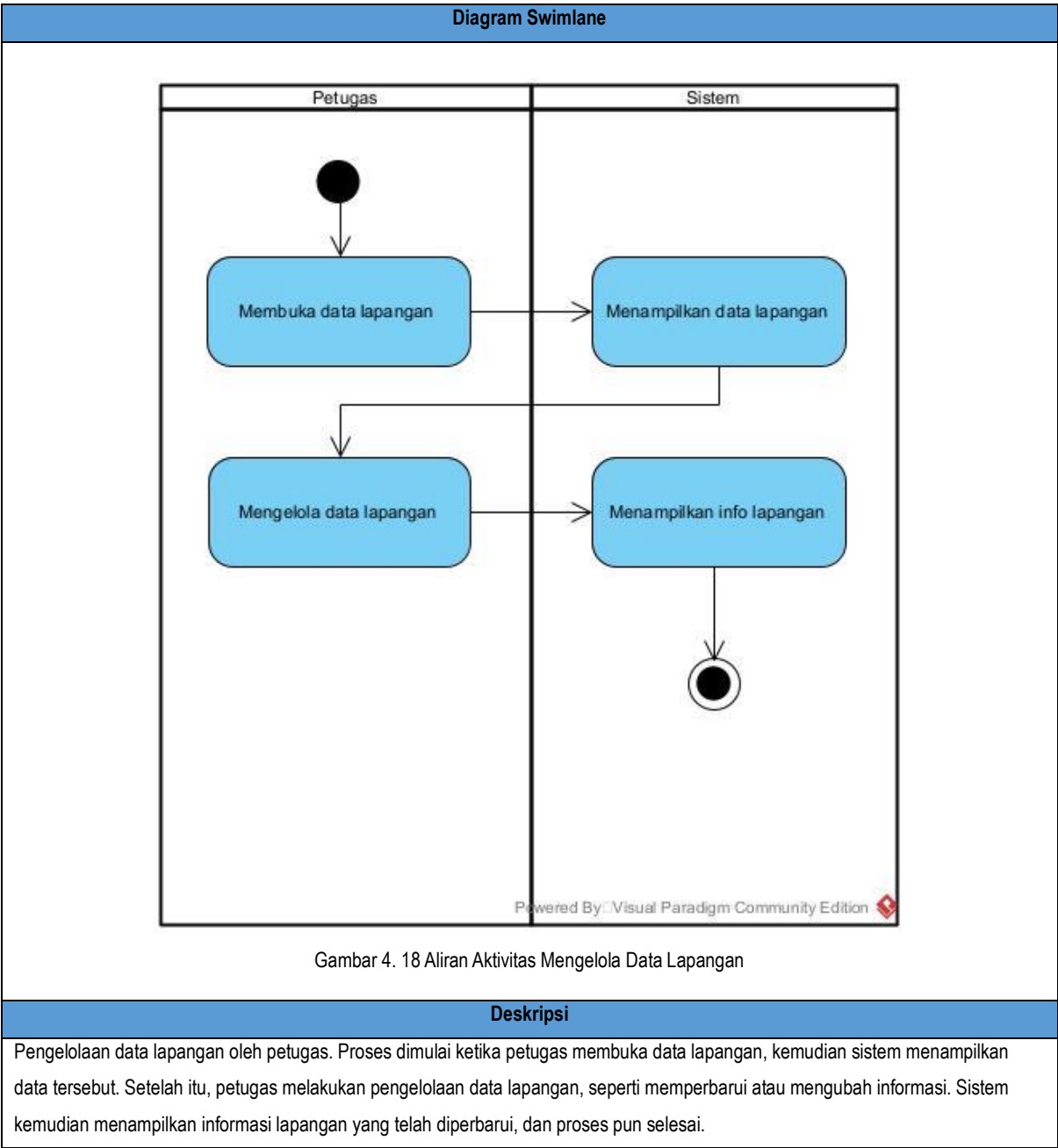
Tabel 4. 17 Aliran Aktivitas Mengelola Data Pemesanan



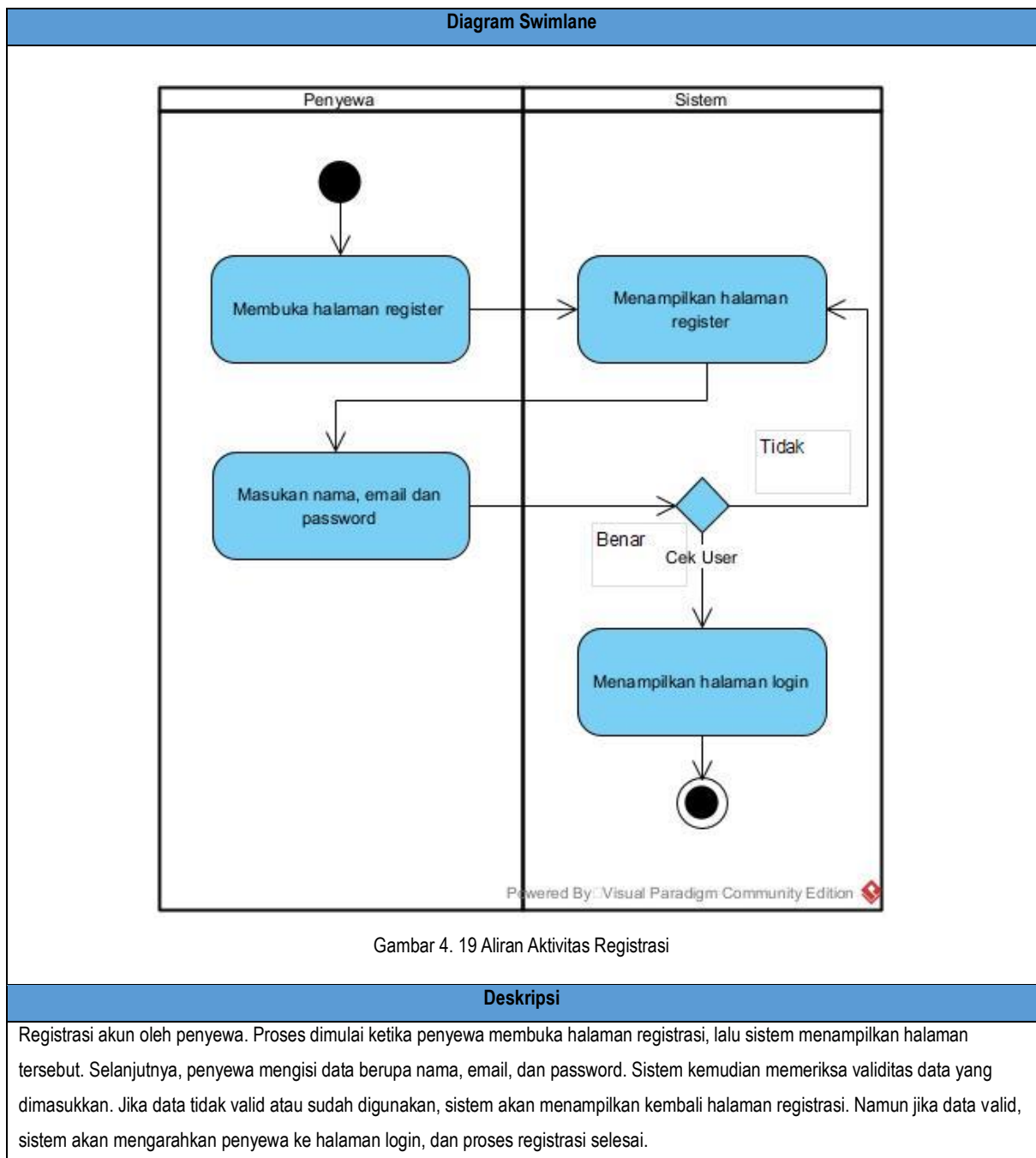
Tabel 4. 18 Aliran Aktivitas Mengelola Data Penyewa



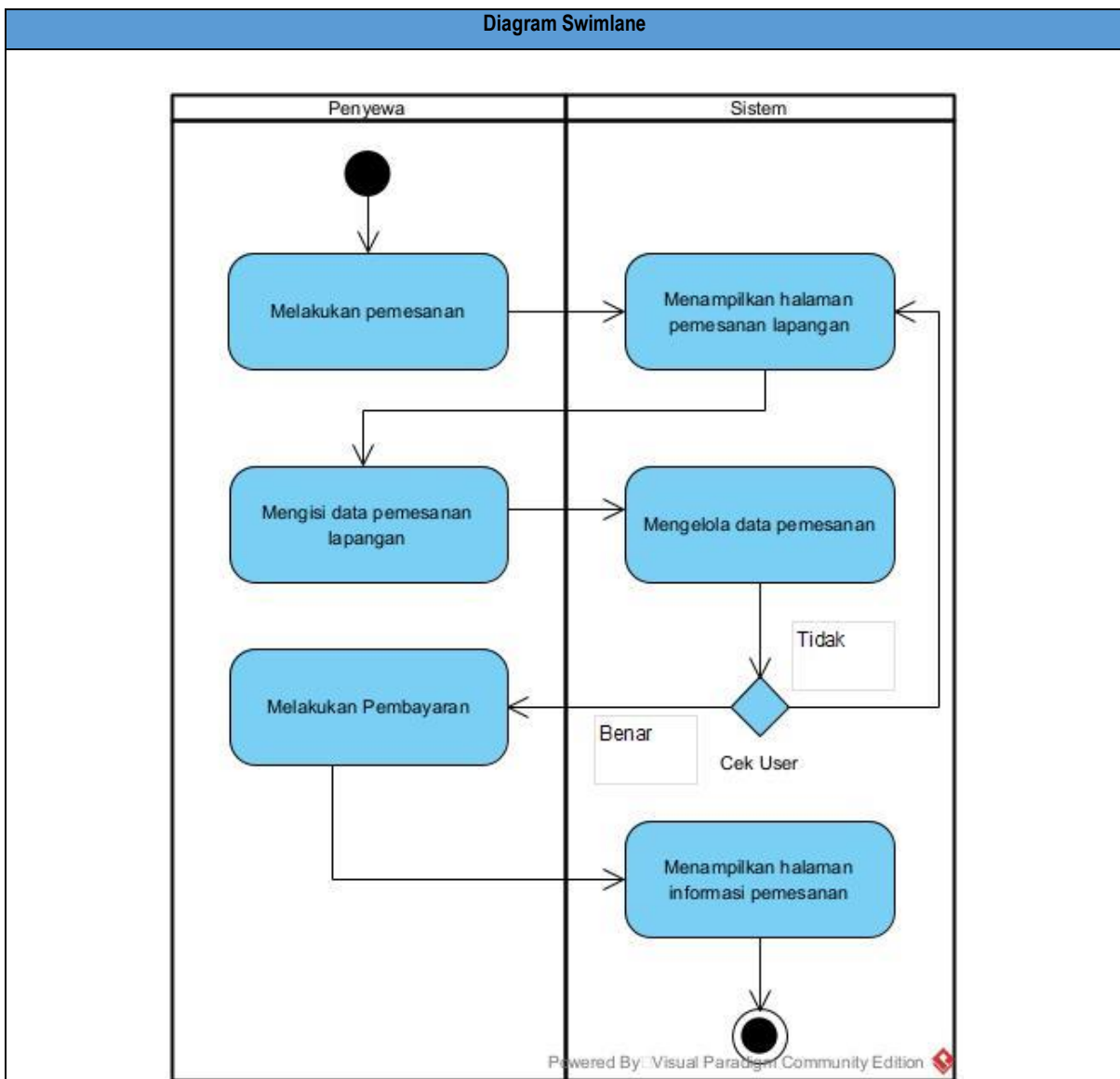
Tabel 4. 19 Aliran Aktivitas Mengelola Data Lapangan



Tabel 4. 20 Aliran Aktivitas Registrasi



Tabel 4. 21 Aliran Aktivitas Pemesanan

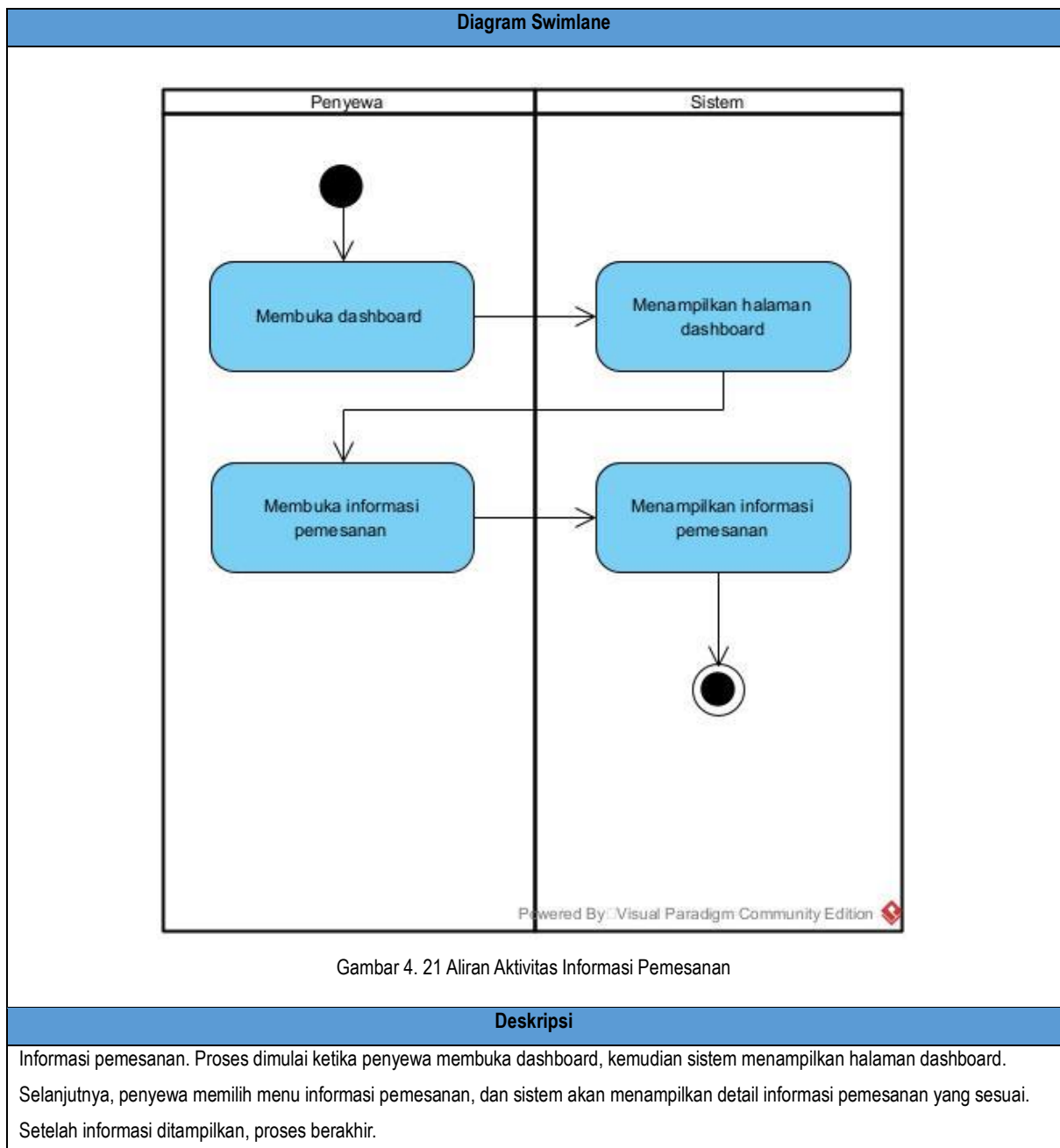


Gambar 4. 20 Aliran Aktivitas Pemesanan

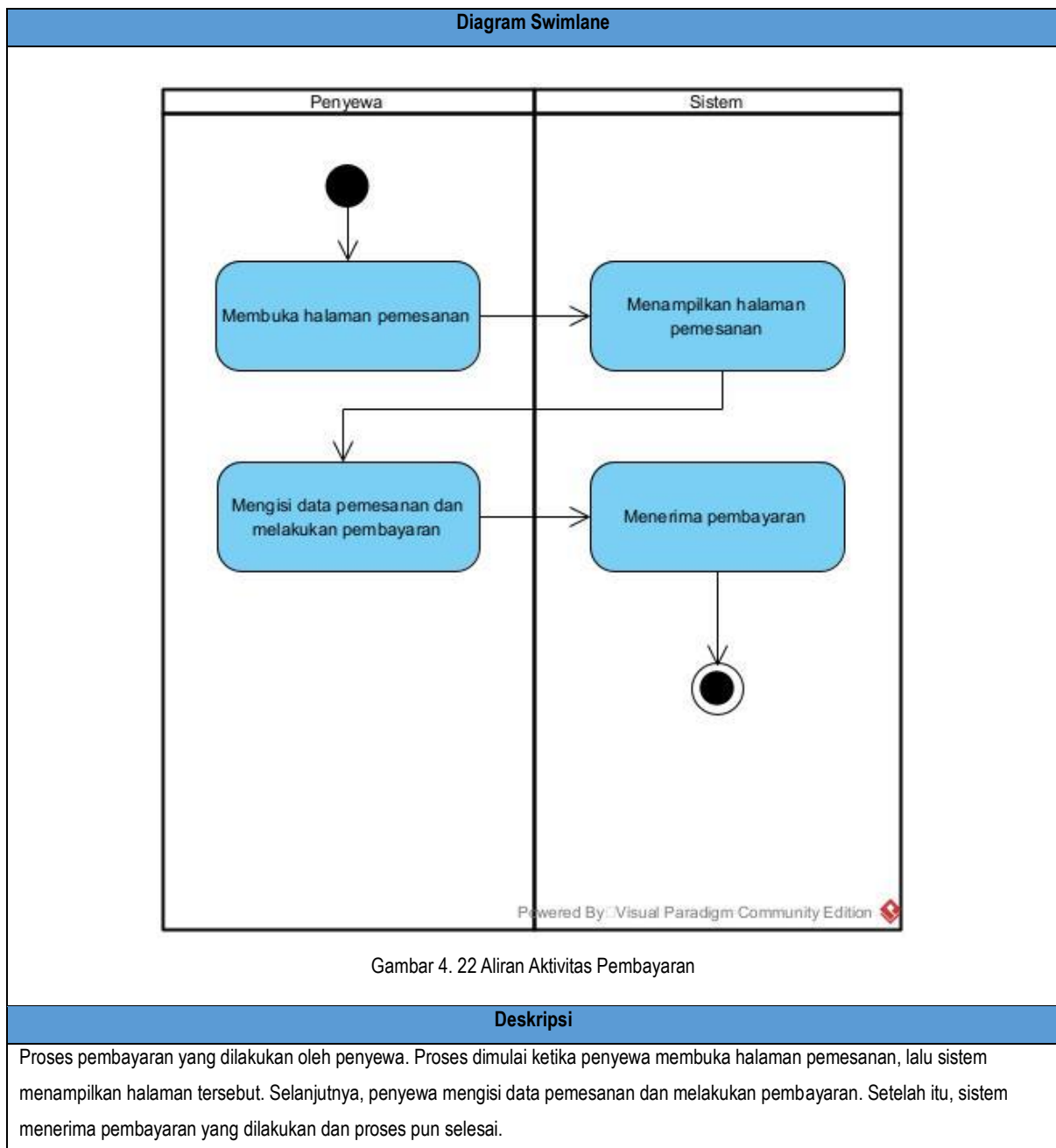
Deskripsi

Pemesanan lapangan oleh penyewa. Proses dimulai saat penyewa melakukan pemesanan, lalu sistem menampilkan halaman pemesanan lapangan. Penyewa kemudian mengisi data pemesanan, dan sistem mengelola data tersebut. Selanjutnya, sistem memverifikasi data pengguna. Jika valid, penyewa dapat melanjutkan ke tahap pembayaran. Setelah pembayaran dilakukan, sistem menampilkan halaman informasi pemesanan, dan proses pun selesai.

Tabel 4. 22 Aliran Aktivitas Informasi Pemesanan



Tabel 4. 23 Aliran Aktivitas Pembayaran



4.2.3 Pemodelan Berbasis Kelas

Pemodelan berbasis kelas memodelkan objek-objek yang akan dimanipulasi oleh sistem perangkat lunak, operasi-operasi yang akan diterapkan, relasi-relasi antar objek, serta memperlihatkan kolaborasi yang terjadi antara kelas yang didefinisikan.

4.2.3.1 Objek Data

Bagian ini adalah dokumentasi perancangan yang mengidentifikasi dan mendeskripsikan entitas yang digunakan dalam basis data. Dalam perancangan, objek data ini sangat penting karena mereka menentukan struktur dari informasi yang akan disimpan dan diproses oleh basis data.

Tabel 4. 24 Objek Data

No.	Nama Objek	Deskripsi
1.	Users	Tabel untuk mngelola data pengguna
2.	Pemesanan	Tabel untuk mengelola transaksi pemesanan lapangan
3.	Pembayarans	Tabel untuk mengelola detail pembayaran
4.	Lapangans	Tabel untuk mengelola informasi lapangan
5.	Jadwals	Tabel untuk mengelola waktu penggunaan lapangan
6.	Personal acces tokens dan migrations	Tabel untuk mendukung autentikasi dan migrasi sistem.

4.2.3.2 Identifikasi Atribut Kelas

Pada umumnya, setiap kelas yang disebutkan memiliki berbagai atribut yang diatur dan diterapkan untuk memastikan aplikasi berfungsi dengan baik. Atribut-atribut yang secara khusus dikembangkan untuk rancangan sistem Pengarsipan berbasis web

Tabel 4. 25 Identifikasi Atribut Kelas

No	Nama	Jenis Kelas	Atribut	Type	Atribut Length / Value	Primary Key	Foreign Key
1	User	Entity	id	bigint	20	✓	
			nama_lengkap	varchar	255		
			username	varchar	255		
			email	varchar	255		
			no_telp	varchar	255		
			password	varchar	255		
			role	enum	penyewa/petugas		
2	Pemesanan	Entity	Id	bigint	20	✓	
			User_id	Bigint	20		✓
			Lapangan_id	Bigint	20		✓
			Jadwal	Longtext	-		
			Tanggal_pemesanan	Date	-		
			Total_harga	Decimal	(10,2)		

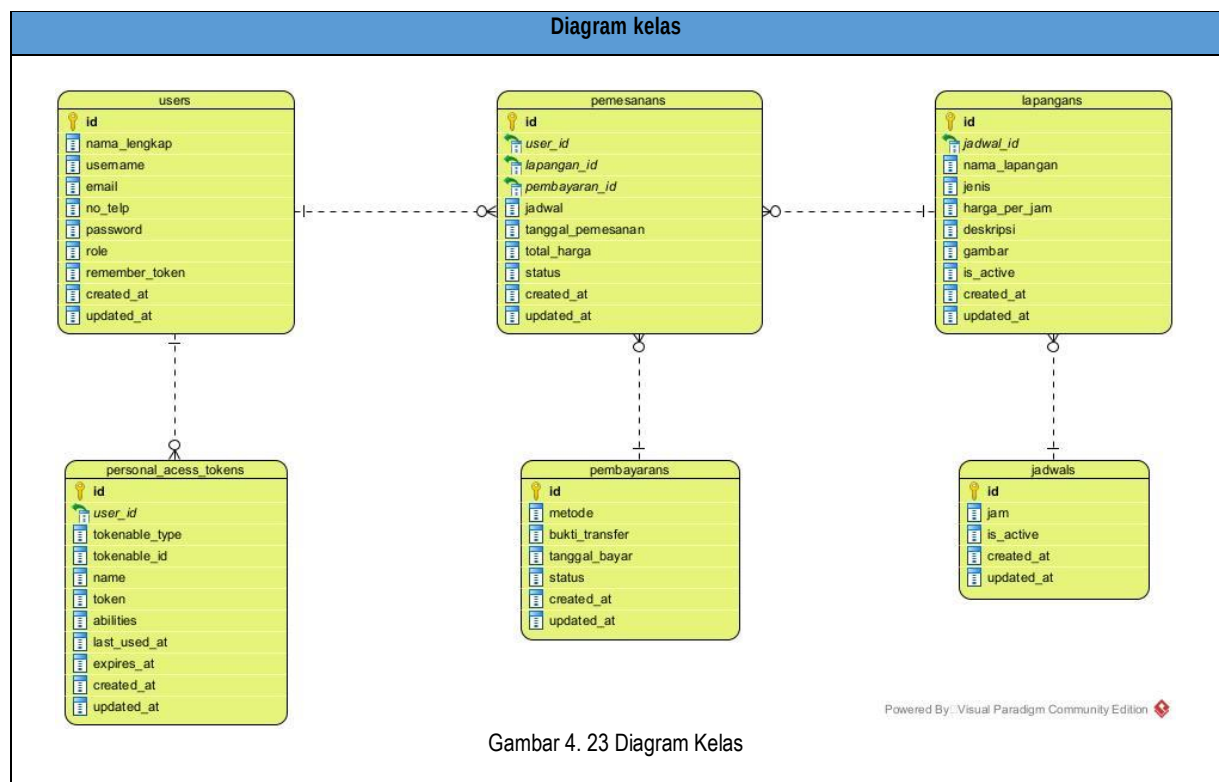
No	Nama	Jenis Kelas	Atribut	Type	Atribut Length / Value	Primary Key	Foreign Key
			status	enum	Pending/sukses/gagal		
3.	Pembayaran	Entity	Id	Bigint	20	✓	
			Pemesanan_id	Bigint	20		✓
			Metode	Varchar	255		
			Bukti_transfer	Varchar	255		
			Tanggal_bayar	Datetime	-		
			status	enum	Pending/valid/invalid		
4	Lapangan	Entity	Id	Bigint	20	✓	
			Nama_lapangan	Varchar	255		
			Jenis	Enum	Futsal/basket/badminton/tenis		
			Harga_per_jam	Decimal	(10,2)		
			Deskripsi	Text	-		
			gambar	Longtext	-		
			Is_active	tinyint	1		
5	Jadwal	Entity	Id	Bigint	20	✓	
			Jam	Time	-		
			Is_active	tinyint	1		
6.	PersonalAccess Tokens	Entity	Id	Bigint	20	✓	
			Tokenable_type	Varchar	255		
			Tokenable_id	Bigint	20		
			Name	Varchar	255		
			Token	Varchar	64		
			Abilities	Text	-		
			Last_used_at	Timestamp	-		
			Expires_at	Timestamp	-		
			Created_at	Timestamp	-		
			Updated_at	timestamp	-		
7	Migrations	Entity	Id	Int	10	✓	
			Migration	Varchar	255		
			batch	int	11		

4.2.3.3 Diagram Kelas

Diagram Kelas merupakan bagian penting dalam tahap perancangan perangkat lunak, yang digunakan untuk menggambarkan struktur sistem secara rinci. Diagram ini menunjukkan hubungan antar kelas dalam sistem, atribut yang dimiliki oleh setiap kelas, serta metode yang akan digunakan untuk menjalankan operasi pada data. Dengan menggunakan Diagram Kelas, pengembang dapat memahami bagaimana berbagai komponen perangkat lunak berinteraksi dan berfungsi satu sama lain dalam sistem berbasis website yang akan dibangun.

Diagram ini juga membantu untuk memvisualisasikan entitas utama dalam sistem, serta relasi antar objek yang relevan, seperti asosiasi, pewarisan, dan agregasi. Berikut adalah Diagram Kelas yang dihasilkan berdasarkan analisis kebutuhan fungsional dan perancangan sistem yang telah dilakukan.

