



**SEKOLAH TINGGI TEKNOLOGI TERPADU NURUL FIKRI**

**PERANCANGAN APLIKASI MANAJEMEN RUANGAN  
BERBASIS WEB DI PT. TIRTA ASASTA DEPOK  
MENGUNAKAN FRAMEWORK LARAVEL**

**TUGAS AKHIR**

**HAIKAL AHMAD ROBBANI  
0110120208**

**PROGRAM STUDI SISTEM INFORMASI  
DEPOK  
AGUSTUS 2024**



**STT TERPADU  
NURUL FIKRI**

**SEKOLAH TINGGI TEKNOLOGI TERPADU NURUL FIKRI**

**PERANCANGAN APLIKASI MANAJEMEN RUANGAN  
BERBASIS WEB DI PT. TIRTA ASASTA DEPOK  
MENGUNAKAN FRAMEWORK LARAVEL**

**TUGAS AKHIR**

**Diajukan sebagai salah satu syarat untuk memperoleh gelar sarjana sistem  
informasi**

**STT - NF**

**HAIKAL AHMAD ROBBANI**

**0110120208**

**PROGRAM STUDI SISTEM INFORMASI**

**DEPOK**

**AGUSTUS 2024**

## HALAMAN PERNYATAAN ORISINALITAS

**Skripsi/Tugas Akhir ini adalah hasil karya penulis, dan semua sumber baik yang dikutip maupun dirujuk telah saya nyatakan dengan benar.**

**Nama : Haikal Ahmad Robbani**

**NIM 0110120208**

Depok, 05 Agustus 2024

Tanda Tangan



Haikal Ahmad Robbani

STT - NF

## HALAMAN PENGESAHAN

Skripsi/Tugas Akhir ini diajukan oleh :

Nama : HAIKAL AHMAD ROBBANI

NIM : 0110120208

Program Studi : Sistem informasi

Judul Skripsi : PERANCANGAN APLIKASI MANAJEMEN RUANGAN  
BERBASIS WEB DI PT. TIRTA ASASTA DEPOK  
MENGUNAKAN FRAMEWORK LARAVEL

Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Sistem Informasi. pada Program Studi Sistem Informasi, Sekolah Tinggi Teknologi Terpadu Nurul Fikri

DEWAN PENGUJI

Pembimbing



(Edi Wibowo, S.E, M.M)

Penguji



(Amalia, S.E., M.Ak)

Ditetapkan di : Depok

Tanggal : 05 Agustus 2024

## KATA PENGANTAR

Puji syukur penulis panjatkan ke hadirat Allah SWT, karena atas berkat dan rahmat-Nya, penulis dapat menyelesaikan Tugas Akhir ini. Penulisan Tugas Akhir ini dilakukan dalam rangka memenuhi salah satu syarat untuk mencapai gelar Sarjana komputer Program Studi Sistem Informasi pada Sekolah Tinggi Teknologi Terpadu Nurul Fikri Penulis menyadari bahwa, bantuan dan bimbingan dari berbagai pihak, dari masa perkuliahan sampai pada penyusunan skripsi ini, sangatlah membantu bagi penulis untuk menyelesaikan tugas akhir ini. Oleh karena itu, penulis mengucapkan terima kasih kepada:

1. Allah SWT.
2. Orang tua dan semua anggota keluarga yang telah memberikan dorongan baik secara moril maupun materil dalam penyelesaian tugas ini.
3. Bapak Dr. Lukman Rosyidi, S.T, M.M., M.T selaku Ketua Sekolah Tinggi Teknologi Terpadu Nurul Fikri.
4. Ibu Misna Asqia, S.Kom, M.Kom selaku Ketua Program Studi Sistem Informasi Sekolah Tinggi Teknologi Terpadu Nurul Fikri.
5. Ibu Dr. Amalia Rahmah, S.T, M.T selaku Dosen Pembimbing Akademik yang telah membimbing penulis selama perkuliahan di Sekolah Tinggi Teknologi Terpadu Nurul Fikri.
6. Bapak Edi Wibowo, S.E, M.M selaku Dosen Pembimbing Tugas Akhir penulis dalam menyelesaikan penulisan ilmiah ini.
7. Ibu Amalia, S.E, M.Ak., selaku Dosen Penguji Tugas Akhir penulis dalam menyelesaikan penulisan ilmiah ini.
8. Para Dosen di lingkungan Sekolah Tinggi Teknologi Terpadu Nurul Fikri yang telah membimbing penulis dalam menuntut ilmu yang telah diberikan.
9. Bapak Wira, Bapak Faris dan Bapak Dudi dari divisi IT di PT. Tirta Asasta Depok beserta karyawan yang tidak dapat saya sebutkan satu per satu yang telah meluangkan waktunya untuk memberikan bantuan selama proses pengembangan aplikasi.

10. Teman-teman yang telah memberikan dorongan dalam proses pembuatan tugas akhir ini.

Dalam penulisan karya ilmiah ini, masih terdapat berbagai kekurangan yang mungkin timbul akibat keterbatasan pengetahuan dan kemampuan penulis. Meskipun demikian, penulis telah berupaya menyelesaikan penulisan ini dengan memberikan yang terbaik sebaik mungkin. Oleh karena itu, penulis dengan rendah hati menerima kritik dan saran dari para pembaca untuk perbaikan lebih lanjut. Akhir kata, penulis berharap semoga Allah SWT membalas segala kebaikan dari semua pihak yang telah membantu. Semoga skripsi ini bermanfaat bagi perkembangan ilmu pengetahuan.

Depok, 05 Agustus 2024



Haikal Ahmad Robbani



STT - NF

**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI  
TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS**

---

Sebagai sivitas akademik Sekolah Tinggi Teknologi Terpadu Nurul Fikri, saya yang bertanda tangan di bawah ini:

Nama : HAIKAL AHMAD ROBBANI

NIM 0110120208

Program Studi : Sistem Informasi

Jenis karya : Skripsi / Tugas Akhir

demikian demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada STT-NF Hak Bebas Royalti Noneksklusif (*Non-exclusive Royalty - Free Right*) atas karya ilmiah saya yang berjudul :

PERANCANGAN APLIKASI MANAJEMEN RUANGAN BERBASIS WEB DI PT. TIRTA ASASTA DEPOK MENGGUNAKAN FRAMEWORK LARAVEL beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini STT-NF berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan mempublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

STT - NF

Dibuat di : Depok

Pada tanggal : 05 Agustus 2024

Yang Menyatakan



( Haikal Ahmad Robbani )

## ABSTRAK

Nama : Haikal Ahmad Robbani  
NIM : 0110120208  
Program Studi : Sistem Informasi  
Judul : Perancangan Aplikasi Manajemen Ruangan Berbasis Website  
Di PT. Tirta Asasta Depok Menggunakan Framework Laravel

Pengelolaan manajemen ruangan di PT. Tirta Asasta Depok saat ini masih dilakukan secara manual dan konvensional. Hal ini sering mengakibatkan kendala dan kurang efektif menghadapi kendala, baik bagi pegawai maupun sekretaris ruangan. Oleh karena itu, perancangan aplikasi manajemen ruangan rapat berbasis *website* di PT. Tirta Asasta diharapkan dapat menjadi solusi untuk kendala pengelolaan manajemen ruangan saat ini. Aplikasi ini dirancang menggunakan *framework* Laravel agar memudahkan proses pengembangan serta memiliki struktur aplikasi yang bagus. Metode pengembangan yang digunakan adalah *Extreme Programming* (XP), yang memungkinkan adaptasi terhadap perubahan yang terjadi selama masa pengembangan.

