

# RANCANG BANGUN SISTEM INFORMASI MANAJEMEN RUANG RAPAT PADA PT XYZ MENGGUNAKAN JAVASCRIPT

Muhammad Izzuddin<sup>1\*</sup>, Nur Cahyo Wibowo<sup>2</sup>, Eka Dyar Wahyuni<sup>3</sup>

<sup>1,2,3</sup>Universitas Pembangunan Nasional Veteran Jawa Timur; Jl. Rungkut Madya, Gn. Anyar, Kec. Gn. Anyar, Surabaya; 60294

Received: 2 Maret 2025  
Accepted: 29 Maret 2025  
Published: 14 April 2025

**Keywords:**

Information System;  
Meeting Room Management;  
Waterfall;  
Black Box Testing.

**Correspondent Email:**

20082010060@student.upnja  
tim.ac.id

**Abstrak.** Perkembangan teknologi digital mendorong kebutuhan akan sistem informasi yang terorganisir, termasuk dalam pengelolaan ruang rapat di perusahaan. Penelitian ini bertujuan untuk merancang dan membangun sistem informasi manajemen ruang rapat berbasis web pada PT XYZ menggunakan teknologi JavaScript. Pengembangan sistem ini mengadopsi metode Waterfall yang mencakup tahapan analisis kebutuhan, desain sistem, implementasi, dan pengujian. Sistem dikembangkan menggunakan React.js untuk front-end, Express.js untuk back-end, dan Prisma sebagai Object-Relational Mapping (ORM) dengan MySQL sebagai basis data. Pengujian dilakukan menggunakan metode Black Box Testing guna memastikan sistem berfungsi sesuai dengan spesifikasi yang telah ditentukan. Hasil implementasi menunjukkan bahwa sistem ini meningkatkan efisiensi dalam pengelolaan ruang rapat, memungkinkan pengguna untuk melakukan reservasi secara real-time, serta mengurangi konflik jadwal akibat pencatatan manual. Dengan adanya sistem ini, PT XYZ dapat mengoptimalkan pemanfaatan ruang rapat dan meningkatkan produktivitas karyawan.

**Abstract.** The advancement of digital technology drives the need for well-organized information systems, including meeting room management in companies. This study aims to design and develop a web-based meeting room management information system at PT XYZ using JavaScript technology. The system development follows the Waterfall method, encompassing requirements analysis, system design, implementation, and testing. The system is built using React.js for the front-end, Express.js for the back-end, and Prisma as the Object-Relational Mapping (ORM) with MySQL as the database. Testing is conducted using the Black Box Testing method to ensure the system functions according to the specified requirements. The implementation results show that this system enhances efficiency in meeting room management, allowing users to make real-time reservations and reducing scheduling conflicts caused by manual record-keeping. With this system, PT XYZ can optimize meeting room utilization and improve employee productivity.

## 1. PENDAHULUAN

Perkembangan teknologi seperti Artificial Intelligence (AI), Internet of Things (IoT), dan Big Data tidak hanya memberikan kemudahan dalam kehidupan sehari-hari, tetapi juga mendorong efisiensi operasional di sektor bisnis [1]. Dalam penelitian yang dilakukan oleh Adrian dan Paeno, yang menunjukkan bahwa adopsi teknologi mengurangi biaya operasional sebesar 30% dan meningkatkan produktivitas dengan mengurangi waktu kerja manual hingga 25% dalam kurun waktu lima tahun terakhir [1]. Teknologi berfungsi sebagai strategi kunci untuk menjaga keberlanjutan dan daya saing perusahaan di era digital.

Ruang rapat memiliki peran penting dalam mendukung kegiatan perusahaan, terutama sebagai ruang kreatif untuk bertukar ide dan menciptakan inovasi. Desain ruang rapat yang nyaman dapat meningkatkan produktivitas dalam bekerja [2]. Faktor-faktor seperti pencahayaan, tata letak furnitur, dan sistem akustik yang baik berperan besar dalam menciptakan suasana kerja yang nyaman [2]. Selain itu, integrasi teknologi seperti sistem konferensi video, dan alat presentasi modern dapat mempercepat proses pengambilan keputusan dan mendukung kolaborasi lintas divisi.

PT XYZ menyediakan fasilitas ruang rapat yang tersebar di lantai 18, 37, dan 38 gedung kantornya, dengan setiap lantai memiliki 4 hingga 6 ruang rapat. Setiap ruang dirancang untuk berbagai kebutuhan pertemuan, mulai dari rapat internal hingga presentasi kepada klien. Teknologi canggih seperti alat presentasi modern dan perangkat konferensi video tersedia untuk mendukung kelancaran komunikasi selama rapat. Meskipun fasilitas ini dapat mendukung operasional perusahaan, dalam pengelolaannya memerlukan sistem yang terorganisasi untuk memastikan penggunaan ruang rapat optimal tanpa adanya konflik jadwal.