Tabel 4. 26 Diagram Kelas

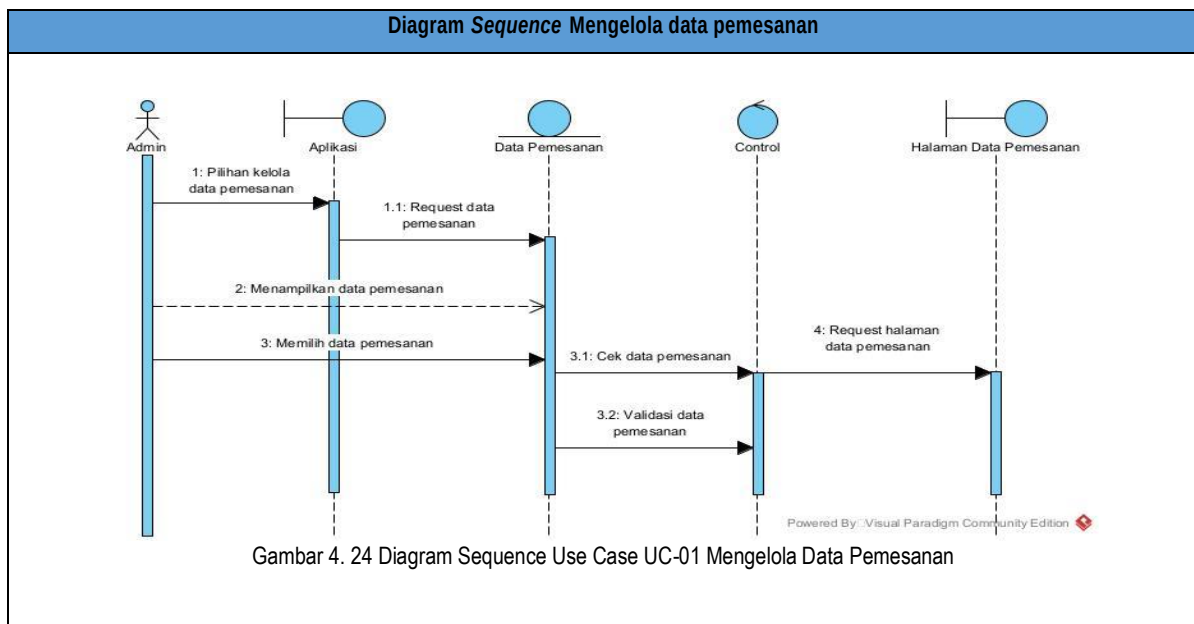


Gambar 4. 23 Diagram Kelas

4.2.3.4 Diagram Sequence

Diagram *sequence* menggambarkan kelakuan atau perilaku objek pada use case dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek. Dapat dilihat pada tabel-tabel berikut ini.

Tabel 4. 27 Diagram Sequence Mengelola Data Pemesanan

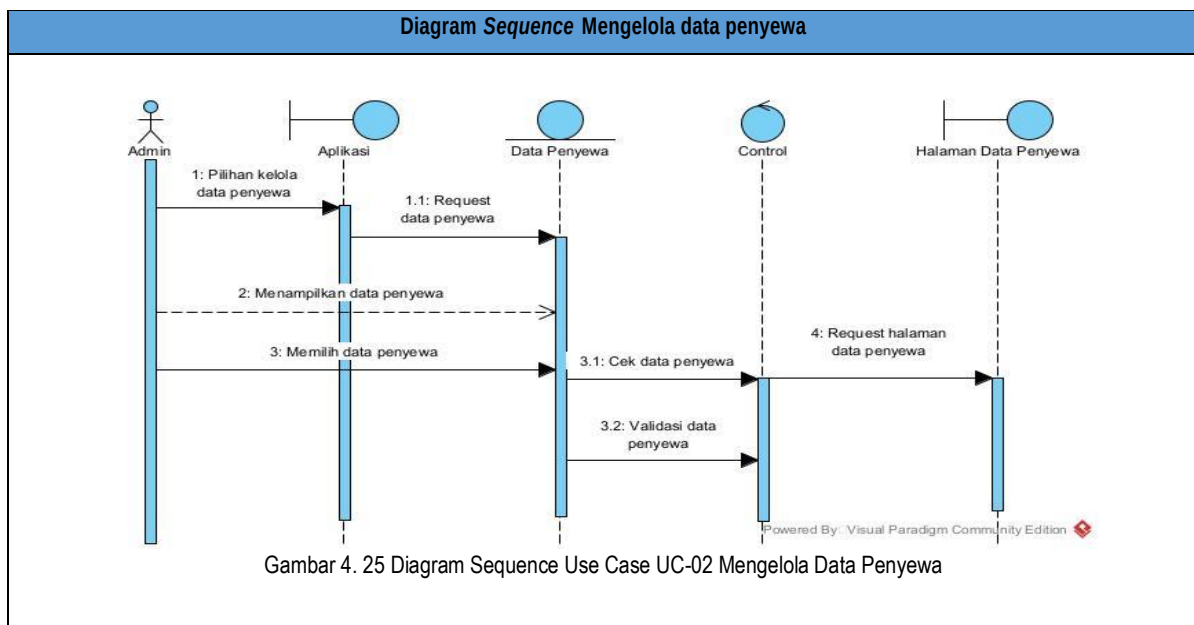


Gambar 4. 24 Diagram Sequence Use Case UC-01 Mengelola Data Pemesanan

Penjelasan

Sequence diagram mengelola data pemesanan menggambarkan proses interaksi antara admin, aplikasi, kontrol, dan halaman data pemesanan dalam pengelolaan informasi reservasi. Alur dimulai saat admin memilih menu *kelola data pemesanan*, kemudian aplikasi menampilkan daftar pemesanan. Admin memilih data yang ingin diperiksa, dan sistem melalui kontrol melakukan validasi serta pengecekan ke database. Setelah data diverifikasi, aplikasi menampilkan halaman detail pemesanan yang siap diperbarui. Diagram ini menunjukkan bagaimana sistem memastikan pengelolaan data pemesanan berjalan terstruktur dan akurat.

Tabel 4. 28 Diagram Sequence Mengelola Data Penyewa

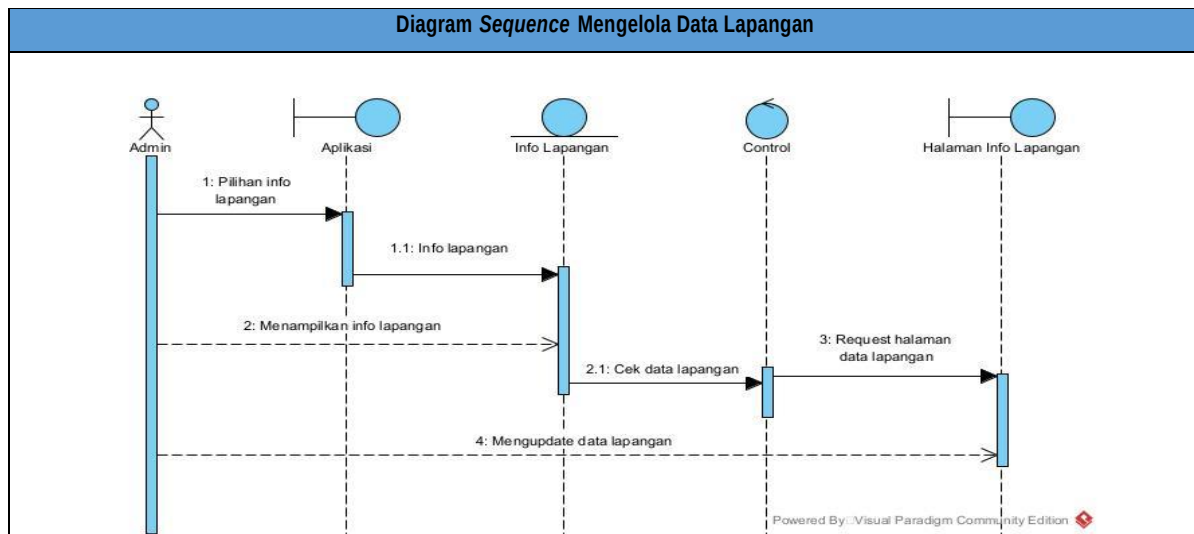


Gambar 4. 25 Diagram Sequence Use Case UC-02 Mengelola Data Penyewa

Penjelasan

Sequence diagram mengelola data penyewa menggambarkan alur interaksi antara admin, aplikasi, kontrol, dan halaman data penyewa dalam proses pengelolaan informasi pengguna. Proses dimulai saat admin memilih menu *kelola data penyewa*, aplikasi menampilkan daftar data penyewa, lalu admin memilih data yang ingin diperbarui. Sistem melalui kontrol melakukan pengecekan dan validasi ke database, kemudian menampilkan halaman detail penyewa untuk dilakukan perubahan. Dengan alur ini, sistem memastikan pengelolaan data penyewa berjalan terstruktur, akurat, dan mudah diakses.

Tabel 4. 29 Diagram Sequence Mengelola Data Lapangan

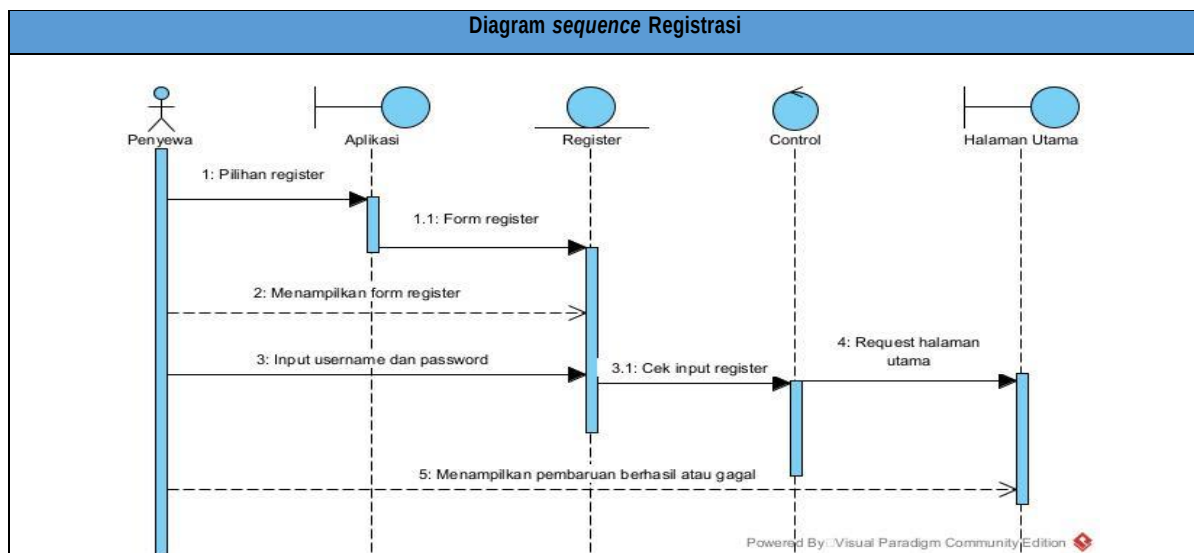


Gambar 4. 26 Diagram Sequence Use Case UC-03 Mengelola Data Lapangan

Penjelasan

Sequence diagram info lapangan menggambarkan alur interaksi antara admin, aplikasi, kontrol, dan halaman info lapangan dalam proses pengelolaan data fasilitas. Proses dimulai saat admin memilih menu *info lapangan*, aplikasi menampilkan informasi lapangan, lalu sistem melakukan pengecekan dan validasi data. Admin dapat memperbarui detail lapangan seperti deskripsi atau status ketersediaan, dan aplikasi menampilkan halaman info lapangan yang sudah diperbarui. Dengan alur ini, sistem memastikan informasi lapangan selalu akurat, terkini, dan mudah diakses oleh penyewa.

Tabel 4. 30 Diagram Sequence Registrasi

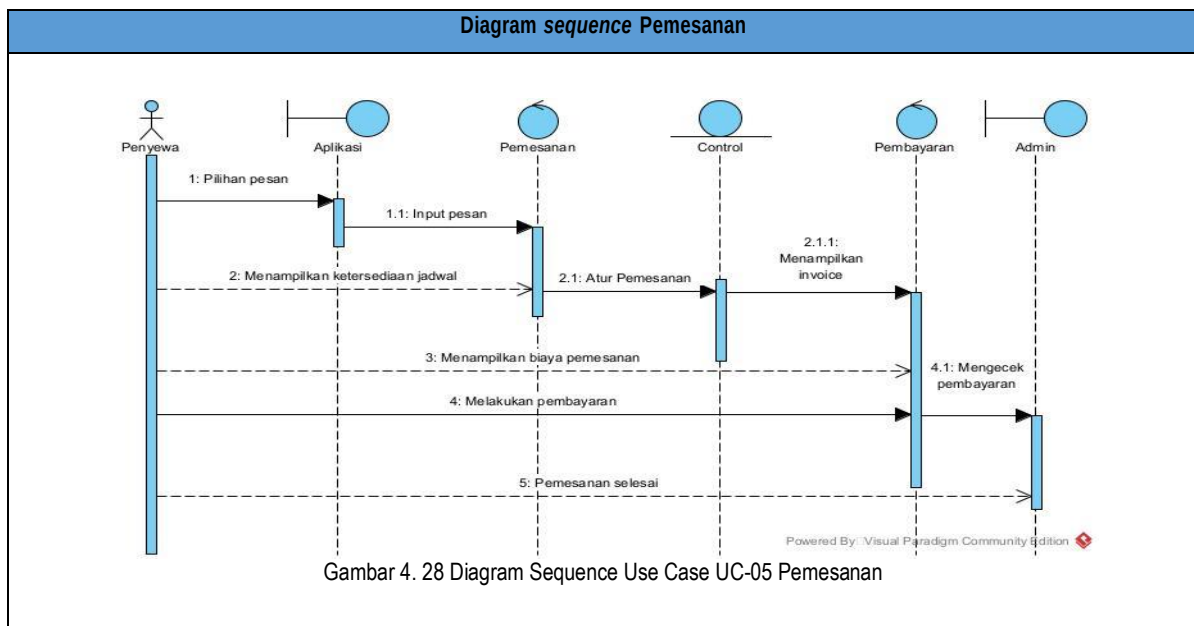


Gambar 4. 27 Diagram Sequence Use Case UC-04 Registrasi

Penjelasan

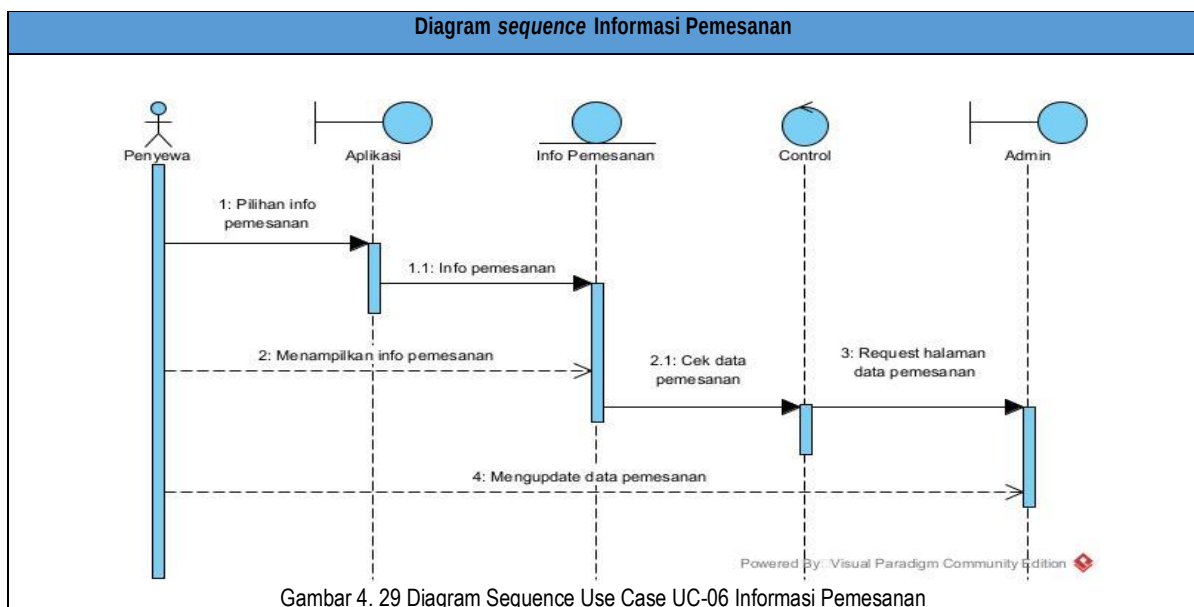
Sequence diagram register menggambarkan alur interaksi antara penyewa, aplikasi, kontrol, dan halaman utama dalam proses pendaftaran akun. Proses dimulai saat penyewa memilih menu *register*, aplikasi menampilkan form pendaftaran, lalu penyewa mengisi username dan password. Sistem melalui kontrol melakukan pengecekan validasi input. Jika data valid, aplikasi mengarahkan ke halaman utama dan menampilkan notifikasi bahwa pendaftaran berhasil; jika tidak valid, sistem menampilkan pesan gagal. Dengan alur ini, sistem memastikan proses registrasi berjalan terstruktur, aman, dan mudah dipahami oleh pengguna.

Tabel 4. 31 Diagram Sequence Pemesanan

**Penjelasan**

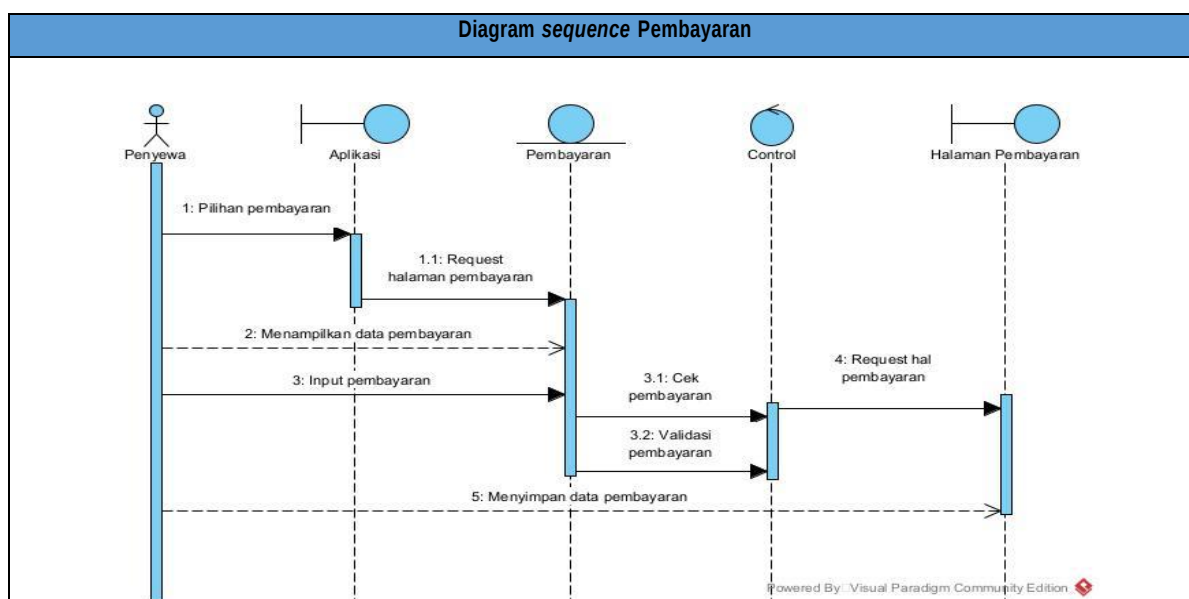
Sequence diagram pemesanan lapangan menggambarkan alur interaksi antara penyewa, aplikasi, modul pemesanan, kontrol, pembayaran, dan admin dalam proses reservasi. Proses dimulai saat penyewa memilih menu *pesan* dan mengisi data pemesanan. Aplikasi menampilkan ketersediaan jadwal, lalu kontrol mengatur reservasi sesuai pilihan. Setelah itu sistem menampilkan invoice dan rincian biaya. Penyewa melakukan pembayaran, modul pembayaran memverifikasi transaksi, dan kontrol memastikan status valid. Jika pembayaran berhasil, sistem menandai pemesanan selesai dan admin dapat memantau data transaksi. Dengan alur ini, proses pemesanan berjalan terstruktur, transparan, dan aman.

Tabel 4. 32 Diagram Sequence Informasi Pemesanan

**Penjelasan**

Sequence diagram info pemesanan menggambarkan alur interaksi antara penyewa, aplikasi, kontrol, dan admin dalam proses pengelolaan informasi reservasi. Proses dimulai saat penyewa memilih menu *info pemesanan*, aplikasi menampilkan detail pemesanan, lalu sistem melakukan pengecekan dan validasi data melalui kontrol. Admin dapat melakukan pembaruan data pemesanan, dan aplikasi menampilkan halaman informasi yang sudah diperbarui. Dengan alur ini, sistem memastikan informasi pemesanan selalu akurat, transparan, dan mudah diakses oleh pengguna.

Tabel 4. 33 Diagram Sequence Pembayaran



Gambar 4. 30 Diagram Sequence Use Case UC-07 Pembayaran

Penjelasan

Sequence diagram pembayaran menggambarkan alur interaksi antara penyewa, aplikasi, kontrol, dan halaman pembayaran dalam proses transaksi. Proses dimulai saat penyewa memilih menu *pembayaran*, aplikasi menampilkan halaman pembayaran, lalu penyewa menginput data pembayaran. Sistem melalui kontrol melakukan pengecekan dan validasi transaksi. Setelah validasi, aplikasi menyimpan data pembayaran dan menampilkan status transaksi kepada penyewa. Dengan alur ini, sistem memastikan proses pembayaran berjalan aman, terstruktur, dan transparan.

BAB 5 IMPLEMENTASI DAN PENGUJIAN

Bab ini berfokus pada tahap konstruksi dalam pengembangan aplikasi, yang mencakup sub bab Coding dan Testing. Sub bab Coding berisi implementasi dari desain yang telah dibuat pada tahap sebelumnya, termasuk penulisan kode program yang sesuai dengan spesifikasi yang telah ditentukan. Proses ini melibatkan penggunaan bahasa pemrograman, alat bantu pengembangan, dan 1 royek-teknik coding yang efisien untuk memastikan aplikasi dapat berjalan dengan baik.

Sub bab Testing berisi aktivitas pengujian yang dilakukan untuk memastikan bahwa aplikasi yang telah dibangun berfungsi sesuai dengan kebutuhan dan bebas dari kesalahan. Pengujian dalam bab ini akan menggunakan metode black-box Testing, yang fokus pada pengujian fungsionalitas aplikasi tanpa memperhatikan struktur internal kode.

5.1 Coding

Sub bab ini menekankan pada tahap implementasi kode dalam pengembangan aplikasi pembelajaran. Pada tahap ini, berbagai alat dan lingkungan pengembangan digunakan untuk memastikan bahwa kode yang ditulis sesuai dengan spesifikasi yang telah ditentukan dan berfungsi dengan baik.

Tabel 5. 1 Spesifikasi Perangkat Keras

No.	Nama Perangkat	Sistem Requirement
1.	Processor	Intel(R) Core(TM) 5 120U (1.40 GHz)
2.	RAM	8,00 GB
3.	Disk Space	477 GB
4.	Operating System	Windows 11 64-bit

Selain spesifikasi perangkat keras yang digunakan sebagaimana ditunjukkan pada Tabel 5.1, penelitian ini juga memanfaatkan sejumlah perangkat lunak pendukung yang dirangkum dalam Tabel 5.2.

Tabel 5. 2 Perangkat lunak yang digunakan dalam pembangunan

No.	Nama Perangkat	Sumber	Deskripsi
1.	Visual Studio Code	https://code.visualstudio.com/	Perangkat lunak yang digunakan sebagai <i>code editor</i> untuk menuliskan <i>code</i> pada saat membangun website.
2.	MySQL	https://www.apachefriends.org/	Perangkat lunak yang digunakan sebagai database.
4.	Laravel	https://laravel.com/	Perangkat lunak yang digunakan sebagai framework dalam tahap pembangunan website.

5.1.1. Struktur Folder

Struktur folder adalah pengaturan direktori dan file dalam 1 proyek aplikasi yang digunakan untuk mengorganisir kode dan sumber daya lainnya. Pengaturan ini sangat penting dalam pengembangan aplikasi berbasis web, karena memungkinkan pengembang untuk dengan mudah menemukan, memodifikasi, dan mengelola berbagai komponen yang dibutuhkan.

Tabel 5. 3 Struktur Folder

No.	Nama Folder	Nama File	Deskripsi
1.	App	Berisi file PHP untuk logika aplikasi (controller, model, dll.)	Tempat utama kode program Laravel, mengatur alur bisnis aplikasi.
2.	Bootstrap	app.php	Mengatur proses awal (bootstrapping) Laravel sebelum aplikasi berjalan.
3.	Config	File konfigurasi (app.php, database.php, dll.)	Menyimpan pengaturan aplikasi seperti database, cache, dan session.
4.	Database	migrations, factories, seeders	Mengatur struktur tabel database (migrations), data dummy (factories), dan data awal (seeders).
5.	Public	index.php, folder img-home	Folder yang bisa diakses publik, berisi file utama aplikasi dan aset (gambar, CSS, JS).
6.	Resources	views, css, js	Menyimpan file tampilan (Blade template), stylesheet, dan script JavaScript.
7.	Routes	web.php, api.php	Menentukan rute URL aplikasi dan menghubungkannya dengan controller.
8.	Storage	bukti_transfer, lapangan	Menyimpan file upload pengguna (misalnya bukti transfer dan gambar lapangan).
9.	Tests	File pengujian otomatis	Digunakan untuk unit testing dan memastikan aplikasi berjalan sesuai harapan.
10.	Vendor	Library hasil instalasi Composer	Berisi dependensi pihak ketiga yang digunakan Laravel.
11.	.env	File konfigurasi environment	Menyimpan pengaturan rahasia seperti database, API key, dan konfigurasi server.
12.	Artisan	File CLI Laravel	Digunakan untuk menjalankan perintah Laravel (migrasi, seeding, dll.).
13.	composer.json	File konfigurasi Composer	Menentukan library/dependensi yang digunakan dalam proyek.
14.	README.md	Dokumentasi proyek	Penjelasan singkat tentang proyek, cara instalasi, dan penggunaan.