Fitur-fitur yang dikembangkan dalam aplikasi ini meliputi pengajuan dan penerimaan peminjaman ruangan, absensi kegiatan, dan pengelolaan data master aplikasi. Pada proses pengujian, digunakan metode *black-box* testing yang meliputi pengujian fungsional dan non-fungsional. Hasil dari pengujian menunjukkan bahwa aplikasi telah sesuai dengan yang diharapkan. Aplikasi manajemen ruangan yang telah dirancang diserahkan kepada perusahaan dengan nama AMARR (Aplikasi Manajemen Ruang Rapat) yang selanjutnya akan diimplementasikan oleh perusahaan. Dengan aplikasi AMARR, diharapkan manajemen ruang rapat di PT. Tirta Asasta Depok menjadi lebih terorganisir dan efisien, serta mengatasi kendala yang selama ini terjadi.

Kata kunci :Manajemen Ruangan, *Website*, PT.Tirta Asasta Depok, *Extreme Programming*, AMARR ( maksimal 5 kata )



## **ABSTRACT**

*Name* : Haikal Ahmad Robbani  
*NIM* : 0110120208  
*Study Program* : Sistem Informasi  
*Title* : Rancangan Aplikasi Manajemen Ruang Berbasis Website Di PT. Tirta Asasta Depok Menggunakan Framework Laravel

*The management of meeting rooms at PT. Tirta Asasta Depok is currently done manually and conventionally, causing inefficiencies and challenges for both employees and secretaries. Therefore, a web-based meeting room management application has been designed to solve these issues. The application uses the Laravel framework for development to ensure a solid application structure. The development method is Extreme Programming (XP), which allows flexibility for changes during the development process. The main features of the application include room booking requests and approvals, attendance tracking, and master data management. Black-box testing, including both functional and non-functional testing, was used to test the application. The results show that the application meets the required expectations. The meeting room management application, named AMARR (Aplikasi Manajemen Ruang Rapat), has been delivered to the company and will be implemented by PT. Tirta Asasta Depok. With AMARR, it is expected that the management of meeting rooms at PT. Tirta Asasta Depok will become more organized and efficient, effectively solving the existing problems.*

*Key words* : management of meeting rooms, Website, PT. Tirta Asasta Depok, Extreme Programming, AMARR

## DAFTAR ISI

HALAMAN PERNYATAAN ORISINALITAS.....	3
HALAMAN PENGESAHAN.....	4
KATA PENGANTAR.....	5
HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI.....	7
TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS.....	7
ABSTRAK.....	8
<i>ABSTRACT</i> .....	9
DAFTAR ISI.....	10
DAFTAR GAMBAR.....	13
DAFTAR TABEL.....	15
BAB I.....	16
PENDAHULUAN.....	16
1.1 Latar belakang.....	16
1.2 Rumusan Masalah.....	18
1.3 Tujuan dan Manfaat Penelitian.....	18
1.3.1 Tujuan.....	18
1.3.2 Manfaat.....	18
1.4 Batasan Masalah.....	19
1.5 Sistematika Penulisan.....	19
BAB II.....	21
KAJIAN LITERATUR.....	21
2.1 Manajemen.....	21
2.2 Manajemen ruangan.....	22
2.3 Website.....	22
2.3.1 HTML.....	23
2.3.2 CSS.....	23
2.4 Javascript.....	24
2.5 <i>Database</i> .....	24

2.6	Mysql .....	25
2.7	PHP .....	25
2.8	<i>Framework</i> .....	26
2.8.1	Laravel.....	26
2.9	UML ( <i>Unified Modeling Language</i> ).....	27
2.9.1	Activity Diagram.....	27
2.9.2	ERD.....	29
2.10	XAMPP.....	31
2.11	Visual studio code.....	32
2.12	Profil Perusahaan .....	32
2.13	Penelitian Terkait .....	33
<b>BAB III METODOLOGI PENELITIAN .....</b>		<b>38</b>
3.1	Tahapan Penelitian .....	38
3.2	Rancangan Penelitian .....	40
3.2.1	Jenis Penelitian.....	40
3.2.2	Metode Analisis Data .....	41
3.2.3	Metode Pengumpulan Data.....	41
3.2.4	Metode Pengujian.....	42
3.2.5	Metode Implementasi dan Evaluasi.....	43
3.2.6	Lingkungan Pengembangan.....	44
<b>BAB IV IMPLEMENTASI DAN EVALUASI.....</b>		<b>45</b>
4.1	Studi Pendahuluan.....	45
4.1.1	Studi Lapangan.....	45
4.1.2	Studi Pustaka.....	51
4.2	Pengembangan Sistem:.....	51
4.2.1	<i>Planning</i> .....	52
4.2.2	<i>Design</i> .....	61
4.2.3	<i>Coding</i> .....	75
4.2.4	<i>Testing</i> .....	77
4.2.5	<i>Release</i> .....	92
4.3	Hasil.....	92
<b>BAB V KESIMPULAN DAN SARAN.....</b>		<b>106</b>

