

BAB 3

METODOLOGI PENELITIAN

3.1 Tools dan Teknologi

1. Framework Laravel: Digunakan untuk mengembangkan aplikasi web.
2. MySQL: Sebagai database untuk menyimpan data pengguna, jadwal lapangan, dan transaksi pemesanan.
3. HTML, CSS, JavaScript: Untuk pengembangan antarmuka pengguna yang responsif.
4. FIFO (First In First Out) : Digunakan sebagai metode pemesanan lapangan untuk memastikan urutan pemesanan berdasarkan kedatangan.

3.2 Tahapan Pengembangan Sistem

1. Kebutuhan Sistem:
 - (a) Pemesanan Lapangan dilakukan secara online.
 - (b) Sistem harus mendukung metode FIFO.
 - (c) Sistem harus menyediakan informasi jadwal lapangan secara real time.
2. Desain Sistem :
 - (a) Desain antarmuka yang mudah digunakan (user-friendly).
 - (b) Desain database untuk mengelola data pengguna, lapangan, dan pemesanan.
3. Implementasi :
 - (a) Pemograman aplikasi berbasis web menggunakan Laravel.
 - (b) Integrasi algoritma *First In First Out (FIFO)* dalam proses pemesanan.
4. Pengujian :
 - (a) Uji unit terhadap fungsi - fungsi pemesanan.
 - (b) Uji integrasi untuk memastikan sistem berjalan dengan baik.

3.3 Perancangan Sistem

Tahap perancangan sistem merupakan bagian penting dalam proses pengembangan perangkat lunak, khususnya dalam membangun aplikasi pemesanan lapangan badminton berbasis web. Perancangan sistem dilakukan setelah tahap analisis kebutuhan selesai, dengan tujuan untuk menjabarkan secara rinci struktur sistem, hubungan antar komponen, serta alur proses dan data yang akan mengalir dalam sistem yang dibangun.

Perancangan dilakukan untuk memastikan bahwa implementasi aplikasi sesuai dengan kebutuhan fungsional pengguna, sekaligus mengoptimalkan efisiensi dalam pengelolaan data dan proses pemesanan lapangan. Dengan pendekatan yang terstruktur, diharapkan sistem dapat meminimalisir kesalahan dalam manajemen jadwal dan mempercepat proses pemesanan.

3.3.1 Tujuan Perancangan

Tujuan utama dari tahap ini adalah:

1. Mendeskripsikan komponen-komponen utama sistem secara teknis dan fungsional.
2. Merancang struktur basis data yang mencakup entitas, relasi, serta atribut penting yang digunakan oleh sistem.
3. Merancang antarmuka pengguna (UI) agar mudah digunakan, responsif, dan sesuai dengan kebutuhan operasional GOR.
4. Merancang logika alur sistem, baik dari sisi pengguna (user) maupun pengelola/admin.
5. Menyusun skema implementasi metode *First In First Out* dalam penjadwalan booking.

3.3.2 Pendekatan Pengembangan Sistem

Aplikasi ini dirancang menggunakan pendekatan *Model-View-Controller* (MVC) yang secara default didukung oleh framework Laravel. Pendekatan ini memungkinkan pemisahan yang jelas antara logika aplikasi, tampilan antarmuka

pengguna, dan pengelolaan data, sehingga mempermudah proses pengembangan dan pemeliharaan sistem.

Dalam pengelolaan pemesanan lapangan, sistem menerapkan prinsip antrian menggunakan struktur data *Queue* (antrian). Setiap permintaan pemesanan yang masuk akan ditempatkan dalam antrian berdasarkan urutan waktu pengajuan. Sistem kemudian memproses pemesanan satu per satu secara berurutan, memastikan bahwa tidak terjadi tabrakan jadwal dan pengguna pertama yang memesan akan diproses terlebih dahulu. Pendekatan ini menjamin keadilan dan efisiensi dalam pengalokasian slot waktu lapangan.

Penggunaan Laravel juga mendukung pengelolaan autentikasi pengguna, validasi input, pengaturan rute (*routing*), serta integrasi dengan database secara efisien. Dengan kombinasi arsitektur MVC dan algoritma antrian *Queue*, sistem mampu memberikan layanan pemesanan yang terstruktur, responsif, dan mudah digunakan oleh pengguna.

3.3.3 Perancangan Struktur Basis Data

Struktur basis data dirancang untuk mendukung proses pencatatan pengguna, data lapangan, proses pemesanan (booking), dan pembayaran (order). Beberapa entitas utama dalam sistem antara lain:

1. Users: Menyimpan informasi akun pengguna dan admin, termasuk nama, email, password, dan peran (role).
2. Fields: Menyimpan data lapangan seperti nama lapangan, harga sewa per jam, dan gambar.
3. Bookings: Mencatat setiap transaksi pemesanan, termasuk referensi ke user dan field, tanggal, jam mulai dan selesai, serta status booking.
4. Orders: Mencatat pembayaran pengguna, bukti transfer, metode pembayaran, dan status validasi oleh admin.

Relasi antar tabel dirancang menggunakan *foreign key* yang menjamin integritas data antar entitas. Diagram relasi antar entitas dijabarkan dalam bentuk *Entity Relationship Diagram* (ERD) pada bagian tersendiri.

3.3.4 Perancangan Alur Sistem

Alur sistem secara umum terbagi menjadi dua bagian: dari sisi pengguna dan dari sisi admin/pengelola GOR.

1. Pengguna:

- (a) Mendaftar dan login ke sistem.
- (b) Melihat daftar lapangan dan mengecek jadwal yang tersedia.
- (c) Melakukan booking satu atau beberapa lapangan berdasarkan jadwal yang masih kosong.
- (d) Mengunggah bukti pembayaran sesuai metode yang dipilih.
- (e) Melihat riwayat booking, status pembayaran, serta progres validasi oleh admin.

2. Admin:

- (a) Login sebagai admin dan mengakses dashboard.
- (b) Melihat daftar pemesanan dan status pembayaran.
- (c) Memvalidasi pembayaran dengan melihat bukti pembayaran yang diunggah pengguna.
- (d) Mengelola data lapangan dan mengatur informasi harga.
- (e) Melihat dan mengekspor laporan pembayaran harian atau bulanan dalam bentuk PDF.

Alur ini divisualisasikan dalam diagram *Flowchart* dan *Data Flow Diagram* (DFD) yang menunjukkan bagaimana data mengalir dan bagaimana proses saling berinteraksi.

