

SISTEM INFORMASI BOOKING LAPANGAN - BookMyField

Mata Kuliah: Pemrograman Web Lanjut



Anggota Kelompok:

Qhaulan Syaqhila (F1D022152)

Izzat Nazhiefa (F1D02310114)

M. Wahyu Hilal Abroor (F1D02310123)

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS TEKNIK

UNIVERSITAS MATARAM

2025

BAB I – DESKRIPSI PROJECT

1. Latar Belakang

Olahraga adalah kegiatan yang dilakukan dengan tujuan tertentu, seperti melatih tubuh agar tetap sehat secara jasmani maupun rohani. Dengan rutin berolahraga, seseorang dapat memperoleh manfaat untuk menjaga kesehatan tubuh. Masyarakat bisa melakukan olahraga di mana saja, tetapi beberapa jenis olahraga memerlukan tempat atau fasilitas khusus yang sesuai dengan kebutuhannya. Contoh olahraga yang membutuhkan sarana khusus antara lain basket, voli, futsal, dan badminton. Beberapa jenis olahraga membutuhkan tempat khusus karena memerlukan fasilitas pendukung, seperti ring, net, gawang, dan lainnya. Untuk memenuhi kebutuhan tersebut, banyak orang memilih menyewa lapangan. Berdasarkan survei yang dilakukan melalui kuesioner terhadap 94 responden, terdiri dari 53,8% mahasiswa, 20,4% siswa SMA, 17,2% karyawan, dan 1,1% lainnya, diketahui bahwa responden pernah menyewa lapangan olahraga. Lapangan yang disewa antara lain futsal, voli, basket, dan badminton. Para responden juga menginginkan adanya inovasi berupa sistem penyewaan lapangan olahraga secara online agar lebih mudah mendapatkan informasi mengenai lapangan yang tersedia dan dapat mempermudah serta mengefisienkan proses penyewaan[1].

Oleh karena itu, responden perlu menyewa lapangan untuk melakukan kegiatan olahraga. Saat ini, sistem pemesanan lapangan sebagian besar masih dilakukan secara manual, yaitu pengguna harus datang langsung ke tempat penyewaan atau menghubungi lewat telepon untuk mengecek ketersediaan lapangan. Namun, pemesanan melalui telepon sering menghadapi kendala terkait validasi yang akurat, sehingga pihak penyedia lapangan tidak selalu menerima pemesanan dengan cara ini. Sistem manual ini cukup merepotkan bagi pengguna karena kurang efisien dari segi waktu, tenaga, dan biaya, karena mereka harus mengecek jadwal dan memesan lapangan satu per satu.

Berdasarkan kondisi tersebut, diperlukan automatisasi sistem dengan merancang dan membuat aplikasi pemesanan lapangan olahraga berbasis web. Sistem ini diharapkan dapat memberikan validasi yang akurat terkait penjadwalan dan pemesanan lapangan, mengefisienkan waktu, serta mengurangi biaya transportasi. Selain itu, pemesanan berbasis web akan memudahkan pengguna karena dapat diakses melalui berbagai perangkat.

2. Tujuan *Project*

Adapun tujuan dari proyek ini adalah sebagai berikut:

- a. Proyek ini bertujuan untuk menggantikan metode pemesanan lapangan yang masih dilakukan secara manual dengan sistem berbasis web, sehingga seluruh proses menjadi lebih terstruktur, dan mudah dikelola.
 - b. Sistem ini diharapkan dapat memberikan kemudahan bagi pengguna untuk memperoleh informasi terkait ketersediaan lapangan secara cepat dan *real-time*, sehingga pengguna tidak perlu lagi melakukan pengecekan secara langsung atau melalui telepon.
 - c. Dengan adanya sistem berbasis web, proses pemesanan lapangan dapat dilakukan dengan lebih efisien, baik dari segi waktu, tenaga, maupun biaya, sehingga pengguna dan pengelola lapangan dapat memanfaatkan sumber daya secara optimal.
 - d. Sistem ini dirancang untuk memberikan validasi yang tepat terhadap penjadwalan dan pemesanan lapangan, sehingga meminimalkan risiko kesalahan atau bentrok jadwal, serta meningkatkan kepercayaan pengguna terhadap layanan.
 - e. Pengguna dapat melakukan pemesanan kapan saja dan dari mana saja melalui berbagai perangkat, seperti komputer, tablet, atau smartphone, sehingga sistem ini mampu mendukung kebutuhan fleksibilitas dan kenyamanan bagi seluruh pengguna.
3. Teknologi yang Digunakan (*Backend, Frontend, Database*)

a. *Backend*:

1) Node.js dan Express.js

Node.js digunakan untuk *runtime* JavaScript di sisi server dan Express.js digunakan sebagai *framework* untuk membangun REST API dengan arsitektur MVC.

2) JWT (JSON Web Token)

JWT digunakan untuk autentikasi dan otorisasi pengguna agar akses API lebih aman.

3) bcrypt

bcrypt digunakan untuk mengenkripsi (hash) kata sandi pengguna sebelum disimpan ke dalam database, sehingga meningkatkan keamanan data pengguna.

4) dotenv

dotenv digunakan untuk mengatur variabel lingkungan, seperti konfigurasi database dan secret key JWT, sehingga tidak perlu dituliskan secara langsung di dalam kode program.

b. *Frontend*:

1) Vue.js 3

Vue 3 digunakan untuk membuat aplikasi Single Page Application (SPA) yang interaktif.

2) Vue Router

Vue Router digunakan untuk mengatur navigasi antar halaman, seperti halaman login, dashboard pengguna, dan halaman *booking* lapangan.

3) Axios

Axios digunakan sebagai library HTTP *client* yang berfungsi menghubungkan *frontend* dengan *backend* melalui API, sehingga proses pengiriman dan penerimaan data dapat dilakukan secara *real-time*.