5.1 Kesimpulan .....	106
5.2 Saran .....	108
DAFTAR PUSTAKA .....	109
Bibliografi .....	109
LAMPIRAN.....	113



STT - NF

## DAFTAR GAMBAR

Gambar 2. 1 Contoh activity diagram .....	29
Gambar 2. 2 contoh erd .....	30
Gambar 3. 1 Tahapan Penelitian .....	38
Gambar 3. 2 Rancangan Penelitian .....	40
Gambar 3. 3 alur kerja XP .....	43
Gambar 4. 1 papan tulis untuk mencatat jadwal peminjaman .....	49
Gambar 4. 2 chat pegawai menanyakan ketersediaan ruangan .....	50
Gambar 4. 3 chat pegawai menanyakan ketersediaan ruangan .....	50
Gambar 4. 4 halaman login .....	61
Gambar 4. 5 halaman register .....	62
Gambar 4. 6 halaman muka .....	63
Gambar 4. 7 halaman data master .....	64
Gambar 4. 8 halaman pengajuan .....	65
Gambar 4. 9 halaman agenda .....	66
Gambar 4. 10 halaman absen .....	67
Gambar 4. 11 halaman absen tamu .....	68
Gambar 4. 12 halaman daftar hadir .....	68
Gambar 4. 13 halaman riwayat .....	69
Gambar 4. 14 Diagram ERD .....	70
Gambar 4. 15 diagram Pengajuan .....	72
Gambar 4. 16 diagram activity pengajuan kegiatan .....	73
Gambar 4. 17 diagram activity absen .....	74
Gambar 4. 18 nama database .....	76
Gambar 4. 19 struktur tabel database .....	76
Gambar 4. 20 struktur tabel database .....	76
Gambar 4. 21 logo AMARR .....	92
Gambar 4. 22 halaman login .....	92
Gambar 4. 23 halaman muka .....	93
Gambar 4. 24 modal keterangan kegiatan .....	93
Gambar 4. 25 modal detail ruangan .....	94

Gambar 4. 26 halaman ruang rapat .....	95
Gambar 4. 27 modal tambah ruangan .....	95
Gambar 4. 28 foto ruangan terkait .....	96
Gambar 4. 29 modal edit ruangan .....	96
Gambar 4. 30 modal hapus.....	96
Gambar 4. 31 halaman permohonan .....	97
Gambar 4. 32 modal detail ajuan .....	98
Gambar 4. 33 halaman agenda .....	99
Gambar 4. 34 modal konfirmasi ajuan.....	99
Gambar 4. 35 notifikasi jumlah ajuan yang harus di konfirmasi .....	100
Gambar 4. 36 halaman riwayat .....	100
Gambar 4. 37 halaman daftar hadir.....	101
Gambar 4. 38 modal daftar absen peserta .....	101
Gambar 4. 39 halaman absen .....	102
Gambar 4. 40 halaman tamu .....	103
Gambar 4. 41 halaman absen tamu .....	103
Gambar 4. 42 halaman muka di akses menggunakan handphone.....	104
Gambar 4. 43 tema gelap aplikasi.....	105
Gambar 4. 44 sidebar .....	105

STT - NF

## DAFTAR TABEL

Tabel 2. 1 Komponen diagram activity .....	28
Tabel 2. 2 Penelitian terkait .....	33
Tabel 4.1 Wawancara.....	45
tabel 4.2 User stories .....	52
tabel 4.3 analisis kebutuhan user.....	53
tabel 4. 4 kebutuhan admin .....	55
tabel 4. 5 kebutuhan admin .....	56
Tabel 4. 6 Testing Databases .....	77
tabel 4. 7 Testing Integrasi template dan Data master .....	78
tabel 4. 8 Testing Iterasi Pengembangan fitur pengajuan .....	82
tabel 4. 9 Iterasi Modul Halaman Utama .....	86
tabel 4. 10 Iterasi role.....	87
tabel 4. 11 Testing Non-fungsional.....	90

STT - NF

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar belakang**

Manajemen ruangan adalah seni dalam mengatur suatu ruangan yang dapat digunakan untuk berbagai di suatu perusahaan, dan dikelola secara efisien dan efektif agar tercapai tujuan yang diinginkan. Ruangan memiliki peran penting dalam suatu perusahaan karena ruangan merupakan tempat dalam menggelar berbagai kegiatan untuk perusahaan. Tentunya kegiatan yang dilakukan memerlukan ruangan sebagai tempat penyelenggaraan yang nyaman dan representatif. Oleh karena itu, kegiatan manajemen ruangan yang dilakukan dengan baik dan efektif dapat berperan dalam memajukan kegiatan perusahaan.

PT. Tirta Asasta merupakan perusahaan daerah air minum yang bergerak di bidang pelayanan masyarakat khususnya penyediaan air bersih di kota Depok. Untuk memaksimalkan tujuan dan pelayanan perusahaan untuk melayani masyarakat, PT. Tirta Asasta memerlukan dukungan infrastruktur kantor untuk membantu kegiatan organisasi, salah satunya adalah penggunaan ruangan untuk berbagai kegiatan. Memiliki 5 ruangan, PT. Tirta Asasta memerlukan manajemen ruangan agar mendukung kegiatan di ruangan yang nyaman dan representatif. Setiap ruangan memiliki masing-masing sekretaris sebagai penanggung jawab ruangan, seperti ruang rapat dirut dikelola oleh sekretaris direktur utama, ruang rapat dirum dikelola oleh sekretaris direktur umum. Secara struktural, masing-masing pengelola ruangan tersebut berada di bawah divisi Rumah Tangga.

Manajemen ruangan di PT. Tirta Asasta saat ini masih dilakukan secara konvensional atau manual, Seperti melakukan pencatatan jadwal, penerimaan peminjaman ruangan, dokumentasi peminjaman, inventaris ruangan dan beberapa hal lainnya. Selain itu, pegawai yang ingin mengajukan peminjaman ruangan harus menghubungi pengelola ruangan yaitu bagian Sekretaris Perusahaan melalui telepon. Setelah mendapatkan informasi ketersediaan ruangan maka peminjam akan mengirimkan surat peminjaman kepada sekretaris perusahaan untuk selanjutnya diproses oleh sekretaris perusahaan agar diberikan persetujuan, ditolak atau di pindahkan ke waktu lain. Berdasarkan hasil wawancara dengan pegawai



perusahaan, cara ini dinilai tidak efisien karena harus menunggu dan beberapa kali bagian sekretaris perusahaan tidak dapat dihubungi. Selain itu penggunaan surat juga dinilai sebagai pemborosan kertas. Permasalahan lain yang terjadi adalah terdapat kesalahan komunikasi antara pengelola ruangan yang mengakibatkan terdapat jadwal saling bersamaan di satu ruangan .

Dengan perkembangan era digital saat ini, penggunaan aplikasi memainkan peranan penting dalam mendukung kegiatan perusahaan, termasuk manajemen ruangan. Solusi yang diberikan peneliti adalah aplikasi berbasis web yang kebutuhan manajemen ruangan di PT. Tirta Asasta. Penggunaan aplikasi *web* akan memudahkan pengguna dalam melakukan peminjaman ruangan secara *online* karena pengguna dapat melihat jadwal peminjaman secara *realtime*. Dalam proses perancangan aplikasi tersebut dibutuhkan teknologi yang dapat mendukung kebutuhan aplikasi, seperti ketepatan waktu, biaya yang ekonomis dan kemudahan dalam proses pembuatan. *Database* yang digunakan untuk menyimpan data aplikasi adalah MySQL. Aplikasi yang dirancang akan menggunakan *framework* laravel versi 8. Laravel merupakan *framework* berbasis MVC yang ditulis dalam bahasa PHP yang memiliki keunggulan efisiensi dan memiliki sintaks yang ekspresif, jelas dan efisien [1].