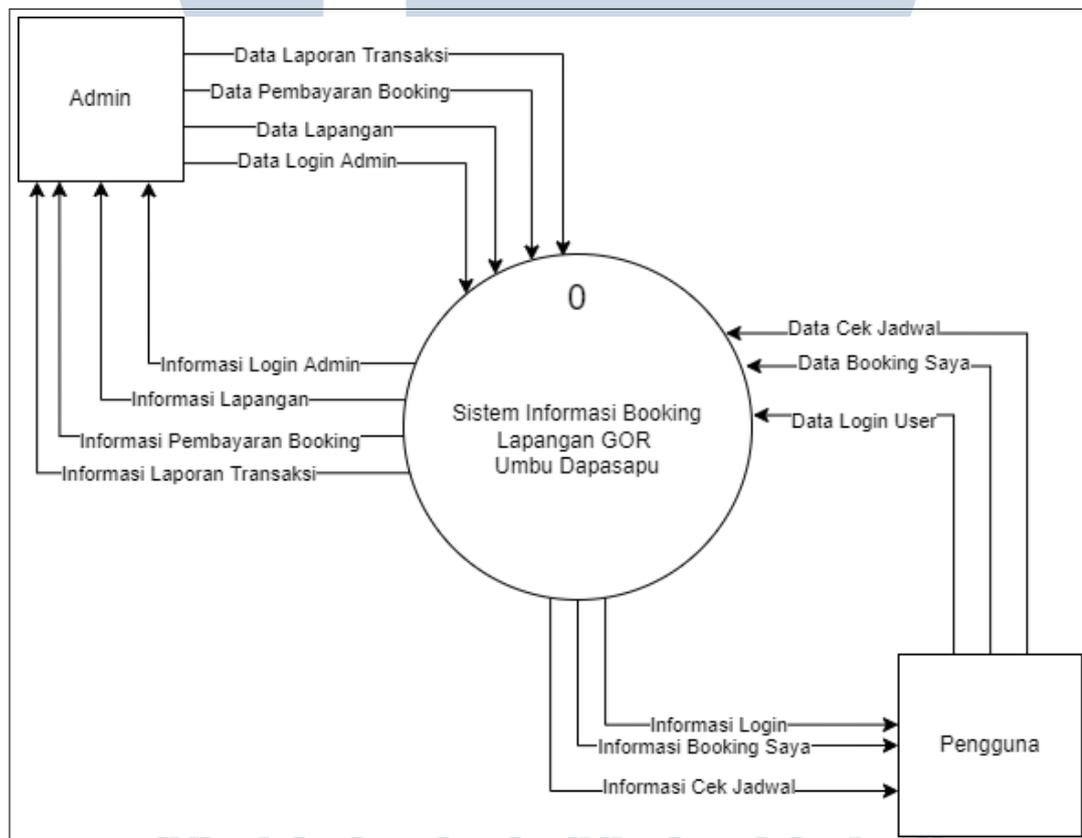
3.4 Data Flow Diagram (DFD)

Data Flow Diagram (DFD) pada aplikasi pemesanan lapangan badminton berbasis web digunakan untuk menggambarkan aliran data antara pengguna, sistem, dan penyimpanan data secara logis. DFD ini terdiri dari entitas eksternal seperti *User* dan *Admin* yang berinteraksi dengan sistem melalui berbagai proses, seperti pendaftaran, login, pemesanan lapangan, pembayaran, dan validasi.

DFD menunjukkan bahwa pengguna dapat melakukan booking lapangan dan mengunggah bukti pembayaran, sedangkan admin bertugas memverifikasi pembayaran serta mengelola data lapangan dan laporan transaksi. Setiap proses dalam sistem terhubung dengan *data store* seperti *Data Lapangan*, *Data Booking*, dan *Data Order* untuk menyimpan dan mengelola informasi yang dibutuhkan secara efisien.

3.4.1 DFD Level 0

DFD Level 0 ini menyajikan gambaran umum interaksi Sistem Informasi Booking Lapangan GOR Umbu Dapasapu dengan dua aktor eksternal utamanya, yaitu Admin dan Pengguna.



Gambar 3.1. DFD Level 0 Sistem Informasi Booking Lapangan

Gambar 3.1 menjelaskan, Admin bertanggung jawab untuk menyediakan data penting seperti laporan transaksi, detail pembayaran booking, informasi lapangan, dan kredensial login admin ke dalam sistem. Sebagai balasan, sistem akan

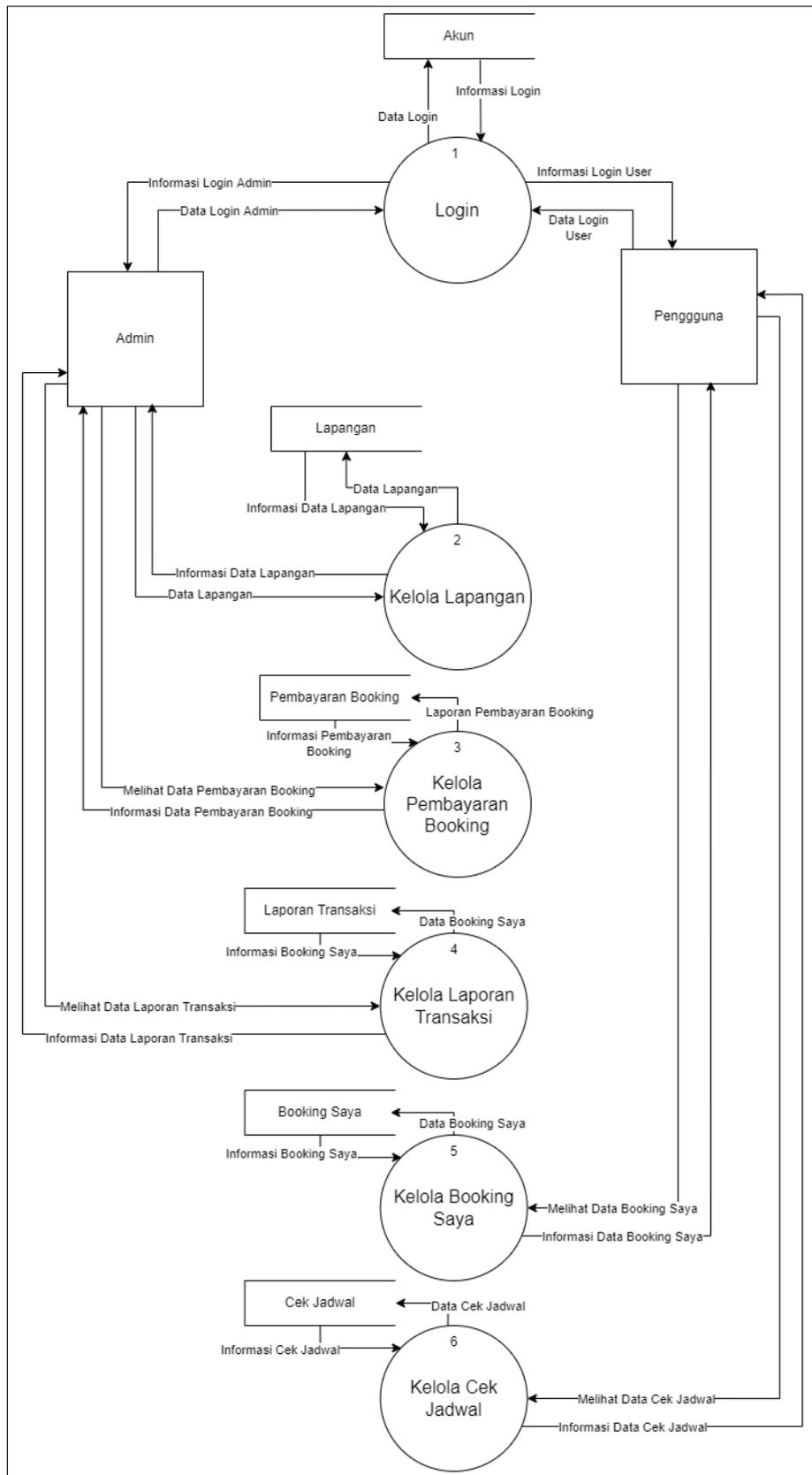
memberikan informasi login, detail lapangan, status pembayaran booking, dan laporan transaksi kepada Admin.