4) Vite

Vite digunakan sebagai alat *build* sekaligus *development* server yang mendukung proses pengembangan yang cepat dan efisien, sehingga mempermudah kegiatan pengujian serta pengembangan antarmuka pengguna.

c. *Database*:

1) MySQL

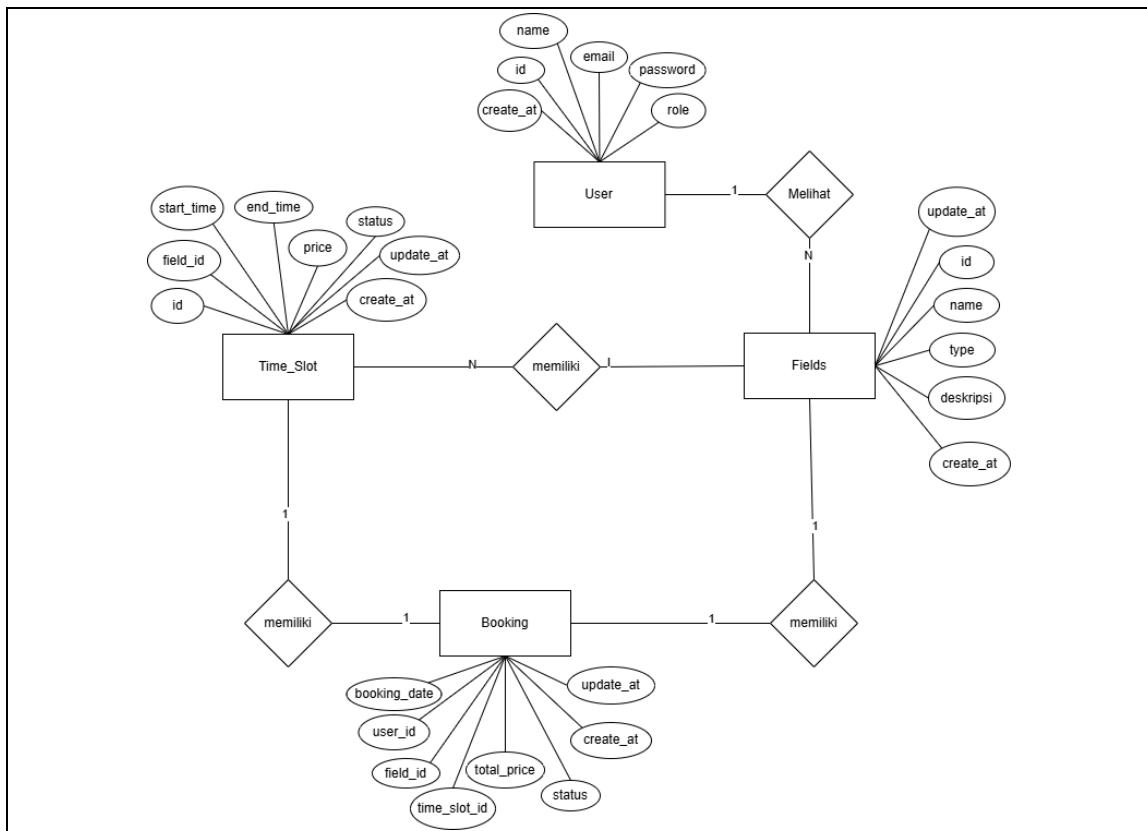
MySQL digunakan sebagai database relasional yang berfungsi menampung dan mengelola berbagai data penting dalam sistem, seperti data pengguna, informasi lapangan, dan data booking. Dengan penerapan struktur tabel yang sistematis serta dukungan perintah SQL, MySQL membantu menjaga keakuratan dan konsistensi data serta meningkatkan efisiensi dalam proses pengolahan dan penyimpanan data aplikasi.

BAB II – PERANCANGAN SISTEM

1. Deskripsi Singkat Sistem

Sistem Manajemen Booking Lapangan berbasis web ini merupakan aplikasi yang memungkinkan pengguna untuk melakukan pemesanan lapangan olahraga secara online dengan mudah dan cepat. Sistem ini menyediakan dua peran utama, yaitu Admin yang dapat mengelola data lapangan dan memantau seluruh booking, serta User yang dapat melakukan booking dan melihat riwayat pemesanan.

2. Desain Database (ERD dan penjelasan singkat tabel)



Penjelasan Tabel:

1. Users

Tabel ini digunakan untuk menyimpan data pengguna, baik itu user maupun admin. Tabel ini berisi kolom id, name, email, password, dan rule (admin/user). Rule disini digunakan untuk menentukan hak akses di sistem (misalnya admin bisa menambah dan menghapus lapangan).

2. Fields

Tabel ini digunakan untuk menyimpan data lapangan yang tersedia untuk booking. Tabel ini berisi kolom id, name, type, deskripsi, price, dan lokasi. Tabel ini juga digunakan untuk menampilkan informasi lapangan pada frontend.

3. Time_Slot

Tabel ini digunakan untuk menyimpan data jam atau slot waktu yang tersedia di lapangan. Tabel ini berisi kolom id, start_time, dan end_time. Tabel ini juga digunakan untuk menentukan kapan lapangan bisa di pesan.

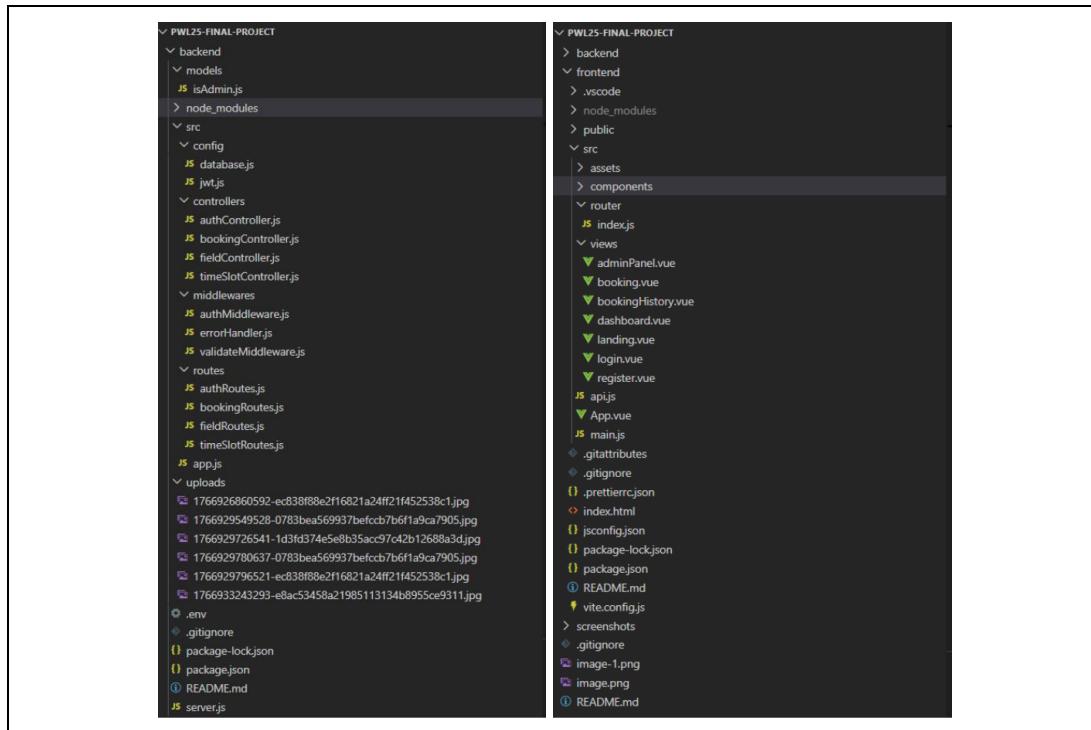
4. Bookings

Tabel ini digunakan untuk menyimpan data pemesanan yang dilakukan pengguna. Tabel ini berisi kolom id, user_id, field_id, time_slot_id, booking_date, dan status. Status bisa berupa aktif atau selesai untuk menandai proses booking.