Solusi sistem berbasis web memungkinkan proses manajemen ruang rapat menjadi lebih terorganisasi dan transparan. Dalam penelitian yang dipublikasikan oleh Afdal dan Amnur, disebutkan bahwa implementasi aplikasi berbasis web dapat meningkatkan pemanfaatan ruang rapat berkat fitur seperti pencatatan notula, pemantauan real-time, dan pengingat jadwal [3]. Sementara itu, penggunaan sistem

berbasis web secara signifikan mengurangi konflik jadwal yang sering terjadi akibat metode pencatatan manual [4]. Aplikasi berbasis web menjadi solusi untuk manajemen ruang rapat dalam lingkungan perusahaan modern.

## 2. TINJAUAN PUSTAKA

### 2.1. Sistem Informasi

Sistem informasi (SI) merupakan suatu kombinasi dari berbagai komponen yang saling terkait dan dirancang untuk mengumpulkan, memproses, menyimpan, serta menyebarkan informasi guna mendukung pengambilan keputusan dan kontrol dalam organisasi [5]. Leavitt Diamond merupakan sebuah model yang menggambarkan interaksi antara empat komponen utama dalam sistem informasi, yaitu teknologi, tugas, orang, dan struktur.

Fungsi utama SI meliputi input, proses, output, dan mekanisme umpan balik [6]. Input mencakup pengumpulan data, proses melibatkan transformasi data menjadi informasi yang berguna, output menyediakan informasi dalam bentuk laporan atau dokumen, dan umpan balik membantu menyesuaikan input atau proses sesuai kebutuhan. Sistem ini juga memiliki tujuan untuk meningkatkan profitabilitas, kualitas pelayanan, dan efektivitas manajemen dalam organisasi [5].

### 2.2. JavaScript

JavaScript adalah salah satu bahasa pemrograman yang paling populer dan digunakan secara luas dalam pengembangan aplikasi web. Pertama kali dikembangkan pada tahun 1995, JavaScript awalnya dirancang sebagai bahasa pemrograman di sisi klien untuk meningkatkan interaktivitas halaman web [7]. Namun, dengan perkembangan teknologi, JavaScript kini digunakan untuk pengembangan sisi klien (front-end) dan sisi server (back-end), menjadikannya bahasa pemrograman full-stack [8].

### 2.3. React.js

React.js adalah pustaka JavaScript open-source yang dirancang khusus untuk membangun antarmuka pengguna (User Interface) yang dinamis dan interaktif. React.js memiliki karakteristik Virtual DOM, yaitu salinan dari DOM yang disimpan di memori [9]. Dengan pendekatan ini, hanya elemen yang berubah yang disinkronkan ke DOM asli, sehingga meningkatkan performa aplikasi.

Selain itu, React.js juga mendukung komponen Reusable, yang memudahkan pemeliharaan dan meningkatkan efisiensi pengembangan dengan membagi antarmuka menjadi komponen kecil yang dapat digunakan kembali [10].

#### 2.4. Express.js

Dibangun di atas ekosistem Node.js, Express.js mendukung arsitektur berbasis JavaScript dan memanfaatkan kemampuan asinkron, sehingga cocok untuk aplikasi berskala besar dengan kebutuhan performa tinggi [13]. Express.js memberikan kebebasan kepada pengembang untuk merancang arsitektur aplikasi sesuai kebutuhan. Dengan routing yang kuat, Express.js memungkinkan pengelolaan rute HTTP dan URL secara dinamis. Selain itu, middleware digunakan untuk menangani permintaan dan respon secara modular, meningkatkan struktur dan efisiensi pengembangan aplikasi [16].

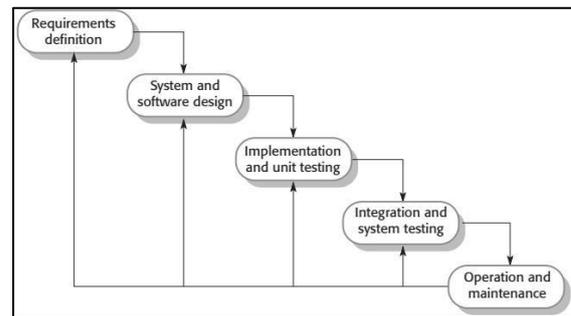
#### 2.5. ORM Prisma

ORM Prisma adalah teknologi yang sering digunakan dalam ekosistem pengembangan perangkat lunak modern untuk manajemen basis data. ORM Prisma adalah Object-Relational Mapping (ORM) generasi baru yang mendukung berbagai basis data relasional dan dirancang untuk mempermudah manipulasi data dengan pendekatan deklaratif. Prisma adalah ORM open-source yang memfasilitasi interaksi dengan basis data menggunakan pendekatan berbasis model. ORM ini memungkinkan pengembang untuk bekerja dengan basis data relasional seperti MySQL, PostgreSQL, dan SQLite [11].