Di sisi lain, Pengguna berinteraksi dengan sistem untuk mengirimkan data login mereka, meminta data kalender ketersediaan, melakukan cek jadwal, dan mengirimkan data booking atau pemesanan lapangan. Sebagai respons, sistem akan mengembalikan informasi login, konfirmasi booking, detail jadwal, dan data kalender kepada Pengguna

3.4.2 DFD Level 1

DFD Level 1 ini menguraikan proses-proses internal yang terjadi di dalam Sistem Informasi Booking Lapangan GOR Umbu Dapasapu berikut penjelasan untuk gambar 3.2.

1. Login
Mengelola otentikasi baik untuk Admin maupun Pengguna. Proses ini menggunakan entitas penyimpanan data *Akun*.
2. Kelola Lapangan
Memungkinkan Admin untuk mengelola informasi lapangan. Interaksi melibatkan penyimpanan data *Lapangan*.
3. Kelola Pembayaran Booking
Menangani pembayaran yang dilakukan oleh Pengguna, dan divalidasi oleh Admin. Data disimpan dalam *Pembayaran Booking*.
4. Kelola Laporan Transaksi
Admin dapat melihat laporan transaksi yang diambil dari penyimpanan data *Laporan Transaksi*.
5. Kelola Booking Saya
Pengguna dapat melihat dan mengelola informasi booking pribadi, tersimpan dalam *Booking Saya*.
6. Kelola Cek Jadwal
Pengguna melakukan pengecekan ketersediaan jadwal, berinteraksi dengan *Cek Jadwal*.



Gambar 3.2. DFD Level 1 Sistem Informasi Booking Lapangan

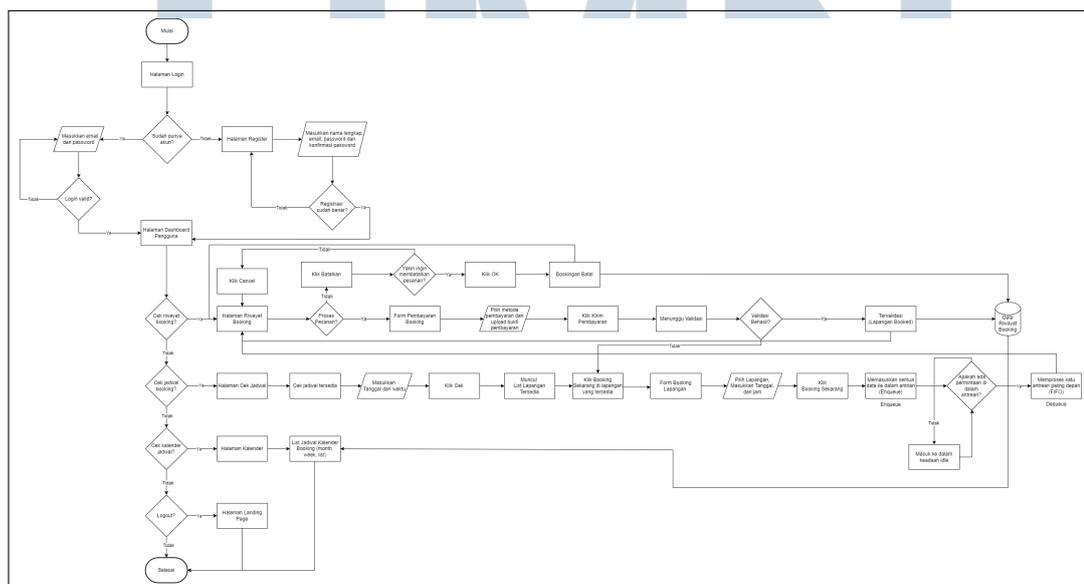
3.5 Flowchart

Flowchart atau diagram alur merupakan representasi grafis dari langkah-langkah atau proses dalam sebuah sistem. Flowchart menyajikan alur kerja secara visual menggunakan simbol-simbol standar seperti persegi panjang untuk proses, belah ketupat untuk pengambilan keputusan, oval untuk awal dan akhir proses, serta panah sebagai penunjuk arah alur. Tujuan utama dari flowchart adalah untuk menggambarkan logika sistem secara sistematis dan mudah dipahami oleh berbagai pihak, termasuk pengembang, analis sistem, dan pengguna non-teknis.

Dengan adanya flowchart, proses analisis dan pengembangan sistem menjadi lebih terstruktur karena setiap tahapan dalam sistem ditampilkan secara runtut dan logis. Flowchart juga memudahkan dalam mendeteksi kesalahan atau redundansi dalam suatu proses sebelum diimplementasikan ke dalam kode program. Oleh karena itu, flowchart sangat berguna sebagai alat bantu dokumentasi dan komunikasi selama proses rekayasa perangkat lunak berlangsung.