5.1.2. Migration

Migration adalah proses mengelola versi skema basis data, yang memungkinkan pengembang untuk membuat perubahan pada skema basis data dengan cara yang terstruktur dan terkelola. Ini adalah pendekatan yang lebih sistematis daripada mengubah skema secara langsung di server basis data.

```

1  <?php
2
3  use Illuminate\Database\Migrations\Migration;
4  use Illuminate\Database\Schema\Blueprint;
5  use Illuminate\Support\Facades\Schema;
6
7  return new class extends Migration
8  {
9      /**
10       * Run the migrations.
11       */
12     public function up(): void
13     {
14         Schema::create('users', function (Blueprint $table) {
15             $table->id();
16             $table->string('nama_lengkap');
17             $table->string('username')->unique();
18             $table->string('email')->unique();
19             $table->string('no_telp')->unique();
20             $table->string('password');
21             $table->enum('role', ['penyewa', 'petugas'])->default('penyewa');
22             $table->rememberToken();
23             $table->timestamps();
24         });
25     }
26
27     /**
28      * Reverse the migrations.
29      */
30     public function down(): void
31     {
32         Schema::dropIfExists('users');
33     }
34 };
35

```

Gambar 5. 1 Migrasi users mendefinisikan tabel identitas pengguna.

```

1  <?php
2
3  use Illuminate\Database\Migrations\Migration;
4  use Illuminate\Database\Schema\Blueprint;
5  use Illuminate\Support\Facades\Schema;
6
7  return new class extends Migration
8  {
9      /**
10       * Run the migrations.
11       */
12     public function up(): void
13     {
14         Schema::create('lapangans', function (Blueprint $table) {
15             $table->id();
16             $table->string('nama_lapangan');
17             $table->enum('jenis', ['futsal', 'basket', 'badminton', 'tenis']);
18             $table->decimal('harga_per_jam', 10, 2);
19             $table->text('deskripsi')->nullable();
20             $table->json('gambar')->nullable();
21             $table->boolean('is_active')->default(true);
22             $table->timestamps();
23         });
24     }
25
26     /**
27      * Reverse the migrations.
28      */
29     public function down(): void
30     {
31         Schema::dropIfExists('lapangans');
32     }
33 };
34

```

Gambar 5. 2 Migrasi lapangans mendefinisikan tabel data lapangan olahraga.

```

1  <?php
2
3  use Illuminate\Database\Migrations\Migration;
4  use Illuminate\Database\Schema\Blueprint;
5  use Illuminate\Support\Facades\DB;
6  use Illuminate\Support\Facades\Schema;
7
8  return new class extends Migration
9  {
10     /**
11      * Run the migrations.
12      */
13     public function up(): void
14     {
15         Schema::create('jadwals', function (Blueprint $table) {
16             $table->id();
17             $table->time('jam')->unique();
18             $table->boolean('is_active')->default(true);
19             $table->timestamps();
20         });
21     }
22
23     /**
24      * Reverse the migrations.
25      */
26     public function down(): void
27     {
28         Schema::dropIfExists('jadwals');
29     }
30 };
31

```

Gambar 5. 3 Migrasi jadwal mendefinisikan tabel jam pemesanan.

```

1  <?php
2
3  use Illuminate\Database\Migrations\Migration;
4  use Illuminate\Database\Schema\Blueprint;
5  use Illuminate\Support\Facades\Schema;
6
7  return new class extends Migration
8  {
9      /**
10       * Run the migrations.
11       */
12     public function up(): void
13     {
14         Schema::create('pemesanans', function (Blueprint $table) {
15             $table->id();
16             $table->foreignId('user_id')->constrained('users')->onDelete('cascade');
17             $table->foreignId('lapangan_id')->constrained('lapangans')->onDelete('cascade');
18             $table->json('jadwal'); // menyimpan jam yang dipilih misal ["08:00", "09:00"] berarti nanti dijadwalkan di jam 08:00 dan 09:00 dan jadi 2 jam di depannya
19             $table->date('tanggal_pemesanan');
20             $table->decimal('total_harga', 10, 2);
21             $table->enum('status', ['pending', 'sukses', 'gagal'])->default('pending');
22             $table->timestamps();
23         });
24     }
25
26     /**
27      * Reverse the migrations.
28      */
29     public function down(): void
30     {
31         Schema::dropIfExists('pemesanans');
32     }
33 };
34

```

Gambar 5. 4 Migrasi pemesanan mendefinisikan tabel pemesanan lapangan.

```

1  <?php
2
3  use Illuminate\Database\Migrations\Migration;
4  use Illuminate\Database\Schema\Blueprint;
5  use Illuminate\Support\Facades\Schema;
6
7  return new class extends Migration
8  {
9      /**
10       * Run the migrations.
11       */
12       public function up(): void
13       {
14           Schema::create('pembayarans', function (Blueprint $table) {
15               $table->id();
16               $table->foreignId('pemesanan_id')->constrained('pemesanans')->onDelete('cascade');
17               $table->string('metode')->nullable(); // metode pembayaran
18               $table->string('bukti_transfer')->nullable(); // gambar yang diupload
19               $table->dateTime('tanggal_bayar')->nullable();
20               $table->enum('status', ['pending', 'valid', 'invalid'])->default('pending');
21               $table->timestamps();
22           });
23       }
24
25       /**
26        * Reverse the migrations.
27        */
28       public function down(): void
29       {
30           Schema::dropIfExists('pembayarans');
31       }
32   };
33

```

Gambar 5. 5 Migrasi pembayarans mendefinisikan tabel pembayaran pemesanan.

5.1.3 Seeder

Seeder adalah proses mengisi data awal ke dalam basis data, yang memungkinkan pengembang untuk mempopulasikan basis data dengan data default atau data sampel yang diperlukan untuk pengujian dan pengembangan.

```

1  <?php
2
3  namespace Database\Seeders;
4
5  // use Illuminate\Database\Console\Seeds\WithoutModelEvents;
6  use Illuminate\Database\Seeder;
7
8  class DatabaseSeeder extends Seeder
9  {
10     /**
11      * Seed the application's database.
12      */
13     public function run(): void
14     {
15         $this->call([
16             UserSeeder::class,
17             JadwalSeeder::class
18         ]);
19     }
20 }
21

```

Gambar 5. 6 DatabaseSeeder mengisi data awal aplikasi.

```

1  <?php
2
3  namespace Database\Seeders;
4
5  use Illuminate\Database\Console\Seeds\WithoutModelEvents;
6  use Illuminate\Database\Seeder;
7  use Illuminate\Support\Facades\DB;
8
9  class JadwalSeeder extends Seeder
10 {
11     /**
12      * Run the database seeds.
13      */
14     public function run()
15     {
16         $data = [];
17
18         // Loop dari jam 00:00 sampai 23:00
19         for ($i = 0; $i < 24; $i++) {
20             $jam = str_pad($i, 2, '0', STR_PAD_LEFT) . ':00';
21
22             $data[] = [
23                 'jam' => $jam,
24                 'is_active' => true,
25                 'created_at' => now(),
26                 'updated_at' => now()
27             ];
28         }
29
30         // Insert ke database
31         DB::table('jadwals')->insert($data);
32     }
33 }
34

```

Gambar 5. 7 JadwalSeeder mengisi tabel jdwals dengan jam otomatis.

```

1  <?php
2
3  namespace Database\Seeders;
4
5  use Illuminate\Database\Console\Seeds\WithoutModelEvents;
6  use Illuminate\Database\Seeder;
7  use App\Models\User;
8  use Illuminate\Support\Facades\Hash;
9
10 class UserSeeder extends Seeder
11 {
12     /**
13      * Run the database seeds.
14      */
15     public function run(): void
16     {
17         // Admin/Petugas Users
18         User::create([
19             'nama_lengkap' => 'Administrator GOR',
20             'username' => 'admin',
21             'email' => 'admin@mail.com',
22             'no_telp' => '081234567890',
23             'password' => Hash::make('admin123123'),
24             'role' => 'petugas',
25         ]);
26
27         // Penyewa Users
28         User::create([
29             'nama_lengkap' => 'Nizar Rahman',
30             'username' => 'nizar',
31             'email' => 'nizar@mail.com',
32             'no_telp' => '082111111111',
33             'password' => Hash::make('nizar123123'),
34             'role' => 'penyewa',
35         ]);
36     }
37 }

```

Gambar 5. 8 UserSeeder mengisi tabel users dengan akun petugas dan penyewa.

5.1.4 Model

Komponen model berhubungan dengan seluruh data logis (data-related logic) yang digunakan pengguna. Data logis adalah data yang relevan dengan konteks tertentu. Misalnya ketika ingin membuat database pegawai, maka data yang dikumpulkan berupa nama pegawai, tanggal lahir, tempat lahir, alamat, dan nomor kontak. Komponen model dapat mewakili data apa saja yang sedang ditransfer antara komponen view dan controller, atau data logis lainnya.

```

1  <?php
2
3  namespace App\Models;
4
5  use Illuminate\Database\Eloquent\Factories\HasFactory;
6  use Illuminate\Database\Eloquent\Model;
7
8  class Jadwal extends Model
9  {
10     use HasFactory;
11
12     protected $guarded = ['id'];
13 }

```

Gambar 5. 9 Model Jadwal merepresentasikan tabel jadwals.

```

1  <?php
2
3  namespace App\Models;
4
5  use Illuminate\Database\Eloquent\Factories\HasFactory;
6  use Illuminate\Database\Eloquent\Model;
7  use Illuminate\Support\Facades\Storage;
8
9  class Lapangan extends Model
10 {
11     use HasFactory;
12
13     protected $fillable = [
14         'nama_lapangan',
15         'jenis',
16         'harga_per_jam',
17         'deskripsi',
18         'gambar',
19         'is_active'
20     ];
21
22     protected $casts = [
23         'gambar' => 'array',
24         'harga_per_jam' => 'decimal:2',
25         'is_active' => 'boolean'
26     ];
27
28     // Relationship
29     public function jadwals()
30     {
31         return $this->hasMany(Jadwal::class);
32     }
33
34     public function jadwalTersedia()
35     {
36         return $this->hasMany(Jadwal::class)->where('status', 'tersedia');
37     }
38
39     // Accessors
40     public function getGambarUrlsAttribute()
41     {
42         if (!$this->gambar) {
43             return [];
44         }
45
46         return collect($this->gambar)->map(function ($gambar) {
47             return Storage::url($gambar);
48         }->toArray();
49     }
50
51     public function getGambarUtamaAttribute()
52     {
53         if (!$this->gambar || empty($this->gambar)) {
54             return asset('images/default-field.jpg');
55         }
56     }
57 }

```

Gambar 5. 10 Model Lapangan merepresentasikan tabel lapangans.

```

56 |
57 |     return Storage::url($this->gambar[0]);
58 | }
59 |
60 | public function getFormattedHargaAttribute()
61 | {
62 |     return 'Rp ' . number_format($this->harga_per_jam, 0, ',', '.');
63 | }
64 |
65 | public function getJenisLabelAttribute()
66 | {
67 |     $labels = [
68 |         'futsal' => 'Futsal',
69 |         'basket' => 'Basket',
70 |         'badminton' => 'Badminton',
71 |         'tenis' => 'Tenis'
72 |     ];
73 |
74 |     return $labels[$this->jenis] ?? ucfirst($this->jenis);
75 | }
76 |
77 | // Scopes
78 | public function scopeActive($query)
79 | {
80 |     return $query->where('is_active', true);
81 | }
82 |
83 | public function scopeByJenis($query, $jenis)
84 | {
85 |     return $query->where('jenis', $jenis);
86 | }
87 |
88 | public function scopeSearch($query, $search)
89 | {
90 |     return $query->where(function ($q) use ($search) {
91 |         $q->where('nama_lapangan', 'like', "%{$search}%")
92 |         ->orWhere('jenis', 'like', "%{$search}%")
93 |         ->orWhere('deskripsi', 'like', "%{$search}%");
94 |     });
95 | }
96 |
97 | // Methods
98 | public function deleteGambar()
99 | {
100 |     if ($this->gambar) {
101 |         foreach ($this->gambar as $gambar) {
102 |             Storage::delete($gambar);
103 |         }
104 |     }
105 | }
106 |
107 | public function isAvailableAt($tanggal, $jam_mulai, $jam_selesai)
108 | {

```

Gambar 5. 11 Model Lapangan merepresentasikan tabel lapangans.

```

109 |     return !$this->jadwals()
110 |         ->where('tanggal', $tanggal)
111 |         ->where('status', 'dipesan')
112 |         ->where(function ($query) use ($jam_mulai, $jam_selesai) {
113 |             $query->where(function ($q) use ($jam_mulai, $jam_selesai) {
114 |                 $q->where('jam_mulai', '<', $jam_selesai)
115 |                 ->where('jam_selesai', '>', $jam_mulai);
116 |             });
117 |         });
118 |     ->exists();
119 | }
120 |
121 | public function pemesanan()
122 | {
123 |     return $this->hasMany(Pemesanan::class);
124 | }
125 | }

```

Gambar 5. 12 Model Lapangan merepresentasikan tabel lapangans.

```

1  <?php
2
3  namespace App\Models;
4
5  use Illuminate\Database\Eloquent\Factories\HasFactory;
6  use Illuminate\Database\Eloquent\Model;
7  use Illuminate\Support\Facades\Storage;
8
9  class Pembayaran extends Model
10 {
11     use HasFactory;
12
13     protected $guarded = ['id'];
14
15     protected $casts = [
16         'tanggal_bayar' => 'datetime'
17     ];
18
19     // Relationships
20     public function pemesanan()
21     {
22         return $this->belongsTo(Pemesanan::class, 'pemesanan_id');
23     }
24
25     // Accessors
26     public function getBuktiTransferUrlAttribute()
27     {
28         if ($this->bukti_transfer) {
29             return Storage::url($this->bukti_transfer);
30         }
31         return null;
32     }
33
34     public function getStatusBadgeAttribute()
35     {
36         $badges = [
37             'pending' => [
38                 'class' => 'bg-yellow-100 text-yellow-800',
39                 'icon' => 'fa-clock',
40                 'text' => 'Pending'
41             ],
42             'valid' => [
43                 'class' => 'bg-green-100 text-green-800',
44                 'icon' => 'fa-check-circle',
45                 'text' => 'Valid'
46             ],
47             'invalid' => [
48                 'class' => 'bg-red-100 text-red-800',
49                 'icon' => 'fa-times-circle',
50                 'text' => 'Invalid'
51             ]
52         ];
53
54         return $badges[$this->status] ?? $badges['pending'];
55     }
56 }

```

Gambar 5. 13 Model Pembayaran merepresentasikan tabel pembayarans.

```

56
57     public function getMetodeFormattedAttribute()
58     {
59         $methods = [
60             'Transfer Bank' => 'Transfer Bank',
61             'Tunai' => 'Tunai',
62             'E-Wallet' => 'E-Wallet'
63         ];
64
65         return $methods[$this->metode] ?? $this->metode;
66     }
67
68     // Scopes
69     public function scopePending($query)
70     {
71         return $query->where('status', 'pending');
72     }
73
74     public function scopeValid($query)
75     {
76         return $query->where('status', 'valid');
77     }
78
79     public function scopeInvalid($query)
80     {
81         return $query->where('status', 'invalid');
82     }
83
84     public function scopeWithProof($query)
85     {
86         return $query->whereNotNull('bukti_transfer');
87     }
88
89     // Methods
90     public function hasBuktiTransfer()
91     {
92         return !empty($this->bukti_transfer);
93     }
94
95     public function isPending()
96     {
97         return $this->status === 'pending';
98     }
99
100    public function isValid()
101    {
102        return $this->status === 'valid';
103    }
104
105    public function isInvalid()
106    {
107        return $this->status === 'invalid';
108    }
109 }

```

Gambar 5. 14 Model Pembayaran merepresentasikan tabel pembayarans.

```

1  <?php
2
3  namespace App\Models;
4
5  use Illuminate\Database\Eloquent\Factories\HasFactory;
6  use Illuminate\Database\Eloquent\Model;
7  use Carbon\Carbon;
8
9  class Pemesanan extends Model
10 {
11     use HasFactory;
12
13     protected $guarded = ['id'];
14
15     protected $casts = [
16         'jadwal' => 'array',
17         'tanggal_pemesanan' => 'date',
18         'total_harga' => 'decimal:2'
19     ];
20
21     // Relationships
22     public function user()
23     {
24         return $this->belongsTo(User::class);
25     }
26
27     public function lapangan()
28     {
29         return $this->belongsTo(Lapangan::class);
30     }
31
32     public function pembayaran()
33     {
34         return $this->hasOne(Pembayaran::class, 'pemesanan_id');
35     }
36
37     // Accessors
38     public function getJadwalFormattedAttribute()
39     {
40         if (is_string($this->jadwal)) {
41             $jadwal = json_decode($this->jadwal, true);
42         } else {
43             $jadwal = $this->jadwal;
44         }
45
46         if (!$jadwal || !is_array($jadwal)) {
47             return [];
48         }
49
50         return array_map(function($jam) {
51             return Carbon::parse($jam)->format('H:i');
52         }, $jadwal);
53     }
54 }

```

Gambar 5. 15 Model Pemesanan merepresentasikan tabel pemesanan.

```

55     public function getStatusBadgeAttribute()
56     {
57         $badges = [
58             'pending' => [
59                 'class' => 'bg-yellow-100 text-yellow-800',
60                 'icon' => 'fa-clock',
61                 'text' => 'Pending'
62             ],
63             'sukses' => [
64                 'class' => 'bg-green-100 text-green-800',
65                 'icon' => 'fa-check-circle',
66                 'text' => 'Sukses'
67             ],
68             'gagal' => [
69                 'class' => 'bg-red-100 text-red-800',
70                 'icon' => 'fa-times-circle',
71                 'text' => 'Gagal'
72             ]
73         ];
74
75         return $badges[$this->status] ?? $badges['pending'];
76     }
77
78     public function getTotalHargaFormattedAttribute()
79     {
80         return 'Rp ' . number_format($this->total_harga, 0, ',', '.');
81     }
82
83     // Scopes
84     public function scopePending($query)
85     {
86         return $query->where('status', 'pending');
87     }
88
89     public function scopeSukses($query)
90     {
91         return $query->where('status', 'sukses');
92     }
93
94     public function scopeGagal($query)
95     {
96         return $query->where('status', 'gagal');
97     }
98
99     public function scopeForUser($query, $userId)
100    {
101        return $query->where('user_id', $userId);
102    }
103
104    public function scopeForLapangan($query, $lapanganId)
105    {
106        return $query->where('lapangan_id', $lapanganId);
107    }
108

```

Gambar 5. 16 Model Pemesanan merepresentasikan tabel pemesanan.

```

109     public function scopeForDate($query, $date)
110     {
111         return $query->whereDate('tanggal_pemesanan', $date);
112     }
113
114     // Methods
115     public function canBeUpdated()
116     {
117         return $this->status === 'pending';
118     }
119
120     public function canBeCanceled()
121     {
122         return $this->status === 'pending';
123     }
124
125     public function needsPaymentProof()
126     {
127         return $this->status === 'pending' &&
128             $this->pembayaran &&
129             $this->pembayaran->status === 'pending' &&
130             !$this->pembayaran->bukti_transfer;
131     }
132
133     public function canBeValidated()
134     {
135         return $this->pembayaran &&
136             $this->pembayaran->bukti_transfer &&
137             $this->pembayaran->status === 'pending';
138     }
139 }

```

Gambar 5. 17 Model Pemesanan merepresentasikan tabel pemesanan.

```

1  <?php
2
3  namespace App\Models;
4
5  use Illuminate\Contracts\Auth\MustVerifyEmail;
6  use Illuminate\Database\Eloquent\Factories\HasFactory;
7  use Illuminate\Foundation\Auth\User as Authenticatable;
8  use Illuminate\Notifications\Notifiable;
9  use Laravel\Sanctum\HasApiTokens;
10
11 class User extends Authenticatable
12 {
13     use HasApiTokens, HasFactory, Notifiable;
14
15     /**
16      * The attributes that are mass assignable.
17      *
18      * @var array<int, string>
19      */
20     protected $guarded = ['id'];
21
22     /**
23      * The attributes that should be hidden for serialization.
24      *
25      * @var array<int, string>
26      */
27     protected $hidden = [
28         'password',
29         'remember_token',
30     ];
31
32     /**
33      * The attributes that should be cast.
34      *
35      * @var array<string, string>
36      */
37     protected $casts = [
38         'email_verified_at' => 'datetime',
39     ];
40
41     public function pemesanan()
42     {
43         return $this->hasMany(Pemesanan::class);
44     }
45 }

```

Gambar 5. 18 Model User merepresentasikan tabel users.

5.1.5 View

Komponen view bertanggung jawab untuk membuat tampilan muka (UI/user interface) pada seluruh situs atau aplikasi. Komponen ini dibuat dari data yang dikumpulkan oleh model, dan diberikan kepada view melalui komponen controller. Namun pada beberapa kasus, view bisa berinteraksi langsung dengan model tanpa bantuan controller.

```

1  <!DOCTYPE html>
2  <html lang="id">
3  <head>
4      <meta charset="UTF-8" />
5      <meta name="viewport" content="width=device-width, initial-scale=1.0"/>
6      <title>Login - GOR EWANGGA</title>
7      <script src="https://cdn.tailwindcss.com"></script>
8      <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.4.0/css/all.min.css">
9  </head>
10 <body class="bg-gray-100 font-sans">
11     <div class="min-h-screen flex items-center justify-center py-12 px-4 sm:px-6 lg:px-8">
12         <div class="max-w-md w-full space-y-8">
13             <!-- Header -->
14             <div class="text-center">
15                 <div class="mx-auto h-20 w-20 flex items-center justify-center rounded-full bg-teal-100 mb-4">
16                     <i class="fas fa-running text-3xl text-teal-600"></i>
17                 </div>
18                 <h2 class="text-3xl font-extrabold text-gray-900">GOR Ewangga</h2>
19                 <p class="mt-2 text-sm text-gray-600">Masuk ke akun Anda</p>
20             </div>
21
22             <!-- Login Form -->
23             <div class="bg-white rounded-lg shadow-md p-8">
24                 @if (session('success'))
25                     <div class="mb-4 p-4 text-green-700 bg-green-100 border border-green-300 rounded-md">
26                         {{ session('success') }}
27                     </div>
28                 @endif
29
30                 @if ($errors->any())
31                     <div class="mb-4 p-4 text-red-700 bg-red-100 border border-red-300 rounded-md">
32                         <ul class="list-disc list-inside text-sm">
33                             @foreach ($errors->all() as $error)
34                                 <li>{{ $error }}</li>
35                             @endforeach
36                         </ul>
37                     </div>
38                 @endif
39
40                 <form method="POST" action="{{ route('login') }}" class="space-y-6">
41                     @csrf
42
43                     <div>
44                         <label for="login" class="block text-sm font-medium text-gray-700 mb-1">
45                             Email atau Username
46                         </label>
47                         <div class="relative">
48                             <div class="absolute inset-y-0 left-0 pl-3 flex items-center pointer-events-none">
49                                 <i class="fas fa-user text-gray-400"></i>
50                             </div>
51                             <input
52                                 id="login"
53                                 name="login"
54                                 type="text"
55                                 autocomplete="username"

```

Gambar 5. 19 Halaman login untuk pengguna melalui username dan password.

```

16         required
17         class="block w-full pl-10 pr-3 py-2 border border-gray-300 rounded-md placeholder-gray-400 focus:outline-none focus:ring-2 focus:ring-teal-500 focus:border-teal-500"
18         placeholder="Masukkan email atau username"
19         value="{{ old('login') }}"
20     )
21 </div>
22
23 <div>
24
25     <label for="password" class="block text-sm font-medium text-gray-700 mb-1">
26         Password
27     </label>
28     <div class="relative">
29         <div class="absolute inset-y-0 left-0 pl-3 flex items-center pointer-events-none">
30             <i class="fas fa-lock text-gray-400"></i>
31         </div>
32         <input
33             id="password"
34             name="password"
35             type="password"
36             autocomplete="current-password"
37             required
38             class="block w-full pl-10 pr-10 py-2 border border-gray-300 rounded-md placeholder-gray-400 focus:outline-none focus:ring-2 focus:ring-teal-500 focus:border-teal-500"
39             placeholder="Masukkan password"
40         >
41         <button type="button" onclick="togglePassword('password')" class="absolute inset-y-0 right-0 pr-3 flex items-center">
42             <i class="password-icon" class="fas fa-eye text-gray-400 hover:text-gray-600"></i>
43         </button>
44     </div>
45
46 </div>
47
48 <div class="flex items-center justify-between">
49     <div class="flex items-center">
50         <input
51             id="remember"
52             name="remember"
53             type="checkbox" class="h-4 w-4 text-teal-600 focus:ring-teal-500 border-gray-300 rounded"
54         >
55         <label for="remember" class="ml-2 block text-sm text-gray-700">
56             Ingat saya
57         </label>
58     </div>
59
60 </div>
61
62 <div>
63     <button
64         type="submit"
65         class="w-full flex items-center justify-center py-2 px-4 border border-transparent rounded-md shadow-sm text-sm font-medium text-white bg-teal-600 hover:bg-teal-700 focus:outline-none focus:ring-2 focus:ring-offset-2 focus:ring-teal-500"
66     >
67         <i class="fas fa-sign-in-alt mr-2"></i>
68         Masuk
69     </button>
70 </div>
71
72 <div class="text-center">
73     <p class="text-sm text-gray-600">
74         Belum punya akun?
75         <a href="{{ route('register') }}" class="text-teal-600 hover:text-teal-500 font-medium">
76             Daftar sekarang
77         </a>
78     </p>
79 </div>
80 </form>
81 </div>
82 </div>
83
84 <script>
85     function togglePassword(fieldId) {
86         const field = document.getElementById(fieldId);
87         const icon = document.getElementById(fieldId + '-icon');
88
89         if (field.type === 'password') {
90             field.type = 'text';
91             icon.classList.remove('fa-eye');
92             icon.classList.add('fa-eye-slash');
93         } else {
94             field.type = 'password';
95             icon.classList.remove('fa-eye-slash');
96             icon.classList.add('fa-eye');
97         }
98     }
99 </script>
100 </body>
101 </html>

```

Gambar 5. 20 Halaman login untuk pengguna melalui username dan password.

```

105 >
106     <i class="fas fa-sign-in-alt mr-2"></i>
107     Masuk
108 </button>
109 </div>
110
111 <div class="text-center">
112     <p class="text-sm text-gray-600">
113         Belum punya akun?
114         <a href="{{ route('register') }}" class="text-teal-600 hover:text-teal-500 font-medium">
115             Daftar sekarang
116         </a>
117     </p>
118 </div>
119 </form>
120 </div>
121 </div>
122 </div>
123
124 <script>
125     function togglePassword(fieldId) {
126         const field = document.getElementById(fieldId);
127         const icon = document.getElementById(fieldId + '-icon');
128
129         if (field.type === 'password') {
130             field.type = 'text';
131             icon.classList.remove('fa-eye');
132             icon.classList.add('fa-eye-slash');
133         } else {
134             field.type = 'password';
135             icon.classList.remove('fa-eye-slash');
136             icon.classList.add('fa-eye');
137         }
138     }
139 </script>
140 </body>
141 </html>

```

Gambar 5. 21 Halaman login untuk pengguna melalui username dan password.

```

1 <!DOCTYPE html>
2 <html lang="id">
3 <head>
4   <meta charset="UTF-8" />
5   <meta name="viewport" content="width=device-width, initial-scale=1.0"/>
6   <title>Register - GOR Ewangga</title>
7   <script src="https://cdn.tailwindcss.com"></script>
8   <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.4.0/css/all.min.css">
9 </head>
10 <body class="bg-gray-100 font-sans">
11   <div class="min-h-screen flex items-center justify-center py-12 px-4 sm:px-6 lg:px-8">
12     <div class="max-w-md w-full space-y-8">
13       <!-- Header -->
14       <div class="text-center">
15         <div class="mx-auto h-20 w-20 flex items-center justify-center rounded-full bg-teal-100 mb-4">
16           <i class="fas fa-running text-3xl text-teal-600"></i>
17         </div>
18         <h2 class="text-3xl font-extrabold text-gray-900">GOR Ewangga</h2>
19         <p class="mt-2 text-sm text-gray-600">Buat akun baru</p>
20       </div>
21
22       <!-- Register Form -->
23       <div class="bg-white rounded-lg shadow-md p-8">
24         <!-- Errors -->
25         <div class="mb-4 p-4 text-red-700 bg-red-100 border border-red-300 rounded-md">
26           <ul class="list-disc list-inside text-sm">
27             <li><!-- Errors --></li>
28             <li><!-- Errors --></li>
29           </ul>
30         </div>
31
32         <!-- Register Form -->
33         <form method="POST" action="{{ route('register') }}" class="space-y-4">
34           <@csrf>
35
36           <div>
37             <label for="nama_lengkap" class="block text-sm font-medium text-gray-700 mb-1">
38               Nama lengkap
39             </label>
40             <div class="relative">
41               <div class="absolute inset-y-0 left-0 pl-3 flex items-center pointer-events-none">
42                 <i class="fas fa-user text-gray-400"></i>
43               </div>
44               <input
45                 id="nama_lengkap"
46                 name="nama_lengkap"
47                 type="text"
48                 required
49                 class="block w-full pl-10 pr-3 py-2 border border-gray-300 rounded-md placeholder-gray-400 focus:outline-none focus:ring-2 focus:ring-teal-500 focus:border-teal-500"
50                 placeholder="Masukkan nama lengkap"
51                 value="{{ old('nama_lengkap') }}"
52               >
53             </div>
54           </div>
55
56