#### 2.6. Metode Waterfall

Metode Waterfall adalah salah satu metodologi pengembangan perangkat lunak yang pertama kali diperkenalkan oleh Winston W. Royce pada tahun 1970 [12]. Metode ini dirancang untuk memberikan kerangka kerja yang lebih terstruktur dalam pengembangan perangkat lunak berskala besar, serta mengatasi permasalahan ketidakjelasan tahapan dalam proses pengembangan perangkat lunak [13]. Model Waterfall mengadopsi pendekatan sekuensial, di mana setiap tahapan harus diselesaikan sepenuhnya sebelum melanjutkan ke tahapan berikutnya [12].

Metode ini memperkenalkan tahapan sistematis mulai dari analisis kebutuhan, desain, implementasi, pengujian, hingga pemeliharaan sesuai pada Gambar 1.



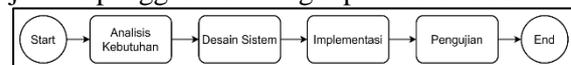
Gambar 1 Tahapan Metode Waterfall

#### 2.7. Black Box Testing

Black Box Testing adalah metode pengujian perangkat lunak yang berfokus pada pengujian fungsi sistem tanpa mempertimbangkan kode sumber atau struktur internal aplikasi [14]. Tujuan utama metode ini adalah memastikan apakah perangkat lunak berfungsi sesuai dengan spesifikasi yang telah ditentukan. Pengujian ini dilakukan dengan memeriksa input dan output dari sistem untuk memastikan hasilnya sesuai dengan yang diharapkan.

### 3. METODE PENELITIAN

Metode penelitian ini menggunakan pendekatan Waterfall, yang merupakan model pengembangan perangkat lunak dengan tahapan berurutan dan terstruktur. Setiap tahapan diselesaikan secara menyeluruh sebelum melanjutkan ke tahap berikutnya. Tahapan dalam penelitian ini terdapat pada Gambar 2 meliputi Analisis Kebutuhan, Desain Sistem, Implementasi, dan Pengujian. Metode ini dipilih untuk memberikan struktur yang sistematis sehingga dapat melakukan pengembangan sistem informasi manajemen ruang rapat yang digunakan untuk mengelola jadwal penggunaan ruang rapat di PT XYZ.



Gambar 2 Tahapan Pengembangan

#### 3.1. Analisis Kebutuhan

Tahap analisis kebutuhan merupakan langkah awal dalam penelitian ini yang bertujuan untuk mengidentifikasi dan mendefinisikan kebutuhan sistem yang akan dikembangkan. Pada tahap ini, dilakukan berbagai kegiatan untuk memahami permasalahan pengelolaan ruang rapat di PT XYZ serta menentukan solusi yang dapat diterapkan melalui sistem informasi berbasis web.

### 3.1.1. Wawancara dan Observasi

Wawancara dan observasi dilakukan untuk memperoleh pemahaman mendalam mengenai permasalahan yang dihadapi oleh PT XYZ dalam pengelolaan ruang rapat. Aktivitas ini bertujuan untuk menggali informasi langsung dari pihak terkait, seperti karyawan, manajer, dan admin yang terlibat dalam proses pemesanan dan pengelolaan ruang rapat.

### 3.1.2. Identifikasi Kebutuhan

Tahap identifikasi kebutuhan bertujuan untuk merumuskan kebutuhan fungsional yang harus dipenuhi oleh sistem informasi manajemen ruang rapat berbasis web di PT XYZ. Kebutuhan ini diidentifikasi berdasarkan hasil wawancara, observasi, dan studi literatur yang dilakukan sebelumnya. Kebutuhan ini akan menjadi pedoman utama dalam proses desain, implementasi, dan pengujian sistem.

## 3.2. Desain Sistem

Tahap desain sistem merupakan langkah penting dalam proses pengembangan sistem informasi manajemen ruang rapat di PT XYZ. Pada tahap ini, dilakukan perancangan alur kerja, antarmuka pengguna, serta basis data untuk memastikan bahwa sistem yang dibangun memenuhi kebutuhan fungsional. Desain sistem ini menjadi dasar dalam implementasi sistem berbasis web menggunakan teknologi React.js dan Express.js.

### 3.2.1. Perancangan Alur Proses

Perancangan alur proses bertujuan untuk memberikan gambaran mengenai bagaimana sistem informasi manajemen ruang rapat di PT XYZ akan beroperasi. Proses ini mencakup identifikasi alur kerja utama seperti pemesanan ruang rapat dan pengelolaan jadwal. Perancangan ini divisualisasikan menggunakan diagram untuk memahami interaksi antar komponen sistem.