3.5.1 Flowchart User

Flowchart user digunakan untuk menjelaskan alur penggunaan aplikasi dari sisi pengguna, mulai dari login, cek jadwal, booking lapangan, hingga upload bukti pembayaran. Tujuannya agar proses dalam sistem lebih mudah dipahami dan mempermudah pengembangan serta penggunaan aplikasi.



Gambar 3.3. Flowchart User

Gambar 3.3 Flowchart sistem ini menggambarkan proses pemesanan lapangan badminton yang dimulai dari pengguna membuka aplikasi, login, melakukan pemesanan lapangan, hingga proses validasi pembayaran. Adapun langkah-langkah secara rinci adalah sebagai berikut:

1. Mulai

Proses dimulai ketika pengguna mengakses sistem atau aplikasi pemesanan lapangan badminton.

2. Halaman Login

Pengguna diarahkan ke halaman login untuk memasukkan email dan password. Sistem akan memverifikasi apakah pengguna sudah memiliki akun.

3. Proses Registrasi

Jika pengguna belum memiliki akun, maka diarahkan ke halaman registrasi untuk mengisi data seperti:

- (a) Nama lengkap
- (b) Alamat email
- (c) Password dan konfirmasi password

Setelah data diisi, sistem akan memverifikasi validitas data. Jika valid, data disimpan ke dalam database. Jika tidak valid, pengguna diminta mengulang proses.

4. Proses Login

Jika pengguna telah memiliki akun, maka dapat langsung login. Sistem akan memverifikasi:

- (a) Jika data valid, pengguna diarahkan ke halaman dashboard pengguna.
- (b) Jika tidak valid, pengguna diminta mengulang proses login.

5. Halaman Dashboard Pengguna

Pada halaman ini, pengguna diberikan beberapa opsi:

- (a) Melihat riwayat booking
- (b) Melihat jadwal booking
- (c) Melihat kalender pemesanan

(d) Logout dari sistem

6. Riwayat Booking

Jika pengguna memilih melihat riwayat booking:

- (a) Sistem menampilkan daftar pemesanan yang telah dilakukan.
- (b) Pengguna dapat membatalkan pemesanan tertentu.
- (c) Sistem akan meminta konfirmasi.
 - i. Jika ya, sistem membatalkan dan menampilkan status *Bookingan Batal*.
 - ii. Jika tidak, sistem kembali ke halaman riwayat booking.

7. Cek Jadwal Booking

Untuk memeriksa ketersediaan lapangan:

- (a) Pengguna memasukkan tanggal dan jam yang diinginkan.
- (b) Menekan tombol *Cek*.
- (c) Sistem menampilkan lapangan yang tersedia.
- (d) Pengguna menekan tombol *Booking Sekarang*.
- (e) Dilanjutkan ke halaman form booking.

8. Form Pembayaran Booking

Setelah booking dikonfirmasi:

- (a) Pengguna memilih metode pembayaran.
- (b) Mengunggah bukti pembayaran.
- (c) Menekan tombol *Kirim Pembayaran*.

9. Validasi Pembayaran

Sistem memproses validasi:

- (a) Jika pembayaran valid, status berubah menjadi *Tervalidasi*.
- (b) Jika tidak valid, pengguna diarahkan ke riwayat untuk mengunggah ulang bukti pembayaran.

10. Cek Kalender Jadwal

Pengguna dapat mengakses tampilan jadwal dalam format:

(a) *Month*

(b) *Week*

(c) *List*

11. Logout

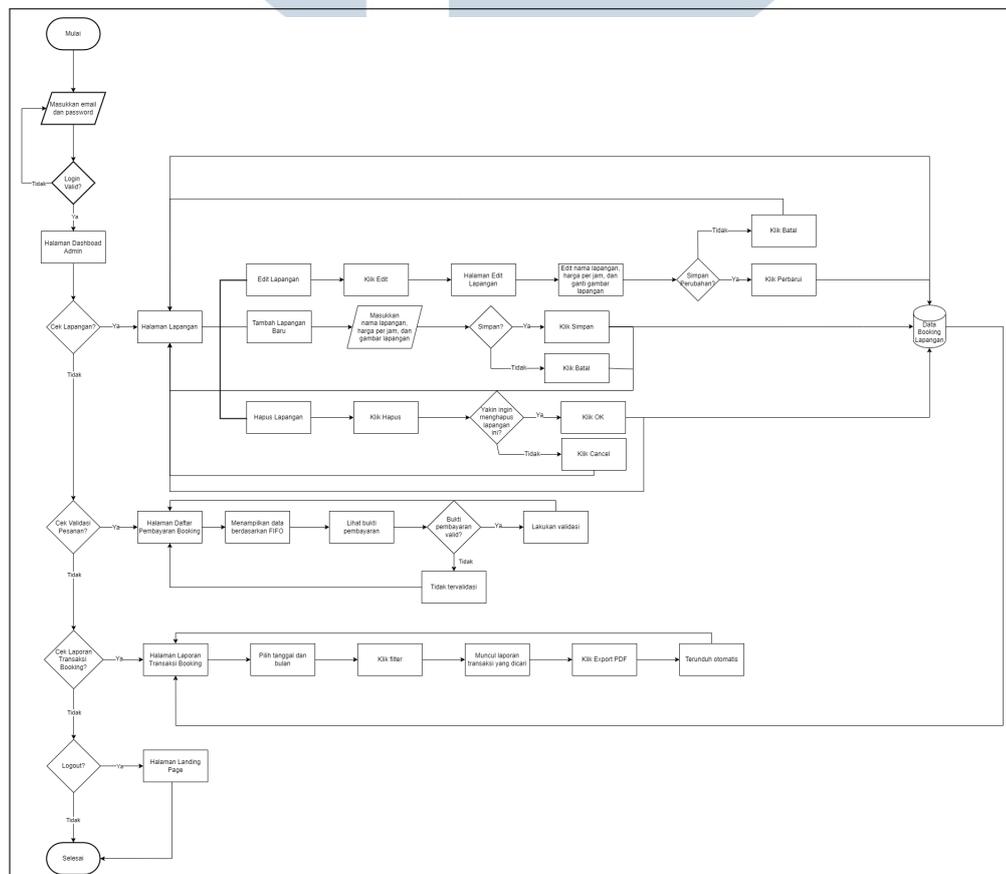
Jika pengguna memilih keluar, sistem mengarahkan ke halaman awal (landing page) dan mengakhiri sesi.

12. Selesai

Menandai berakhirnya proses interaksi pengguna dalam satu sesi pemesanan.