BAB III – IMPLEMENTASI

1. Implementasi *Backend* (struktur folder, API, middleware)

a. Struktur folder



Dari gambar di atas dapat dilihat struktur folder *backend* dan *frontend* pada sistem informasi *booking* lapangan, pada struktur folder backend disesuaikan dengan struktur MVC begitupun *frontend* juga memiliki struktur folder yang mirip dengan MVC.

b. API

Adapun daftar API *endpoint* adalah sebagai berikut:

1) Auth (/api/auth)

Method	Endpoint	Keterangan
POST	/register	Register user baru
POST	/login	Login user dan dapat token

2) Fields (/api/fields)

Method	Endpoint	Keterangan
GET	/	Ambil semua lapangan
POST	/	Tambah lapangan (Admin)
PUT	/:id	Update lapangan (Admin)
DELETE	/:id	Hapus lapangan (Admin)

3) Bookings (/api/bookings)

Method	Endpoint	Keterangan
GET	/	Ambil semua booking
POST	/	Booking baru (user)
PUT	/:id/status	Update status booking (Admin)
GET	/fields	Ambil daftar lapangan untuk booking

GET	/time-slots	Ambil daftar time slot tersedia
GET	/user	Riwayat booking user
PUT	/:id/complete	Tandai booking selesai & buka slot (Admin)

4) Time Slots (/api/time-slots)

Method	Endpoint	Keterangan
GET	/	Ambil semua time slot
POST	/	Tambah time slot (Admin)
PUT	/:id	Update time slot (Admin)
DELETE	/:id	Hapus time slot (Admin)

c. Middleware

Untuk meningkatkan keamanan API, digunakan *middleware* autentikasi berbasis JSON Web Token (JWT). Middleware ini bertugas memvalidasi token yang dikirim oleh klien melalui header *Authorization* sebelum memberikan izin akses ke *endpoint* yang bersifat terbatas.

1) authMiddleware.js

- a) verifyToken digunakan untuk memeriksa apakah request membawa JWT yang valid
- b) isAdmin digunakan untuk memeriksa apakah user memiliki role admin

2) validateMiddleware.js

Validasi input body sesuai field yang wajib diisi, berikut implementasinya:

```
router.post('/', verifyToken, isAdmin, validate(['name', 'type']), addField)
```

Dari potongan kode tersebut dapat disimpulkan bahwa akses ke *route* dibatasi hanya untuk pengguna yang telah login dengan peran admin. Selain itu, request yang dikirimkan wajib menyertakan data *name* dan *type* pada body. Jika salah satu syarat tersebut tidak terpenuhi, maka akses ke *route* tidak akan diizinkan.

2. Implementasi *Frontend* (halaman, *routing*, konsumsi API)

Frontend dibangun menggunakan Vue 3 (Composition API) dan berfungsi sebagai antarmuka pengguna untuk melakukan *booking* lapangan, melihat riwayat *booking*, serta panel admin untuk pengelolaan data.

a. Halaman

- 1) Halaman dashboard publik
- 2) Halaman login dan registrasi
- 3) Halaman dashboard awal setelah login

- 4) Halaman *booking* lapangan
- 5) Halaman *view history booking*
- 6) Halaman dashboard admin yang terdiri dari CRUD Lapangan, *Time slot* dan *booking*

b. *Routing*

Routing menggunakan Vue Router untuk navigasi antar halaman. Berikut ini adalah kode penggunaan *routing*.

```
const routes = [
  { path: '/', component: Landing }, // Landing bebas akses
  { path: '/login', component: Login },
  { path: '/register', component: Register },
  { path: '/dashboard', component: Dashboard, meta: {
    requiresAuth: true } },
  { path: '/booking', component: Booking, meta: { requiresAuth:
    true } },
  { path: '/admin', component: AdminPanel, meta: { requiresAuth:
    true, requiresAdmin: true } },
  { path: '/booking-history', component: BookingHistory, meta: {
    requiresAuth: true } }
]
```

Kode routes tersebut adalah konfigurasi *routing* Vue Router yang menentukan halaman apa (*component*) yang ditampilkan berdasarkan URL (path). Halaman seperti /, /login, dan /register bisa diakses bebas tanpa login, sedangkan /dashboard, /booking, dan /booking-history hanya bisa diakses oleh *user* yang sudah login karena memiliki “meta.requiresAuth”. Khusus halaman /admin, selain harus login juga membutuhkan hak akses admin (meta.requiresAdmin), sehingga user biasa tidak dapat membukanya. Properti meta digunakan sebagai penanda tambahan yang kemudian dicek pada *navigation guard* untuk membatasi akses halaman sesuai status login dan *role* pengguna.

c. Konsumsi API

Komunikasi data antara frontend dan backend dilakukan menggunakan library Axios.

Berikut contoh implemntasi API untuk *request update* status *booking*.

```
async function completeBooking(booking) {
  if (!confirm('Tandai booking ini sebagai selesai?')) return
  const res = await
  fetch(`http://localhost:3000/api/bookings/${booking.id}/status`, {
    method: 'PUT',
    headers: {
      'Content-Type': 'application/json',
      'Authorization': `Bearer ${token}`
    },
    body: JSON.stringify({ status: 1 }) // 1 = complete
  })
  const data = await res.json()
  alert(data.message)
```

```
    loadBookings()
    loadTimeSlots()
}
```

Fungsi `completeBooking` ini digunakan untuk mengubah status *booking* melalui *backend*. Ketika admin menekan tombol, sistem terlebih dahulu menampilkan konfirmasi agar tidak terjadi perubahan tidak sengaja. Jika disetujui, *frontend* mengirim request PUT ke *endpoint backend* dengan menyertakan “`booking.id`” pada URL dan token autentikasi pada *header* untuk memastikan hanya pengguna yang berwenang yang dapat mengaksesnya. Data status: 1 dikirim dalam body sebagai penanda bahwa booking telah selesai. Setelah *backend* memproses permintaan dan mengembalikan respons, *frontend* menampilkan pesan hasil proses dan memuat ulang data booking serta time slot agar tampilan langsung mencerminkan perubahan status terbaru.

BAB IV – PENGUJIAN

1. Pengujian API (Postman/Thunder Client)

The screenshot shows the Postman application interface with two separate requests displayed.