### 3.2.2. Perancangan Antarmuka

Perancangan antarmuka pengguna (User Interface) bertujuan untuk menciptakan sistem yang mudah digunakan (user-friendly) dan mampu memenuhi kebutuhan dalam pengelolaan ruang rapat di PT XYZ. Desain antarmuka ini menggunakan pendekatan responsive design, sehingga dapat diakses dari berbagai perangkat, seperti komputer, tablet, dan smartphone.

### 3.2.3. Perancangan Basis Data

Perancangan basis data bertujuan untuk menyimpan dan mengelola data yang

diperlukan oleh sistem informasi manajemen ruang rapat di PT XYZ secara terstruktur, aman, dan efisien. Basis data ini akan menjadi pusat penyimpanan informasi seperti data pengguna, ruang rapat, jadwal, dan pemesanan, yang mendukung operasional sistem secara keseluruhan.

## 3.3. Implementasi

Tahap implementasi merupakan proses merealisasikan desain sistem informasi manajemen ruang rapat ke dalam bentuk aplikasi berbasis web yang berfungsi sesuai dengan kebutuhan. Implementasi dilakukan dengan menggunakan teknologi React.js untuk pengembangan front-end, Express.js untuk back-end, dan Prisma sebagai ORM. Tahap ini memastikan bahwa sistem dapat dijalankan sesuai spesifikasi yang telah dirancang.

### 3.3.1. Implementasi Basis Data

Pada tahap implementasi basis data, sistem informasi manajemen ruang rapat menggunakan MySQL sebagai RDBMS untuk menyimpan data yang berelasi dengan aman dan ORM Prisma untuk mempermudah interaksi antara aplikasi back-end dan basis data. Prisma dipilih karena kemudahan penggunaannya dalam mengelola basis data, seperti query, migrasi, dan pengelolaan skema secara deklaratif.

### 3.3.2. Pengkodean Back-end

Pada tahap ini, pengkodean back-end dilakukan menggunakan Express.js untuk menangani logika bisnis, komunikasi dengan basis data MySQL melalui Prisma ORM, dan pengelolaan RESTful API. Back-end dirancang untuk memastikan kelancaran operasional fitur utama, seperti autentikasi pengguna, pemesanan, dan pengelolaan ruang rapat.

### 3.3.3. Pengkodean Front-end

Pada tahap ini, pengembangan front-end sistem informasi manajemen ruang rapat dilakukan menggunakan React.js untuk menciptakan antarmuka yang interaktif, responsif, dan user-friendly. Front-end dirancang berdasarkan kebutuhan fungsional yang telah ditentukan sebelumnya, seperti pemesanan dan pengelolaan ruang rapat.

## 3.4. Pengujian

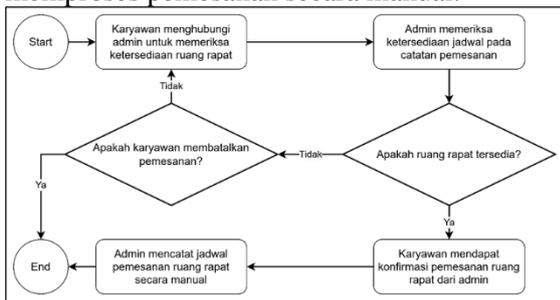
Tahap pengujian dilakukan untuk memastikan bahwa sistem informasi manajemen ruang rapat di PT XYZ berfungsi sesuai dengan kebutuhan. Pengujian ini

menggunakan metode Black Box Testing, yang berfokus pada pengujian fungsionalitas sistem tanpa memperhatikan kode sumber. Pengujian dilakukan pada antarmuka pengguna untuk memastikan sistem memberikan keluaran yang sesuai dengan masukan yang diberikan.

**4. HASIL DAN PEMBAHASAN**

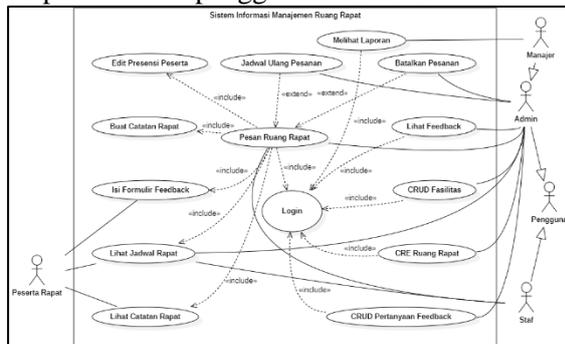
Berdasarkan hasil wawancara dengan karyawan PT XYZ, ditemukan bahwa tidak adanya sistem terintegrasi untuk memesan ruang rapat, sehingga pemesanan dilakukan secara manual melalui komunikasi verbal atau dokumen sederhana. Hal ini sering kali menyebabkan konflik jadwal antar pengguna.

Berikut adalah Gambar 3 yang menggambarkan proses pemesanan ruang rapat yang berlangsung sebelum sistem informasi manajemen ruang rapat diimplementasikan. Flowchart ini menunjukkan bagaimana karyawan harus memeriksa ketersediaan ruang rapat melalui admin, yang kemudian memproses pemesanan secara manual.