3.5.2 FLOWchart Admin

Flowchart ini menjelaskan alur kerja admin dalam sistem pemesanan lapangan berbasis web. Admin bertugas untuk mengelola data lapangan, memverifikasi pembayaran, serta menghasilkan laporan transaksi.



Gambar 3.4. Flowchart Admin

Gambar 3.4 Flowchart sistem admin berfungsi untuk menggambarkan seluruh alur proses kerja admin dalam mengelola sistem pemesanan lapangan badminton, mulai dari login, pengelolaan data lapangan, validasi pembayaran, hingga pelaporan transaksi.

1. Mulai
Menandakan sistem mulai dijalankan oleh admin.
2. Masukkan Email dan Password
Admin memasukkan kredensial login (email dan password) untuk masuk ke dalam sistem.
3. Login Valid?
Sistem memverifikasi kredensial login:
 - (a) Jika valid, admin diarahkan ke *Halaman Dashboard Admin*.
 - (b) Jika tidak valid, sistem meminta input ulang.
4. Halaman Dashboard Admin
Setelah login, admin dapat memilih menu:
 - (a) Kelola data lapangan
 - (b) Validasi pembayaran
 - (c) Laporan transaksi
 - (d) Logout
5. Cek Lapangan?
Jika memilih menu data lapangan:
 - (a) Sistem menampilkan daftar lapangan
 - (b) Admin dapat:
 - i. Tambah lapangan baru:
 - A. Isi nama, harga per jam, dan upload gambar
 - B. Simpan ke database *Master Booking Lapangan*
 - ii. Edit lapangan:
 - A. Ubah nama, harga, dan gambar
 - B. Simpan atau batalkan perubahan

iii. Hapus lapangan:

A. Konfirmasi penghapusan

B. Jika ya, lapangan dihapus dari database

6. Cek Validasi Pesanan?

Jika admin memilih validasi pembayaran:

(a) Sistem menampilkan daftar bukti pembayaran

(b) Admin melakukan validasi:

i. Jika valid, status berubah menjadi *Tervalidasi*

ii. Jika tidak, status tetap *Tidak Tervalidasi*

7. Cek Laporan Transaksi Booking?

Jika memilih laporan transaksi:

(a) Masuk ke halaman laporan

(b) Pilih tanggal dan bulan

(c) Klik *Filter* untuk menampilkan data

(d) Klik *Export PDF* untuk mengunduh laporan

8. Logout?

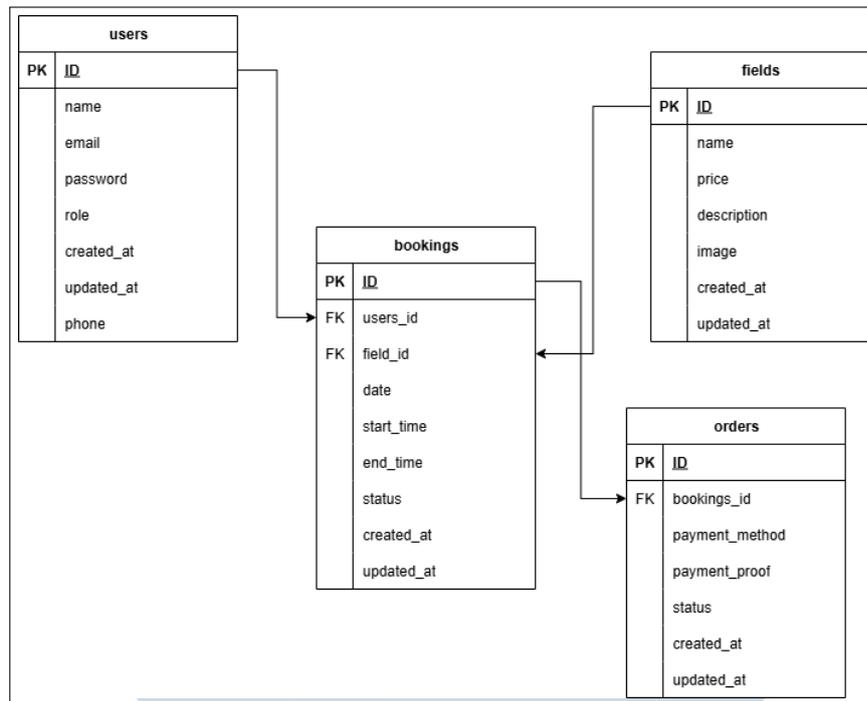
Admin mengakhiri sesi dengan klik logout, dan diarahkan ke *Halaman Landing Page*.

9. Selesai

Proses administrasi selesai.

3.5.3 Database Schema

Database schema adalah rancangan struktur logis dari basis data yang menggambarkan bagaimana tabel-tabel saling berelasi, termasuk kolom, tipe data, dan kunci primer/asing. Schema membantu dalam merancang alur data secara efisien dan menjaga integritas data. Dalam sistem pemesanan lapangan, schema melibatkan tabel seperti *users*, *bookings*, *fields*, dan *orders* yang saling terhubung. Misalnya, satu pengguna dapat memiliki banyak pemesanan, dan setiap pemesanan dapat terkait dengan satu pembayaran. Perancangan schema yang baik sangat penting untuk mempermudah pengelolaan data dan memastikan sistem berjalan optimal.



Gambar 3.5. Database Schema

Gambar 3.5 menunjukkan rancangan basis data sistem pemesanan lapangan GOR Umbu Dapasapu yang terdiri dari empat entitas utama, yaitu *Users*, *Fields*, *Bookings*, dan *Orders*. Berikut adalah penjelasan masing-masing relasi antartabel:

1. Relasi antara *Users* dan *Bookings*

Relasi antara tabel *Users* dan *Bookings* adalah *one-to-many*, yang berarti satu pengguna dapat melakukan banyak pemesanan. Hubungan ini diwakili oleh *foreign key* `user_id` pada tabel *Bookings* yang merujuk ke `id` pada tabel *Users*.

2. Relasi antara *Fields* dan *Bookings*

Tabel *Fields* juga memiliki relasi *one-to-many* dengan tabel *Bookings*, di mana satu lapangan dapat dipesan oleh banyak pengguna di waktu berbeda. Hal ini direpresentasikan oleh *foreign key* `fields_id` pada tabel *Bookings* yang merujuk ke `id` pada tabel *Fields*.

3. Relasi antara *Bookings* dan *Orders*

Relasi ini bersifat *one-to-one*, artinya satu pemesanan hanya memiliki satu pembayaran. Tabel *Orders* memiliki *foreign key* `booking_id` yang mengarah ke `id` pada tabel *Bookings*, memastikan bahwa setiap transaksi pembayaran terkait dengan satu pemesanan yang valid.