Request 1: PUT /bookings/5/status

- Method:** PUT
- URL:** `http://localhost:3000/api/bookings/5/status`
- Headers:** (10)
- Body:** raw JSON
- ```
1 {
2 "status": 0
3 }
```
- Response:** 200 OK | 79 ms | 315 B | Save Response
- ```
1 {
2   "message": "Status booking berhasil diperbarui"
3 }
```

Request 2: PUT /fields/4

- Method:** PUT
- URL:** `http://localhost:3000/api/fields/4`
- Headers:** (10)
- Body:** raw JSON
- ```
1 {
2 "name": "Lapangan Padel A",
3 "type": "Indoor",
4 "deskripsi": "Permukaan Vinyl, Struktur panoramic"
5 }
```
- Response:** 200 OK | 36 ms | 298 B | Save Response
- ```
1 {
2   "message": "Lapangan diupdate"
3 }
```

The screenshot shows the Postman application interface with two requests displayed.

Request 1: Put_Update status booking dan status time slot-Admin

- Method: PUT
- URL: <http://localhost:3000/api/bookings/9/complete>
- Headers:
 - Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6Ikp...
- Body (JSON):

```
1 {  
2   "message": "Booking selesai & slot dibuka kembali"  
3 }
```
- Response: 200 OK | 149 ms | 318 B | Save Response

Request 2: Put_Update Slot time di field tertentu

- Method: PUT
- URL: <http://localhost:3000/api/time-slots/2>
- Headers: (10)
- Body (raw JSON):

```
1 {  
2   "start_time": "13:00:00",  
3   "end_time": "15:15:00",  
4   "price": 720000  
5 }
```
- Response: 200 OK | 140 ms | 311 B | Save Response

The screenshot shows the Postman application interface. At the top, there are navigation links: 'Search Postman' (with a magnifying glass icon), 'Ctrl + K', 'Invite' (with a person icon), 'Upgrade' (with a gear icon), and environment dropdowns ('No environment'). Below the header, a list of API endpoints is shown: 'GET Get data fields', 'POST Post data login', 'POST Post data register', 'POST Post_Add data field' (which is selected and highlighted in orange), and 'POST Post_Delete data field'. The main workspace shows a 'Test API_Tubes PWL_BookingField / Post_Add data field' test. The 'Body' tab is selected, showing the raw JSON payload sent to the endpoint:

```
1 {
2     "name": "Lapangan Padel A",
3     "type": "Indoor",
4     "deskripsi": "Permukaan Vinyl, dinding akrilik"
5 }
```

The response status is '200 OK' with a duration of '57 ms' and a size of '301 B'. Below the body, there's a preview of the response JSON:

```
1 {
2     "message": "Lapangan ditambahkan"
3 }
```

The screenshot shows the Postman application interface. At the top, there's a search bar with 'Search Postman' and a 'Ctrl + K' hotkey. To the right are 'Invite', 'Settings', 'Bell', and 'Upgrade' buttons. The main navigation bar includes 'POST Pos', 'GET Get_V', 'GET Get_V', 'GET Get_U', 'PUT Put_U', 'GET Get_L', 'POST Pos', and a '+' button. A dropdown menu says 'No environment'. Below the bar, the URL is 'Test API_Tubes PWL_BookingField / Post_Add time slot untuk field'.

The request details are as follows:

- Method:** POST
- URL:** <http://localhost:3000/api/time-slots>
- Body:** (raw) JSON
- Body Content:**

```
1 {  
2     "field_id": 1,  
3     "start_time": "19:00",  
4     "end_time": "21:00",  
5     "price": 650000  
6 }
```

The response details are:

- Status:** 200 OK
- Time:** 307 ms
- Size:** 312 B
- Save Response** button

The response body is displayed below:

```
1 {  
2     "message": "Slot waktu berhasil ditambahkan"  
3 }
```

The image shows two separate instances of the Postman application interface, each displaying an API request and its response.

Top Instance (Post data register):

- Request:** POST <http://localhost:3000/api/auth/register>
- Body (raw JSON):**

```
1 {
2   "name": "jihe",
3   "email": "jihe@gmail.com",
4   "password": "jihe123"
5 }
```

- Response:** 200 OK (238 ms, 298 B)
{"message": "Register berhasil"}

Bottom Instance (Post data login Pengguna):

- Request:** POST <http://localhost:3000/api/auth/login>
- Body (raw JSON):**

```
1 {
2   "email": "jihe@gmail.com",
3   "password": "jihe123"
4 }
```

- Response:** 200 OK (342 ms, 436 B)
{"token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.\n eyJpZC10Mywicm9sZSI6InVzZXIxIiLCJpXXQiOjE3NjY4MTg0MTYsImV4cCI6MTc2Njk0NDgxNn0.\n _Ju0GfCLuHup1Ajk1Z4xsh2mJAhFpCTKf01_WBTvzD0"}

The image shows two separate instances of the Postman application interface.

Top Instance (Post data login):

- Request:** POST <http://localhost:3000/api/auth/login>
- Body:** raw JSON
- Body Content:**

```
1 {
2   "email": "admin@gmail.com",
3   "password": "admin123"
4 }
```

- Response:** 200 OK
{"token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZC16MSwicm9sZSI6ImFkbWluIiwiaWF0IjoxNzY2ODE1NTk4LCJleHAiOjE3NjY5MDE5OTh9.v9Gsz2Dr9c0fhZ-_U4LBQilttS3doktMmJdZXbxYB6M"}

Bottom Instance (Post_data booking-Pengguna):

- Request:** POST <http://localhost:3000/api/bookings>
- Body:** raw JSON
- Body Content:**

```
1 {
2   "field_id": 2,
3   "date": "2025-12-28",
4   "time_slot_id": 3
5 }
6
```

- Response:** 200 OK
{"message": "Booking berhasil", "total_price": 450000}

The image displays two separate instances of the Postman application interface, both showing a GET request to `http://localhost:3000/api/time-slots`.

Request 1 (Top):

- Headers:** Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6Ikp...
- Body:** JSON (Previews two objects)
- Response:** 200 OK, 13 ms, 1.29 KB (JSON response shown below)

```
[{"id": 1, "field_id": 1, "start_time": "06:00:00", "end_time": "08:15:00", "price": 650000, "status": 0, "created_at": "2025-12-26T02:34:42.000Z", "updated_at": "2025-12-27T07:39:22.000Z", "field_name": "Lapangan Futsal A1"}, {"id": 2, "field_id": 2, ...}]
```

Request 2 (Bottom):

- Headers:** None selected
- Body:** none (radio button selected)
- Response:** 200 OK, 61 ms, 1.49 KB (JSON response shown below)

```
[{"id": 1, "field_id": 1, "start_time": "06:00:00", "end_time": "08:15:00", "price": 650000, "status": 0, "created_at": "2025-12-26T02:34:42.000Z", "updated_at": "2025-12-27T10:57:42.000Z", "field_name": "Lapangan Futsal A1"}, {"id": 2, "field_id": 2, ...}]
```

Test API_Tubes PWL_BookingField / Get_View Data field-Admin

GET http://localhost:3000/api/bookings/fields

Headers (10)

Key	Value	Description
Authorization	Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6Ikp...	