Salah satu fitur penting laravel adalah Eloquent. Eloquent adalah sebuah ORM (*Object-Relational Mapping*) yaitu teknik untuk berinteraksi dengan databases menggunakan objek dan model. Fitur Eloquent dapat memberikan kemudahan pengembang seperti mengakses data dari *database*, melakukan operasi CRUD (*create, read, update, delete*) tanpa harus menulis *query* SQL secara manual. [2]. Selain itu laravel juga memiliki beberapa *library* yang dapat digunakan untuk kebutuhan aplikasi, salah satunya adalah carbon yang berfungsi untuk memanipulasi tanggal dan waktu. Berdasarkan hal tersebut, peneliti membuat penelitian yang berjudul *Perancangan Aplikasi Manajemen Rruangan berbasis Web di PT. Tirta Asasta Depok menggunakan Framework Laravel.*

## 1.2 Rumusan Masalah

1. Bagaimana proses perancangan aplikasi manajemen ruangan?
2. Apa saja fitur-fitur yang terdapat di aplikasi manajemen ruangan?
3. Bagaimana hasil pengujian fitur-fitur yang terdapat di aplikasi manajemen ruangan?

## 1.3 Tujuan dan Manfaat Penelitian

### 1.3.1 Tujuan

Tujuan dari penelitian ini adalah :

1. Merancang dan mengembangkan aplikasi manajemen ruangan di PT. Tirta Asasta Depok.
2. Mengetahui fitur-fitur pada aplikasi manajemen ruangan.
3. Mengetahui hasil pengujian fitur-fitur pada aplikasi manajemen ruangan.

### 1.3.2 Manfaat

Pihak yang diharapkan mendapatkan manfaat dari penelitian ini adalah :

1. PT. Tirta Asasta Depok

Manfaat yang didapat oleh perusahaan adalah dapat menggunakan aplikasi manajemen ruangan untuk solusi menyelesaikan permasalahan manajemen ruangan di perusahaan.

2. Penulis

Manfaat yang di dapatkan untuk penulis adalah memberikan pengalaman dalam pengembangan aplikasi berbasis *website*. Serta mengasah kemampuan dalam menyelesaikan permasalahan yang ada di lingkungan.

3. Akademik

Penelitian ini diharapkan memberikan manfaat kepada akademik berupa pedoman dan tolak ukur untuk menilai efektivitas proses pembelajaran yang dapat digunakan sebagai bahan evaluasi.

#### 4. Pembaca

Dapat digunakan sebagai tambahan pengetahuan dan referensi apabila melakukan penelitian terkait.

### 1.4 Batasan Masalah

Berikut adalah batasan-batasan masalah pada penelitian ini:

1. Aplikasi menangani pengelolaan data ruangan, data pengajuan peminjaman serta data kegiatan peminjaman ruangan.
2. Aplikasi memiliki dua *role* yaitu *user* dan *admin*.
3. Aplikasi tidak meliputi pada bagian keamanan.
4. *Framework* yang digunakan adalah laravel 8

### 1.5 Sistematika Penulisan

Untuk memudahkan proses penulisan penelitian ini maka diperlukan sistematika penulisan yang terdiri dari bab dan sub-bab. Sistematika penulisan tugas akhir ini adalah sebagai berikut :

#### **BAB I      Pendahuluan**

Bab ini merupakan bab pertama yang menjelaskan mengenai latar belakang masalah. Pada bab pendahuluan juga membahas sub-bab rumusan masalah, tujuan dan manfaat, batasan masalah dan sistematika penulisan.

#### **BAB II     Kajian Literatur**

Bab ini memberikan penjelasan mengenai apa saja definisi variabel yang digunakan dalam penelitian, Teori untuk menganalisis permasalahan penelitian serta penjelasan mengenai teknologi atau *tools* yang digunakan untuk menyelesaikan permasalahan penelitian. Bab ini juga memiliki penelitian terkait yang disajikan dalam bentuk tabel perbandingan.

#### **BAB III    Metodologi penelitian**

Bab ini menjelaskan terkait metode penelitian, tahapan-tahapan penelitian serta rancangan penelitian. pada bab IV juga memberikan penjelasan mengenai metode pengumpulan data penelitian serta lingkup pengembangan.

#### **BAB IV IMPLEMENTASI DAN EVALUASI**

Bab IV membahas terkait hasil implementasi perancangan aplikasi yang disertai dengan *flowchart*, bukti nyata seperti *screenshot*, foto, dan sebagainya. Selain itu, pada Bab IV terdapat mengenai hasil data penelitian dan evaluasi hasil penelitian.

#### **BAB V Kesimpulan Dan Saran**

Pada Bab V berisikan mengenai kesimpulan dari hasil penelitian yang dilakukan serta memberikan saran terkait hal-hal yang dapat di kembangkan untuk penelitian terkait yang akan datang.



STT - NF

## BAB II

### KAJIAN LITERATUR

#### 2.1 Manajemen

Dalam pengertian secara umum, manajemen dapat diartikan sebagai sebuah seni dalam mengatur suatu hal agar hal yang di manajemen tersebut mencapai tujuan yang diinginkan melalui upaya bersama dengan sejumlah orang atau sumber milik organisasi. Secara etimologi, manajemen berasal dari bahasa Inggris *to manage* yang memiliki makna mengurus, mengawasi dan membimbing. Kata *manage* sendiri berasal dari bahasa Italia yaitu *menaggio*, yang berarti pelaksanaan atau penanganan suatu hal [3].

Menurut Andrew F. Skila, manajemen adalah aktivitas yang melibatkan perencanaan, pengaturan, pengorganisasian, pelaksanaan, dan pengambilan keputusan yang dijalankan oleh suatu organisasi untuk mengelola sumber daya yang dimilikinya. Sedangkan Henry Fayol mengemukakan bahwa manajemen mencakup lima konsep utama, yaitu perencanaan, koordinasi, perintah, pengaturan, pengorganisasian, dan pengendalian. Kedua pendapat ini menunjukkan bahwa manajemen adalah proses yang kompleks yang bertujuan untuk memastikan bahwa sumber daya organisasi digunakan secara efektif untuk mencapai tujuan yang ditetapkan. [4].