```

Gambar 5. 22 Halaman register untuk pembuatan akun baru.

```

56
57
58   <div>
59     <label for="username" class="block text-sm font-medium text-gray-700 mb-1">
60       Username
61     </label>
62     <div class="relative">
63       <div class="absolute inset-y-0 left-0 pl-3 flex items-center pointer-events-none">
64         <i class="fas fa-at text-gray-400"></i>
65       </div>
66       <input
67         id="username"
68         name="username"
69         type="text"
70         required
71         class="block w-full pl-10 pr-3 py-2 border border-gray-300 rounded-md placeholder-gray-400 focus:outline-none focus:ring-2 focus:ring-teal-500 focus:border-teal-500"
72         placeholder="Masukkan username"
73         value="{{ old('username') }}"
74       >
75     </div>
76
77     <div>
78       <label for="email" class="block text-sm font-medium text-gray-700 mb-1">
79         Email
80       </label>
81       <div class="relative">
82         <div class="absolute inset-y-0 left-0 pl-3 flex items-center pointer-events-none">
83           <i class="fas fa-envelope text-gray-400"></i>
84         </div>
85         <input
86           id="email"
87           name="email"
88           type="email"
89           autocomplete="email"
90           required
91           class="block w-full pl-10 pr-3 py-2 border border-gray-300 rounded-md placeholder-gray-400 focus:outline-none focus:ring-2 focus:ring-teal-500 focus:border-teal-500"
92           placeholder="Masukkan email"
93           value="{{ old('email') }}"
94         >
95       </div>
96     </div>
97
98     <div>
99       <label for="no_telp" class="block text-sm font-medium text-gray-700 mb-1">
100         Nomor Telepon
101       </label>
102       <div class="relative">
103         <div class="absolute inset-y-0 left-0 pl-3 flex items-center pointer-events-none">
104           <i class="fas fa-phone text-gray-400"></i>
105         </div>

```

Gambar 5. 23 Halaman register untuk pembuatan akun baru.

```

106 |         <input
107 |           id="no_telp"
108 |           name="no_telp"
109 |           type="tel"
110 |           required
111 |           class="block w-full pl-10 pr-3 py-2 border border-gray-300 rounded-md placeholder-gray-400 focus:outline-none focus:ring-2 focus:ring-teal-500 focus:border-teal-500"
112 |           placeholder="Masukkan nomor telepon"
113 |           value="{{ old('no_telp') }}"
114 |         >
115 |       </div>
116 |     </div>
117 |
118 |     <div>
119 |       <label for="password" class="block text-sm font-medium text-gray-700 mb-1">
120 |         Password
121 |       </label>
122 |       <div class="relative">
123 |         <div class="absolute inset-y-0 left-0 pl-3 flex items-center pointer-events-none">
124 |           <i class="fas fa-lock text-gray-400"></i>
125 |         </div>
126 |         <input
127 |           id="password"
128 |           name="password"
129 |           type="password"
130 |           autocomplete="new-password"
131 |           required
132 |           class="block w-full pl-10 pr-10 py-2 border border-gray-300 rounded-md placeholder-gray-400 focus:outline-none focus:ring-2 focus:ring-teal-500 focus:border-teal-500"
133 |           placeholder="Masukkan password (min. 8 karakter)"
134 |         >
135 |         <button type="button" onclick="togglePassword('password')" class="absolute inset-y-0 right-0 pr-3 flex items-center">
136 |           <i id="password-icon" class="fas fa-eye text-gray-400 hover:text-gray-600"></i>
137 |         </button>
138 |       </div>
139 |     </div>
140 |
141 |     <div>
142 |       <label for="password_confirmation" class="block text-sm font-medium text-gray-700 mb-1">
143 |         Konfirmasi Password
144 |       </label>
145 |       <div class="relative">
146 |         <div class="absolute inset-y-0 left-0 pl-3 flex items-center pointer-events-none">
147 |           <i class="fas fa-lock text-gray-400"></i>
148 |         </div>
149 |         <input
150 |           id="password_confirmation"
151 |           name="password_confirmation"
152 |           type="password"
153 |           autocomplete="new-password"
154 |           required
155 |           class="block w-full pl-10 pr-10 py-2 border border-gray-300 rounded-md placeholder-gray-400 focus:outline-none focus:ring-2 focus:ring-teal-500 focus:border-teal-500"
156 |         >

```

Gambar 5. 24 Halaman register untuk pembuatan akun baru.

```

156 |         placeholder="Masukkan password"
157 |       >
158 |       <button type="button" onclick="togglePassword('password_confirmation')" class="absolute inset-y-0 right-0 pr-3 flex items-center">
159 |         <i id="password_confirmation-icon" class="fas fa-eye text-gray-400 hover:text-gray-600"></i>
160 |       </button>
161 |     </div>
162 |   </div>
163 |
164 |   <div class="flex items-center">
165 |     <input
166 |       id="terms"
167 |       name="terms"
168 |       type="checkbox"
169 |       required
170 |       class="h-4 w-4 text-teal-600 focus:ring-teal-500 border-gray-300 rounded"
171 |     >
172 |     <label for="terms" class="ml-2 block text-sm text-gray-700">
173 |       Saya setuju dengan <a href="#">"Kebijakan Privasi"</a> dan <a href="#">"Syarat dan ketentuan"</a>
174 |     </label>
175 |   </div>
176 |
177 |   <div class="pt-2">
178 |     <button
179 |       type="submit"
180 |       class="w-full flex items-center justify-center py-2 px-4 border border-transparent rounded-md shadow-sm text-sm font-medium text-white bg-teal-600 hover:bg-teal-700 focus:outline-none focus:ring-2 focus:ring-offset-2 focus:ring-teal-500"
181 |     >
182 |       <i class="fas fa-user-plus w-2"></i>
183 |     </button>
184 |     <button
185 |       type="button"
186 |       class="w-full flex items-center justify-center py-2 px-4 border border-transparent rounded-md shadow-sm text-sm font-medium text-gray-500 hover:text-gray-600"
187 |     >
188 |       Sudah punya akun?
189 |       <a href="{{ route('login') }}" class="text-teal-600 hover:text-teal-500 font-medium">
190 |         Masuk sekarang
191 |       </a>
192 |     </button>
193 |   </div>
194 |
195 |   </div>
196 | </div>
197 | </div>
198 |
199 | <script>
200 |   function togglePassword(fieldId) {
201 |     const field = document.getElementById(fieldId);
202 |     const icon = document.getElementById(fieldId + "-icon");
203 |     if (field.type === "password") {
204 |       field.type = "text";
205 |       icon.classList.remove('fa-eye-slash');
206 |       icon.classList.add('fa-eye');
207 |     } else {
208 |       field.type = "password";
209 |       icon.classList.remove('fa-eye');
210 |       icon.classList.add('fa-eye-slash');
211 |     }
212 |   }
213 |
214 | </script>
215 | </body>
216 | </html>

```

Gambar 5. 25 Halaman register untuk pembuatan akun baru.

```

206 |         field.type = 'text';
207 |         icon.classList.remove('fa-eye-slash');
208 |         icon.classList.add('fa-eye');
209 |       } else {
210 |         field.type = 'password';
211 |         icon.classList.remove('fa-eye-slash');
212 |         icon.classList.add('fa-eye');
213 |       }
214 |     }
215 |   }
216 | </script>
217 | </body>
218 | </html>

```

Gambar 5. 26 Halaman register untuk pembuatan akun baru.

```

1  @extends('layout.main')
2  @section('content')
3  <main class="flex-1 overflow-y-auto p-4">
4    <!-- Overview Title -->
5    <h2 class="text-xl font-semibold text-gray-800 mb-6">Dashboard Overview</h2>
6
7    <!-- Cards -->
8    <div class="grid grid-cols-1 md:grid-cols-2 lg:grid-cols-4 gap-6 mb-8">
9      <!-- Card 1: Total Users -->
10     <div class="bg-white rounded-lg shadow p-6 flex flex-col justify-between">
11       <div class="flex items-center justify-between">
12         <div>
13           <p class="text-sm text-gray-500">Total Penyewa</p>
14           <p class="text-2xl font-semibold text-gray-800">{{ number_format($stats['total_users'] ?? 0) }}</p>
15         </div>
16         <div class="w-[50px] h-[50px] flex justify-center items-center rounded-full bg-teal-100 text-teal-600">
17           <i class="fas fa-users"></i>
18         </div>
19       </div>
20       <div class="mt-4 text-sm">
21         <span class="text-green-500 font-medium">+12.5%</span>
22         <span class="text-gray-500 ml-1">dari bulan lalu</span>
23       </div>
24     </div>
25
26     <!-- Card 2: Total Pemesanan -->
27     <div class="bg-white rounded-lg shadow p-6 flex flex-col justify-between">
28       <div class="flex items-center justify-between">
29         <div>
30           <p class="text-sm text-gray-500">Total Pemesanan</p>
31           <p class="text-2xl font-semibold text-gray-800">{{ number_format($stats['total_pemesanan'] ?? 0) }}</p>
32         </div>
33         <div class="w-[50px] h-[50px] flex justify-center items-center rounded-full bg-green-100 text-green-600">
34           <i class="fas fa-calendar-check"></i>
35         </div>
36       </div>
37       <div class="mt-4 text-sm">
38         <span class="text-green-500 font-medium">+8.1%</span>
39         <span class="text-gray-500 ml-1">dari minggu lalu</span>
40       </div>
41     </div>
42
43     <!-- Card 3: Revenue -->
44     <div class="bg-white rounded-lg shadow p-6 flex flex-col justify-between">
45       <div class="flex items-center justify-between">
46         <div>
47           <p class="text-sm text-gray-500">Total Pendapatan</p>
48           <p class="text-2xl font-semibold text-gray-800">Rp {{ number_format($stats['total_pendapatan'] ?? 0, 0, ',', '.') }}</p>
49         </div>
50         <div class="w-[50px] h-[50px] flex justify-center items-center rounded-full bg-yellow-100 text-yellow-600">
51           <i class="fas fa-money-bill-wave"></i>
52         </div>
53       </div>
54       <div class="mt-4 text-sm">
55         <span class="text-green-500 font-medium">+15.4%</span>

```

Gambar 5. 27 Halaman dashboard menampilkan halaman utama aplikasi.

```

56         <span class="text-gray-500 ml-1">dari bulan lalu</span>
57     </div>
58 </div>
59
60 <!-- Card 4: Pending Orders -->
61 <div class="bg-white rounded-lg shadow p-6 flex flex-col justify-between">
62     <div class="flex items-center justify-between">
63         <div>
64             <p class="text-sm text-gray-500">Pemesanan Pending</p>
65             <p class="text-2xl font-semibold text-gray-800">{{ number_format($stats['pemesanan_pending'] ?? 0) }}</p>
66         </div>
67         <div class="w-[50px] h-[50px] flex justify-center items-center rounded-full bg-red-100 text-red-600">
68             <i class="fas fa-clock"></i>
69         </div>
70     </div>
71     <div class="mt-4 text-sm">
72         <span class="text-red-500 font-medium">-3.2%</span>
73         <span class="text-gray-500 ml-1">dari minggu lalu</span>
74     </div>
75 </div>
76 </div>
77
78 <!-- Charts -->
79 <div class="grid grid-cols-1 lg:grid-cols-2 gap-6 mb-8">
80     <div class="bg-white rounded-lg shadow p-4">
81         <h3 class="text-lg font-semibold text-gray-800 mb-4">Tren Pemesanan per Bulan</h3>
82         <div class="chart-container" style="position: relative; height:300px;">
83             <canvas id="orderslineChart"></canvas>
84         </div>
85     </div>
86     <div class="bg-white rounded-lg shadow p-4">
87         <h3 class="text-lg font-semibold text-gray-800 mb-4">Pemesanan per Jenis Lapangan</h3>
88         <div class="chart-container" style="position: relative; height:300px;">
89             <canvas id="fieldBarChart"></canvas>
90         </div>
91     </div>
92 </div>
93
94 <!-- Recent Orders Table -->
95 <div class="bg-white rounded-lg shadow overflow-hidden">
96     <div class="px-6 py-4 border-b border-gray-200">
97         <div class="flex items-center justify-between">
98             <h3 class="text-lg font-semibold text-gray-800">Pemesanan Terbaru</h3>
99             <a href="{{ route('laporan.index') }}" class="text-teal-600 hover:text-teal-800 text-sm font-medium">
100                 Lihat Semua
101             </a>
102         </div>
103     </div>
104     <div class="overflow-x-auto">
105         <table class="min-w-full divide-y divide-gray-200">
106             <thead class="bg-gray-50">

```

Gambar 5. 28 Halaman dashboard menampilkan halaman utama aplikasi.

```

107         <tr>
108             <th class="px-6 py-3 text-xs font-medium text-gray-500 uppercase text-left">ID Pemesanan</th>
109             <th class="px-6 py-3 text-xs font-medium text-gray-500 uppercase text-left">Penyewa</th>
110             <th class="px-6 py-3 text-xs font-medium text-gray-500 uppercase text-left">Lapangan</th>
111             <th class="px-6 py-3 text-xs font-medium text-gray-500 uppercase text-left">Tanggal</th>
112             <th class="px-6 py-3 text-xs font-medium text-gray-500 uppercase text-left">Total</th>
113             <th class="px-6 py-3 text-xs font-medium text-gray-500 uppercase text-left">Status</th>
114             <th class="px-6 py-3 text-xs font-medium text-gray-500 uppercase text-left">Aksi</th>
115         </tr>
116     </thead>
117     <tbody class="bg-white divide-y divide-gray-200">
118         @forelse($recentOrders as $order)
119             <tr class="hover:bg-gray-50">
120                 <td class="px-6 py-4 text-sm font-medium text-gray-900">
121                     {{ str_pad($order->id, 4, '0', STR_PAD_LEFT) }}
122                 </td>
123                 <td class="px-6 py-4 text-sm text-gray-500">
124                     <div class="font-medium text-gray-900">{{ $order->user->nama_lengkap }}</div>
125                     <div class="text-xs text-gray-500">{{ $order->user->no_telp }}</div>
126                 </td>
127                 <td class="px-6 py-4 text-sm text-gray-500">
128                     <div class="font-medium text-gray-900">{{ $order->lapangan->nama_lapangan }}</div>
129                     <div class="text-xs text-gray-500">{{ $order->lapangan->jenis_label }}</div>
130                 </td>
131                 <td class="px-6 py-4 text-sm text-gray-500">
132                     {{ $order->tanggal_pemesanan->format('d/m/Y') }}
133                 </td>
134                 <td class="px-6 py-4 text-sm font-medium text-gray-900">
135                     {{ $order->total_harga_formatted }}
136                 </td>
137                 <td class="px-6 py-4">
138                     <span class="inline-flex items-center px-2.5 py-0.5 rounded-full text-xs font-medium {{ $order->status_badge['class'] }}">
139                         <i class="fas {{ $order->status_badge['icon'] }} mr-1"></i>
140                         {{ $order->status_badge['text'] }}
141                     </span>
142                 </td>
143                 <td class="px-6 py-4 text-sm font-medium">
144                     <a href="#" onclick="viewDetail('{{ $order->id }}')" class="text-teal-600 hover:text-teal-900">
145                         Detail
146                     </a>
147                 </td>
148             </tr>
149         @empty
150             <tr>
151                 <td colspan="7" class="px-6 py-8 text-center text-gray-500">
152                     <div class="flex flex-col items-center">
153                         <i class="fas fa-inbox text-4xl text-gray-300 mb-2"></i>
154                         <p>Belum ada pemesanan</p>
155                     </div>
156                 </td>
157             </tr>

```

Gambar 5. 29 Halaman dashboard menampilkan halaman utama aplikasi.

```

157     </tr>
158     @endforelse
159 </tbody>
160 </table>
161 </div>
162 </div>
163 </main>
164 @endsection
165
166 @push('scripts')
167 <script src="https://cdn.jsdelivr.net/npm/chart.js"></script>
168 <script>
169 // Variabel untuk menyimpan instance chart
170 let ordersChart = null;
171 let fieldChart = null;
172
173 // Fungsi untuk menginisialisasi chart
174 function initializeCharts() {
175     // Pastikan data chart tersedia
176     const chartData = @json($chartData ?? []);
177     const fieldData = @json($fieldData ?? []);
178
179     // Hancurkan chart sebelumnya jika ada
180     if (ordersChart) {
181         ordersChart.destroy();
182     }
183     if (fieldChart) {
184         fieldChart.destroy();
185     }
186
187     // Chart 1: Line Chart - Tren Pemesanan per Bulan
188     const ctx1 = document.getElementById('ordersLineChart');
189     if (ctx1 && chartData && chartData.length > 0) {
190         ordersChart = new Chart(ctx1, {
191             type: 'line',
192             data: {
193                 labels: chartData.map(item => item.label || 'N/A'),
194                 datasets: [{
195                     label: 'Pemesanan',
196                     data: chartData.map(item => item.value || 0),
197                     borderColor: '#0d9488',
198                     backgroundColor: 'rgba(13, 148, 136, 0.1)',
199                     borderWidth: 3,
200                     fill: true,
201                     tension: 0.4,
202                     pointBackgroundColor: '#0d9488',
203                     pointBorderColor: 'ffffff',
204                     pointBorderWidth: 2,
205                     pointRadius: 5,
206                     pointHoverRadius: 7
207                 }]
208             },
209             options: {
210                 responsive: true,
211                 maintainAspectRatio: false,
212                 plugins: {
213                     legend: {
214                         display: false
215                     }
216                 },
217                 scales: {
218                     y: {
219                         beginAtZero: true,
220                         grid: {
221                             borderDash: [2, 2]
222                         },
223                         ticks: {
224                             stepSize: 1
225                         }
226                     },
227                     x: {
228                         grid: {
229                             display: false
230                         }
231                     }
232                 }
233             }
234         });
235     }
236 }

```

Gambar 5. 30 Halaman dashboard menampilkan halaman utama aplikasi.

```

233     animation: {
234       duration: 1000,
235       easing: 'easeOutQuart'
236     }
237   }
238   });
239   } else if (ctx1) {
240     // Jika tidak ada data, tampilkan pesan
241     ctx1.style.display = 'none';
242     ctx1.parentElement.innerHTML += '<p class="text-center text-gray-500 py-8">Belum ada data pemesanan</p>';
243   }
244
245   // Chart 2: Doughnut Chart - Pemesanan per Jenis Lapangan
246   const ctx2 = document.getElementById('fieldBarChart');
247   if (ctx2 && fieldData && fieldData.length > 0) {
248     // Warna untuk setiap jenis lapangan
249     const fieldColors = {
250       'futsal': '#108998',
251       'basket': '#F59600',
252       'badminton': '#8B5CF6',
253       'tenis': '#F06292'
254     };
255
256     const fieldLabels = {
257       'futsal': 'Futsal',
258       'basket': 'Basket',
259       'badminton': 'Badminton',
260       'tenis': 'Tenis'
261     };
262
263     fieldChart = new Chart(ctx2, {
264       type: 'doughnut',
265       data: {
266         labels: fieldData.map(item => fieldLabels[item.jenis] || item.jenis),
267         datasets: [{
268           data: fieldData.map(item => item.total || 0),
269           backgroundColor: fieldData.map(item => fieldColors[item.jenis] || '#607288'),
270           borderColor: 'ffffff',
271           borderWidth: 2,
272           hoverOffset: 4
273         }]
274       },
275       options: {
276         responsive: true,
277         maintainAspectRatio: false,
278         plugins: {
279           legend: {
280             position: 'bottom',
281             labels: {
282               padding: 20,
283               usePointStyle: true,
284               font: {
285                 size: 12
286               }
287             }
288           }
289         },
290         animation: {
291           duration: 1000,
292           easing: 'easeOutQuart'
293         }
294       }
295     });
296   } else if (ctx2) {
297     // Jika tidak ada data, tampilkan pesan
298     ctx2.style.display = 'none';
299     ctx2.parentElement.innerHTML += '<p class="text-center text-gray-500 py-8">Belum ada data lapangan</p>';
300   }
301 }
302
303 // Inisialisasi chart saat dokumen siap
304 document.addEventListener('DOMContentLoaded', function() {
305   initializeCharts();
306 });
307
308 // Function untuk view detail
309 function viewDetail(orderId) {
310   alert('Detail pemesanan #' + orderId + '\n\nFitur ini bisa dihubungkan ke halaman detail pemesanan.');
```

Gambar 5. 31 Halaman dashboard menampilkan halaman utama aplikasi.

```

1  @extends('layout.main')
2
3  @section('page-title', 'Kalender Jadwal')
4
5  @section('content')
6  <main class="flex-1 overflow-y-auto p-4">
7    <div class="mb-4">
8      <div class="flex flex-col md:flex-row md:items-center md:justify-between">
9        <div>
10         <h2 class="text-xl font-semibold text-gray-800">Kalender Jadwal</h2>
11         <p class="text-gray-600 mt-1">lihat jadwal pemesanan lapangan yang telah dikonfirmasi</p>
12       </div>
13
14       <!-- Legend untuk warna berdasarkan jenis lapangan -->
15       <div class="mt-4 md:mt-0">
16         <div class="flex flex-wrap gap-2">
17           <div class="flex items-center text-xs">
18             <div class="w-3 h-3 rounded mr-1" style="background-color: #800080;"></div>
19             <span>Futsal</span>
20           </div>
21           <div class="flex items-center text-xs">
22             <div class="w-3 h-3 rounded mr-1" style="background-color: #FFA500;"></div>
23             <span>Basket</span>
24           </div>
25           <div class="flex items-center text-xs">
26             <div class="w-3 h-3 rounded mr-1" style="background-color: #800080;"></div>
27             <span>Badminton</span>
28           </div>
29           <div class="flex items-center text-xs">
30             <div class="w-3 h-3 rounded mr-1" style="background-color: #FF4500;"></div>
31             <span>Tenis</span>
32           </div>
33         </div>
34       </div>
35     </div>
36
37     <div class="bg-white rounded-lg shadow overflow-hidden">
38       <div class="p-4">
39         <div id="calendar"></div>
40       </div>
41     </div>
42
43     <!-- Modal untuk detail event -->
44     <div id="eventModal" class="fixed inset-0 z-50 hidden overflow-y-auto aria-labelledby="modal-title" role="dialog" aria-modal="true">
45       <div class="flex items-end justify-center min-h-screen pt-4 px-4 pb-20 text-center sm:block sm:p-0">
46         <div class="fixed inset-0 bg-gray-500 bg-opacity-75 transition-opacity" aria-hidden="true"></div>
47         <div class="hidden sm:inline-block sm:align-middle sm:h-screen" aria-hidden="true"></div>
48         <div class="bg-white px-4 pt-5 pb-4 sm:p-6 sm:pb-4">
49           <div class="sm:flex sm:items-start">
50             <div class="mt-3 text-center sm:mt-0 sm:text-left w-full">
51               <h3 class="text-lg leading-6 font-medium text-gray-900" id="modal-title">
52                 Detail Jadwal Pemesanan
53               </h3>
54               <div class="mt-4" id="modal-content">
55                 <!-- Content will be filled by JavaScript -->
56               </div>
57             </div>
58           </div>
59         </div>
60       </div>
61       <div class="bg-gray-50 px-4 py-3 sm:px-6 sm:flex sm:flex-row-reverse">
62         <button type="button" id="closeModal" class="w-full inline-flex justify-center rounded-md border border-transparent shadow-sm px-4 py-2 bg-blue-600 text-base fo
63           Tutup
64         </button>
65       </div>
66     </div>
67
68     </div>
69
70   </main>
71 </section>
72
73 @push('styles')
74 <link href="https://cdn.jsdelivr.net/npm/fullcalendar@5.11.3/main.min.css" rel="stylesheet">
75 <style>
76   .fc-toolbar-title {
77     font-size: 1.25rem;
78     font-weight: 600;
79     color: #374181 !important;
80   }
81

```

Gambar 5. 32 Halaman jadwal menampilkan daftar pemesanan lapangan.

```

82 }
83 .fc-button {
84   background-color: #fff;
85   border: 1px solid #ddd;
86   color: #337ab7;
87   padding: 0.4rem 0.8rem;
88   border-radius: 0.375rem;
89   font-size: 0.875rem;
90   transition: background-color 0.2s;
91 }
92 .fc-button:hover {
93   background-color: #f3f4f6;
94 }
95 .fc-button-primary {
96   background-color: #007bff;
97   border-color: #007bff;
98   color: #fff;
99 }
100 .fc-button-primary:hover {
101   background-color: #0070c0;
102 }
103 .fc-event {
104   border-radius: 0.375rem;
105   padding: 2px 4px;
106   cursor: pointer;
107   font-size: 0.875rem;
108 }
109 .fc-event:hover {
110   opacity: 0.8;
111 }
112 .fc-daygrid-event {
113   border-radius: 4px;
114 }
115 .fc-timegrid-event {
116   border-radius: 4px;
117 }
118 </style>
119 @endpush
120
121 @push('scripts')
122 <script src="https://cdn.jsdelivr.net/npm/fullcalendar@5.11.3/main.min.js"></script>
123 <script src="https://cdn.jsdelivr.net/npm/fullcalendar@5.11.3/locales/id.min.js"></script>
124 <script>
125   document.addEventListener('DOMContentLoaded', function() {
126     let calendarEl = document.getElementById('calendar');
127     let modal = document.getElementById('eventModal');
128     let closeModalBtn = document.getElementById('closeModal');
129
130     let calendar = new FullCalendar.Calendar(calendarEl, {
131       locale: 'id',
132       initialView: 'dayGridMonth',
133       headerToolbar: {
134         left: 'prev,next today',
135         center: 'title',
136         right: 'dayGridMonth,timeGridWeek,timeGridDay,listWeek'
137       },
138       nowIndicator: true,
139       navlinks: true,
140       dayMaxEvents: 3, // batasi event per hari di month view
141       moreLinkText: function(num) {
142         return '+' + num + ' lainnya';
143       },
144       height: 'auto',
145       eventTimeFormat: {
146         hour: '2-digit',
147         minute: '2-digit',
148         hour12: false
149       },
150       slotLabelFormat: {
151         hour: '2-digit',
152         minute: '2-digit',
153         hour12: false
154       },
155       events: function(info, successCallback, failureCallback) {
156         fetch('/jadwal/events', {
157           method: 'GET',
158           headers: {
159             'Content-Type': 'application/json',
160             'X-Requested-With': 'XMLHttpRequest',