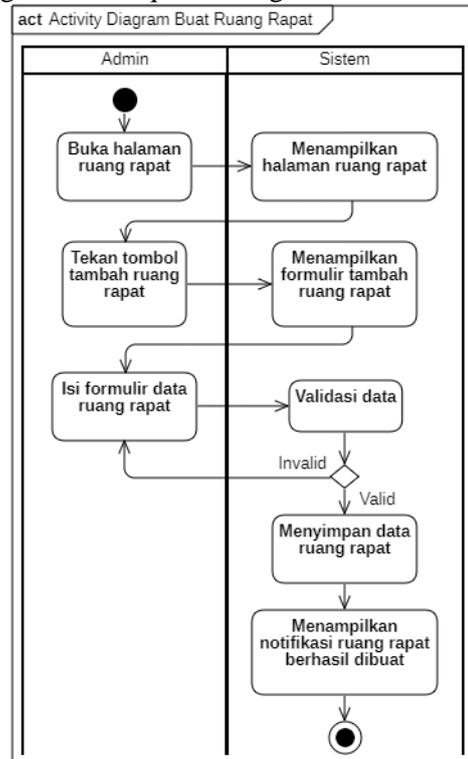
**Gambar 3 Flowchart Proses Pemesanan**

Berikut adalah Gambar 4.2 diagram untuk menggambarkan interaksi antara aktor dengan fitur utama sistem, seperti memesan ruang rapat, melihat jadwal, mengelola ruang rapat, dan mengakses catatan. Diagram use case ini menunjukkan bagaimana masing-masing aktor berperan dalam penggunaan sistem.



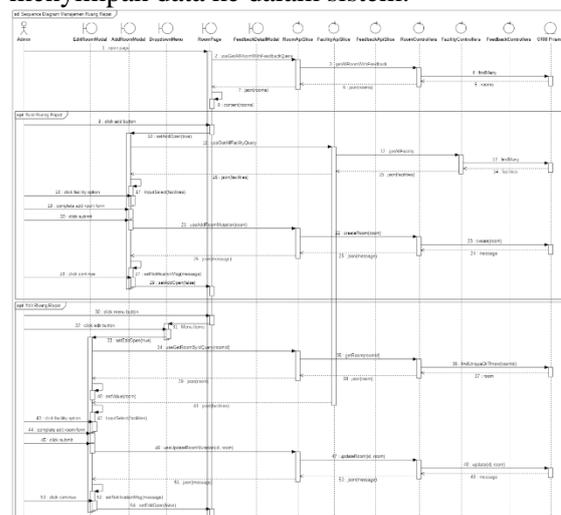
**Gambar 4 Use Case Diagram**

Bentuk visualisasi dari analisis dan desain alur kerja melibatkan interaksi admin dan pengguna terhadap sistem. Hasil perancangan alur proses ditunjukkan dalam bentuk activity diagram dan sequence diagram.



**Gambar 5 Activity Diagram Buat Ruang Rapat**

Activity Diagram pada Gambar 5 menunjukkan proses admin menambahkan ruang rapat baru. Admin mengisi detail ruang, seperti nama, kapasitas, dan fasilitas, kemudian menyimpan data ke dalam sistem.



**Gambar 6 Sequence Diagram Proses Manajemen Ruang Rapat**

Sequence Diagram pada Gambar 6 menjelaskan bagaimana admin mengelola data ruang rapat. Admin dapat memilih untuk menambah, mengedit, atau menghapus ruang rapat melalui antarmuka. Sistem front-end mengirimkan data tersebut ke back-end, yang kemudian memperbarui data.

Antarmuka dirancang dengan fokus pada pengalaman pengguna untuk memastikan semua fitur dapat diakses dengan cepat. Desain antarmuka mencakup tata letak halaman, navigasi, serta elemen-elemen visual lainnya yang dirancang untuk mempermudah interaksi pengguna dengan sistem.

Modal pada Gambar 7 digunakan untuk menambahkan ruang rapat baru. Admin dapat memasukkan nama ruang, lokasi, kapasitas, dan fasilitas melalui form yang sederhana.

Gambar 7 Modal Add Room

Basis data dirancang menggunakan MySQL sebagai RDBMS dan Prisma ORM untuk mempermudah interaksi antara aplikasi back-end dan basis data. Desain basis data disusun berdasarkan hasil analisis kebutuhan yang telah dilakukan, sehingga dapat mendukung sistem dan meminimalkan potensi redundansi data.

Skema Gambar 8 menggambarkan relasi antar tabel dalam sistem informasi manajemen ruang rapat menggunakan pendekatan ORM Prisma dalam format UML.