Berdasarkan penjelasan di atas, dapat disimpulkan bahwa manajemen merupakan sebuah kegiatan yang memiliki seni dalam merencanakan, mengatur dan merancang yang dilakukan oleh sebuah organisasi. Kegiatan tersebut dilakukan agar pengelolaan sumber daya yang dimiliki dapat digunakan secara efektif dan efisien sesuai tujuan awal.

## 2.2 Manajemen ruangan

Berdasarkan penjelasan manajemen sebelumnya, manajemen ruangan dapat diartikan sebagai kegiatan untuk mengelola ruangan sebagai sumber daya organisasi yang di dalamnya terdiri dari merencanakan, mengatur dan merancang kebutuhan ruangan di organisasi atau perusahaan tersebut. Ruangan memiliki peran penting dalam suatu perusahaan karena ruangan merupakan tempat dalam menggelar berbagai kegiatan untuk perusahaan. Tentunya kegiatan yang dilakukan memerlukan ruangan sebagai tempat penyelenggaraan yang nyaman dan representatif. Dengan kegiatan manajemen ruangan, maka ruangan yang nyaman representatif dapat di dapat dengan ketentuan manajemen ruangan dilakukan dengan benar dan efektif.

## 2.3 Website

*Website* atau *world wide web* atau bisa disebut web merupakan suatu halaman digital yang terdiri dari satu halaman atau lebih dan dapat diakses menggunakan internet. Pertama kali *website* tersedia tahun 1991 dan diluncurkan oleh ahli komputer asal Inggris bernama Sir Timothy John “Tim” Berners-Lee. Pada awalnya, Sir Timothy membuat website dengan tujuan sebagai sarana bertukar dan pembaharuan informasi antara peneliti di tempat kerjanya. Seiring perkembangan teknologi Saat ini, *website* memiliki dapat digunakan orang lain yang dapat memberikan informasi, berhubungan dengan orang lain, atau menjadi suatu basis aplikasi yang dapat memberikan layanan tertentu. Halaman website terdiri dari berupa teks, gambar, audio dan bahkan video yang tersedia secara digital.

Untuk membuat suatu halaman *website* dapat menggunakan komponen HTML dan CSS. HTML adalah singkatan dari *hypertext markup language* yang berfungsi sebagai kerangka dari suatu halaman *website*. CSS merupakan singkatan dari *Cascading Style Sheet* yaitu bahasa yang digunakan untuk memanipulasi tampilan HTML agar lebih indah dan cantik [5]. Selain itu, penggunaan bahasa javascript dalam proses pengembangan

aplikasi web sebagai kombinasi dari ketiga komponen tersebut dapat menciptakan aplikasi web yang lebih terstruktur, bergaya dan interaktif.

### 2.3.1 HTML

HTML merupakan singkatan dari *Hypertext Markup Language* yaitu bahasa standar yang digunakan untuk membangun *website* dan menampilkan konten pada suatu halaman *website* [6]. HTML memanfaatkan tag-tag yang telah di deklarasikan untuk elemen di suatu konten dan dapat saling berhubungan dengan dokumen HTML lainnya atau sering disebut dengan *link*. [7]. Sebagian besar, situs web yang ada di Internet saat ini masih menggunakan HTML sebagai teknologi utama *website* mereka. Jika dibandingkan dengan tubuh manusia, HTML dapat dianggap sebagai rangka tulang yang menjadi fondasi dan struktur utama dalam membangun tubuh tersebut.

### 2.3.2 CSS

CSS atau singkatan dari *cascading style sheet* adalah dokumen bahasa yang membantu pengembang untuk memberikan keindahan dan formating halaman *website*. CSS biasanya diterapkan ke dalam elemen HTML dalam pendefinisian *style website*. CSS dapat mengubah ukuran *text* dapat diperbesar atau diperkecil, dapat mengatur besar kecil suatu gambar, memberikan warna border tabel atau warna pada suatu elemen tombol HTML, dan masih banyak kemampuan lain CSS dalam mengatur tampilan HTML. Jika dibandingkan dengan tubuh manusia, CSS dapat dianggap sebagai kulit dan pakaian yang memberikan penampilan dan gaya yang menarik, sekaligus melindungi dan menyempurnakan struktur yang dibangun oleh HTML.

## 2.4 Javascript

Javascript adalah bahasa pemrograman yang bersifat *client-side* yang artinya baris perintah pada bahasa ini di eksekusi di sisi browser [8]. Javascript diluncurkan pada tahun 1995 oleh Netscape dan dinamakan “LiveScript” yang berfungsi sebagai bahasa sederhana untuk browser. Oleh karena itu Javascript dapat digunakan untuk mendeteksi dan bereaksi terhadap event-event yang di sebabkan di *client-side*. Seperti melakukan memvalidasi kapasitas ruangan dengan jumlah peserta, memunculkan modal, menambah interaktif pada suatu *button* di halaman. Hal-hal tersebut dapat digunakan tanpa perlu *user* melakukan *reload* halaman sehingga halaman website menjadi lebih interaktif dan menghemat waktu penggunaan. Jika diibaratkan dengan tubuh manusia, Javascript dapat dianggap sebagai otak dari website yang memberikan tambahan interaktif kepada *user*.

## 2.5 Database

*Database* atau disebut basis data adalah tempat disimpannya kumpulan kumpulan informasi dari suatu website yang kemudian disusun di perangkat lunak agar dapat diakses dengan mudah. *Database* merupakan salah satu komponen penting yang harus diperhatikan dalam perancangan aplikasi berbasis web [9]. Dengan *database*, pengembang aplikasi web dapat menyimpan berbagai data dan informasi, seperti data ruangan, data pengajuan, data fasilitas dan beberapa data lainnya yang terkait. Salah satu jenis *database* yang banyak digunakan adalah *relational database*. Dengan menggunakan konsep *relational model* atau yang berarti data disimpan dalam tabel-tabel yang saling terhubung melalui kunci utama dan kunci asing [10].



## 2.6 Mysql

Mysql merupakan salah sistem manajemen basis data atau DBMS yang menggunakan bahasa SQL atau *Structured Query Language* sebagai bahasa perintahnya. Mysql bersifat *open source* dan memiliki dua bentuk lisensi yaitu *free software* dan *shareware*. Selain itu Mysql juga termasuk sebagai *relational database* yang merenungkan pengguna *database* untuk lebih mudah dan cepat dalam melakukan manipulasi data di *database* [11]. Karena bersifat *open source*, tentunya hal ini merupakan keunggulan utama dari Mysql karena memiliki banyak pengguna dan dukungan komunitas di seluruh dunia.