```

Gambar 5. 33 Halaman jadwal menampilkan daftar pemesanan lapangan.

```

161     'X-CSRF-TOKEN': document.querySelector('meta[name="csrf-token"]').getAttribute('content')
162   }
163   })
164   .then(response => response.json())
165   .then(data => {
166     | successCallback(data);
167   })
168   .catch(error => {
169     | console.error('Error fetching events:', error);
170     | failureCallback(error);
171   });
172 },
173 eventClick: function(info) {
174   | showEventModal(info.event);
175 },
176 eventBidMount: function(info) {
177   | // Tambahkan tooltip
178   | info.el.setAttribute('title',
179   |   info.event.extendedProps.lapangan + ' - ' +
180   |   info.event.extendedProps.penyewa
181   | );
182 }
183 });
184
185 calendar.render();
186
187 // Function untuk menampilkan modal detail
188 function showEventModal(event) {
189   const props = event.extendedProps;
190   const startTime = event.start.toISOString('id-ID', {
191     hour: '2-digit',
192     minute: '2-digit',
193     hour12: false
194   });
195   const endTime = event.end.toISOString('id-ID', {
196     hour: '2-digit',
197     minute: '2-digit',
198     hour12: false
199   });
200   const eventDate = event.start.toISOString('id-ID', {
201     weekday: 'long',
202     year: 'numeric',
203     month: 'long',
204     day: 'numeric'
205   });
206
207   document.getElementById('modal-content').innerHTML = `
208     <div class="space-y-3">
209       <div class="bg-gray-50 p-3 rounded-lg">
210         <h4 class="font-sans font-weight-bold text-gray-900 mb-2">Informasi Lapangan</h4>
211         <div class="grid grid-cols-1 gap-2 text-sm">
212           <div><span class="font-medium">Lapangan:</span> ${props.lapangan}</div>
213           <div><span class="font-medium">Jenis:</span> ${props.jenis}</div>
214         </div>
215       </div>
216       <div class="bg-blue-50 p-3 rounded-lg">
217         <h4 class="font-sans font-weight-bold text-gray-900 mb-2">Jadwal</h4>
218         <div class="grid grid-cols-1 gap-2 text-sm">
219           <div><span class="font-medium">Tanggal:</span> ${eventDate}</div>
220           <div><span class="font-medium">Waktu:</span> ${startTime} - ${endTime}</div>
221           <div><span class="font-medium">Durasi:</span> ${props.durasi}</div>
222         </div>
223       </div>
224       <div class="bg-green-50 p-3 rounded-lg">
225         <h4 class="font-sans font-weight-bold text-gray-900 mb-2">Informasi Penyewa</h4>
226         <div class="grid grid-cols-1 gap-2 text-sm">
227           <div><span class="font-medium">Nama:</span> ${props.penyewa}</div>
228           <div><span class="font-medium">No. Telp:</span> ${props.phone}</div>
229         </div>
230       </div>
231       <div class="bg-yellow-50 p-3 rounded-lg">
232         <h4 class="font-sans font-weight-bold text-gray-900 mb-2">Pembayaran</h4>
233         <div class="grid grid-cols-1 gap-2 text-sm">
234           <div><span class="font-medium">Total:</span> ${props.total_harga}</div>
235         </div>
236       </div>
237     </div>

```

Gambar 5. 34 Halaman jadwal menampilkan daftar pemesanan lapangan.

```

238     <div><span class="font-medium">Metode:</span> ${props.metode_bayar}</div>
239     <div><span class="font-medium text-green-600">Status: Lunas</span></div>
240   </div>
241 </div>
242 `;
243
244 modal.classList.remove('hidden');
245
246 // Event listeners untuk modal
247 closeModalBtn.addEventListener('click', function() {
248   modal.classList.add('hidden');
249 });
250
251 modal.addEventListener('click', function(e) {
252   if (e.target === modal) {
253     modal.classList.add('hidden');
254   }
255 });
256
257 // Close modal with ESC key
258 document.addEventListener('keydown', function(e) {
259   if (e.key === 'Escape' && !modal.classList.contains('hidden')) {
260     modal.classList.add('hidden');
261   }
262 });
263
264 });
265
266 </script>
267 @endpush

```

Gambar 5. 35 Halaman jadwal menampilkan daftar pemesanan lapangan.

```

1  @extends('layout.main')
2
3  @section('page-title', 'Manajemen Lapangan')
4
5  @section('content')
6  <main class="flex-1 overflow-y-auto p-4">
7    <!-- Page Header -->
8    <div class="mb-6">
9      <div class="flex flex-col md:flex-row md:items-center md:justify-between">
10        <div>
11          <h2 class="text-2xl font-semibold text-gray-800">Manajemen Lapangan</h2>
12          <p class="text-gray-600 mt-1">Kelola data lapangan olahraga</p>
13        </div>
14        <div class="mt-4 md:mt-0">
15          <a href="{{ route('lapangan.create') }}">
16            <div class="inline-flex items-center px-4 py-2 bg-teal-600 hover:bg-teal-700 text-white font-medium rounded-lg transition-colors">
17              <i class="fas fa-plus mr-2"></i>
18              Tambah Lapangan
19            </div>
20          </a>
21        </div>
22      </div>
23
24    <!-- Flash Messages -->
25    @if(session('success'))
26      <div class="mb-6 bg-green-100 border border-green-400 text-green-700 px-4 py-3 rounded relative" role="alert">
27        <span class="block sm:inline">{{ session('success') }}</span>
28        <button type="button" class="absolute top-0 bottom-0 right-0 px-4 py-3" onclick="this.parentElement.style.display='none'">
29          <i class="fas fa-times"></i>
30        </button>
31      </div>
32    @endif
33
34    @if(session('error'))
35      <div class="mb-6 bg-red-100 border border-red-400 text-red-700 px-4 py-3 rounded relative" role="alert">
36        <span class="block sm:inline">{{ session('error') }}</span>
37        <button type="button" class="absolute top-0 bottom-0 right-0 px-4 py-3" onclick="this.parentElement.style.display='none'">
38          <i class="fas fa-times"></i>
39        </button>
40      </div>
41    @endif
42
43    <!-- Stats Cards -->
44    <div class="grid grid-cols-1 md:grid-cols-2 lg:grid-cols-4 gap-6 mb-8">
45      <div class="bg-white rounded-lg shadow p-6">
46        <div class="flex items-center justify-between">
47          <div>
48            <p class="text-sm text-gray-500">Total Lapangan</p>
49            <p class="text-2xl font-semibold text-gray-800">{{ $stats['total'] }}</p>
50          </div>
51          <div class="w-12 h-12 bg-blue-100 rounded-full flex items-center justify-center">
52            <i class="fas fa-map text-blue-600"></i>
53          </div>
54        </div>
55      </div>
56
57      <div class="bg-white rounded-lg shadow p-6">
58        <div class="flex items-center justify-between">
59          <div>
60            <p class="text-sm text-gray-500">Lapangan Aktif</p>
61            <p class="text-2xl font-semibold text-green-600">{{ $stats['active'] }}</p>
62          </div>
63          <div class="w-12 h-12 bg-green-100 rounded-full flex items-center justify-center">
64            <i class="fas fa-check-circle text-green-600"></i>
65          </div>
66        </div>
67      </div>
68
69      <div class="bg-white rounded-lg shadow p-6">
70        <div class="flex items-center justify-between">
71          <div>
72            <p class="text-sm text-gray-500">Futsal</p>
73            <p class="text-2xl font-semibold text-orange-600">{{ $stats['futsal'] }}</p>
74          </div>
75          <div class="w-12 h-12 bg-orange-100 rounded-full flex items-center justify-center">
76            <i class="fas fa-futbol text-orange-600"></i>
77          </div>
78        </div>
79      </div>
80
81    </div>
82    <div class="bg-white rounded-lg shadow p-6">

```

Gambar 5. 36 Halaman lapangan menampilkan daftar lapangan olahraga.

```

82         <div class="flex items-center justify-between">
83             <div>
84                 <p class="text-sm text-gray-500">Basket</p>
85                 <p class="text-2xl font-semibold text-purple-600">{{ $stats['basket'] }}</p>
86             </div>
87             <div class="w-12 h-12 bg-purple-100 rounded-full flex items-center justify-center">
88                 <i class="fas fa-basketball-ball text-purple-600"></i>
89             </div>
90         </div>
91     </div>
92 </div>
93
94 <!-- Lapangan Table -->
95 <div class="bg-white rounded-lg shadow overflow-hidden">
96     <div class="px-6 py-4 border-b border-gray-200">
97         <div class="flex flex-col md:flex-row md:items-center md:justify-between">
98             <h3 class="text-lg font-semibold text-gray-800">Data Lapangan</h3>
99             <div class="mt-4 md:mt-0 flex flex-wrap items-center gap-3">
100                 <!-- Search Box -->
101                 <form method="GET" class="flex items-center space-x-2">
102                     <div class="relative">
103                         <input type="text"
104                             name="search"
105                             value="{{ request('search') }}"
106                             placeholder="Cari lapangan..."
107                             class="pl-10 pr-4 py-2 border border-gray-300 rounded-lg focus:ring-2 focus:ring-teal-500 focus:border-transparent">
108                         <i class="fas fa-search absolute left-3 top-3 text-gray-400"></i>
109                     </div>
110                 </form>
111
112                 <!-- Filter Jenis -->
113                 <select name="jenis" class="px-4 py-2 border border-gray-300 rounded-lg focus:ring-2 focus:ring-teal-500 focus:border-transparent">
114                     <option value="">Semua Jenis</option>
115                     <option value="futsal" {{ request('jenis') == 'futsal' ? 'selected' : '' }}>Futsal</option>
116                     <option value="basket" {{ request('jenis') == 'basket' ? 'selected' : '' }}>Basket</option>
117                     <option value="badminton" {{ request('jenis') == 'badminton' ? 'selected' : '' }}>Badminton</option>
118                     <option value="tenis" {{ request('jenis') == 'tenis' ? 'selected' : '' }}>Tenis</option>
119                 </select>
120
121                 <!-- Filter Status -->
122                 <select name="status" class="px-4 py-2 border border-gray-300 rounded-lg focus:ring-2 focus:ring-teal-500 focus:border-transparent">
123                     <option value="">Semua Status</option>
124                     <option value="active" {{ request('status') == 'active' ? 'selected' : '' }}>Aktif</option>
125                     <option value="inactive" {{ request('status') == 'inactive' ? 'selected' : '' }}>Tidak Aktif</option>
126                 </select>
127
128                 <button type="submit" class="px-4 py-2 bg-teal-600 hover:bg-teal-700 text-white rounded-lg transition-colors">
129                     <i class="fas fa-filter mr-1"></i>
130                     Filter
131                 </button>
132
133                 @if(request()->hasAny(['search', 'jenis', 'status']))
134                     <a href="{{ route('lapangan.index') }}" class="px-4 py-2 bg-gray-500 hover:bg-gray-600 text-white rounded-lg transition-colors">
135                         <i class="fas fa-times mr-1"></i>
136                         Reset
137                     </a>
138                 @endif
139             </div>
140         </div>
141     </div>
142
143     <div class="overflow-x-auto">
144         <table class="min-w-full divide-y divide-gray-200" id="lapanganTable">
145             <thead class="bg-gray-50">
146                 <tr>
147                     <th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">Lapangan</th>
148                     <th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">Jenis</th>
149                     <th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">Harga/jam</th>
150                     <th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">Status</th>
151                     <th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">Aksi</th>
152                 </tr>
153             </thead>
154             <tbody class="bg-white divide-y divide-gray-200">
155                 @foreach($lapangans as $lapangan)
156                     <tr class="hover:bg-gray-50">
157                         <td class="px-6 py-4 whitespace-nowrap">
158                             <div class="flex items-center">
159                                 <div class="h-12 w-12 flex-shrink-0">

```

Gambar 5. 37 Halaman lapangan menampilkan daftar lapangan olahraga.

```

160 nama_lapangan }}"
163       onerror="this.src='https://placehold.co/48x48/c5e7eb/6b7280?text=No+Image'"
164     />
165 <div class="ml-4">
166   <div class="text-sm font-medium text-gray-900">{{ $lapangan->nama_lapangan }}</div>
167   <div class="text-sm text-gray-500">
168     {{ Str::limit($lapangan->deskripsi, 50) }}
169   </div>
170 </div>
171 </td>
172 <td class="px-6 py-4 whitespace-nowrap">
173   <span class="inline-flex px-2 py-1 text-xs font-semibold rounded-full
174     @if($lapangan->jenis == 'futsal') bg-green-100 text-green-800
175     @elseif($lapangan->jenis == 'basket') bg-orange-100 text-orange-800
176     @elseif($lapangan->jenis == 'badminton') bg-blue-100 text-blue-800
177     @else bg-purple-100 text-purple-800 @endif">
178     @if($lapangan->jenis == 'futsal')
179       <i class="fas fa-futbol mr-1"></i>
180     @elseif($lapangan->jenis == 'basket')
181       <i class="fas fa-basketball-ball mr-1"></i>
182     @elseif($lapangan->jenis == 'badminton')
183       <i class="fas fa-table-tennis mr-1"></i>
184     @else
185       <i class="fas fa-tennis-ball mr-1"></i>
186     @endif
187     {{ $lapangan->jenis_label }}
188   </span>
189 </td>
190 <td class="px-6 py-4 whitespace-nowrap text-sm text-gray-900 font-medium">
191   {{ $lapangan->formatted_harga }}
192 </td>
193 <td class="px-6 py-4 whitespace-nowrap">
194   <button onclick="toggleStatus('{{ $lapangan->id }}')"
195     class="inline-flex items-center px-2 py-1 text-xs font-semibold rounded-full transition-colors
196     {{ $lapangan->is_active ? 'bg-green-100 text-green-800 hover:bg-green-200' : 'bg-red-100 text-red-800 hover:bg-red-200' }}"
197     <i class="fas {{ $lapangan->is_active ? 'fa-check-circle' : 'fa-times-circle' }} mr-1"></i>
198     {{ $lapangan->is_active ? 'Aktif' : 'Tidak Aktif' }}
199   </button>
200 </td>
201 <td class="px-6 py-4 whitespace-nowrap text-sm font-medium">
202   <div class="flex items-center space-x-2">
203     <a href="{{ route('lapangan.show', $lapangan) }}"
204       class="inline-flex items-center px-3 py-1 bg-blue-600 hover:bg-blue-700 text-white text-xs font-medium rounded-md transition-colors">
205       <i class="fas fa-eye mr-1"></i>
206       Detail
207     </a>
208     <a href="{{ route('lapangan.edit', $lapangan) }}"
209       class="inline-flex items-center px-3 py-1 bg-yellow-600 hover:bg-yellow-700 text-white text-xs font-medium rounded-md transition-colors">
210       <i class="fas fa-edit mr-1"></i>
211       Edit
212     </a>
213     <button onclick="confirmDelete('{{ $lapangan->id }}', '{{ $lapangan->nama_lapangan }}')"
214       class="inline-flex items-center px-3 py-1 bg-red-600 hover:bg-red-700 text-white text-xs font-medium rounded-md transition-colors">
215       <i class="fas fa-trash mr-1"></i>
216       Hapus
217     </button>
218   </div>
219 </td>
220 </tr>
221 <tr>
222   <td colspan="6">
223     <div class="flex flex-col items-center">
224       <div class="flex flex-col items-center">
225         <i class="fas fa-map text-4xl text-gray-300 mb-4"></i>
226         <p class="text-gray-500">Tidak ada data lapangan</p>
227       </div>
228       <a href="{{ route('lapangan.create') }}"
229         class="mt-4 inline-flex items-center px-4 py-2 bg-teal-600 hover:bg-teal-700 text-white font-medium rounded-lg transition-colors">
230         <i class="fas fa-plus mr-2"></i>
231         Tambah Lapangan Pertama
232       </a>
233     </div>
234 </td>
235 </tr>

```

Gambar 5. 38 Halaman lapangan menampilkan daftar lapangan olahraga.

```

236         @endforelse
237     </tbody>
238 </table>
239 </div>
240
241 <!-- Pagination -->
242 @if($lapangans->hasPages())
243 <div class="px-6 py-4 border-t border-gray-200">
244     {{ $lapangans->appends(request()->query()->links() ) }}
245 </div>
246 @endif
247 </div>
248 </main>
249
250 <!-- Delete Confirmation Modal -->
251 <div id="deleteModal" class="fixed inset-0 bg-black bg-opacity-50 flex items-center justify-center hidden z-50">
252 <div class="bg-white rounded-lg p-6 max-w-sm w-full mx-4">
253 <div class="flex items-center mb-4">
254 <div class="w-12 h-12 bg-red-100 rounded-full flex items-center justify-center mr-4">
255 <i class="fas fa-exclamation-triangle text-red-600"></i>
256 </div>
257 <h3 class="text-lg font-semibold text-gray-900">Konfirmasi Hapus</h3>
258 </div>
259 <p class="text-gray-600 mb-6">Apakah Anda yakin ingin menghapus lapangan <span id="lapanganName" class="font-semibold"></span>? Tindakan ini tidak dapat dibatalkan.</p>
260 <div class="flex justify-end space-x-3">
261 <button onclick="closeDeleteModal()" class="px-4 py-2 text-gray-600 hover:bg-gray-100 rounded-md transition-colors">
262     Batal
263 </button>
264 <form id="deleteForm" method="POST" class="inline">
265     @csrf
266     @method('DELETE')
267     <button type="submit" class="px-4 py-2 bg-red-600 hover:bg-red-700 text-white rounded-md transition-colors">
268         Hapus
269     </button>
270 </form>
271 </div>
272 </div>
273 </div>
274 @endsection
275
276 @push('scripts')
277 <script>
278 function toggleStatus(lapanganId) {
279     fetch('/lapangan/{lapanganId}/toggle-status', {
280         method: 'PATCH',
281         headers: {
282             'X-CSRF-TOKEN': document.querySelector('meta[name="csrf-token"]').getAttribute('content'),
283             'Content-Type': 'application/json',
284             'Accept': 'application/json'
285         }
286     })
287     .then(response => response.json())
288     .then(data => {
289         if (data.success) {
290             // Reload page to update UI
291             window.location.reload();
292         } else {
293             alert(data.message || 'Terjadi kesalahan');
294         }
295     })
296     .catch(error => {
297         console.error('Error:', error);
298         alert('Terjadi kesalahan saat mengubah status');
299     });
300 }
301
302 function confirmDelete(lapanganId, lapanganName) {
303     document.getElementById('lapanganName').textContent = lapanganName;
304     document.getElementById('deleteForm').action = '/lapangan/{lapanganId}';
305     document.getElementById('deleteModal').classList.remove('hidden');
306 }
307
308 function closeDeleteModal() {
309     document.getElementById('deleteModal').classList.add('hidden');
310 }
311

```

Gambar 5. 39 Halaman lapangan menampilkan daftar lapangan olahraga.

```

312 // Close modal when clicking outside
313 document.getElementById('deleteModal').addEventListener('click', function(e) {
314     if (e.target.id === 'deleteModal') {
315         closeDeleteModal();
316     }
317 });
318
319 // Close modal with Escape key
320 document.addEventListener('keydown', function(e) {
321     if (e.key === 'Escape') {
322         closeDeleteModal();
323     }
324 });
325 </script>
326 @endpush

```

Gambar 5. 40 Halaman lapangan menampilkan daftar lapangan olahraga.

```

1  @extends('layout.main')
2
3  @section('page-title', 'Laporan Pemesanan')
4
5  @section('content')
6  <main class="flex-1 overflow-y-auto p-4">
7    <!-- Header -->
8    <div class="mb-6">
9      <div class="flex flex-col md:flex-row md:items-center md:justify-between">
10        <div>
11          <h2 class="text-xl font-semibold text-gray-800">Laporan Pemesanan</h2>
12          <p class="text-gray-600 mt-1">Kelola dan pantau semua data pemesanan lapangan</p>
13        </div>
14        <div class="mt-4 md:mt-0">
15          <button onclick="cetakLaporan()" class="bg-red-600 hover:bg-red-700 text-white px-4 py-2 rounded-lg inline-flex items-center">
16            <i class="fas fa-file-pdf mr-2"></i>
17            Cetak Laporan PDF
18          </button>
19        </div>
20      </div>
21    </div>
22
23    <!-- Statistics Cards -->
24    <div class="grid grid-cols-1 md:grid-cols-2 lg:grid-cols-5 gap-4 mb-6">
25      <div class="bg-white p-4 rounded-lg shadow">
26        <div class="flex items-center">
27          <div class="p-2 bg-blue-100 rounded-lg">
28            <i class="fas fa-calendar-alt text-blue-600"></i>
29          </div>
30          <div class="ml-3">
31            <p class="text-sm text-gray-500">Total Pemesanan</p>
32            <p class="text-lg font-semibold text-gray-900">{{ number_format($stats['total_pemesanan']) }}</p>
33          </div>
34        </div>
35      </div>
36
37      <div class="bg-white p-4 rounded-lg shadow">
38        <div class="flex items-center">
39          <div class="p-2 bg-green-100 rounded-lg">
40            <i class="fas fa-check-circle text-green-600"></i>
41          </div>
42          <div class="ml-3">
43            <p class="text-sm text-gray-500">Sukses</p>
44            <p class="text-lg font-semibold text-gray-900">{{ number_format($stats['pemesanan_sukses']) }}</p>
45          </div>
46        </div>
47      </div>
48
49      <div class="bg-white p-4 rounded-lg shadow">
50        <div class="flex items-center">
51          <div class="p-2 bg-yellow-100 rounded-lg">
52            <i class="fas fa-clock text-yellow-600"></i>
53          </div>
54          <div class="ml-3">
55            <p class="text-sm text-gray-500">Pending</p>
56            <p class="text-lg font-semibold text-gray-900">{{ number_format($stats['pemesanan_pending']) }}</p>
57          </div>
58        </div>
59      </div>
60
61      <div class="bg-white p-4 rounded-lg shadow">
62        <div class="flex items-center">
63          <div class="p-2 bg-purple-100 rounded-lg">
64            <i class="fas fa-money-bill-wave text-purple-600"></i>
65          </div>
66          <div class="ml-3">
67            <p class="text-sm text-gray-500">Total Pendapatan</p>
68            <p class="text-lg font-semibold text-gray-900">Rp {{ number_format($stats['total_pendapatan'], 0, ',', '.') }}</p>
69          </div>
70        </div>
71      </div>
72
73      <div class="bg-white p-4 rounded-lg shadow">
74        <div class="flex items-center">
75          <div class="p-2 bg-teal-100 rounded-lg">
76            <i class="fas fa-calendar-day text-teal-600"></i>
77          </div>
78          <div class="ml-3">
79            <p class="text-sm text-gray-500">Hari Ini</p>
80            <p class="text-lg font-semibold text-gray-900">{{ number_format($stats['pemesanan_hari_ini']) }}</p>
81          </div>

```

Gambar 5. 41 Halaman laporan menampilkan daftar laporan pemesanan.

```

82     </div>
83   </div>
84 </div>
85
86   <!-- Filter Section -->
87   <div class="bg-white rounded-lg shadow mb-6">
88     <div class="p-4 border-b border-gray-200">
89       <h3 class="text-lg font-medium text-gray-900">Filter & Pencarian</h3>
90     </div>
91     <div class="p-4">
92       <form method="GET" action="{{ route('laporan.index') }}" class="grid grid-cols-1 md:grid-cols-2 lg:grid-cols-4 gap-4">
93         <div>
94           <label class="block text-sm font-medium text-gray-700 mb-1">Pencarian</label>
95           <input type="text" name="search" value="{{ request('search') }}"
96             placeholder="Nama penyewa atau lapangan..."
97             class="w-full px-3 py-2 border border-gray-300 rounded-md focus:outline-none focus:ring-2 focus:ring-teal-500">
98         </div>
99
100        <div>
101          <label class="block text-sm font-medium text-gray-700 mb-1">Status Pemesanan</label>
102          <select name="status" class="w-full px-3 py-2 border border-gray-300 rounded-md focus:outline-none focus:ring-2 focus:ring-teal-500">
103            <option value="">Semua Status</option>
104            <option value="pending" {{ request('status') == 'pending' ? 'selected' : '' }}>Pending</option>
105            <option value="sukses" {{ request('status') == 'sukses' ? 'selected' : '' }}>Sukses</option>
106            <option value="gagal" {{ request('status') == 'gagal' ? 'selected' : '' }}>Gagal</option>
107          </select>
108        </div>
109
110        <div>
111          <label class="block text-sm font-medium text-gray-700 mb-1">Status Pembayaran</label>
112          <select name="status_bayar" class="w-full px-3 py-2 border border-gray-300 rounded-md focus:outline-none focus:ring-2 focus:ring-teal-500">
113            <option value="">Semua Status</option>
114            <option value="pending" {{ request('status_bayar') == 'pending' ? 'selected' : '' }}>Pending</option>
115            <option value="valid" {{ request('status_bayar') == 'valid' ? 'selected' : '' }}>Valid</option>
116            <option value="invalid" {{ request('status_bayar') == 'invalid' ? 'selected' : '' }}>Invalid</option>
117          </select>
118        </div>
119
120        <div>
121          <label class="block text-sm font-medium text-gray-700 mb-1">Jenis Lapangan</label>
122          <select name="jenis" class="w-full px-3 py-2 border border-gray-300 rounded-md focus:outline-none focus:ring-2 focus:ring-teal-500">
123            <option value="">Semua Jenis</option>
124            <option value="futsal" {{ request('jenis') == 'futsal' ? 'selected' : '' }}>Futsal</option>
125            <option value="basket" {{ request('jenis') == 'basket' ? 'selected' : '' }}>Basket</option>
126            <option value="badminton" {{ request('jenis') == 'badminton' ? 'selected' : '' }}>Badminton</option>
127            <option value="tenis" {{ request('jenis') == 'tenis' ? 'selected' : '' }}>Tenis</option>
128          </select>
129        </div>
130
131        <div>
132          <label class="block text-sm font-medium text-gray-700 mb-1">Tanggal Dari</label>
133          <input type="date" name="tanggal_dari" value="{{ request('tanggal_dari') }}"
134            class="w-full px-3 py-2 border border-gray-300 rounded-md focus:outline-none focus:ring-2 focus:ring-teal-500">
135        </div>
136
137        <div>
138          <label class="block text-sm font-medium text-gray-700 mb-1">Tanggal Sampai</label>
139          <input type="date" name="tanggal_sampai" value="{{ request('tanggal_sampai') }}"
140            class="w-full px-3 py-2 border border-gray-300 rounded-md focus:outline-none focus:ring-2 focus:ring-teal-500">
141        </div>
142
143        <div class="flex items-end space-x-2">
144          <button type="submit" class="bg-teal-600 hover:bg-teal-700 text-white px-4 py-2 rounded-md">
145            <i class="fas fa-search mr-1"></i> Filter
146          </button>
147          <a href="{{ route('laporan.index') }}" class="bg-gray-500 hover:bg-gray-600 text-white px-4 py-2 rounded-md">
148            <i class="fas fa-times mr-1"></i> Reset
149          </a>
150        </div>
151      </form>
152    </div>
153  </div>
154
155  <!-- Data Table -->
156  <div class="bg-white rounded-lg shadow overflow-hidden">
157    <div class="overflow-x-auto">
158      <table class="w-full">