Gambar 8 Skema ORM Prisma (UML)

Implementasi basis data menggunakan Prisma sebagai ORM dan MySQL sebagai sistem manajemen basis data. Implementasi ini mencakup pembuatan skema basis data menggunakan Prisma Schema, migrasi skema ke MySQL, serta konfigurasi koneksi antara Prisma dan MySQL.

Model Facility pada Gambar 9 digunakan untuk mendefinisikan fasilitas yang tersedia di ruang rapat. Model ini memiliki atribut id (primary key) untuk mengidentifikasi fasilitas secara unik dan name yang menyimpan nama fasilitas, seperti proyektor, AC, atau whiteboard. Model ini juga memiliki relasi dengan model Room melalui model transaksional Room Facility, yang memungkinkan setiap fasilitas digunakan di lebih dari satu ruang rapat.

```

12 model tb_m_facility {
13   id String @id @default(uuid()) @db.VarChar(36)
14   name String @unique @db.VarChar(28)
15   rooms tb_tr_room_facility[]
16 }
17
18 model tb_tr_room_facility {
19   id String @id @default(uuid()) @db.VarChar(36)
20   room tb_m_room @relation(fields: [room_id], references: [id])
21   room_id String @db.VarChar(36)
22   facility tb_m_facility @relation(fields: [facility_id], references: [id])
23   facility_id String @db.VarChar(36)
24   quantity Int @db.Smallint
25
26   @@unique([room_id, facility_id])
27 }
    
```

Gambar 9 Skema Prisma Model Facility

Model Room pada Gambar 10 dirancang untuk mengelola informasi ruang rapat. Atribut penting dalam model ini mencakup id sebagai primary key, name untuk nama ruang, location untuk lokasi ruang, capacity untuk kapasitas maksimum, dan isAvailable untuk status ketersediaan. Relasi penting mencakup hubungan dengan model Booking untuk mencatat pemesanan ruang dan model Room Facility untuk mendata fasilitas yang ada di ruang tersebut.

```

1 model tb_m_room {
2   id      String      @id @default(uuid()) @db.VarChar(36)
3   name    String      @unique @db.VarChar(20)
4   location String      @db.VarChar(30)
5   capacity Int         @db.SmallInt
6   is_available Boolean @default(true)
7   photo   String      @db.Text()
8   facilities tb_tr_room_facility[]
9   bookings tb_m_booking[]
10 }
    
```

**Gambar 10 Skema Prisma Model Room**

Pengkodean back-end dilakukan menggunakan Express.js sebagai framework utama untuk membangun API RESTful yang berfungsi sebagai jembatan antara front-end dan basis data. Setiap fitur sistem, diimplementasikan sebagai endpoint yang dapat diakses melalui metode HTTP (GET, POST, PUT, DELETE).

Endpoint ini memungkinkan admin menambahkan ruang rapat baru. Data seperti nama, kapasitas, dan lokasi diterima pada Gambar 11 melalui body request, lalu disimpan ke tabel Room. Respon berupa data ruang yang berhasil dibuat.

```

1 const room = await PrismaClientROOME.tb_m_room.create({
2   data: {
3     name,
4     location,
5     capacity,
6     photo,
7     facilities: {
8       create: facilities?.map((e) => ({
9         facility: ...,
10        quantity: e.quantity,
11      })),
12     },
13   },
14   include,
15 })
    
```

**Gambar 11 Potongan Kode Back-end**

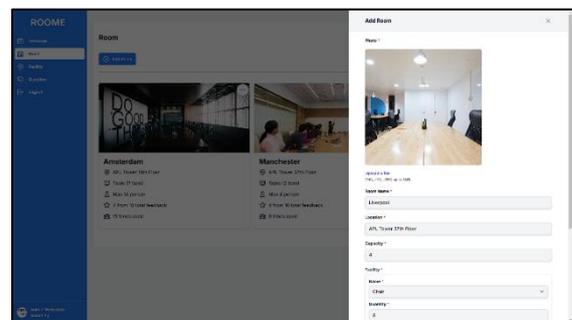
Pengkodean front-end menggunakan React.js untuk membangun antarmuka yang responsif dan dinamis. Komponen utama yang dikembangkan meliputi halaman jadwal rapat, formulir pemesanan, dan pengelolaan ruang rapat. Dirancang menggunakan framework CSS seperti Tailwind untuk memastikan tampilan responsif di berbagai perangkat.