Selain itu, Mysql juga dapat terintegrasi dengan berbagai bahasa pemrograman yang dapat memudahkan dalam proses pembuatan aplikasi. Kelebihan lainnya adalah memiliki struktur yang tabel yang fleksibel dan memiliki *query* (perintah untuk manipulasi data) yang mudah dipahami. Berdasarkan keunggulan tersebut membuat Mysql sangat cocok untuk digunakan dalam pengembangan aplikasi berbasis web [12].

## 2.7 PHP

PHP atau singkatan dari *Hypertext Preprocessor* adalah bahasa pemrograman skrip populer yang saat ini banyak digunakan untuk pengembangan aplikasi berbasis web. Di Indonesia, sebanyak 729,822 website menggunakan PHP sebagai bahasa utama mereka [13]. Awalnya, PHP diciptakan oleh insinyur perangkat lunak yaitu Rasmus Lerdorf dan digunakan sebagai alat untuk mencatat jumlah pengunjung di situs web pribadinya [14]. Alasan mengapa PHP digunakan untuk pengembangan web adalah karena bahasa pemrograman PHP bersifat *server-side scripting*. Ini artinya, Proses *parsing* kode PHP di dalam *web server* dan terjadi ketika kode tersebut dipanggil dari *web browser* dan diterjemahkan dalam dokumen HTML, yang selanjutnya ditampilkan kembali *oleh web server*.

## 2.8 *Framework*

*Framework* suatu kerangka kerja yang sudah tersusun secara lengkap dan generik yang berfungsi untuk pembuatan ataupun pengembangan aplikasi maupun sistem. *Framework* pada umumnya sudah dilengkapi dengan fitur-fitur yang siap pakai sehingga pengembang dapat lebih fokus dengan fitur-fitur aplikasi yang akan dibuat sehingga tidak perlu menulis baris kode dari nol atau awal.

Dengan menggunakan *framework* dapat mempermudah, mempercepat dan mengorganisir penulisan kode secara sistematis [15]. Keunggulan lainnya adalah *framework* menyediakan library atau fungsi yang dapat digunakan langsung, contohnya fungsi migrate pada Laravel yang mempermudah pembuatan tabel pada database dan memberikan struktur yang baik dalam aplikasi yang dibuat [16]. Oleh karena itu *framework* termasuk kedalam peran penting dalam membantu pengembang dalam pengerjaan aplikasi. Mulai dari efisiensi waktu, tenaga hingga biaya.

### 2.8.1 **Laravel**

Laravel adalah sebuah *framework* PHP yang dirilis dibawah lisensi MIT. *Framework* laravel bersifat *open source* dan diciptakan oleh Taylor Otwell. Laravel dibangun dengan konsep arsitektur MVC yang memisahkan kode aplikasi menjadi tiga bagian yaitu *Model, View dan Controller*. Laravel saat ini menjadi salah satu *framework* yang banyak digemari oleh *developer* web, berdasarkan data dari stackshare.io, 23348 menggunakan Laravel. Penggunaan Laravel dalam pengembangan website dirancang untuk mengurangi biaya pengembangan awal dan biaya pemeliharaan tanpa mengurangi kualitas aplikasi [17]. Hal ini didukung dengan laravel menyediakan solusi untuk beberapa fitur umum yang dibutuhkan di aplikasi web modern. Solusi ini diberikan dalam bentuk fitur-fitur yang sudah terinstal atau *package* yang dapat diinstall sesuai dengan kebutuhan aplikasi, beberapa *packages* tersebut seperti *Authentication, Authorization, Notifications & Mail, Database Migrations* dan masih banyak *packages* lainnya [18].






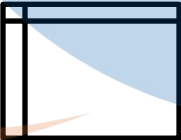
## 2.9 UML (*Unified Modeling Language*)

UML merupakan singkatan dari *Unified Modeling Language*. Ketika melakukan perancangan aplikasi maka dibutuhkan dokumentasi yang baik agar dapat menggambarkan kerangka dasar dari keseluruhan aplikasi. UML merupakan salah satu *tools* yang dijadikan untuk menggambarkan kerangka tersebut dengan baik. Dengan UML, suatu aplikasi dapat diwakili dengan diagram sebagai visual dari rancangan aplikasi itu sendiri. Selain itu dengan UML, pengembang dapat membuat representasi sistem aplikasi yang jelas dan baik sehingga menjadi acuan dalam proses pengembangan aplikasi. Terdapat beberapa macam dari jenis UML, namun beberapa yang paling sering digunakan seperti, *Use case diagram*, *Sequence diagram*, *Class diagram* dan yang terakhir merupakan jenis yang digunakan oleh peneliti yaitu *Activity Diagram*.

### 2.9.1 Activity Diagram

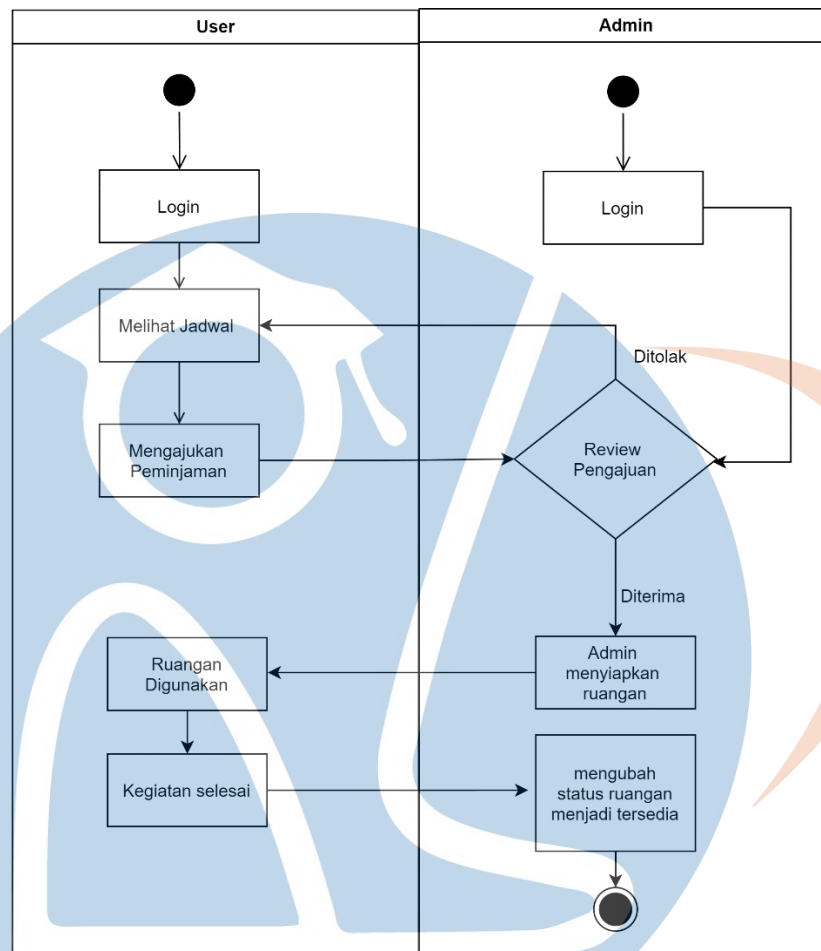
Salah satu kunci penting dalam merancang aplikasi adalah melakukan visualisasi bagaimana alur penggunaan aplikasi ketika aplikasi tersebut digunakan oleh *user*. Hal ini penting dikarenakan pengembang ingin ketika aplikasi di tangan *user* maka dijalankan sesuai rencana dan tujuan dibuat aplikasi tersebut. Penggunaan *Activity diagram* merupakan salah satu solusi dalam melakukan penggambaran awal tersebut melalui visual bentuk diagram. *Activity diagram* mengilustrasikan urutan aktivitas yang bisa berupa urutan menu atau proses bisnis yang terdapat dalam aplikasi tersebut. Diagram ini menjelaskan proses yang sedang berlangsung dengan menggambarkan secara vertikal untuk mendeskripsikan aktor, tindakan yang dilakukan, dan kapan tindakan tersebut harus terjadi [19]. Komponen-komponen yang terdapat di dalam *activity diagram* adalah sebagai berikut :