```

Gambar 5. 42 Halaman laporan menampilkan daftar laporan pemesanan.

```

159 <thead class="bg-gray-50">
160 <tr>
161 <th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">No</th>
162 <th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">Penyawa</th>
163 <th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">Lapangan</th>
164 <th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">Tanggal</th>
165 <th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">Jadwal</th>
166 <th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">Total</th>
167 <th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">Status</th>
168 <th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">Pembayaran</th>
169 <th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">Aksi</th>
170 </tr>
171 </thead>
172 <tbody class="bg-white divide-y divide-gray-200">
173 @foreach($pemesanan as $index => $pemesanan)
174 <tr class="hover:bg-gray-50">
175 <td class="px-6 py-4 whitespace-nowrap text-sm text-gray-900">
176 {{ $pemesanan->firstItem() + $index }}
177 </td>
178 <td class="px-6 py-4 whitespace-nowrap">
179 <div class="flex flex-col">
180 <div class="text-sm font-medium text-gray-900">{{ $pemesanan->user->nama_lengkap }}</div>
181 <div class="text-sm text-gray-500">{{ $pemesanan->user->no_telp }}</div>
182 </div>
183 </td>
184 <td class="px-6 py-4 whitespace-nowrap">
185 <div class="flex flex-col">
186 <div class="text-sm font-medium text-gray-900">{{ $pemesanan->lapangan->nama_lapangan }}</div>
187 <div class="text-sm text-gray-500">{{ $pemesanan->lapangan->jenis_label }}</div>
188 </div>
189 </td>
190 <td class="px-6 py-4 whitespace-nowrap text-sm text-gray-900">
191 {{ $pemesanan->tanggal_pemesanan->format('d/m/Y') }}
192 </td>
193 <td class="px-6 py-4 whitespace-nowrap text-sm text-gray-900">
194 @if($pemesanan->jadwal_formatted)
195 {{ implode(' ', $pemesanan->jadwal_formatted) }}
196 @else
197 -
198 @endif
199 </td>
200 <td class="px-6 py-4 whitespace-nowrap text-sm font-medium text-gray-900">
201 {{ $pemesanan->total_harga_formatted }}
202 </td>
203 <td class="px-6 py-4 whitespace-nowrap">
204 <span class="inline-flex items-center px-2.5 py-0.5 rounded-full text-xs font-medium {{ $pemesanan->status_badge['class'] }}">
205 <i class="fas {{ $pemesanan->status_badge['icon'] }}" mr-1"></i>
206 {{ $pemesanan->status_badge['text'] }}
207 </span>
208 </td>
209 <td class="px-6 py-4 whitespace-nowrap">
210 @if($pemesanan->pembayaran)
211 <span class="inline-flex items-center px-2.5 py-0.5 rounded-full text-xs font-medium {{ $pemesanan->pembayaran->status_badge['class'] }}">
212 <i class="fas {{ $pemesanan->pembayaran->status_badge['icon'] }}" mr-1"></i>
213 {{ $pemesanan->pembayaran->status_badge['text'] }}
214 </span>
215 @else
216 <span class="text-gray-400 text-xs">Tidak ada</span>
217 @endif
218 </td>
219 <td class="px-6 py-4 whitespace-nowrap text-sm font-medium">
220 <button onclick="cetakTiket('{{ $pemesanan->id }})"
221 class="bg-blue-600 hover:bg-blue-700 text-white px-3 py-1 rounded text-xs inline-flex items-center">
222 <i class="fas fa-ticket-alt mr-1"></i>
223 Cetak Tiket
224 </button>
225 </td>
226 </tr>
227 @empty
228 <tr>
229 <td colspan="9" class="px-6 py-4 text-center text-gray-500">
230 <div class="flex flex-col items-center">
231 <i class="fas fa-inbox text-4xl text-gray-300 mb-2"></i>
232 <p>Tidak ada data pemesanan</p>
233 </div>
234 </td>
235 </tr>

```

Gambar 5. 43 Halaman laporan menampilkan daftar laporan pemesanan.

```

236         @endforelse
237     </tbody>
238 </table>
239 </div>
240
241 <!-- Pagination -->
242 @if($pemesanans->hasPages())
243     <div class="px-6 py-3 border-t border-gray-200">
244         {{ $pemesanans->withQueryString()->links() }}
245     </div>
246 @endif
247 </div>
248 </main>
249 @endsection
250
251 @push('scripts')
252 <script>
253     function cetakTiket(id) {
254         // Buat form hidden untuk mengirim request POST
255         let form = document.createElement('form');
256         form.method = 'GET';
257         form.action = `/laporan/cetak-pdf/${id}`;
258         form.target = '_blank';
259
260         document.body.appendChild(form);
261         form.submit();
262         document.body.removeChild(form);
263     }
264
265     function cetakLaporan() {
266         // Ambil semua parameter filter yang sedang aktif
267         const url = new URL(window.location);
268         const params = new URLSearchParams(url.search);
269
270         // Buat URL untuk cetak laporan dengan filter yang sama
271         let cetakUrl = '/laporan/cetak-pdf';
272         if (params.toString()) {
273             cetakUrl += '?' + params.toString();
274         }
275
276         // Buka di tab baru
277         window.open(cetakUrl, '_blank');
278     }
279 </script>
280 @endpush

```

Gambar 5. 44 Halaman laporan menampilkan daftar laporan pemesanan.

```

1  @extends('layout.main')
2
3  @section('page-title', 'Daftar Pemesanan')
4
5  @section('content')
6  <main class="flex-1 overflow-y-auto p-4">
7      <!-- Page Header -->
8      <div class="mb-6">
9          <div class="flex flex-col md:flex-row md:items-center md:justify-between">
10             <div>
11                 <h2 class="text-xl font-semibold text-gray-800">Daftar Pemesanan</h2>
12                 <p class="text-gray-600 mt-1">Kelola semua pemesanan lapangan</p>
13             </div>
14             <div class="mt-4 md:mt-0">
15                 <a href="{{ route('pemesanan.create') }}"
16                 class="inline-flex items-center px-4 py-2 bg-teal-600 hover:bg-teal-700 text-white font-medium rounded-lg transition-colors">
17                     <i class="fas fa-plus mr-2"></i>
18                     Buat Pemesanan
19                 </a>
20             </div>
21         </div>
22     </div>
23
24     <!-- Flash Messages -->
25     @if(session('success'))
26         <div class="mb-6 bg-green-100 border border-green-400 text-green-700 px-4 py-3 rounded relative" role="alert">
27             <span class="block sm:inline">{{ session('success') }}</span>
28             <button type="button" class="absolute top-0 bottom-0 right-0 px-4 py-3" onclick="this.parentElement.style.display='none'">
29                 <i class="fas fa-times"></i>
30             </button>
31         </div>
32     @endif
33
34     @if(session('error'))
35         <div class="mb-6 bg-red-100 border border-red-400 text-red-700 px-4 py-3 rounded relative" role="alert">
36             <span class="block sm:inline">{{ session('error') }}</span>
37             <button type="button" class="absolute top-0 bottom-0 right-0 px-4 py-3" onclick="this.parentElement.style.display='none'">
38                 <i class="fas fa-times"></i>
39             </button>
40         </div>
41     @endif
42
43     <!-- Pemesanan Table -->
44     <div class="bg-white rounded-lg shadow overflow-hidden">
45         <div class="px-6 py-4 border-b border-gray-200">
46             <h3 class="text-lg font-semibold text-gray-800">Data Pemesanan</h3>
47         </div>
48
49         <div class="overflow-x-auto">
50             <table class="min-w-full divide-y divide-gray-200">
51                 <thead class="bg-gray-50">
52                     <tr>
53                         <th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">Pemesanan</th>
54                         <th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">Lapangan</th>
55                         <th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">Jadwal</th>
56                         <th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">Total</th>
57                         <th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">Status</th>
58                         <th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">Pembayaran</th>
59                         <th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">Aksi</th>
60                     </tr>
61                 </thead>
62                 <tbody class="bg-white divide-y divide-gray-200">
63                     @foreach($pemesanans as $pemesanan)
64                         <tr class="hover:bg-gray-50">
65                             <td class="px-6 py-4 whitespace-nowrap">
66                                 <div>
67                                     <div class="text-sm font-medium text-gray-900">
68                                         {{ $pemesanan->user->name }}
69                                     </div>
70                                     <div class="text-sm text-gray-500">
71                                         {{ $pemesanan->tanggal_pemesanan->format('d/m/Y') }}
72                                     </div>
73                                     <div class="text-xs text-gray-400">
74                                         {{ $pemesanan->created_at->format('d/m/Y H:i') }}
75                                     </div>
76                                 </div>
77                             </td>
78                             <td class="px-6 py-4 whitespace-nowrap">
79                                 <div class="text-sm font-medium text-gray-900">
80                                     {{ $pemesanan->lapangan->nama_lapangan }}
81                                 </div>

```

Gambar 5. 45 Halaman pemesanan menampilkan pemesanan lapangan.

```

82 <div class="text-sm text-gray-500">
83   {{ ucfirst($pemesanan->lapangan->jenis) }}
84 </div>
85 </td>
86 <td class="px-6 py-4">
87   <div class="flex flex-wrap gap-1">
88     @foreach($json, decode($pemesanan->jadwal) as $jam)
89       <span class="inline-flex px-2 py-1 text-xs font-medium bg-blue-100 text-blue-800 rounded-md">
90         {{ \Carbon\Carbon::parse($jam)->format('H:i') }}
91       </span>
92     @endforeach
93   </div>
94 </td>
95 <td class="px-6 py-4 whitespace-nowrap text-sm font-medium text-gray-900">
96   Rp {{ number_format($pemesanan->total_harga, 0, ',', '.') }}
97 </td>
98 <td class="px-6 py-4 whitespace-nowrap">
99   <span class="inline-flex px-2 py-1 text-xs font-semibold rounded-full items-center">
100     @if($pemesanan->status == 'sukses') bg-green-100 text-green-800
101     @elseif($pemesanan->status == 'pending') bg-yellow-100 text-yellow-800
102     @else bg-red-100 text-red-800 @endif
103     @if($pemesanan->status == 'sukses')
104       <i class="fas fa-check-circle mr-1"></i>
105     @elseif($pemesanan->status == 'pending')
106       <i class="fas fa-clock mr-1"></i>
107     @else
108       <i class="fas fa-times-circle mr-1"></i>
109     @endif
110     {{ ucfirst($pemesanan->status) }}
111   </span>
112 </td>
113 <td class="px-6 py-4 whitespace-nowrap">
114   <div class="text-sm">
115     <div class="font-medium text-gray-900">
116       {{ ($pemesanan->pembayaran->metode ?? 'N/A') }}
117     </div>
118   </div>
119 </td>
120 <td class="px-6 py-4 whitespace-nowrap text-sm font-medium">
121   <div class="flex items-center space-x-2">
122     <a href="{{ route('pemesanan.show', $pemesanan) }}"
123       class="inline-flex items-center px-3 py-1 bg-blue-600 hover:bg-blue-700 text-white text-xs font-medium rounded-md transition-colors">
124       <i class="fas fa-eye mr-1"></i>
125       Detail
126     </a>
127
128     @if($pemesanan->status == 'pending' && $pemesanan->pembayaran->status == 'pending' && ($pemesanan->pembayaran->bukti_transfer))
129       <button onclick="openUploadModal('{{ $pemesanan->id }}')"
130         class="inline-flex items-center px-3 py-1 bg-green-600 hover:bg-green-700 text-white text-xs font-medium rounded-md transition-colors">
131         <i class="fas fa-upload mr-1"></i>
132         Upload Bukti
133       </button>
134     @endif
135
136     @if($pemesanan->pembayaran->bukti_transfer && $pemesanan->pembayaran->status == 'pending' && auth()->user()->role == 'admin')
137       <button onclick="openValidasiModal('{{ $pemesanan->id }}')"
138         class="inline-flex items-center px-3 py-1 bg-purple-600 hover:bg-purple-700 text-white text-xs font-medium rounded-md transition-colors">
139         <i class="fas fa-check mr-1"></i>
140         Validasi
141       </button>
142     @endif
143
144     @if($pemesanan->status == 'pending' && (auth()->user()->id == $pemesanan->user_id || auth()->user()->role == 'admin'))
145       <button onclick="confirmDelete('{{ $pemesanan->id }}', '{{ $pemesanan->lapangan->nama_lapangan }}')"
146         class="inline-flex items-center px-3 py-1 bg-red-600 hover:bg-red-700 text-white text-xs font-medium rounded-md transition-colors">
147         <i class="fas fa-times mr-1"></i>
148         Batal
149       </button>
150     @endif
151
152     <button onclick="cetakTiket('{{ $pemesanan->id }}')"
153       class="bg-blue-600 hover:bg-blue-700 text-white px-3 py-1 rounded text-xs inline-flex items-center">
154       <i class="fas fa-ticket-alt mr-1"></i>
155       Cetak tiket
156     </button>
157   </div>

```

Gambar 5. 46 Halaman pemesanan menampilkan pemesanan lapangan.

```

158         </td>
159     </tr>
160     @empty
161     <tr>
162         <td colspan="7" class="px-6 py-12 text-center">
163             <div class="flex flex-col items-center">
164                 <i class="fas fa-calendar-alt text-4xl text-gray-300 mb-4"></i>
165                 <p class="text-gray-500">Tidak ada data pemesanan</p>
166                 <a href="{ {{ route('pemesanan.create') }}"
167                     class="mr-4 inline-flex items-center px-4 py-2 bg-teal-600 hover:bg-teal-700 text-white font-medium rounded-lg transition-colors">
168                     <i class="fas fa-plus mr-2"></i>
169                     Buat Pemesanan Pertama
170                 </a>
171             </div>
172         </td>
173     </tr>
174     @endiforelse
175 </tbody>
176 </table>
177 </div>
178
179 <!-- Pagination -->
180 @if($pemesanans->hasPages())
181     <div class="px-6 py-4 border-t border-gray-200">
182         {{ $pemesanans->links() }}
183     </div>
184 @endif
185 </div>
186 </main>
187
188 <!-- Upload Bukti Modal -->
189 <div id="uploadModal" class="fixed inset-0 bg-black bg-opacity-50 flex items-center justify-center hidden z-50">
190     <div class="bg-white rounded-lg p-6 max-w-md w-full mx-4">
191         <div class="flex items-center mb-4">
192             <div class="w-12 h-12 bg-green-100 rounded-full flex items-center justify-center mr-4">
193                 <i class="fas fa-upload text-green-600"></i>
194             </div>
195             <h3 class="text-lg font-semibold text-gray-900">Upload Bukti Transfer</h3>
196         </div>
197
198         <form id="uploadForm" method="POST" enctype="multipart/form-data">
199             @csrf
200             <div class="mb-4">
201                 <label for="bukti_transfer" class="block text-sm font-medium text-gray-700 mb-2">
202                     Pilih File Bukti Transfer
203                 </label>
204                 <input type="file"
205                     id="bukti_transfer"
206                     name="bukti_transfer"
207                     accept="image/*"
208                     required
209                     class="w-full px-3 py-2 border border-gray-300 rounded-lg focus:ring-2 focus:ring-green-500 focus:border-transparent">
210             </div>
211             <p class="text-sm text-gray-500">Format: JPG, PNG, GIF. Maksimal 2MB</p>
212
213             <div class="flex justify-end space-x-3">
214                 <button type="button" onclick="closeUploadModal()" class="px-4 py-2 text-gray-600 hover:bg-gray-100 rounded-md transition-colors">
215                     Batal
216                 </button>
217                 <button type="submit" class="px-4 py-2 bg-green-600 hover:bg-green-700 text-white rounded-md transition-colors">
218                     Upload
219                 </button>
220             </div>
221         </form>
222     </div>
223 </div>
224
225 <!-- Validasi Modal -->
226 <div id="validasiModal" class="fixed inset-0 bg-black bg-opacity-50 flex items-center justify-center hidden z-50">
227     <div class="bg-white rounded-lg p-6 max-w-md w-full mx-4">
228         <div class="flex items-center mb-4">
229             <div class="w-12 h-12 bg-purple-100 rounded-full flex items-center justify-center mr-4">
230                 <i class="fas fa-check text-purple-600"></i>
231             </div>
232             <h3 class="text-lg font-semibold text-gray-900">Validasi Pembayaran</h3>
233         </div>

```

Gambar 5. 47 Halaman pemesanan menampilkan pemesanan lapangan.

```

234 </div>
235
236 <p class="text-gray-600 mb-4">Pilih status pembayaran untuk pemesanan ini:</p>
237
238 <form id="validasiForm" method="POST">
239   @csrf
240   <div class="mb-4">
241     <div class="space-y-2">
242       <label class="flex items-center">
243         <input type="radio" name="status_pembayaran" value="valid" class="mr-2" required>
244         <span class="text-green-600">Valid - Pembayaran diterima</span>
245       </label>
246       <label class="flex items-center">
247         <input type="radio" name="status_pembayaran" value="invalid" class="mr-2" required>
248         <span class="text-red-600">Invalid - Pembayaran ditolak</span>
249       </label>
250     </div>
251   </div>
252
253   <div class="flex justify-end space-x-3">
254     <button type="button" onclick="closeValidasiModal()" class="px-4 py-2 text-gray-600 hover:bg-gray-300 rounded-md transition-colors">
255       Batal
256     </button>
257     <button type="submit" class="px-4 py-2 bg-purple-600 hover:bg-purple-700 text-white rounded-md transition-colors">
258       Validasi
259     </button>
260   </div>
261 </form>
262 </div>
263 </div>
264 </div>
265
266 <!-- Delete Confirmation Modal -->
267 <div id="deleteModal" class="fixed inset-0 bg-black bg-opacity-50 flex items-center justify-center hidden z-50">
268   <div class="bg-white rounded-lg p-8 max-w-sm w-full mx-4">
269     <div class="flex items-center mb-4">
270       <div class="w-12 h-12 bg-red-100 rounded-full flex items-center justify-center mr-4">
271         <i class="fas fa-exclamation-triangle text-red-600"></i>
272       </div>
273       <h3 class="text-lg font-semibold text-gray-900">Konfirmasi Batalkan</h3>
274     </div>
275     <p class="text-gray-600 mb-6">Apakah Anda yakin ingin membatalkan pemesanan dengan id="lapanganName" class="font-italic">? Tindakan ini tidak dapat dibatalkan.</p>
276     <div class="flex justify-end space-x-3">
277       <button type="button" onclick="closeDeleteModal()" class="px-4 py-2 text-gray-600 hover:bg-gray-300 rounded-md transition-colors">
278         Batal
279       </button>
280       <button type="button" onclick="confirmDelete()" class="px-4 py-2 text-red-600 hover:bg-red-700 text-white rounded-md transition-colors">
281         Ya, Batalkan
282       </button>
283     </div>
284   </div>
285 </div>
286 </div>
287 </div>
288 </div>
289 </div>
290 @endsection
291
292 @push('scripts')
293   <script>
294     function cetakTiket(id) {
295       // Buat form hidden untuk mengirim request POST
296       let form = document.createElement('form');
297       form.method = 'GET';
298       form.action = '/laporan/cetak-pdf/' + id;
299       form.target = "_blank";
300
301       document.body.appendChild(form);
302       form.submit();
303       document.body.removeChild(form);
304     }
305
306     function openUploadModal(pemesananId) {
307       document.getElementById('uploadForm').action = '/pemesanan/' + pemesananId + '/upload-bukti';
308       document.getElementById('uploadModal').classList.remove('hidden');
309     }
310
311     function closeUploadModal() {
312       document.getElementById('uploadModal').classList.add('hidden');
313       document.getElementById('uploadForm').reset();
314     }
315
316     function openValidasiModal(pemesananId) {
317       document.getElementById('validasiForm').action = '/pemesanan/' + pemesananId + '/validasi-pembayaran';
318       document.getElementById('validasiModal').classList.remove('hidden');
319     }
320
321     function closeValidasiModal() {
322       document.getElementById('validasiModal').classList.add('hidden');
323       document.getElementById('validasiForm').reset();
324     }
325
326     function confirmDelete(pemesananId, lapanganName) {
327       document.getElementById('lapanganName').textContent = lapanganName;
328       document.getElementById('deleteForm').action = '/pemesanan/' + pemesananId;
329       document.getElementById('deleteModal').classList.remove('hidden');
330     }
331
332     function closeDeleteModal() {
333       document.getElementById('deleteModal').classList.add('hidden');
334     }
335
336     // Close modals when clicking outside
337     document.addEventListener('click', function(e) {
338       if (e.target.id === 'uploadModal') closeUploadModal();
339       if (e.target.id === 'validasiModal') closeValidasiModal();
340       if (e.target.id === 'deleteModal') closeDeleteModal();
341     });
342
343     // Close modals with Escape key
344     document.addEventListener('keydown', function(e) {
345       if (e.key === 'Escape') {
346         closeUploadModal();
347         closeValidasiModal();
348         closeDeleteModal();
349       }
350     });
351   </script>
352 @endpush

```

Gambar 5. 48 Halaman pemesanan menampilkan pemesanan lapangan.

```

312 document.getElementById('uploadModal').classList.add('hidden');
313 document.getElementById('uploadForm').reset();
314 }
315
316 function openValidasiModal(pemesananId) {
317   document.getElementById('validasiForm').action = '/pemesanan/' + pemesananId + '/validasi-pembayaran';
318   document.getElementById('validasiModal').classList.remove('hidden');
319 }
320
321 function closeValidasiModal() {
322   document.getElementById('validasiModal').classList.add('hidden');
323   document.getElementById('validasiForm').reset();
324 }
325
326 function confirmDelete(pemesananId, lapanganName) {
327   document.getElementById('lapanganName').textContent = lapanganName;
328   document.getElementById('deleteForm').action = '/pemesanan/' + pemesananId;
329   document.getElementById('deleteModal').classList.remove('hidden');
330 }
331
332 function closeDeleteModal() {
333   document.getElementById('deleteModal').classList.add('hidden');
334 }
335
336 // Close modals when clicking outside
337 document.addEventListener('click', function(e) {
338   if (e.target.id === 'uploadModal') closeUploadModal();
339   if (e.target.id === 'validasiModal') closeValidasiModal();
340   if (e.target.id === 'deleteModal') closeDeleteModal();
341 });
342
343 // Close modals with Escape key
344 document.addEventListener('keydown', function(e) {
345   if (e.key === 'Escape') {
346     closeUploadModal();
347     closeValidasiModal();
348     closeDeleteModal();
349   }
350 });
351 </script>
352 @endpush

```

Gambar 5. 49 Halaman pemesanan menampilkan pemesanan lapangan.

```

1 @extends('layout.main')
2
3 @section('page-title', 'Manajemen Pengguna')
4
5 @section('content')
6 <main class="flex-1 overflow-y-auto p-4">
7   <!-- Page Header -->
8   <div class="mb-6">
9     <h2 class="text-xl font-semibold text-gray-800">Manajemen Pengguna</h2>
10    <p class="text-gray-600 mt-1">Kelola data pengguna sistem</p>
11  </div>
12
13  <!-- Stats Cards -->
14  <div class="grid grid-cols-1 md:grid-cols-3 gap-6 mb-8">
15    <div class="bg-white rounded-lg shadow p-6">
16      <div class="flex items-center justify-between">
17        <div>
18          <p class="text-sm text-gray-500">Total Pengguna</p>
19          <p class="text-2xl font-semibold text-gray-800">{{ $users->total() }}</p>
20        </div>
21        <div class="w-12 h-12 bg-blue-100 rounded-full flex items-center justify-center">
22          <i class="fas fa-users text-blue-600"></i>
23        </div>
24      </div>
25    </div>
26
27    <div class="bg-white rounded-lg shadow p-6">
28      <div class="flex items-center justify-between">
29        <div>
30          <p class="text-sm text-gray-500">Penyewa</p>
31          <p class="text-2xl font-semibold text-gray-800">{{ $users->where('role', 'penyewa')->count() }}</p>
32        </div>
33        <div class="w-12 h-12 bg-green-100 rounded-full flex items-center justify-center">
34          <i class="fas fa-user text-green-600"></i>
35        </div>
36      </div>
37    </div>
38
39    <div class="bg-white rounded-lg shadow p-6">
40      <div class="flex items-center justify-between">
41        <div>
42          <p class="text-sm text-gray-500">Petugas</p>
43          <p class="text-2xl font-semibold text-gray-800">{{ $users->where('role', 'petugas')->count() }}</p>
44        </div>
45        <div class="w-12 h-12 bg-orange-100 rounded-full flex items-center justify-center">
46          <i class="fas fa-user-tie text-orange-600"></i>
47        </div>
48      </div>
49    </div>
50  </div>
51
52  <!-- Users Table -->
53  <div class="bg-white rounded-lg shadow overflow-hidden">
54    <div class="px-6 py-4 border-b border-gray-200">
55      <div class="flex flex-col md:flex-row md:items-center md:justify-between">
56        <h3 class="text-lg font-semibold text-gray-800">Data Pengguna</h3>
57        <div class="mt-4 md:mt-0 flex items-center space-x-3">
58          <!-- Search Box -->
59          <div class="relative">
60            <input type="text"
61              placeholder="Cari pengguna..."
62              class="pl-10 pr-4 py-2 border border-gray-300 rounded-lg focus:ring-2 focus:ring-teal-500 focus:border-transparent"
63              id="searchInput">
64            <i class="fas fa-search absolute left-3 top-3 text-gray-400"></i>
65          </div>
66          <!-- Filter Role -->
67          <select class="px-4 py-2 border border-gray-300 rounded-lg focus:ring-2 focus:ring-teal-500 focus:border-transparent" id="roleFilter">
68            <option value="">Semua Role</option>
69            <option value="penyewa">Penyewa</option>
70            <option value="petugas">Petugas</option>
71          </select>
72        </div>
73      </div>
74    </div>
75
76    <div class="overflow-x-auto">
77      <table class="min-w-full divide-y divide-gray-200" id="usersTable">
78        <thead class="bg-gray-50">
79          <tr>
80            <th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">ID</th>
81            <th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">Nama Lengkap</th>

```

Gambar 5. 50 Halaman user menampilkan daftar akun pengguna beserta identitas.

```

82 <th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">Username</th>
83 <th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">Email</th>
84 <th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">No. Telepon</th>
85 <th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">Role</th>
86 <th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">Tanggal Daftar</th>
87 <th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">Aksi</th>
88 </tr>
89 </thead>
90 <tbody class="bg-white divide-y divide-gray-200">
91 <@foreach($users as $user)>
92 <tr class="hover:bg-gray-50 user-row" data-role="{{ $user->role }}">
93 <td class="px-6 py-4 whitespace-nowrap text-sm font-medium text-gray-900">{{ str_pad($user->id, 4, '0', STR_PAD_LEFT) }}</td>
94 <td class="px-6 py-4 whitespace-nowrap">
95 <div class="flex items-center">
96 nama_lengkap }}">
97 <div class="ml-4">
98 <div class="text-sm font-medium text-gray-900">{{ $user->nama_lengkap }}</div>
99 </div>
100 </td>
101 <td class="px-6 py-4 whitespace-nowrap text-sm text-gray-500">{{ $user->username }}</td>
102 <td class="px-6 py-4 whitespace-nowrap text-sm text-gray-500">{{ $user->email }}</td>
103 <td class="px-6 py-4 whitespace-nowrap text-sm text-gray-500">{{ $user->no_telp }}</td>
104 <td class="px-6 py-4 whitespace-nowrap">
105 <div class="px-6 py-4 whitespace-nowrap">
106 <@if($user->role === 'petugas')>
107 <span class="inline-flex px-2 py-1 text-xs font-medium rounded-full bg-orange-100 text-orange-800">
108 <i class="fas fa-user-tie mr-1"></i>
109 Petugas
110 </span>
111 <@else>
112 <span class="inline-flex px-2 py-1 text-xs font-medium rounded-full bg-blue-100 text-blue-800">
113 <i class="fas fa-user mr-1"></i>
114 Penyewa
115 </span>
116 </div>
117 </td>
118 <td class="px-6 py-4 whitespace-nowrap text-sm text-gray-500">{{ $user->created_at->format('d M Y') }}</td>
119 <td class="px-6 py-4 whitespace-nowrap text-sm font-medium">
120 <button onclick="showUserDetail('{{ $user->id }}')"
121 <div class="inline-flex items-center px-3 py-1 bg-teal-600 hover:bg-teal-700 text-white text-xs font-medium rounded-md transition-colors">
122 <i class="fas fa-eye mr-1"></i>
123 Detail
124 </button>
125 </td>
126 </tr>
127 <@empty>
128 <tr>
129 <td colspan="8" class="px-6 py-12 text-center">
130 <div class="flex flex-col items-center">
131 <i class="fas fa-users text-4xl text-gray-300 mb-4"></i>
132 <p class="text-gray-500">Tidak ada data pengguna</p>
133 </div>
134 </td>
135 </tr>
136 </tbody>
137 </table>
138 </div>
139 </div>
140 <!-- Pagination -->
141 <@if($users->hasPages())>
142 <div class="px-6 py-4 border-t border-gray-200">
143 <div class="text-gray-500">{{ $users->links() }}</div>
144 </div>
145 </div>
146 </div>
147 </div>
148 </main>
149 </div>
150 <!-- Modal Detail User -->
151 <div id="userDetailModal" class="fixed inset-0 bg-black bg-opacity-50 flex items-center justify-center hidden z-50">
152 <div class="bg-white rounded-lg max-w-2xl w-full mx-4 max-h-[90vh] overflow-y-auto">
153 <div class="flex items-center justify-between p-6 border-b border-gray-200">
154 <div class="text-lg font-medium text-gray-900">Detail Pengguna</div>
155 <button onclick="closeUserDetail()" class="text-gray-400 hover:text-gray-600">
156 <i class="fas fa-times text-xl"></i>
157 </button>