Struktur ERD ini mendukung proses pemesanan yang efisien, validasi pembayaran yang akurat, serta pengelolaan data yang terstruktur dengan baik. Setiap entitas memiliki atribut penting seperti nama pengguna, informasi lapangan, waktu booking, dan bukti pembayaran yang saling terhubung melalui relasi yang konsisten dan logis.

3.5.4 Struktur Database

Skema Database adalah struktur yang mendefinisikan bagaimana data disimpan dalam basis data seperti misalnya nama tabel, kolom, tipe data, relasi antar tabel, dan lain-lain.

Tabel 3.1. Skema Tabel Users

Kolom	Tipe Data	Keterangan
id	BIGINT	Primary Key, Auto Increment
name	VARCHAR	Nama pengguna
email	VARCHAR	Alamat email pengguna
password	VARCHAR	Password terenkripsi
role	ENUM	admin atau user
phone	VARCHAR	Nomor telepon pengguna
created_at	TIMESTAMP	Tanggal dibuat otomatis
updated_at	TIMESTAMP	Tanggal terakhir diperbarui

Pada Tabel 3.1 menyimpan informasi penting tentang setiap pengguna yang terdaftar dalam sistem. Setiap entri dalam tabel ini diidentifikasi secara unik melalui kolom *id* yang berfungsi sebagai *primary key* dan bersifat auto-increment. Informasi pengguna mencakup nama lengkap (*name*), alamat email (*email*), serta *password* yang telah dienkripsi untuk menjamin keamanan autentikasi. Kolom *role* digunakan untuk membedakan jenis pengguna, yaitu antara *admin* dan *user*. Selain itu, terdapat juga kolom *phone* yang menyimpan nomor telepon pengguna, serta kolom *created_at* dan *updated_at* yang mencatat waktu pembuatan dan pembaruan data secara otomatis oleh sistem. Tabel ini berperan penting dalam manajemen hak akses serta pelacakan aktivitas pengguna dalam aplikasi pemesanan lapangan.

Tabel 3.2. Skema Tabel Fields

Kolom	Tipe Data	Keterangan
id	int	Primary Key, AUTO_INCREMENT
name	varchar(100)	Nama lapangan
price	integer	Harga sewa per jam
image	varchar(255)	Nama file gambar lapangan
created_at	timestamp	Tanggal dibuat (otomatis)
updated_at	timestamp	Tanggal diperbarui (otomatis)

Tabel 3.2 digunakan untuk menyimpan data mengenai lapangan bulu tangkis yang tersedia di sistem. Kolom *id* merupakan *primary key* yang bersifat auto-increment. Kolom *name* menyimpan nama lapangan, sedangkan *price* menunjukkan biaya sewa per jam. Kolom *image* digunakan untuk menyimpan nama file gambar lapangan guna ditampilkan di antarmuka pengguna. Sistem juga secara otomatis mencatat waktu pembuatan dan pembaruan data melalui kolom *created_at* dan *updated_at*.

Tabel 3.3. Skema Tabel Bookings

Kolom	Tipe Data	Keterangan
id	int	Primary Key, AUTO_INCREMENT
user_id	foreignId	Relasi ke tabel users
field_id	foreignId	Relasi ke tabel fields
date	date	Tanggal booking dilakukan
start_time	time	Jam mulai sewa lapangan
end_time	time	Jam selesai sewa lapangan
status	enum	Status booking: pending, approved, canceled
created_at	timestamp	Tanggal dibuat (otomatis)
updated_at	timestamp	Tanggal diperbarui (otomatis)

Tabel 3.3 menyimpan informasi mengenai transaksi pemesanan lapangan oleh pengguna. Setiap entri memiliki *id* unik sebagai *primary key*. Kolom *user_id* dan *field_id* masing-masing menjadi *foreign key* yang menghubungkan pemesanan dengan data pengguna dan lapangan. Kolom *date*, *start_time*, dan *end_time* digunakan untuk mencatat jadwal sewa lapangan. Status dari pemesanan disimpan dalam kolom *status* dengan nilai seperti *pending*, *approved*, atau *canceled*, dan

sistem juga mencatat waktu pembuatan serta pembaruan data melalui *created_at* dan *updated_at*.

Tabel 3.4. Skema Tabel Orders

Kolom	Tipe Data	Keterangan
id	int	Primary Key, AUTO_INCREMENT
booking_id	foreignId	Relasi ke tabel bookings
status	enum	pending, paid, confirmed, canceled
payment_method	varchar(255)	Metode pembayaran: QRIS, Transfer, atau Tunai
payment_proof	varchar(255)	Nama file bukti pembayaran (jika ada)
note	text	Catatan tambahan dari user (opsional)
created_at	timestamp	Waktu data dibuat
updated_at	timestamp	Waktu data terakhir diperbarui

Tabel 3.4 digunakan untuk mencatat informasi pembayaran yang dilakukan oleh pengguna atas pemesanan lapangan yang tercatat pada tabel bookings. Kolom *booking_id* menjadi *foreign key* yang merujuk ke pemesanan terkait. Kolom *status* menunjukkan status pembayaran, seperti *pending*, *paid*, *confirmed*, atau *canceled*. Informasi terkait metode pembayaran dicatat melalui kolom *payment_method*, sedangkan file bukti pembayaran tersimpan dalam *payment_proof*. Kolom *note* menyimpan catatan tambahan dari pengguna, jika ada. Atribut *created_at* dan *updated_at* digunakan untuk melacak waktu pembuatan dan perubahan data.