Tabel 2. 1 Komponen diagram activity

Simbol	Nama	Keterangan
	<i>Initial State</i>	Dimulainya alur kegiatan pada diagram.
	<i>Final State</i>	Bagian akhir dari alur kegiatan pada diagram
	Aktivitas	Aktivitas yang sedang terjadi di aplikasi.
	Percabangan/ <i>Decision</i>	Pada bagian percabangan terdapat pilihan aktivitas lebih dari satu atau aktivitas yang menyebabkan terdapat dua hasil.
	<i>Flow line</i>	Garis yang menghubungkan antar aktivitas
	<i>Swimlane</i>	Bagian yang memisahkan aktor yang bertanggung jawab terhadap aktivitas yang terjadi

STT - NF

Untuk lebih jelasnya, berikut merupakan contoh dari *activity* diagram :



Gambar 2. 1 Contoh activity diagram

Contoh gambar di atas memberikan ilustrasi singkat mengenai alur aplikasi peminjaman ruangan yang terdapat dua *Swimlane* yaitu admin dan *user*.

### 2.9.2 ERD

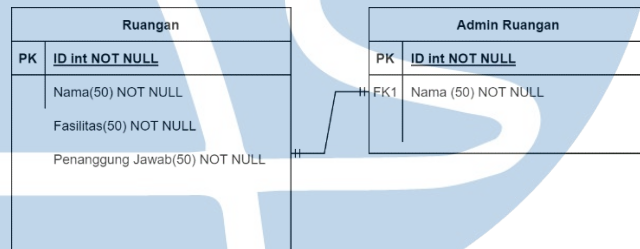
ERD atau *Entity Relationship Diagram* adalah sebuah diagram struktural yang digunakan untuk merancang sebuah *database* [20]. ERD memiliki manfaat agar pengembang dapat memiliki gambaran dan lebih memahami terkait struktur *database* yang akan dibuat. Dengan ERD pengembang dapat mengetahui tabel-tabel yang memiliki hubungan atau relasi dengan tabel lainnya. Tiga komponen ERD adalah :

*Entity* : Objek yang memiliki properti atau atribut yang disimpan di *database*. Seperti *user*, ruangan rapat dan divisi.

*Relationship* : *Relationship* atau relasi merupakan komponen yang memberikan gambar hubungan antara satu *entity* dengan *entity* lainya atau lebih. *Relationship* memiliki jenis sebagai berikut :

- *One-to-one* : Satu *entity* memiliki hubungan langsung dengan satu *entity* lainya, contohnya *user* dengan absen
- *One-to-many* : Satu *entity* yang memiliki hubungan dengan lebih dari satu *entity* lainya, contohnya pengajuan peminjaman dengan peserta rapat
- *Many-to-many* : *entity* yang memiliki atau berhubungan dengan lebih dari satu *entity* dan juga sebaliknya, contohnya adalah ruang rapat dengan pengajuan peminjaman

ERD menggunakan garis hubung sebagai representasi untuk menunjukkan hubungan antar entitas.



Gambar 2. 2 contoh erd

Gambar di atas memberikan ilustrasi relasi "*One-to-One*" di antara kedua tabel ini yang ditandai dengan garis hubungan dari kolom Penanggung Jawab pada tabel Ruangan ke kolom ID pada tabel Admin Ruangan, yang dapat diartikan bahwa setiap ruangan ditangani oleh hanya satu admin ruangan tertentu

## 2.10 XAMPP

XAMPP adalah *software* yang menyediakan fitur menjadi *web server* untuk kebutuhan pengembangan aplikasi web. *Software* ini bersifat *open source* dan dapat digunakan gratis oleh para pengembang aplikasi web. Selain itu XAMPP juga disebut sebagai *Standalone server* atau server yang dapat berdiri sendiri memudahkan pengembang dalam menjalankan proses pengeditan dan pengembangan aplikasi tanpa harus terhubung ke jaringan atau server lainnya [21]. Nama XAMPP adalah singkatan dari fitur-fitur yang tersedia, yaitu :

X : mengartikan aplikasi ini tersedia untuk berbagai sistem operasi, seperti *Windows, Linux, MacOS* dan juga *Solaris*.

A : mewakili Apache, yaitu aplikasi web server yang berfungsi untuk menghasilkan halaman web yang benar kepada *user* berdasarkan kode yang ditulis pengembang [22].

M : yaitu Mysql, digunakan sebagai *database* penyimpanan data aplikasi web. Mysql merupakan *database* yang bersifat *relational database*. Konsep *relational model* atau yang berarti data disimpan dalam tabel-tabel yang saling terhubung melalui kunci utama dan kunci asing.

P : PHP, merupakan bahasa pemrograman yang bersifat *server-side* dan digunakan untuk pengembangan aplikasi web.

P : Perl, adalah bahasa pemrograman serba guna yang awalnya dikembangkan untuk manipulasi teks dan sekarang digunakan untuk berbagai kebutuhan, seperti administrasi sistem, pengembangan web dan pengembangan GUI. Dikarenakan aplikasi di atas sudah termasuk ke dalam XAMPP maka tidak perlu lagi melakukan konfigurasi secara manual karena sudah di konfigurasi otomatis ketika menginstall XAMPP.