Body

```
[{"id": 1, "name": "Lapangan Futsal A1", "type": "Indoor"}, {"id": 2, "name": "Lapangan Mini Soccer B", "type": "outdoor"}, {"id": 4, ...}]
```

200 OK 41 ms 433 B Save Response

Test API_Tubes PWL_BookingField / Get data fields

GET http://localhost:3000/api/fields

Headers (7)

Key	Value	Description
Authorization	Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6Ikp...	

Body

```
[{"id": 1, "name": "Lapangan Futsal A1", "type": "Indoor", "deskripsi": "permukaan vinyl", "created_at": "2025-12-26T02:33:43.000Z", "updated_at": "2025-12-26T16:02:41.000Z"}, {"id": 2, "name": "Lapangan Mini Soccer B", "type": "outdoor", "deskripsi": "permukaan rumput sintetis", "created_at": "2025-12-26T07:25:11.000Z", "updated_at": "2025-12-26T16:03:00.000Z"}]
```

200 OK 52 ms 611 B Save Response

Collection deleted

Test API_Tubes PWL_BookingField / Get_data booking-Admin

GET http://localhost:3000/api/bookings

Headers (8)

Key	Value
Authorization	Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6Ikp...

Body

```
[{"id": 5, "user_name": "jj", "field_name": "Lapangan Futsal A1", "booking_date": "2025-12-27T16:00:00.000Z", "start_time": "06:00:00", "end_time": "08:15:00", "status": 1}]
```

200 OK

Test API_Tubes PWL_BookingField / DEL_data time slot

DELETE http://localhost:3000/api/time-slots/9

Headers (8)

Key	Value
Authorization	Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6Ikp...

Body

```
{"message": "Slot waktu berhasil dihapus"}
```

200 OK

Test API_Tubes PWL_BookingField / Delete_Hapus data field-admin

DELETE http://localhost:3000/api/fields/5

Headers (8)

none	form-data	x-www-form-urlencoded	raw	binary	GraphQL	JSON
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