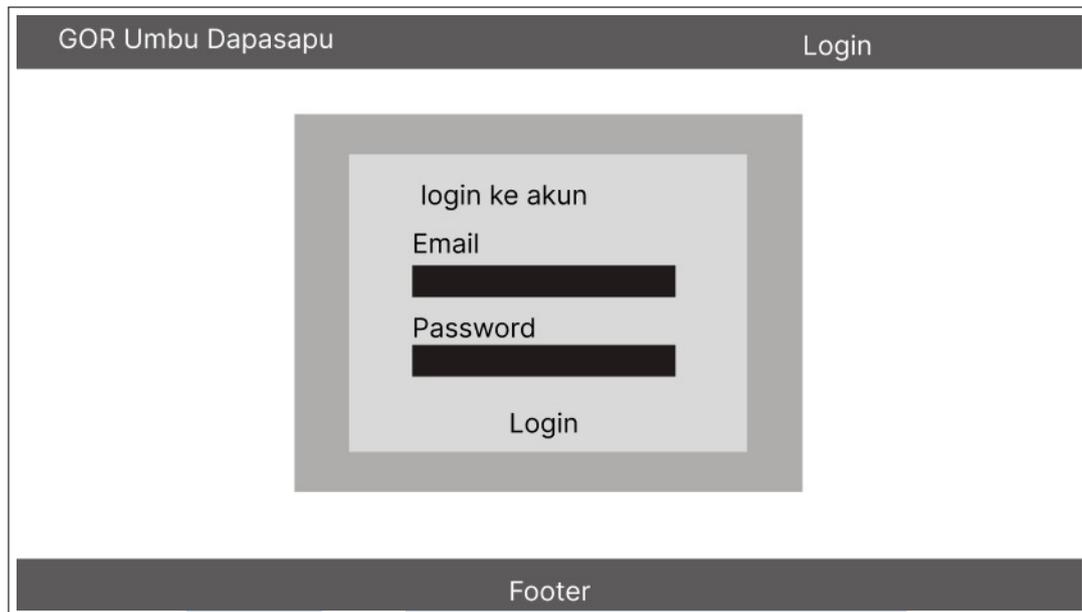
3.6 Perancangan Antarmuka Pengguna (User Interface)

Tahap ini menjelaskan perancangan tampilan antarmuka *website* yang nanti akan digunakan untuk dasar tampilan pada implementasi. proses perancangan menggunakan software Figma.

Beberapa tampilan antarmuka utama antara lain:

A Halaman awal dan Informasi GOR dan Login/register

Halaman ini adalah perancangan tampilan dimana pengguna jika ingin melakukan pemesanan harus melakukan *login* terlebih dahulu, jika belum memiliki akun pengguna bisa melakukan pendaftaran akun terlebih dahulu.



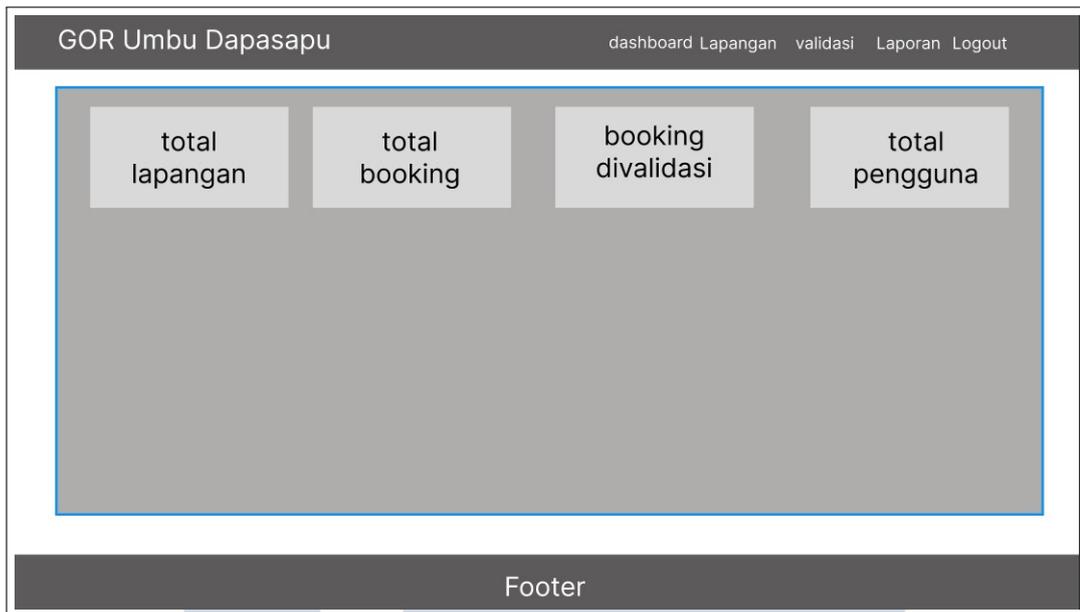
Gambar 3.6. Tampilan Home

Pada Gambar 3.6 menggambarkan rancangan antar muka pada halaman awal dari aplikasi booking lapangan pada Gor GBU.

B Halaman Dashboard

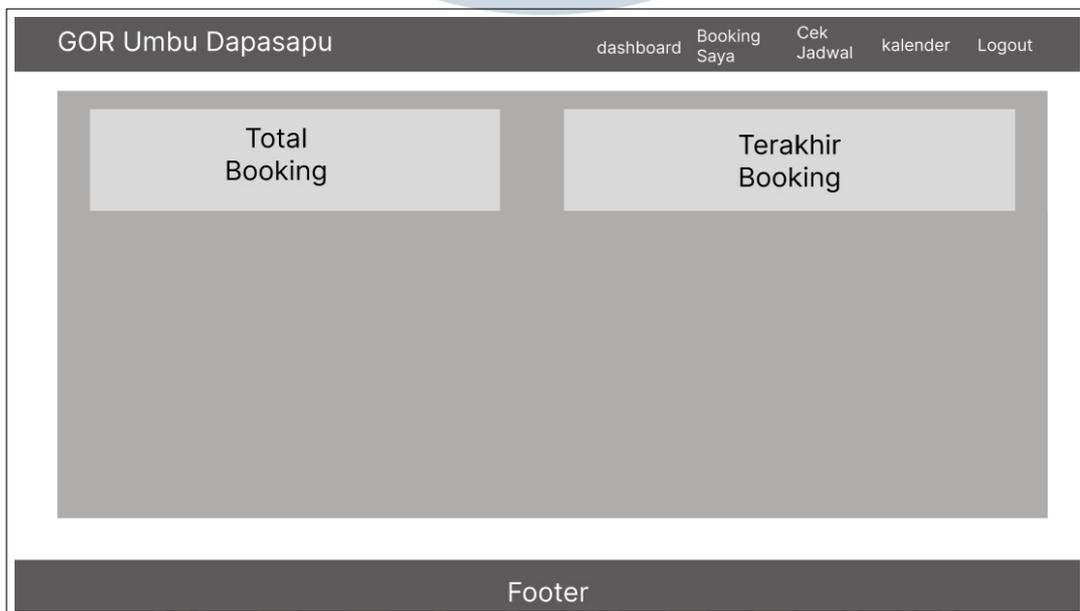
Halaman ini adalah perancangan tampilan awal Dashboard untuk pengguna dan Admin.

UMMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.7. Dashboard Admin

Gambar 3.7 diatas adalah gambar rancangan dari tampilan dashboard dari admin.