Modal ini memungkinkan admin menambahkan ruang rapat baru. Form pada Gambar 13 berisi input seperti nama, kapasitas, dan lokasi ruang, kemudian data yang baru dikirim ke API untuk disimpan ke basis data sesuai Gambar 12.

```

1 const [addRoom, { isError, isSuccess, error, data }] = useAddRoomMutation()
2
3 const onSubmit = async (data) => {
4   try {
5     setIsSubmitting(true)
6     await addRoom(
7       createFormData({
8         name: data.name,
9         location: data.location,
10        capacity: parseInt(data.capacity, 10),
11        facilities: data.facilities.map((e) => ({
12          id: e.id,
13          quantity: parseInt(e.quantity, 10),
14        })),
15        photo: data.photoFile[0],
16      })
17    )
    }
    }
    
```

**Gambar 12 Potongan Kode Front-end**



**Gambar 13 Modal Web**

Tahap pengujian dilakukan menggunakan metode Black Box Testing, yang berfokus pada pengujian fungsionalitas. Setiap fitur diuji dengan memberikan berbagai input dan memeriksa output yang dihasilkan. Pengujian dilakukan untuk memastikan integritas data antara front-end dan back-end.

Endpoint Tabel 1 diuji untuk menambahkan ruang rapat baru. Data ruang yang valid berhasil disimpan dengan status kode 201. Jika input tidak lengkap, API mengembalikan status kode 400 dengan pesan error.

**Tabel 1 Tabel Pengujian API**

Skenario Pengujian	Hasil yang Diharapkan	Hasil Sebenarnya	Status
Membuat ruang rapat dengan data yang valid	Ruang rapat berhasil dibuat	Ruang rapat berhasil dibuat	Sesuai
Membuat ruang rapat dengan data yang tidak valid	Menampilkan respon error "Required"	Menampilkan respon error "Required"	Sesuai
Membuat ruang rapat dengan nama yang sudah ada	Menampilkan respon error "Already used"	Menampilkan respon error "Already used"	Sesuai
Membuat ruang rapat dengan role selain admin	Menampilkan respon error "Unauthorized"	Menampilkan respon error "Unauthorized"	Sesuai

Modal Tabel 4.38 diuji untuk memastikan admin dapat menambahkan ruang rapat baru. Input seperti nama ruang, kapasitas, dan lokasi

diuji untuk validasi, dan data berhasil disimpan ke database setelah tombol "Save" ditekan.

**Tabel 2 Tabel Pengujian Modal Web**

Skenario Pengujian	Hasil yang Diharapkan	Hasil Sebenarnya	Status
Menambahkan ruang rapat dengan data yang valid	Berhasil menambahkan ruang rapat baru	Berhasil menambahkan ruang rapat baru	Sesuai
Menambahkan ruang rapat dengan data yang tidak valid	Menampilkan notifikasi error validasi	Menampilkan notifikasi error validasi	Sesuai
Menambahkan ruang rapat dengan nama yang sudah ada	Menampilkan notifikasi error "Already used"	Menampilkan notifikasi error "Already used"	Sesuai
Menambahkan ruang rapat tanpa memiliki role admin	Menampilkan halaman error "Unauthorized"	Menampilkan halaman error "Unauthorized"	Sesuai

## 5. KESIMPULAN

Berdasarkan hasil implementasi dan pengujian, sistem informasi manajemen ruang rapat berbasis web JavaScript di PT XYZ memungkinkan pengguna untuk memantau ketersediaan ruang secara real-time dan memesan ruang rapat secara daring. Sistem ini dirancang menggunakan React.js sebagai framework front-end untuk membangun antarmuka yang interaktif dan responsif, serta Express.js sebagai framework back-end untuk mengelola logika server dan integrasi dengan basis data. Untuk pengelolaan basis data, digunakan Prisma sebagai ORM, dengan MySQL sebagai sistem manajemen basis data. Selain itu, sistem dilengkapi dengan fitur tambahan, seperti pengingat email otomatis, pencatatan notula rapat, dan formulir feedback yang terpusat. Mengadopsi metode pengembangan berbasis Waterfall, sistem ini dikembangkan secara terstruktur melalui tahapan analisis kebutuhan, desain, implementasi, dan pengujian. Hasil pengujian menggunakan metode Black Box Testing menunjukkan bahwa sistem berfungsi sesuai dengan berbagai skenario pengujian dan mampu memenuhi tujuan penelitian. Penelitian ini menunjukkan bahwa penerapan teknologi full-stack JavaScript dalam pengembangan sistem berbasis web yang kompleks serta pemanfaatan ORM Prisma, yang memungkinkan pengembang berinteraksi dengan basis data secara terstruktur dan konsisten.

## UCAPAN TERIMA KASIH

Penelitian ini tidak terlepas dari dukungan banyak pihak. Penulis mengucapkan terima kasih kepada orang tua yang telah memberikan dukungan baik materi maupun moral. Berkat

doa, motivasi, dan dukungan mereka yang tiada henti, penulis dapat menjalani proses penelitian ini dengan lebih tenang dan percaya diri. Penulis ingin mengungkapkan rasa terima kasih yang mendalam kepada dosen pembimbing atas kesabaran dan wawasan yang telah diberikan. Bimbingan yang diberikan sangat berperan dalam membantu penulis mengatasi berbagai tantangan selama proses penelitian. Meskipun penelitian ini masih memiliki keterbatasan, penulis berharap hasilnya tetap dapat memberikan kontribusi yang berarti bagi perkembangan ilmu pengetahuan.