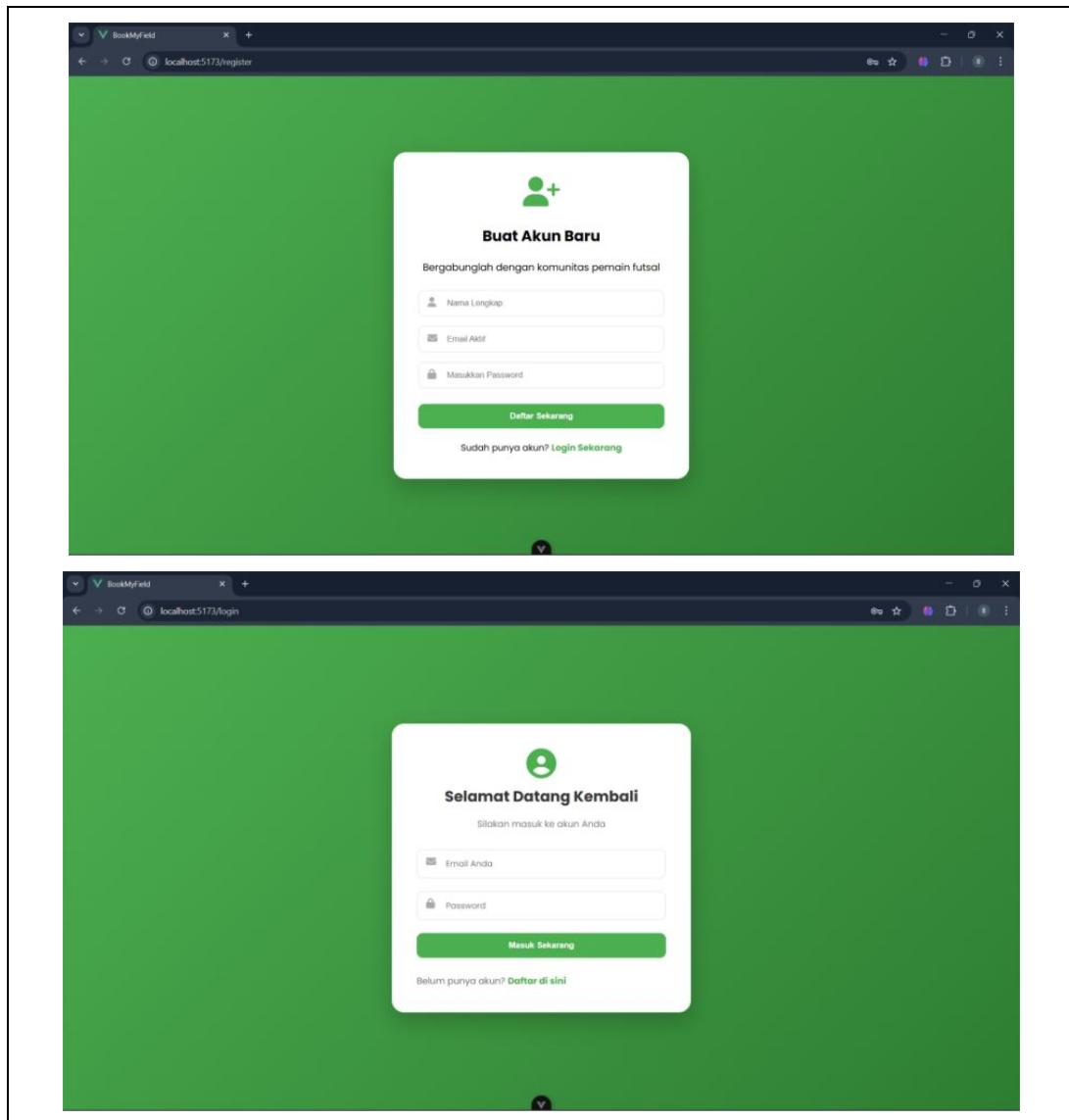
Body

```
1 Ctrl+Alt+P to Ask AI
```

200 OK

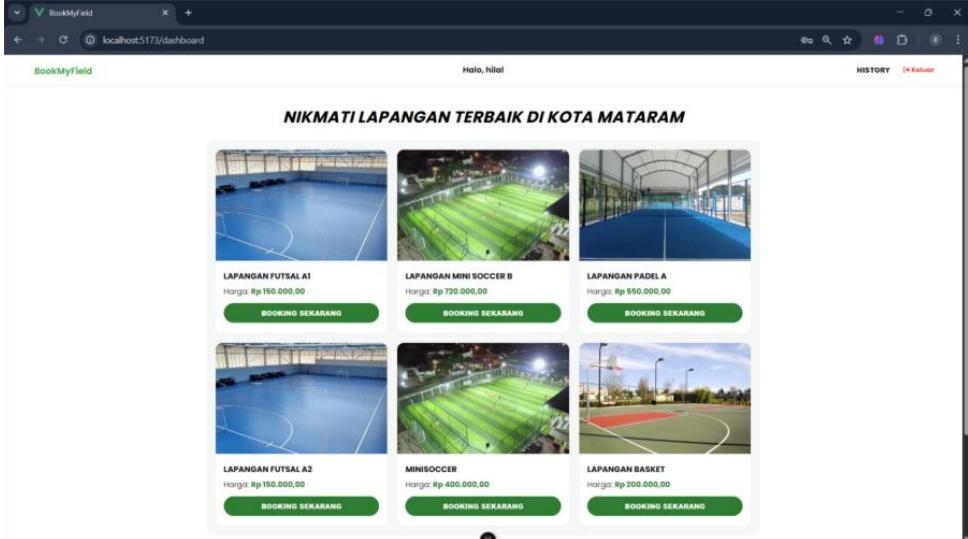
2. Pengujian Aplikasi (Screenshot fitur utama)

a. Halaman Login dan Registrasi



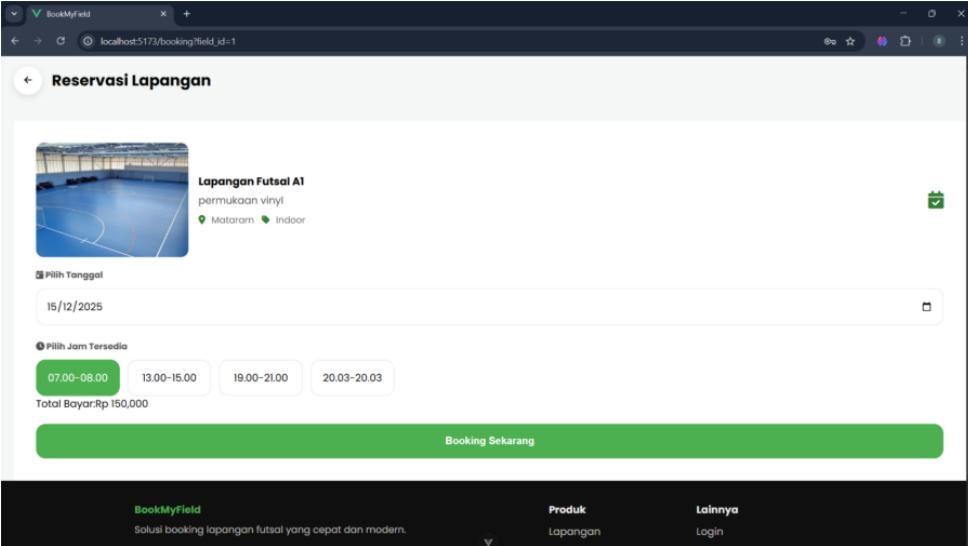
Dari gambar di atas dapat diketahui bahwa sebelum masuk ke dalam website, pengguna perlu melakukan registrasi jika belum memiliki akun, setelah melakukan registrasi pengguna perlu melakukan login untuk masuk kedalam website

b. Booking Lapangan



The screenshot shows the BookMyField dashboard with the heading "NIKMATI LAPANGAN TERBAIK DI KOTA MATARAM". It displays six sports fields with their names and prices:

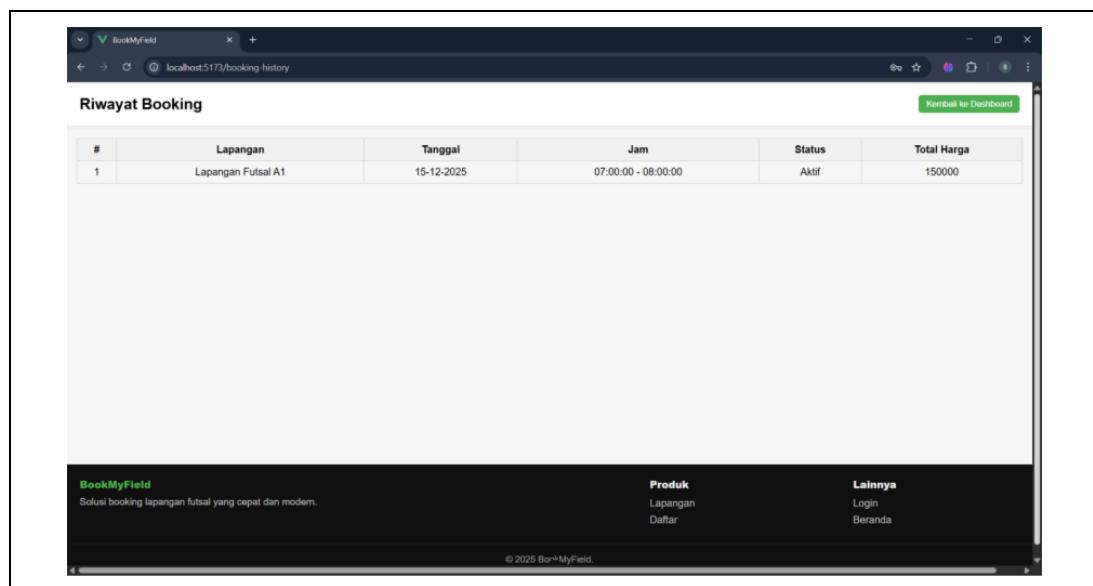
- LAPANGAN FUTSAL A1: Harga: Rp 150.000,00. Button: BOOKING SEKARANG.
- LAPANGAN MINI SOCCER B: Harga: Rp 720.000,00. Button: BOOKING SEKARANG.
- LAPANGAN PADEL A: Harga: Rp 550.000,00. Button: BOOKING SEKARANG.
- LAPANGAN FUTSAL A2: Harga: Rp 150.000,00. Button: BOOKING SEKARANG.
- MINISOCER: Harga: Rp 400.000,00. Button: BOOKING SEKARANG.
- LAPANGAN BASKET: Harga: Rp 200.000,00. Button: BOOKING SEKARANG.



The screenshot shows the "Reservasi Lapangan" (Booking) page for "Lapangan Futsal A1". It includes the field image, name, surface type (permukaan vinyl), location (Mataram, Indoor), date selection (15/12/2025), time selection (07.00-08.00, 13.00-15.00, 19.00-21.00, 20.03-20.03), total price (Total Bayar: Rp 150,000), and a large green "Booking Sekarang" button.