```

Gambar 5. 51 Halaman user menampilkan daftar akun pengguna beserta identitas.

```

158 </div>
159
160 <div class="p-6">
161   <!-- Loading State -->
162   <div id="modalLoading" class="flex items-center justify-center py-12">
163     <div class="animate-spin rounded-full h-8 w-8 border-b-2 border-teal-600"></div>
164     <span class="ml-3 text-gray-600">Memuat data...</span>
165   </div>
166
167   <!-- User Detail Content -->
168   <div id="userDetailContent" class="hidden">
169     <div class="flex items-center mb-6">
170       
171       <div class="ml-6">
172         <h4 id="userName" class="text-xl font-semibold text-gray-900"></h4>
173         <p id="userEmail" class="text-gray-600"></p>
174         <span id="userRole" class="inline-flex px-2 py-1 text-xs font-semibold rounded-full mt-2"></span>
175       </div>
176     </div>
177
178     <div class="grid grid-cols-1 md:grid-cols-2 gap-6">
179       <div>
180         <h5 class="text-sm font-medium text-gray-500 uppercase tracking-wide mb-3">Informasi Pribadi</h5>
181         <div class="space-y-3">
182           <div>
183             <label class="block text-sm font-medium text-gray-700">ID Pengguna</label>
184             <p id="userId" class="mt-1 text-sm text-gray-900"></p>
185           </div>
186           <div>
187             <label class="block text-sm font-medium text-gray-700">Username</label>
188             <p id="userUsername" class="mt-1 text-sm text-gray-900"></p>
189           </div>
190           <div>
191             <label class="block text-sm font-medium text-gray-700">No. Telepon</label>
192             <p id="userPhone" class="mt-1 text-sm text-gray-900"></p>
193           </div>
194         </div>
195       </div>
196
197       <div>
198         <h5 class="text-sm font-medium text-gray-500 uppercase tracking-wide mb-3">Informasi Akun</h5>
199         <div class="space-y-3">
200           <div>
201             <label class="block text-sm font-medium text-gray-700">Tanggal Daftar</label>
202             <p id="userCreatedAt" class="mt-1 text-sm text-gray-900"></p>
203           </div>
204           <div>
205             <label class="block text-sm font-medium text-gray-700">Terakhir Diperbarui</label>
206             <p id="userUpdatedAt" class="mt-1 text-sm text-gray-900"></p>
207           </div>
208         </div>
209       </div>
210     </div>
211   </div>
212 </div>
213
214 <div class="flex justify-end px-6 py-4 border-t border-gray-200">
215   <button onclick="closeUserDetail()" class="px-4 py-2 bg-gray-300 hover:bg-gray-400 text-gray-800 rounded-md transition-colors">
216     Tutup
217   </button>
218 </div>
219 </div>
220 </div>
221 @endsection
222
223 @push('scripts')
224 <script>
225   function showUserDetail(userId) {
226     const modal = document.getElementById('userDetailModal');
227     const loading = document.getElementById('modalLoading');
228     const content = document.getElementById('userDetailContent');
229
230     // Show modal and loading
231     modal.classList.remove('hidden');
232     loading.classList.remove('hidden');
233     content.classList.add('hidden');
234
235     // Fetch user data

```

Gambar 5. 52 Halaman user menampilkan daftar akun pengguna beserta identitas.

```

235 // Fetch user data
236 fetch('/pengguna/${userId}', {
237   headers: {
238     'X-Requested-With': 'XMLHttpRequest',
239     'X-CSRF-TOKEN': document.querySelector('meta[name="csrf-token"]').getAttribute('content')
240   }
241 })
242 .then(response => response.json())
243 .then(data => {
244   // Hide loading and show content
245   loading.classList.add('hidden');
246   content.classList.remove('hidden');
247
248   // Populate modal with user data
249   document.getElementById('userId').textContent = `${String(data.id).padStart(4, '0')}`;
250   document.getElementById('userName').textContent = data.nama_lengkap;
251   document.getElementById('userUsername').textContent = data.username;
252   document.getElementById('userEmail').textContent = data.email;
253   document.getElementById('userPhone').textContent = data.no_telp;
254   document.getElementById('userCreatedAt').textContent = data.created_at;
255   document.getElementById('userUpdatedAt').textContent = data.updated_at;
256
257   // Set role badge
258   const roleElement = document.getElementById('userRole');
259   if (data.role === 'petugas') {
260     roleElement.className = 'inline-flex px-2 py-1 text-xs font-sans rounded-full bg-orange-100 text-orange-800';
261     roleElement.innerHTML = 'i class="fas fa-user-tie mr-1"></i>Petugas';
262   } else {
263     roleElement.className = 'inline-flex px-2 py-1 text-xs font-sans rounded-full bg-blue-100 text-blue-800';
264     roleElement.innerHTML = 'i class="fas fa-user mr-1"></i>Penyewa';
265   }
266
267   // Update avatar (you can customize this based on your needs)
268   document.getElementById('userAvatar').src = `https://placeholder.co/80/2dd4bf/ffffff/text-${data.nama_lengkap.charAt(0)}`;
269 })
270 .catch(error => {
271   console.error('Error:', error);
272   loading.innerHTML = '<div class="text-red-600">i class="fas fa-exclamation-triangle mr-2"></i>Terjadi kesalahan saat memuat data</div>';
273 });
274
275 function closeUserDetail() {
276   document.getElementById('userDetailModal').classList.add('hidden');
277 }
278
279 // Search functionality
280 document.getElementById('searchInput').addEventListener('input', function() {
281   const searchTerm = this.value.toLowerCase();
282   const rows = document.querySelectorAll('.user-row');
283
284   rows.forEach(row => {
285     const text = row.textContent.toLowerCase();
286     if (text.includes(searchTerm)) {
287       row.style.display = '';
288     } else {
289       row.style.display = 'none';
290     }
291   });
292 });
293
294 // Role filter functionality
295 document.getElementById('roleFilter').addEventListener('change', function() {
296   const selectedRole = this.value;
297   const rows = document.querySelectorAll('.user-row');
298
299   rows.forEach(row => {
300     const userRole = row.getAttribute('data-role');
301     if (selectedRole === '' || userRole === selectedRole) {
302       row.style.display = '';
303     } else {
304       row.style.display = 'none';
305     }
306   });
307 });
308
309 // Close modal when clicking outside
310 document.getElementById('userDetailModal').addEventListener('click', function(e) {
311   if (e.target.id === 'userDetailModal') {
312     closeUserDetail();
313   }

```

Gambar 5. 53 Halaman user menampilkan daftar akun pengguna beserta identitas.

```

314   }
315 });
316
317 // Close modal with Escape key
318 document.addEventListener('keydown', function(e) {
319   if (e.key === 'Escape') {
320     closeUserDetail();
321   }
322 });
323 </script>
324 @endpush

```

Gambar 5. 54 Halaman user menampilkan daftar akun pengguna beserta identitas.

5.1.6 Controller

Komponen controller berfungsi sebagai penghubung antara model dengan view. Tugas controller hanyalah memproses data dan permintaan yang masuk, kemudian memberitahu komponen model apa yang harus dikerjakan, dan hasilnya akan ditampilkan oleh komponen view.

```

1  <?php
2
3  namespace App\Http\Controllers;
4
5  use App\Models\User;
6  use Illuminate\Http\Request;
7  use Illuminate\Support\Facades\Auth;
8  use Illuminate\Support\Facades\Hash;
9  use Illuminate\Support\Facades\Validator;
10
11 class AuthController extends Controller
12 {
13     /**
14      * Show login form
15      */
16     public function showLoginForm()
17     {
18         return view('auth.login');
19     }
20
21     /**
22      * Handle login request
23      */
24     public function login(Request $request)
25     {
26         $validator = Validator::make($request->all(), [
27             'login' => 'required|string',
28             'password' => 'required|string',
29         ], [
30             'login.required' => 'Email/Username harus diisi',
31             'password.required' => 'Password harus diisi',
32         ]);
33
34         if ($validator->fails()) {
35             return back()->withErrors($validator)->withInput();
36         }
37
38         $credentials = $this->getCredentials($request);
39
40         if (Auth::attempt($credentials, $request->filled('remember'))) {
41             $request->session()->regenerate();
42             // Redirect semua user ke dashboard setelah login
43             return redirect()->intended('/dashboard')->with('success', 'Login berhasil!');
44         }
45
46         return back()->withErrors([
47             'login' => 'Email/Username atau password salah.',
48         ])->withInput();
49     }
50
51     /**
52      * Get credentials based on login type (email or username)
53      */
54     protected function getCredentials(Request $request)
55     {
56         $login = $request->input('login');
57         $password = $request->input('password');
58
59         $field = filter_var($login, FILTER_VALIDATE_EMAIL) ? 'email' : 'username';
60
61         return [
62             $field => $login,
63             'password' => $password
64         ];
65     }
66
67     /**
68      * Show register form
69      */
70     public function showRegisterForm()
71     {
72         return view('auth.register');
73     }
74
75     /**
76      * Handle register request
77      */
78     public function register(Request $request)
79     {
80         $validator = Validator::make($request->all(), [

```

Gambar 5. 55 AuthController mengatur autentikasi pengguna dalam aplikasi.

```

82         'nama_lengkap' => 'required|string|max:255',
83         'username' => 'required|string|max:255|unique:users,username',
84         'email' => 'required|string|email|max:255|unique:users,email',
85         'no_telp' => 'required|string|max:20|unique:users,no_telp',
86         'password' => 'required|string|min:8|confirmed',
87     ], [
88         'nama_lengkap.required' => 'Nama lengkap harus diisi',
89         'username.required' => 'Username harus diisi',
90         'username.unique' => 'Username sudah digunakan',
91         'email.required' => 'Email harus diisi',
92         'email.email' => 'Format email tidak valid',
93         'email.unique' => 'Email sudah digunakan',
94         'no_telp.required' => 'Nomor telepon harus diisi',
95         'no_telp.unique' => 'Nomor telepon sudah digunakan',
96         'password.required' => 'Password harus diisi',
97         'password.min' => 'Password minimal 8 karakter',
98         'password.confirmed' => 'Konfirmasi password tidak cocok',
99     ]
100 );
101 if ($validator->fails()) {
102     return back()->withErrors($validator)->withInput();
103 }
104
105 $user = User::create([
106     'nama_lengkap' => $request->nama_lengkap,
107     'username' => $request->username,
108     'email' => $request->email,
109     'no_telp' => $request->no_telp,
110     'password' => Hash::make($request->password)
111 ]);
112
113 Auth::login($user);
114
115 return redirect('/dashboard')->with('success', 'Registrasi berhasil! Selamat datang.');
```

Gambar 5. 56 AuthController mengatur autentikasi pengguna dalam aplikasi.

```

1  <?php
2
3  namespace App\Http\Controllers;
4
5  use Illuminate\Foundation\Auth\Access\AuthorizesRequests;
6  use Illuminate\Foundation\Validation\ValidatesRequests;
7  use Illuminate\Routing\Controller as BaseController;
8
9  class Controller extends BaseController
10 {
11     use AuthorizesRequests, ValidatesRequests;
12 }
13
```

Gambar 5. 57 Controller berisi kelas dasar aplikasi.

```

1  <?php
2
3  namespace App\Http\Controllers;
4
5  use App\Models\Pemesanan;
6  use Illuminate\Http\Request;
7  use Carbon\Carbon;
8
9  class JadwalController extends Controller
10 {
11     public function index()
12     {
13         return view('jadwal.index');
14     }
15
16     public function getEvents(Request $request)
17     {
18         // Ambil pemesanan dengan status sukses dan pembayaran valid
19         $pemesanans = Pemesanan::with(['lapangan', 'user', 'pembayaran'])
20             ->where('status', 'sukses')
21             ->whereHas('pembayaran', function($query) {
22                 $query->where('status', 'valid');
23             })
24             ->get();
25
26         $events = [];
27
28         foreach ($pemesanans as $pemesanan) {
29             // Parse jadwal JSON
30             $jadwalArray = is_string($pemesanan->jadwal)
31                 ? json_decode($pemesanan->jadwal, true)
32                 : $pemesanan->jadwal;
33
34             if ($jadwalArray && is_array($jadwalArray) && count($jadwalArray) >= 2) {
35                 // Ambil jam mulai dan jam selesai
36                 $jamMulai = $jadwalArray[0]; // "07:00:00"
37                 $jamSelesai = end($jadwalArray); // jam terakhir dalam array
38
39                 // Tambahkan 1 jam ke jam selesai untuk durasi
40                 $jamSelesaiPlusOne = Carbon::parse($jamSelesai)->addHour()->format('H:i:s');
41
42                 // Buat event untuk FullCalendar
43                 $events[] = [
44                     'id' => $pemesanan->id,
45                     'title' => $pemesanan->lapangan->nama_lapangan . ' - ' . $pemesanan->user->nama_lengkap,
46                     'start' => $pemesanan->tanggal_pemesanan->format('Y-m-d') . 'T' . $jamMulai,
47                     'end' => $pemesanan->tanggal_pemesanan->format('Y-m-d') . 'T' . $jamSelesaiPlusOne,
48                     'backgroundColor' => $this->getEventColor($pemesanan->lapangan->jenis),
49                     'borderColor' => $this->getEventColor($pemesanan->lapangan->jenis),
50                     'textColor' => 'ffffff',
51                     'extendedProps' => [
52                         'lapangan' => $pemesanan->lapangan->nama_lapangan,
53                         'jenis' => $pemesanan->lapangan->jenis_label,
54                         'penyewa' => $pemesanan->user->nama_lengkap,
55                         'phone' => $pemesanan->user->no_telp,
56                         'total_harga' => $pemesanan->total_harga_formatted,
57                         'metode_bayar' => $pemesanan->pembayaran->metode_formatted,
58                         'jadwal_detail' => $pemesanan->jadwal_formatted,
59                         'durasi' => count($jadwalArray) . ' jam'
60                     ]
61                 ];
62             }
63         }
64
65         return response()->json($events);
66     }
67
68     private function getEventColor($jenis)
69     {
70         $colors = [
71             'futsal' => '#108080', // green
72             'basket' => '#FFD700', // yellow
73             'badminton' => '#800080', // purple
74             'tenis' => '#FF0000', // red
75         ];
76
77         return $colors[$jenis] ?? '#667788'; // default gray
78     }
79 }

```

Gambar 5. 58 JadwalController mengatur pengelolaan jadwal pemesanan.

```

1  <?php
2
3  namespace App\Http\Controllers;
4
5  use App\Models\Lapangan;
6  use App\Models\Jadwal;
7  use Illuminate\Http\Request;
8  use Illuminate\Support\Facades\Storage;
9  use Illuminate\Support\Facades\DB;
10 use Illuminate\Support\Str;
11
12 class LapanganController extends Controller
13 {
14     public function index(Request $request)
15     {
16         $query = Lapangan::query();
17
18         // Search functionality
19         if ($request->has('search') && !empty($request->search)) {
20             $query->search($request->search);
21         }
22
23         // Filter by jenis
24         if ($request->has('jenis') && !empty($request->jenis)) {
25             $query->byJenis($request->jenis);
26         }
27
28         // Filter by status
29         if ($request->has('status')) {
30             if ($request->status == 'active') {
31                 $query->active();
32             } elseif ($request->status == 'inactive') {
33                 $query->where('is_active', false);
34             }
35         }
36
37         $lapangans = $query->latest()->paginate(10);
38
39         // Statistics
40         $stats = [
41             'total' => Lapangan::count(),
42             'active' => Lapangan::active()->count(),
43             'inactive' => Lapangan::where('is_active', false)->count(),
44             'futsal' => Lapangan::byJenis('futsal')->count(),
45             'basket' => Lapangan::byJenis('basket')->count(),
46             'badminton' => Lapangan::byJenis('badminton')->count(),
47             'tenis' => Lapangan::byJenis('tenis')->count(),
48         ];
49
50         if ($request->ajax()) {
51             return response()->json([
52                 'html' => view('lapangan.partials.table', compact('lapangans'))->render(),
53                 'pagination' => $lapangans->links()->render()
54             ]);
55         }
56
57         return view('lapangan.index', compact('lapangans', 'stats'));
58     }
59
60     public function create()
61     {
62         return view('lapangan.create');
63     }
64
65     public function store(Request $request)
66     {
67         $request->validate([
68             'nama_lapangan' => 'required|string|max:255',
69             'jenis' => 'required|in:futsal,basket,badminton,tenis',
70             'harga_per_jam' => 'required|numeric|min:0',
71             'deskripsi' => 'nullable|string',
72             'gambar.*' => 'nullable|image|mimes:jpeg,png,jpg,gif|max:2048',
73             'is_active' => 'boolean'
74         ]);
75
76         try {
77             DB::beginTransaction();
78
79             $data = $request->only(['nama_lapangan', 'jenis', 'harga_per_jam', 'deskripsi']);
80             $data['is_active'] = $request->has('is_active');
81

```

Gambar 5. 59 LapanganController mengatur pengelolaan data lapangan.

```

82 // Handle multiple image uploads
83 if ($request->hasFile('gambar')) {
84     $gambarPaths = [];
85     foreach ($request->file('gambar') as $file) {
86         $path = $file->store('lapangan', 'public');
87         $gambarPaths[] = $path;
88     }
89     $data['gambar'] = $gambarPaths;
90 }
91
92 $lapangan = Lapangan::create($data);
93
94 DB::commit();
95
96 return redirect()->route('lapangan.index')
97     ->with('success', 'Lapangan berhasil ditambahkan!');
98
99 } catch (\Exception $e) {
100     DB::rollBack();
101
102     // Delete uploaded files if error occurs
103     if (isset($gambarPaths)) {
104         foreach ($gambarPaths as $path) {
105             Storage::disk('public')->delete($path);
106         }
107     }
108
109     return back()->withInput()
110         ->with('error', 'Terjadi kesalahan: ' . $e->getMessage());
111 }
112
113 public function show(Lapangan $lapangan)
114 {
115     $lapangan = Lapangan::findOrFail($lapangan->id);
116
117     return view('lapangan.show', compact('lapangan'));
118 }
119
120 public function edit(Lapangan $lapangan)
121 {
122     return view('lapangan.edit', compact('lapangan'));
123 }
124
125 public function update(Request $request, Lapangan $lapangan)
126 {
127     $request->validate([
128         'nama_lapangan' => 'required|string|max:255',
129         'jenis' => 'required|in:futsal,basket,badminton,tenis',
130         'harga_per_jam' => 'required|numeric|min:0',
131         'deskripsi' => 'nullable|string',
132         'gambar.*' => 'nullable|image|mimes:jpeg,png,jpg,gif|max:2048',
133         'is_active' => 'boolean',
134         'existing_images' => 'nullable|array'
135     ]);
136
137     try {
138         DB::beginTransaction();
139
140         $data = $request->only(['nama_lapangan', 'jenis', 'harga_per_jam', 'deskripsi']);
141         $data['is_active'] = $request->has('is_active');
142
143         // Handle image updates
144         $existingImages = $request->input('existing_images', []);
145         $gambarPaths = [];
146
147         // Keep selected existing images
148         if ($lapangan->gambar) {
149             foreach ($lapangan->gambar as $oldImage) {
150                 if (in_array($oldImage, $existingImages)) {
151                     $gambarPaths[] = $oldImage;
152                 } else {
153                     // Delete unselected images
154                     Storage::disk('public')->delete($oldImage);
155                 }
156             }
157         }
158
159         // Add new images

```

Gambar 5. 60 LapanganController mengatur pengelolaan data lapangan.

```

161         if ($request->hasFile('gambar')) {
162             foreach ($request->file('gambar') as $file) {
163                 $path = $file->store('lapangan', 'public');
164                 $gambarPaths[] = $path;
165             }
166         }
167
168         $data['gambar'] = $gambarPaths;
169         $lapangan->update($data);
170
171         DB::commit();
172
173         return redirect()->route('lapangan.show', $lapangan)
174             ->with('success', 'Lapangan berhasil diperbarui!');
175     } catch (\Exception $e) {
176         DB::rollBack();
177         return back()->withInput()
178             ->with('error', 'Terjadi kesalahan: ' . $e->getMessage());
179     }
180 }
181
182 public function destroy(Lapangan $lapangan)
183 {
184     try {
185         DB::beginTransaction();
186
187         // Check if lapangan has any bookings
188         $hasBookings = $lapangan->jadwals()->exists();
189         if ($hasBookings) {
190             return back()->with('error', 'Lapangan tidak dapat dihapus karena masih ada pemesanan aktif!');
191         }
192
193         // Delete images
194         $lapangan->deleteGambar();
195
196         // Delete lapangan (jadwals will be deleted automatically due to cascade)
197         $lapangan->delete();
198
199         DB::commit();
200
201         return redirect()->route('lapangan.index')
202             ->with('success', 'Lapangan berhasil dihapus!');
203     } catch (\Exception $e) {
204         DB::rollBack();
205         return back()->with('error', 'Terjadi kesalahan: ' . $e->getMessage());
206     }
207 }
208
209 public function toggleStatus(Lapangan $lapangan)
210 {
211     try {
212         $lapangan->update([
213             'is_active' => !$lapangan->is_active
214         ]);
215
216         $status = $lapangan->is_active ? 'diaktifkan' : 'dinonaktifkan';
217
218         return response()->json([
219             'success' => true,
220             'message' => "Lapangan berhasil ($status)!",
221             'is_active' => $lapangan->is_active
222         ]);
223     } catch (\Exception $e) {
224         return response()->json([
225             'success' => false,
226             'message' => 'Terjadi kesalahan: ' . $e->getMessage()
227         ], 500);
228     }
229 }
230
231 public function deleteImage(Request $request, Lapangan $lapangan)
232 {
233     $request->validate([
234         'image_path' => 'required|string'
235     ]);
236 }

```

Gambar 5. 61 LapanganController mengatur pengelolaan data lapangan.

```

240     try {
241         $imagePath = $request->image_path;
242         $gambarArray = $lapangan->gambar ?? [];
243
244         if (($key = array_search($imagePath, $gambarArray)) !== false) {
245             // Remove from array
246             unset($gambarArray[$key]);
247
248             // Delete physical file
249             Storage::disk('public')->delete($imagePath);
250
251             // Update database
252             $lapangan->update(['gambar' => array_values($gambarArray)]);
253
254             return response()->json([
255                 'success' => true,
256                 'message' => 'Gambar berhasil dihapus!'
257             ]);
258         }
259
260         return response()->json([
261             'success' => false,
262             'message' => 'Gambar tidak ditemukan!'
263         ], 404);
264     } catch (\Exception $e) {
265         return response()->json([
266             'success' => false,
267             'message' => 'Terjadi kesalahan: ' . $e->getMessage()
268         ], 500);
269     }
270 }
271
272 }

```

Gambar 5. 62 LapanganController mengatur pengelolaan data lapangan.

```

1  <?php
2
3  namespace App\Http\Controllers;
4
5  use App\Models\Pemesanan;
6  use Illuminate\Http\Request;
7  use Barryvdh\DomPDF\Facade\Pdf;
8  use Carbon\Carbon;
9
10 class LaporanController extends Controller
11 {
12     public function index(Request $request)
13     {
14         $query = Pemesanan::with(['lapangan', 'user', 'pembayaran'])
15             ->orderBy('tanggal_pemesanan', 'desc')
16             ->orderBy('created_at', 'desc');
17
18         $user = auth()->user();
19         if ($user->role === 'penyewa') {
20             $query->where('user_id', $user->id);
21         }
22
23         // Filter berdasarkan status pemesanan
24         if ($request->filled('status')) {
25             $query->where('status', $request->status);
26         }
27
28         // Filter berdasarkan status pembayaran
29         if ($request->filled('status_bayar')) {
30             $query->whereHas('pembayaran', function($q) use ($request) {
31                 $q->where('status', $request->status_bayar);
32             });
33         }
34
35         // Filter berdasarkan tanggal
36         if ($request->filled('tanggal_dari')) {
37             $query->whereDate('tanggal_pemesanan', '>=', $request->tanggal_dari);
38         }
39
40         if ($request->filled('tanggal_sampai')) {
41             $query->whereDate('tanggal_pemesanan', '<=', $request->tanggal_sampai);
42         }
43
44         // Filter berdasarkan jenis lapangan
45         if ($request->filled('jenis')) {
46             $query->whereHas('lapangan', function($q) use ($request) {
47                 $q->where('jenis', $request->jenis);
48             });
49         }
50
51         // Search berdasarkan nama penyewa atau lapangan
52         if ($request->filled('search')) {
53             $search = $request->search;
54             $query->where(function($q) use ($search) {
55                 $q->whereHas('user', function($subQ) use ($search) {
56                     $subQ->where('nama_lengkap', 'like', "%{$search}%")
57                         ->orWhere('username', 'like', "%{$search}%");
58                 })->orWhereHas('lapangan', function($subQ) use ($search) {
59                     $subQ->where('nama_lapangan', 'like', "%{$search}%");
60                 });
61             });
62         }
63
64         $pemesanans = $query->paginate(15);
65
66         // Statistik untuk dashboard
67         $stats = [
68             'total_pemesanan' => Pemesanan::count(),
69             'pemesanan_sukses' => Pemesanan::where('status', 'sukses')->count(),
70             'pemesanan_pending' => Pemesanan::where('status', 'pending')->count(),
71             'total_pendapatan' => Pemesanan::where('status', 'sukses')
72                 ->whereHas('pembayaran', function($q) {
73                     $q->where('status', 'valid');
74                 })->sum('total_harga'),
75             'pemesanan_hari_ini' => Pemesanan::whereDate('tanggal_pemesanan', today())->count()
76         ];
77
78         return view('laporan.index', compact('pemesanans', 'stats'));
79     }
80
81     public function cetakPdf(Request $request, $id = null)