Gambar 3.8. Dashboard Pengguna

Gambar 3.8 diatas adalah gambar rancangan dari tampilan dashbord dari pengguna.

C Halaman Cek jadwal Dan Pemesanan Lapangan (Pengguna)

Halaman ini menunjukkan rancangan pengguna untuk melakukan pemesanan lapangan.



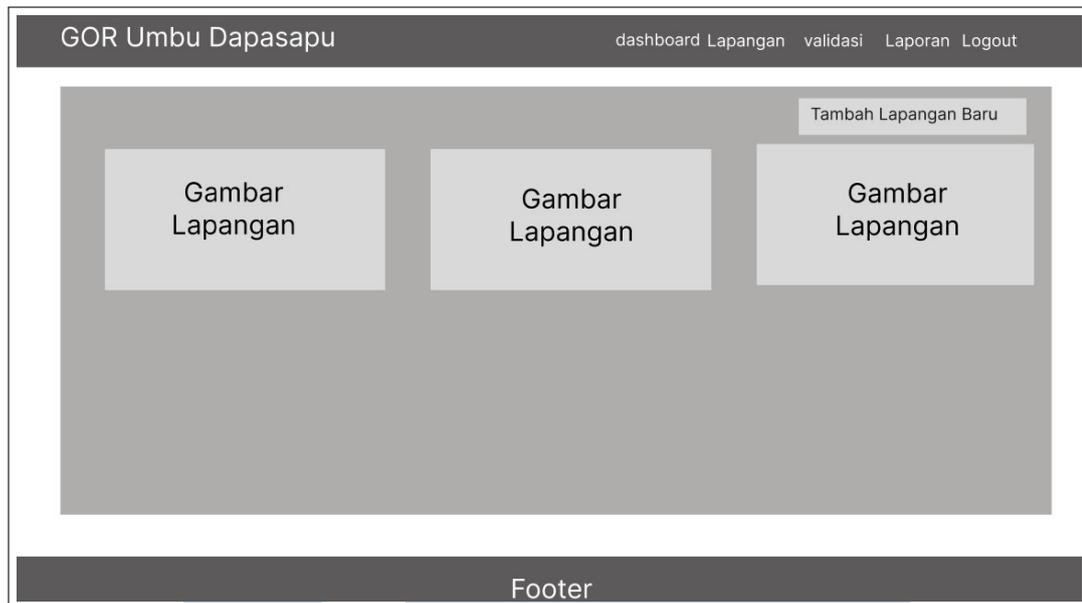
Gambar 3.9. Halaman Pemesanan Lapangan

Gambar 3.9 diatas adalah gambar rancangan dari tampilan untuk pengguna melakukan pemesanan Lapangan.

D Halaman Kelola Lapangan (Admin)

Menampilkan daftar lapangan yang dapat dikelola (tambah, edit, hapus). Informasi meliputi nama, harga, deskripsi, dan gambar lapangan. Tersedia tombol aksi CRUD pada setiap data.

UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.10. Halaman Kelola Lapangan Admin

Gambar 3.10 diatas menunjukkan halaman dimana Admin melakukan kelola lapangannya.

E Halaman Riwayat Booking (Pengguna)

Halaman ini menampilkan seluruh riwayat booking milik pengguna dalam bentuk tabel. Informasi meliputi nama lapangan, tanggal, jam, status pembayaran, dan progress bar status.

UMMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.11. Halaman Riwayat Booking Lapangan

Gambar 3.11 Diatas menggambarkan Halaman dimana pengguna bisa melihat riwayat pemesanan lapangannya dan aksi - aksi yang dia bisa lakukan.

F Halaman Form pengguna melakukan pembayaran

Pengguna dapat mengunggah bukti transfer (gambar) sebagai konfirmasi pembayaran. Setelah itu, status order akan menjadi “menunggu validasi” hingga dikonfirmasi oleh admin.

UMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA

GOR Umbu Dapasapu

Booking Saya Cek Jadwal kalender Logout

Lapangan yang dipilih

Tanggal yang dipilih

Waktu yang dipilih

Metode Pembayaran

Uploud Bukti Pembayaran

Kirim Pembayaran

Footer

Gambar 3.12. Halaman Form Pembayaran

Gambar 3.12 diatas menunjukkan halaman dimana pengguna bisa mengisi form pembayaran seperti mengunggah bukti transfer.

G Halaman Validasi Pembayaran (Admin)

Admin dapat melihat daftar semua order (transaksi), lengkap dengan bukti pembayaran, metode pembayaran, dan tombol aksi untuk melakukan validasi atau tanda centang jika sudah tervalidasi.

UIN
UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.13. Halaman Validasi Pembayaran Admin

Gambar 3.13 diatas menggambarkan halaman dimana admin bisa melihat bukti pembayaran, melakukan validasi pembayaran.

H Halaman Laporan Keuangan (Admin)

Admin dapat memfilter laporan berdasarkan tanggal atau bulan, melihat total pemasukan dari booking yang tervalidasi, dan mengekspor laporan dalam bentuk PDF.

UMMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.14. Halaman Laporan Keuangan Admin

Gambar 3.14 diatas menunjukkan halaman dimana admin bisa melihat laporan keuangan berdasarkan tanggal bulan yang admin inginkan.

3.7 Penerapan Algoritma FIFO dalam Sistem

Dalam sistem pemesanan lapangan badminton ini, algoritma *First In First Out* (FIFO) diterapkan untuk mengatur antrian pemesanan yang masuk berdasarkan waktu pengajuan booking. FIFO merupakan salah satu bentuk algoritma antrian di mana data yang pertama kali masuk akan diproses terlebih dahulu. Pendekatan ini memastikan bahwa pengguna yang lebih dulu melakukan pemesanan memiliki prioritas lebih tinggi dalam mendapatkan slot waktu lapangan yang tersedia.

Secara implementatif, setiap permintaan booking yang masuk akan disimpan ke dalam basis data dengan mencatat waktu pengajuan secara otomatis menggunakan timestamp. Ketika sistem memverifikasi ketersediaan jadwal, sistem akan menyeleksi data booking berdasarkan urutan waktu masuk secara menaik (ascending). Dengan demikian, jika terdapat lebih dari satu permintaan untuk jadwal dan lapangan yang sama, maka sistem akan memberikan hak booking kepada pengguna yang lebih dahulu mengajukan permintaan. Pendekatan ini membantu mencegah terjadinya konflik jadwal dan menjaga keadilan dalam sistem antrian pemesanan.