Dari gambar di atas dapat diketahui bahwa setelah berhasil login, pengguna masuk ke dalam halaman awal, dihalaman tersebut pengguna dapat melihat lapangan yang tersedia, pengguna dapat melakukan booking dengan klik button booking, setelah klik button tersebut pengguna akan masuk ke halaman booking yang berisi detail booking yang dapat di isi oleh pengguna.

c. History Booking



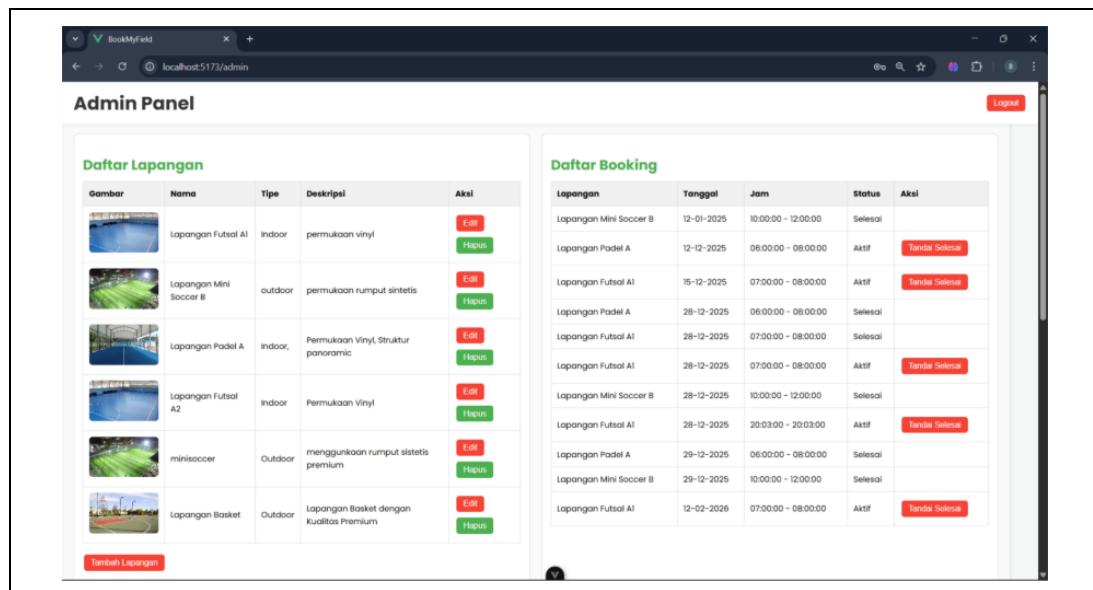
The screenshot shows a web browser window titled "Riwayat Booking" (Booking History) from the "BookMyField" website. The URL in the address bar is "localhost:5173/booking history". The main content displays a table with one row of data:

#	Lapangan	Tanggal	Jam	Status	Total Harga
1	Lapangan Futsal A1	15-12-2025	07:00:00 - 08:00:00	Aktif	150000

At the bottom of the page, there is a footer with the BookMyField logo, a slogan "Solusi booking lapangan futsal yang cepat dan modern.", navigation links for "Produk" (Lapangan, Daftar), "Lainnya" (Login, Beranda), and copyright information "© 2025 BookMyField".

Berdasarkan gambar, sistem menyediakan fitur history booking yang memungkinkan pengguna melihat riwayat pemesanan sebelumnya, termasuk detail lapangan, tanggal, waktu, dan status booking. Fitur ini memudahkan pemantauan aktivitas dan pengecekan ulang pemesanan.

d. Halaman Admin Panel



The screenshot shows the "Admin Panel" of the BookMyField website. The URL in the address bar is "localhost:5173/admin". The page has two main sections: "Daftar Lapangan" (List of Fields) on the left and "Daftar Booking" (List of Bookings) on the right.

Daftar Lapangan (List of Fields):

Gambar	Nama	Tipe	Deskripsi	Aksi
	Lapangan Futsal A1	Indoor	permukaan vinyl	Edit Hapus
	Lapangan Mini Soccer B	outdoor	permukaan rumput sintetis	Edit Hapus
	Lapangan Padel A	indoor	Permukaan Vinyl, Struktur panoramic	Edit Hapus
	Lapangan Futsal A2	Indoor	Permukaan Vinyl	Edit Hapus
	minisoccer	Outdoor	menggunakan rumput sintetis premium	Edit Hapus
	Lapangan Basket	Outdoor	Lapangan Basket dengan Kualitas Premium	Edit Hapus

Daftar Booking (List of Bookings):

Lapangan	Tanggal	Jam	Status	Aksi
Lapangan Mini Soccer B	12-01-2025	10:00:00 - 12:00:00	Selesai	
Lapangan Padel A	12-12-2025	06:00:00 - 08:00:00	Aktif	Tanda Selesai
Lapangan Futsal A1	15-12-2025	07:00:00 - 08:00:00	Aktif	Tanda Selesai
Lapangan Padel A	28-12-2025	06:00:00 - 08:00:00	Selesai	
Lapangan Futsal A1	28-12-2025	07:00:00 - 08:00:00	Aktif	Tanda Selesai
Lapangan Mini Soccer B	28-12-2025	10:00:00 - 12:00:00	Selesai	
Lapangan Futsal A1	28-12-2025	20:00:00 - 20:00:00	Aktif	Tanda Selesai
Lapangan Padel A	29-12-2025	06:00:00 - 08:00:00	Selesai	
Lapangan Mini Soccer B	29-12-2025	10:00:00 - 12:00:00	Selesai	
Lapangan Futsal A1	12-02-2026	07:00:00 - 08:00:00	Aktif	Tanda Selesai

Dari gambar di atas dapat diketahui bahwa setelah admin login, terlihat admin panel seperti pada gambar tersebut yang didalamnya berisi detail informasi dari website.

e. Admin CRUD Lapangan

The screenshot shows the BookMyField Admin Panel interface. On the left, there is a sidebar with navigation links: Home, Daftar Lapangan, Daftar Booking, Pengguna, Log Out. The main content area has two sections: 'Admin Panel' and 'Daftar Lapangan'.

Admin Panel: This section contains a table of fields and a modal for adding new fields.