```

Gambar 5. 63 LaporanController mengatur pencetakan laporan pemesanan.

```

82 {
83     if ($id) {
84         // Cetak satu pemesanan (struk/tiket)
85         $pemesanan = Pemesanan::with(['lapangan', 'user', 'pembayaran'])
86             ->findOrFail($id);
87
88         $pdf = Pdf::loadView('laporan.pdf.tiket', compact('pemesanan'))
89             ->setPaper('a5', 'portrait');
90
91         $filename = 'Tiket-Pemesanan-' . $pemesanan->id . '-' . date('YmdHis') . '.pdf';
92
93         return $pdf->download($filename);
94     } else {
95         // Cetak laporan keseluruhan
96         $query = Pemesanan::with(['lapangan', 'user', 'pembayaran'])
97             ->orderBy('tanggal_pemesanan', 'desc');
98
99         // Apply same filters as index
100         if ($request->filled('status')) {
101             $query->where('status', $request->status);
102         }
103         if ($request->filled('status_bayar')) {
104             $query->whereHas('pembayaran', function($q) use ($request) {
105                 $q->where('status', $request->status_bayar);
106             });
107         }
108         if ($request->filled('tanggal_dari')) {
109             $query->whereDate('tanggal_pemesanan', '>=', $request->tanggal_dari);
110         }
111         if ($request->filled('tanggal_sampai')) {
112             $query->whereDate('tanggal_pemesanan', '<=', $request->tanggal_sampai);
113         }
114         if ($request->filled('jenis')) {
115             $query->whereHas('lapangan', function($q) use ($request) {
116                 $q->where('jenis', $request->jenis);
117             });
118         }
119
120         $pemesanans = $query->get();
121
122         $stats = [
123             'total_pemesanan' => $pemesanans->count(),
124             'total_pendapatan' => $pemesanans->where('status', 'sukses')
125                 ->filter(function($p) {
126                     return $p->pembayaran && $p->pembayaran->status == 'valid';
127                 })->sum('total_harga'),
128             'periode' => [
129                 'dari' => $request->tanggal_dari ? 'Semua',
130                 'sampai' => $request->tanggal_sampai ? 'Semua'
131             ]
132         ];
133
134         $pdf = Pdf::loadView('laporan.pdf.laporan', compact('pemesanans', 'stats'))
135             ->setPaper('a4', 'landscape');
136
137         $filename = 'Laporan-Pemesanan-' . date('YmdHis') . '.pdf';
138
139         return $pdf->download($filename);
140     }
141 }
142 }

```

Gambar 5. 64 LaporanController mengatur pencetakan laporan pemesanan.

```

1  <?php
2
3  namespace App\Http\Controllers;
4
5  use App\Models\Jadwal;
6  use App\Models\Lapangan;
7  use App\Models\Pembayaran;
8  use App\Models\Pemesanan;
9  use Illuminate\Http\Request;
10 use Illuminate\Support\Facades\Auth;
11 use Illuminate\Support\Facades\DB;
12 use Illuminate\Support\Facades\Storage;
13
14 class PemesananController extends Controller
15 {
16     /**
17      * Display a listing of the resource.
18      */
19     public function index()
20     {
21         $user = Auth::user();
22
23         $query = Pemesanan::with(['lapangan', 'user', 'pembayaran'])
24             ->latest();
25
26         if ($user->role === 'penyewa') {
27             $query->where('user_id', $user->id);
28         }
29
30         $pemesanans = $query->paginate(10);
31
32         return view('pemesanan.index', compact('pemesanans'));
33     }
34
35     /**
36      * Show the form for creating a new resource.
37      */
38     public function create()
39     {
40         $lapangans = Lapangan::where('is_active', true)->get();
41         $jadwals = Jadwal::where('is_active', true)->orderBy('jam')->get();
42
43         return view('pemesanan.create', compact('lapangans', 'jadwals'));
44     }
45
46     /**
47      * Store a newly created resource in storage.
48      */
49     public function store(Request $request)
50     {
51         $request->validate([
52             'lapangan_id' => 'required|exists:lapangans,id',
53             'tanggal_pemesanan' => 'required|date|after_or_equal:today',
54             'jadwal' => 'required|array|min:1',
55             'jadwal.*' => 'required|exists:jadwals,id',
56             'metode_pembayaran' => 'required|string|in:Transfer Bank,Tunai,E-Wallet',
57         ]);
58
59         DB::beginTransaction();
60
61         try {
62             $lapangan = Lapangan::findOrFail($request->lapangan_id);
63             $user = Auth::user();
64             $tanggal = $request->tanggal_pemesanan;
65
66             // Ambil jam dari ID jadwal yang dipilih
67             $jadwalDipilih = Jadwal::whereIn('id', $request->jadwal)
68                 ->orderBy('jam')
69                 ->get(['id', 'jam'])
70                 ->pluck('jam')
71                 ->toArray();
72
73             // Validasi: Cek bentrok jadwal dengan pemesanan yang sudah ada
74             $existingPemesanan = Pemesanan::where('lapangan_id', $lapangan->id)
75                 ->where('tanggal_pemesanan', $tanggal)
76                 ->whereIn('status', ['pending', 'sukses'])
77                 ->get();
78
79             foreach ($existingPemesanan as $pemesanan) {
80                 $existingJadwal = json_decode($pemesanan->jadwal, true) ?? [];
81                 $intersection = array_intersect($jadwalDipilih, $existingJadwal);

```

Gambar 5. 65 PemesananController mengatur proses pemesanan lapangan.

```

82
83         if (empty($intersection)) {
84             $conflictJam = implode(' ', $intersection);
85             return back();
86             ->with('error', "Jadwal pada jam {$conflictJam} sudah dipesan untuk tanggal {$tanggal}.");
87             ->withInput();
88         }
89     }
90
91     // Hitung total harga
92     $totalHarga = $lapangan->harga_per_jam * count($jadwalDipilih);
93
94     // Buat pemesanan baru
95     $pemesanan = Pemesanan::create([
96         'user_id' => $user->id,
97         'lapangan_id' => $lapangan->id,
98         'jadwal' => json_encode($jadwalDipilih), // Simpan sebagai JSON
99         'tanggal_pemesanan' => $tanggal,
100         'total_harga' => $totalHarga,
101         'status' => 'pending'
102     ]);
103
104     // Buat pembayaran terkait
105     Pembayaran::create([
106         'pemesanan_id' => $pemesanan->id,
107         'metode' => $request->metode_pembayaran,
108         'status' => 'pending'
109     ]);
110
111     DB::commit();
112
113     return redirect()->route('pemesanan.index')
114         ->with('success', 'Pemesanan berhasil dibuat. Silakan lakukan pembayaran.');
```

```

115     } catch (\Exception $e) {
116         DB::rollBack();
117         return back();
118         ->with('error', 'Terjadi kesalahan: ' . $e->getMessage());
119         ->withInput();
120     }
121 }
122
123 /**
124  * Display the specified resource.
125  */
126 public function show(Pemesanan $pemesanan)
127 {
128     $this->authorize('view', $pemesanan);
129
130     $pemesanan->load(['lapangan', 'user', 'pembayaran']);
131
132     return view('pemesanan.show', compact('pemesanan'));
133 }
134
135 /**
136  * Update the specified resource in storage.
137  */
138 public function update(Request $request, Pemesanan $pemesanan)
139 {
140     $this->authorize('update', $pemesanan);
141
142     $request->validate([
143         'status' => 'required|in:pending,sukses,gagal'
144     ]);
145
146     DB::beginTransaction();
147
148     try {
149         $pemesanan->update(['status' => $request->status]);
150
151         // Update status pembayaran sesuai status pemesanan
152         if ($request->status === 'sukses') {
153             $pemesanan->pembayaran()->update(['status' => 'valid']);
154         } elseif ($request->status === 'gagal') {
155             $pemesanan->pembayaran()->update(['status' => 'invalid']);
156         }
157     }
158
159     DB::commit();
160 }

```

Gambar 5. 66 PemesananController mengatur proses pemesanan lapangan.

```

161         return back()->with('success', 'Status pemesanan berhasil diperbarui');
162     } catch (\Exception $e) {
163         DB::rollBack();
164         return back()->with('error', 'Terjadi kesalahan: ' . $e->getMessage());
165     }
166 }
167
168 /**
169  * Remove the specified resource from storage.
170  */
171 public function destroy(Pemesanan $pemesanan)
172 {
173     $this->authorize('delete', $pemesanan);
174
175     // Hapus bukti transfer jika ada
176     if ($pemesanan->pembayaran && $pemesanan->pembayaran->bukti_transfer) {
177         Storage::disk('public')->delete($pemesanan->pembayaran->bukti_transfer);
178     }
179
180     $pemesanan->delete();
181
182     return redirect()->route('pemesanan.index')
183         ->with('success', 'Pemesanan berhasil dibatalkan');
184 }
185
186 /**
187  * Check available schedules
188  */
189 public function checkJadwal(Request $request)
190 {
191     $request->validate([
192         'lapangan_id' => 'required|exists:lapangans,id',
193         'tanggal' => 'required|date|after_or_equal:today'
194     ]);
195
196     $lapanganId = $request->lapangan_id;
197     $tanggal = $request->tanggal;
198
199     // Dapatkan semua jadwal yang sudah dipesan pada tanggal dan lapangan tersebut
200     $bookedJadwals = [];
201     $pemesanans = Pemesanan::where('lapangan_id', $lapanganId)
202         ->where('tanggal_pemesanan', $tanggal)
203         ->whereIn('status', ['pending', 'sukses'])
204         ->get();
205
206     foreach ($pemesanans as $pemesanan) {
207         $jadwal = json_decode($pemesanan->jadwal, true) ?? [];
208         $bookedJadwals = array_merge($bookedJadwals, $jadwal);
209     }
210
211     // Hilangkan duplikat
212     $bookedJadwals = array_unique($bookedJadwals);
213
214     // Dapatkan ID jadwal yang sudah dipesan
215     $bookedJadwalIds = Jadwal::whereIn('jam', $bookedJadwals)->pluck('id')->toArray();
216
217     return response()->json([
218         'booked' => $bookedJadwalIds
219     ]);
220 }
221
222 /**
223  * Upload bukti transfer
224  */
225 public function uploadBukti(Request $request, Pemesanan $pemesanan)
226 {
227     $this->authorize('update', $pemesanan);
228
229     $request->validate([
230         'bukti_transfer' => 'required|image|mimes:jpeg,png,jpg,gif|max:2048'
231     ]);
232
233     DB::beginTransaction();
234
235     try {
236         if ($request->hasFile('bukti_transfer')) {
237             // Hapus bukti transfer lama jika ada
238             if ($pemesanan->pembayaran->bukti_transfer) {
239                 Storage::disk('public')->delete($pemesanan->pembayaran->bukti_transfer);
240             }
241         }
242     } catch (\Exception $e) {
243         DB::rollBack();
244         return back()->with('error', 'Terjadi kesalahan: ' . $e->getMessage());
245     }
246
247     $pemesanan->pembayaran->bukti_transfer = $request->file('bukti_transfer');
248     $pemesanan->pembayaran->save();
249
250     return back()->with('success', 'Bukti transfer berhasil diupload');
251 }

```

Gambar 5. 67 PemesananController mengatur proses pemesanan lapangan.

```

240         Storage::disk('public')->delete($pesanan->payment->bukti_transfer);
241     }
242
243     // Upload bukti transfer baru
244     $path = $request->file('bukti_transfer')->store('bukti_transfer', 'public');
245
246     // Update pembayaran
247     $pesanan->payment->update([
248         'bukti_transfer' => $path,
249         'tanggal_bayar' => now(),
250         'status' => 'pending' // Status menunggu validasi admin
251     ]);
252
253     DB::commit();
254
255     return back()->with('success', 'Bukti transfer berhasil diupload. Menunggu validasi petugas.');
```

```

256     }
257
258     return back()->with('error', 'Gagal mengupload bukti transfer');
```

```

259     } catch (\Exception $e) {
260         DB::rollBack();
261         return back()->with('error', 'Terjadi kesalahan: ' . $e->getMessage());
262     }
263 }
264
265 /**
266  * Validasi pembayaran (untuk admin)
267  */
268 public function validasiPembayaran(Request $request, Pesanan $pesanan)
269 {
270     $this->authorize('update', $pesanan);
271
272     $request->validate([
273         'status_pembayaran' => 'required|in:valid,invalid'
274     ]);
275
276     DB::beginTransaction();
277
278     try {
279         $statusPesanan = $request->status_pembayaran === 'valid' ? 'sukses' : 'gagal';
280
281         // Update status pesanan
282         $pesanan->update(['status' => $statusPesanan]);
283
284         // Update status pembayaran
285         $pesanan->payment->update(['status' => $request->status_pembayaran]);
286
287         DB::commit();
288
289         $message = $request->status_pembayaran === 'valid'
290             ? 'Pembayaran berhasil divalidasi'
291             : 'Pembayaran ditolak';
292
293         return back()->with('success', $message);
294     } catch (\Exception $e) {
295         DB::rollBack();
296         return back()->with('error', 'Terjadi kesalahan: ' . $e->getMessage());
297     }
298 }
299
300 }
301

```

Gambar 5. 68 PemesananController mengatur proses pemesanan lapangan.

```

1  <?php
2
3  namespace App\Http\Controllers;
4
5  use App\Models\User;
6  use Illuminate\Http\Request;
7
8  class UserController extends Controller
9  {
10     public function index()
11     {
12         $users = User::orderBy('created_at', 'desc')->paginate(10);
13         return view('users.index', compact('users'));
14     }
15
16     public function show(User $user)
17     {
18         return response()->json([
19             'id' => $user->id,
20             'nama_lengkap' => $user->nama_lengkap,
21             'username' => $user->username,
22             'email' => $user->email,
23             'no_telp' => $user->no_telp,
24             'role' => $user->role,
25             'created_at' => $user->created_at->format('d M Y H:i'),
26             'updated_at' => $user->updated_at->format('d M Y H:i')
27         ]);
28     }
29 }

```

Gambar 5. 69 UserController mengatur pengelolaan data pengguna.

5.2 Testing

Pengujian dilakukan untuk memverifikasi kesesuaian aplikasi dengan harapan serta meningkatkan kualitas sistem. Metode yang digunakan adalah alpha testing dengan pendekatan black box, yang berfokus pada pengujian fungsionalitas aplikasi. Berikut hasil pengujian yang diperoleh.

Tabel 5. 4 Alpha testing menggunakan black box testing

Test Scenario	Test Case	Test Step	Expected	Actual	Pass/Fail
TS-01 Cek fungsi registrasi akun	TC-1.1 Memastikan pengguna dapat melakukan registrasi akun	Buka aplikasi Pilih menu register Input data akun baru Klik submit	Sistem menyimpan data akun baru dan menampilkan notifikasi "registrasi berhasil"	Sistem menyimpan data akun baru dan menampilkan notifikasi "registrasi berhasil"	Pass
	TC-1.2 Memastikan validasi input registrasi	Input data tidak lengkap Klik submit	Sistem menolak input dan menampilkan pesan error validasi	Sistem menolak input dan menampilkan pesan error validasi	Pass
TS-02 Cek fungsi login	TC-2.1 Memastikan pengguna dapat login dengan akun valid	Buka aplikasi Pilih menu login Input username dan password valid Klik submit	Sistem menampilkan dashboard sesuai profil pengguna	Sistem menampilkan dashboard sesuai profil pengguna	Pass
	TC-2.2 Memastikan sistem menolak login dengan akun tidak valid	Input username dan password salah Klik submit	Sistem menampilkan pesan "username/password salah"	Sistem menampilkan pesan "username/password salah"	Pass
TS-03 Cek fungsi pemesanan lapangan	TC-3.1 Memastikan penyewa dapat melakukan pemesanan	Login sebagai penyewa Pilih menu <i>pemesanan</i> Pilih lapangan dan jadwal Klik pesan	Sistem menampilkan invoice dan status pemesanan pending	Sistem menampilkan invoice dan status pemesanan pending	Pass
	TC-3.2 Memastikan sistem menolak pemesanan jadwal yang sudah terisi	Pilih lapangan dan jadwal yang sudah dipesan Klik pesan	Sistem menolak pemesanan dan menampilkan pesan "jadwal tidak tersedia"	Sistem menolak pemesanan dan menampilkan pesan "jadwal tidak tersedia"	Pass
TS-04 Cek fungsi pembayaran	TC-4.1 Memastikan penyewa dapat	Login sebagai penyewa	Sistem memvalidasi bukti pembayaran	Sistem memvalidasi bukti pembayaran	Pass

Test Scenario	Test Case	Test Step	Expected	Actual	Pass/Fail
	mengunggah bukti pembayaran	Pilih menu pembayaran Upload bukti transfer Klik submit	dan mengubah status menjadi valid	dan mengubah status menjadi valid	
TS-05 Cek fungsi informasi pemesanan	TC-5.1 Memastikan penyewa dapat melihat detail pemesanan	Login sebagai penyewa Pilih menu info pemesanan	Sistem menampilkan detail reservasi terkini	Sistem menampilkan detail reservasi terkini	Pass
TS-06 Cek fungsi admin mengelola data	TC-6.1 Memastikan admin dapat melihat daftar penyewa	Login sebagai admin Pilih menu data penyewa	Sistem menampilkan daftar seluruh penyewa	Sistem menampilkan daftar seluruh penyewa	Pass
	TC-6.2 Memastikan admin dapat mengelola data lapangan	Login sebagai admin Pilih menu data lapangan Tambah/Ubah data lapangan	Sistem menyimpan perubahan data lapangan	Sistem menyimpan perubahan data lapangan	Pass

Setelah skenario pengujian dijelaskan pada Tabel 5.4, hasil pelaksanaan alpha testing ditunjukkan pada Tabel 5.5 yang memuat jumlah test case serta persentase keberhasilan uji.

Tabel 5. 5 Hasil Alpha testing

No	Test Scenario	Jumlah Test Case	Persentasi Lulus Uji
1	TS-01 Cek fungsi registrasi akun	2	100%
2	TS-02 Cek fungsi login	2	100%
3	TS-03 Cek fungsi pemesanan lapangan	2	100%
4	TS-04 Cek fungsi pembayaran	1	100%
5	TS-05 Cek fungsi informasi pemesanan	1	100%
6	TS-06 Cek fungsi admin mengelola data	2	100%

Berdasarkan hasil rekapitulasi pengujian alpha testing menggunakan metode Black Box Testing, seluruh skenario uji yang telah dirancang dapat dijalankan dengan baik. Dari total test scenario yang diuji, sebagian besar berhasil lolos dengan persentase keberhasilan 100%, menunjukkan bahwa fitur-fitur utama aplikasi seperti registrasi, login, pemesanan lapangan, pembayaran, serta pengelolaan data oleh admin telah berfungsi sesuai kebutuhan. Meskipun demikian, terdapat beberapa fungsi yang masih perlu diperbaiki agar integrasi sistem dengan basis data dapat berjalan lebih optimal. Secara keseluruhan, hasil pengujian ini membuktikan bahwa aplikasi pemesanan online lapangan berbasis web

yang dikembangkan sudah layak digunakan dan mampu mendukung proses penyewaan lapangan di GOR Ewangga Kuningan dengan lebih efisien.

BAB 6

KESIMPULAN DAN SARAN

Bab ini menjelaskan mengenai kesimpulan yang didapatkan dari penelitian yang telah dilakukan serta saran yang diberikan oleh penulis untuk perkembangan pada penelitian selanjutnya.

6.1 Kesimpulan

Penelitian ini berhasil merancang dan membangun aplikasi pemesanan online lapangan berbasis web menggunakan framework Laravel yang sesuai dengan kebutuhan GOR Ewangga Kuningan. Sistem yang dikembangkan mampu mengatasi kelemahan metode manual, seperti kesalahan jadwal, keterbatasan informasi, dan kurangnya efisiensi pelayanan. Fitur utama berupa registrasi pengguna, pemesanan lapangan, pengecekan jadwal real-time, pembayaran digital, serta pengelolaan data oleh admin telah berjalan dengan baik. Dengan penerapan metode Waterfall, pengembangan sistem menjadi lebih terstruktur dan sistematis. Secara keseluruhan, aplikasi ini meningkatkan efisiensi operasional, transparansi informasi, dan kepuasan pengguna.

6.2 Saran

Untuk pengembangan lebih lanjut, aplikasi ini dapat ditingkatkan dengan menambahkan fitur notifikasi otomatis agar pengguna mendapat pengingat jadwal dan status pembayaran. Integrasi dengan layanan pembayaran online juga disarankan guna mempermudah transaksi. Selain itu, fitur laporan keuangan dan statistik pemesanan akan membantu admin dalam memantau kinerja GOR. Tampilan antarmuka sebaiknya dioptimalkan agar lebih responsif di berbagai perangkat, khususnya smartphone. Terakhir, pengujian keamanan data perlu dilakukan secara berkala untuk menjaga keandalan sistem dan melindungi informasi pengguna.

DAFTAR PUSTAKA

- [AJI21] Hidayat, A. N., Iskandar, D., & Nofiyati, N. (2021). Sistem Informasi Marketplace Penyewaan Barang Berbasis Web Dengan Framework Laravel. *Jurnal Ilmu Komputer Dan Informatika*, 1(2), 75-98.
- [AHM19] Hidayat, A., & Piliang, F. (2019). Rancang bangun sistem informasi penyewaan lahan parkir berbasis web gis. *Jurnal Sistem Informasi Dan Sains Teknologi*, 1(1).
- [CHA23] Kesuma, C. (2023). Rancang Bangun Sistem Informasi Penyewaan Mobil Berbasis Web. *Jurnal ICT: Information Communication & Technology*, 23(1), 124-128.
- [NAT24] Natasya, N., & Putri, R. N. (2024). Rancang Bangun Sistem Informasi Penyewaan dan Rekomendasi Lahan Pertanian Menggunakan Metode KNN. *Jurnal Sistem Informasi dan Teknik Informasi*, 4(1).
- [PRE10] Pressman, R. S. (2010). A practitioner's approach. *Software Engineering*, 2, 41-42.
- [PUT23] Putra, B. J. M., Purwatama, A., & Purnamasari, P. O. D. A. (2023). Pengembangan Aplikasi Reservasi Salon Keshoo Berbasis Web. *Journal of Electrical, Electronic, Mechanical, Informatic and Social Applied Science*, 2(2), 24-30.
- [RAH19] Rahmi, F. C., & Pramono, M. (2019). Manajemen pengelolaan fasilitas olahraga gedung serbaguna di gelangang olahraga (GOR) delta Sidoarjo. *Jurnal Kesehatan Olahraga*, 7(2), 1-6.
- [SHO22] Sholihin, N., & Ardiansyah, M. (2022). *Membangun Web Dengan Framework Laravel 8*. Pascal Books.
- [SOM11] Sommerville, I. (2011). *Software engineering 9th Edition*. ISBN-10, 137035152(18), 2.
- [WIR19] Wirawan, A., Prasetyo, D. H., & Absari, D. T. (2019). Pembuatan Website Penyewaan Prasarana Olahraga Berbasis Lapangan. *CALYPTRA*, 7(2), 4696-4710.

LAMPIRAN



PEMERINTAH DAERAH KABUPATEN KUNINGAN
DINAS PEMUDA OLAHRAHA DAN PARIWISATA

Jl. Raya Cilowa No. 40 A. Telp. (0232) 871378 Kramatmulya
 KUNINGAN

Kode POS 45553

REKOMENDASI

NOMOR : 000.9.6 / 579 / SE/PA/2025

Menindaklanjuti surat Ketua Prodi Teknik Informatika Universitas Pasundan Bandung Nomor 203/Unpas.FT.IF/Q/VII/2025 tanggal 29 Juli 2025 hal Penelitian Tugas Akhir, untuk bahan Penyusunan Skripsi Mahasiswa diharuskan melaksanakan Penelitian.

Sehubungan hal tersebut, dengan ini kami memberikan rekomendasi dan mengijinkan untuk melaksanakan kegiatan dimaksud pada Dinas Pemuda Olahraga dan Pariwisata Kabupaten Kuningan, dengan data Identitas sebagai berikut :

Nama Mahasiswa : **MOCHAMMAD NIZAR ALBAEHAQI**
 NRP : 203040035
 Program Studi : Teknik Informatika

Demikian rekomendasi ini untuk dipergunakan sebagaimana mestinya.

Kuningan, 3 Agustus 2025

KEPALA DINAS
PEMUDA OLAHRAHA DAN PARIWISATA
KABUPATEN KUNINGAN



Drs. ASEP BUDI SETIAWAN, M.Si
 Pembina Utama Muda
 NIP. 19660513 199203 1 007



PEMERINTAH DAERAH KABUPATEN KUNINGAN
DINAS PEMUDA OLAHRAGA DAN PARIWISATA

Jl. Raya Cilowa No. 40A. Telp. (0232) 871378 Kramatmulya

Kode Pos: Kuningan 45553

**REKOMENDASI PENGGUNAAN GEDUNG OLAHRAGA EWANGGA
 KABUPATEN KUNINGAN**

NOMOR : 426.21/ ³⁸³ /UPTD PORAPAR

Menindaklanjuti Surat dari ASOSIASI FUTSAL KABUPATEN KUNINGAN, Nomor 0171/AFKAB-KNG/XII/2023, Tanggal 1 Desember 2023 Hal Permohonan Izin dan Rekomendasi Penggunaan Tempat di Gedung Olahraga Ewangga Kuningan untuk kegiatan "**Kejuaraan Futsal SUPER CUP Tahun 2023**", pada prinsipnya kami mendukung dan memberikan Ijin Penggunaan Gedung Olahraga Ewangga Kuningan pada :

Hari : Rabu – Jum'at
 Tanggal : 3 - 12 Januari 2024
 Waktu : Pukul 11.00 WIB s.d Selesai
 Tempat : Gedung Olahraga Ewangga Kuningan

Dengan tetap mematuhi dan memperhatikan hal-hal sebagai berikut :

1. Menjaga keamanan, ketertiban, dan kebersihan fasilitas;
2. Menjaga kelestarian alam dan lingkungan sekitar;
3. Menempuh prosedur perijinan dari instansi terkait;
4. Apabila ada kepentingan acara Bupati dan Wakil Bupati serta kegiatan Pemerintah Kabupaten Kuningan, maka pihak penyewa Gedung Olahraga Ewangga harus mengikuti agenda kegiatan tersebut;
5. Hal-hal yang berkenaan dengan pelaksanaan kegiatan sepenuhnya menjadi tanggung jawab penyelenggara;
6. Pihak penyewa harus tunduk dan taat pada aturan pengguna gedung/tempat dan hukum yang berlaku.
7. Agar maklum dan menjadi bahan seperlunya.

Ditetapkan : DI KUNINGAN
 Pada tanggal : 20 Desember 2023

KEPALA DINAS
 PEMUDA OLAHRAGA DAN PARIWISATA
 KABUPATEN KUNINGAN



Dr. CARLAN, S.Pd., M.MPd
 Pembina Tingkat I
 NIP. 19720312 200604 1011

Tembusan :

- Yth. 1. Kepala UPTD Porapar Kab. Kuningan
 2. Kapolsek Kuningan





**PANITIA PELAKSANA
TURNAMEN BASKET BALL SE KABUPATEN KUNINGAN
PERBASI CUP I
TAHUN 2025**

Sekretariat : Gg. Angsana, Kuningan, Kec. Kuningan, Kabupaten Kuningan, Jawa Barat 45511



**INDONESIA
BASKETBALL**

Nomor : 001/PanPel/PERBASI Cup I /V/2025

Lampiran : -

Perihal : PERMOHONAN IZIN TEMPAT

Kuningan, 22 Mei 2025

Kepada

Yth.
di.
Tempat

Dengan hormat,

Dipermaklumkan dengan hormat kami atas nama Panitia Pelaksana Turnamen Basket Ball Se Kabupaten Kuningan Perbasi Cup I Tahun 2025 dan sesuai dengan hasil musyawarah pihak Panitia Turnamen Basket Ball Perbasi Cup dengan Pengurus club serta kepelatihan tingkat pelajar Sekabupaten Kuningan, bahwa kegiatan tersebut diatas akan dilaksanakan pada :

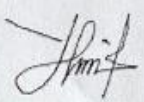
Hari, Tanggal : Senin, 18 Agustus 2025 S/d 24 Agustus 2025

Tempat : Gedung Olahraga Ewangga Kabupaten Kuningan

Dengan ini kami memohon ijin tempat sarana dan prasarana untuk berlangsungnya kegiatan tersebut.

Demikian surat permohonan izin tempat ini kami sampaikan, atas kerjasama dan perhatiannya kami ucapkan terima kasih.

Ketua Pelaksana



Yayan Ariyana, S.Pd



Sekretaris



Rubi Mahrulizal, S.Kep., Ners