## DAFTAR PUSTAKA

- [1] R. E. Adrian and Paeno, "Analisis Dampak Teknologi Terhadap Efisiensi Industri di Era Digital," *Prosiding Seminar Nasional Manajemen*, vol. 4, no. 1, pp. 656–660, Dec. 2024, [Online]. Available: <http://openjournal.unpam.ac.id/index.php/PSM/index>
- [2] Maulinda, T. Yuningsih, and Y. Yanuar, "Dampak tata ruang kantor abad-21 terhadap produktivitas karyawan di lingkungan perkantoran," *INTEGRATED Journal Of Information Technology And Vocational Education*, vol. 5, pp. 1–12, 2023, doi: 10.17509/integrated.v%vi%i.66091.
- [3] Afdal and H. Amnur, "MRAPAT Untuk Sistem Manajemen Ruang Rapat, Absensi, dan Notulen di PT PLN Unit Wilayah Sumbar," *JITSI: Jurnal Ilmiah Teknologi Sistem Informasi*, vol. 1, no. 2, pp. 43–52, Jun. 2020, doi: 10.30630/jitsi.1.2.6.
- [4] E. A. Dirfa, J. A. N. Mita, and S. P. Adithama, "Pembangunan Sistem Informasi Manajemen Rapat (Studi Kasus: Universitas Atma Jaya Yogyakarta)," *Jurnal Informatika Atma Jogja*, vol. 3, no. 2, pp. 82–89, Nov. 2022, doi: 10.24002/jiaj.v3i2.6779.
- [5] L. Zemmouchi-Ghomari, "Basic Concepts of Information Systems," in *Contemporary Issues in Information Systems - A Global Perspective*, IntechOpen, 2022. doi: 10.5772/intechopen.97644.
- [6] Y. Zheng, "Do Information Systems Today Play a Strategic Role in Business?," *Journal of Soft Computing and Decision Analytics*, vol. 2, no. 1, pp. 65–70, Jan. 2024, doi: 10.31181/jscda21202432.
- [7] K. A. Yasa, K. Q. Fredlina, and I. G. J. E. P. Putra, "Pembangunan Sistem Informasi Inventory Berbasis Web (Studi Kasus: CV Penerbit Nilacakra)," *Smart Techno (Smart Technology, Informatics and*

- Technopreneurship*), vol. 5, no. 1, pp. 13–21, Feb. 2023, doi: 10.59356/smart-techno.v5i1.74.
- [8] A. Shukla, “Modern JavaScript Frameworks and JavaScript’s Future as a FullStack Programming Language,” *Journal of Artificial Intelligence & Cloud Computing*, pp. 1–5, Dec. 2023, doi: 10.47363/JAICC/2023(2)144.
- [9] Dr. T. V. Lakshmi and L. Rakshitha, “The Power of React JS for Business Applications,” *International Research Journal on Advanced Engineering and Management (IRJAEM)*, vol. 2, no. 05, pp. 1637–1639, May 2024, doi: 10.47392/IRJAEM.2024.0229.
- [10] D. Bhatt, K. Parekh, M. Minat, and B. Patel, “ReactJS: A Comprehensive Analysis of its features, Performance, and Suitability for Modern Web Development,” *International Journal of Scientific Research in Engineering and Management (IJSREM)*, vol. 07, no. 09, pp. 1–7, Sep. 2023, doi: 10.55041/IJSREM25667.
- [11] Inc. Prisma Data, “Prisma Docs.” Accessed: Feb. 26, 2024. [Online]. Available: <https://www.prisma.io/docs>
- [12] M. F. Hidayattullah and Y. Hapsari, “Implementasi Metode Waterfall pada Rancang Bangun Sistem Informasi Kerja Praktik Industri,” *Ultima InfoSys: Jurnal Ilmu Sistem Informasi*, vol. 11, no. 2, pp. 85–89, Dec. 2020, doi: 10.31937/si.v11i2.1594.
- [13] M. Irfan, Moch. A. G. N. Rosid, and A. Lutfiyani, “Perancangan Sistem Absensi Berbasis Website dengan Metode Waterfall di BAPPEDA Kebumen,” *Jurnal Kridatama Sains dan Teknologi*, vol. 5, no. 01, pp. 75–88, Mar. 2023, doi: 10.53863/kst.v5i01.702.
- [14] A. Amalia, S. W. Putri Hamidah, and T. Kristanto, “Pengujian Black Box Menggunakan Teknik Equivalence Partitions Pada Aplikasi E-Learning Berbasis Web,” *Building of Informatics, Technology and Science (BITS)*, vol. 3, no. 3, pp. 269–274, Dec. 2021, doi: 10.47065/bits.v3i3.1062.