Gambar	Nama	Tipe	Deskripsi	Aksi
Lapangan Futsal A1	Lapangan Futsal A1	Indoor	permukaan vinyl	<button>Edit</button> <button>Hapus</button>
Lapangan Mini Soccer B	Lapangan Mini Soccer B	Indoor	permukaan vinyl	<button>Edit</button> <button>Hapus</button>
Lapangan Padel A	Lapangan Padel A	Indoor	permukaan vinyl	<button>Edit</button> <button>Hapus</button>
Lapangan Futsal A2	Lapangan Futsal A2	Indoor	permukaan vinyl	<button>Edit</button> <button>Hapus</button>
minisoccer	minisoccer	Outdoor	menggunakan rumput sintetis premium	<button>Edit</button> <button>Hapus</button>
Lapangan Basket	Lapangan Basket	Outdoor	Lapangan Basket dengan Kualitas Premium	<button>Edit</button> <button>Hapus</button>

Tambah Lapangan (Add Field) Modal:

- Nama lapangan: Lapangan Futsal A1
- Tipe (Indoor/Outdoor): Indoor
- Deskripsi: permukaan vinyl
- File: Choose File (No file chosen)
- Action Buttons: Tambah (Add) and Batalkan (Cancel)

Daftar Lapangan (List Fields) Table:

Gambar	Nama	Tipe	Deskripsi	Aksi
Lapangan Futsal A1	Lapangan Futsal A1	Indoor	permukaan vinyl	<button>Edit</button> <button>Hapus</button>
Lapangan Mini Soccer B	Lapangan Mini Soccer B	Indoor	permukaan vinyl	<button>Edit</button> <button>Hapus</button>
Lapangan Padel A	Lapangan Padel A	Indoor	permukaan vinyl	<button>Edit</button> <button>Hapus</button>
Lapangan Futsal A2	Lapangan Futsal A2	Indoor	permukaan vinyl	<button>Edit</button> <button>Hapus</button>
minisoccer	minisoccer	Outdoor	menggunakan rumput sintetis premium	<button>Edit</button> <button>Hapus</button>
Lapangan Basket	Lapangan Basket	Outdoor	Lapangan Basket dengan Kualitas Premium	<button>Edit</button> <button>Hapus</button>

Daftar Booking (List Bookings) Table:

Lapangan	Tanggal	Jam	Status	Aksi
Lapangan Mini Soccer B	29-12-2025	10:00:00 - 12:00:00	Selesai	<button>Tanda Selesai</button>
Lapangan Futsal A1	12-02-2026	07:00:00 - 08:00:00	Aktif	<button>Tanda Selesai</button>

Berdasarkan gambar tersebut, dapat disimpulkan bahwa pengguna dengan peran admin memiliki hak akses penuh untuk mengelola data lapangan. Admin dapat melakukan seluruh operasi CRUD, yaitu membuat data lapangan baru, melihat dan menampilkan data yang ada, memperbarui informasi lapangan, serta menghapus data lapangan sesuai kebutuhan sistem. Admin CRUD Time Slot

The screenshot shows the Admin Panel interface for BookMyField. A modal window titled "Tambah Time Slot" (Add Time Slot) is open, prompting the user to input details for a new slot. The form fields include "Lapangan" (Field), "Jam Mulai" (Start Time), "Jam Selesai" (End Time), and "Harga" (Price). The background table lists various fields and their current status (e.g., Tersedia, Blocked).

Lapangan	Jam Mulai	Jam Selesai	Harga	Status
Lapangan Mini Soccer B	13:00:00	15:00:00	Rp 465000	Tersedia
Lapangan Padel A	06:00:00			
Lapangan Futsal A2	07:00:00			
Lapangan Futsal A2	08:00:00			
minisoccer	09:00:00			
Lapangan Basket	08:00:00	09:00:00	Rp 200000	Blocked

The screenshot shows the Admin Panel interface for BookMyField. A modal window titled "Edit Time Slot" (Edit Time Slot) is open, allowing the admin to update an existing slot. The form fields include "Lapangan" (Field), "Jam Mulai" (Start Time), "Jam Selesai" (End Time), "Harga" (Price), and "Status" (Status). The background table lists various fields and their current status (e.g., Tersedia, Blocked).

Lapangan	Jam Mulai	Jam Selesai	Harga	Status
Lapangan Futsal A1	07:00:00			
Lapangan Futsal A1	13:00:00			
Lapangan Futsal A1	19:00:00			
Lapangan Futsal A1	20:00:00			
Lapangan Mini Soccer B	13:00:00	15:00:00	Rp 720000	Tersedia
Lapangan Mini Soccer B	10:00:00	12:00:00	Rp 460000	Tersedia

Berdasarkan sistem yang ditunjukkan, admin memiliki kemampuan untuk mengelola data timeslot secara penuh melalui operasi CRUD. Admin dapat membuat timeslot baru sesuai jadwal yang tersedia, membaca atau menampilkan daftar timeslot yang ada, memperbarui informasi timeslot jika terjadi perubahan waktu atau status, serta menghapus timeslot yang tidak lagi digunakan. Dengan demikian, admin dapat memastikan pengelolaan timeslot berjalan tertib dan sesuai kebutuhan operasional.

f. Admin Edit Status Booking

The screenshot shows the BookMyField Admin Panel interface. On the left, there's a table titled "Daftar Lapangan" (List of Fields) with columns: Gambar (Image), Nama (Name), Tipe (Type), Deskripsi (Description), and Aksi (Actions). It lists several fields: Lapangan Futsal A1 (Indoor, permukaan vinyl), Lapangan Mini Soccer B (outdoor, permukaan rumput sintetis), Lapangan Padel A (Indoor, Permukaan Vinyl, Struktur panoramic), Lapangan Futsal A2 (Indoor, Permukaan Vinyl), minisoccer (Outdoor, menggunakan rumput sintetis premium), and Lapangan Basket (Outdoor, Lapangan basket dengan kualitas Premium). On the right, there's a table titled "Bukti Booking" (Booking Proof) with columns: Lapangan, Tanggal, Jam, Status, and Aksi. It shows bookings for various fields at different times, with some entries having a "Tanda Selesai" (Completed) button in the actions column. A modal dialog box is centered over the tables, displaying the message "localhost:5173 says: tanda booking ini sebagai selesai?" with "OK" and "Cancel" buttons.

Berdasarkan sistem yang ditunjukkan, admin memiliki kemampuan untuk mengubah **status booking** yang telah dibuat oleh pengguna. Hal ini memungkinkan admin untuk memperbarui status pemesanan dari “aktif” menjadi “selesai”, sehingga memudahkan pengelolaan dan pemantauan transaksi. Fitur ini berhubungan dengan lapangan. Ketika booking selesai time slot pada lapangan tersebut akan berubah menjadi tersedia yang mana sebelumnya masih berstatus booked.

LAMPIRAN

Link GitHub Repository: <https://github.com/hilalabroor/pwl25-final-project.git>

DAFTAR PUSTAKA

- [1] H. Rasikhah and A. R. Adriansyah, “Jurnal Informatika Terpadu PERANCANGAN DAN IMPLEMENTASI BOOKING SYSTEM LAPANGAN MENGGUNAKAN FRAMEWORK MVC BERBASIS WEB,” *Jurnal Informatika Terpadu*, vol. 8, no. 1, pp. 8–12, [Online]. Available: <https://journal.nurulfikri.ac.id/index.php/JIT>