## 2.11 Visual studio code

Visual studio code atau VScode adalah aplikasi kode editor yang dikembangkan oleh Microsoft dan tersedia untuk sistem operasi Windows, Linux dan Mac Os [20]. Aplikasi ini adalah salah satu yang populer dalam digunakan untuk melakukan code editor dalam proses pengembangan aplikasi. Kelebihan aplikasi ini mendukung berbagai bahasa mulai dari PHP, HTML, JavaScript, Python dan lainnya. Selain itu beberapa fitur seperti *intellisense*, yaitu fitur semacam *autocorrect* pada keyboard *smartphone*, *debugging*, github integration dan marketplace yang menyediakan berbagai *extension* [23] menjadi alasan mengapa kode editor ini cukup populer.

## 2.12 Profil Perusahaan

PT. Tirta Asasta Depok merupakan perusahaan perseroan daerah atau perseroda yang bergerak di bidang pelayanan air bersih di wilayah kota Depok. Pada awalnya PT. Tirta Asasta merupakan perusahaan daerah minum yang tergabung dengan PDAM kahuripan Bogor. Pada tahun 2015 dilakukan pemindahan aset dan pegawai tirta kahuripan ke PDAM tirta asasta. Selanjutnya pada tahun 2021, telah terjadi perubahan badan hukum perusahaan daerah air minum tirta asasta kota Depok menjadi PT.Tirta Asasta Depok (Perseroda). Memiliki visi Menjadi Perusahaan Air Minum dengan kinerja terbaik melalui pelayanan prima dan berwawasan lingkungan. Guna mendukung visi tersebut, kegiatan perusahaan yang nyaman dan representatif menjadi salah satu hal yang penting. Saat ini PT. Tirta Asasta Depok memiliki lima ruangan yang dapat digunakan untuk kebutuhan kegiatan perusahaan dan dikelola oleh bagian sekretaris perusahaan. Dengan kegiatan yang cukup banyak sementara ketersediaan ruangan yang terbatas, maka sudah seharusnya PT. Tirta Asasta memiliki sistem manajemen ruangan yang lebih baik.



## 2.13 Penelitian Terkait

Tabel 2. 2 Penelitian terkait

No.	Nama dan Tahun	Judul	Hasil	Metode Pengujian	Kesimpulan
1	Muhammad Amin, 2019	Booking Meeting Management System Berbasis Framework Laravel	Aplikasi Booking ruangan	BlackBox	Berdasarkan hasil pengujian <i>black-box</i> , fungsi pada sistem bekerja sudah sesuai dengan yang diharapkan. Saran untuk penelitian ini adalah melakukan pengembangan lagi dengan menambah fitur tampilan yang lebih baik dan dapat memproses melalui <i>mobile application</i> .
2	Wulung Wahyu Witono, Faishal Mufied Al Anshary, Nia Ambarsari, 2023	Aplikasi Berbasis Website Untuk Booking Hotel Surya Andesa Klaten Menggunakan Framework Laravel Dengan Metode Waterfall	Aplikasi Booking Hotel	UAT( <i>user acceptance test</i> )	Berdasarkan hasil UAT dapat disimpulkan Aplikasi website untuk booking hotel surya andesa dibangun dengan <i>framework laravel</i> dapat terselesaikan dan menyelesaikan permasalahan terkait pemesanan kamar serta manipulasi data oleh karyawan.
3	Nofiati, 2020	Sistem Informasi Perpustakaan Berbasis Web Dengan Framework Laravel	Sistem Informasi Perpustakaan	Black Box	Sistem perpustakaan di SMP Muhammadiyah 7 Semarang perpustakaan telah membantu dalam mempermudah pengelolaan data, mengamankan penyimpanan data, mengurangi kesalahan

					dalam pencatatan, dan mempercepat proses pencarian informasi.
4	Fitri Bimantoro, 2021	Sistem Informasi Manajemen Ruang Program Studi Teknik Informatika Menggunakan Framework Laravel	Sistem Informasi Manajemen Ruang	Black Box	Sistem Informasi manajemen ruangan berbasis website di Program Studi Teknik informatika sudah berjalan cukup baik karena pegawai telah mampu mengoperasikan sistem. Selain itu uji black box sudah cukup baik dan mendapat Mean Opinion Scores sebesar 69.6.
5	Mochammad Alif Pratama, 2022	Rancang Bangun Sistem Pemesanan Meeting Room Berbasis Web Menggunakan Framework Laravel	Sistem Pemesanan Meeting Room	UAT(user acceptance test)	Aplikasi Booking Meeting Room PT. Vivo Mobile Indonesia membantu pemesanan ruang meeting secara efektif dan efisien. Aplikasi ini dapat diakses kapan saja dan di mana saja. kekurangan aplikasi ini adalah tampilan yang kurang menarik dan waktu <i>loading</i> yang masih cukup lama.

Beberapa penelitian terkait yang menjadi rujukan dan referensi dalam penelitian ini :

1. Penelitian dilakukan oleh Muhammad Amin dan Yeremia Alfa Susetyo, M.Cs. (2019) yang berjudul “*Booking Meeting Management System Berbasis Framework Laravel*”. Hasil dari penelitian ini adalah aplikasi *system meeting management* berbasis web menggunakan *framework laravel* dengan arsitektur MVC yang dapat memberikan kemudahan dan efisien dalam proses *booking* dan menggantikan cara *booking* konvensional menjadi secara *online*. Alasan penelitian ini sebagai referensi adalah terdapat kesamaan dalam penggunaan *framework* aplikasi yang di bangun. Selain itu aplikasi juga di tunjukan untuk melakukan *booking* suatu ruangan.

2. Penelitian dilakukan oleh Nia Ambarsari, Faishal Mufied Al Anshary, Wulung Wahyu Witono (2023) yang berjudul “Aplikasi Berbasis Website Untuk Booking Hotel Surya Andesa Klaten Menggunakan Framework Laravel Dengan Metode Waterfall”. Hasil dari penelitian ini adalah aplikasi berbasis website yang dapat melakukan booking per hari dan per jam, mengelola detail kamar dan mengelola user karyawan, melakukan pembayaran cashless dan membuat laporan. Alasan penelitian ini sebagai referensi adalah terdapat kesamaan dalam penggunaan framework aplikasi yang di bangun. Selain itu aplikasi juga di tunjukan untuk melakukan booking suatu kamar.

3. Penelitian dilakukan oleh Nofianti (2020) yang berjudul “Sistem Informasi Perpustakaan Berbasis Web dengan Framework Laravel di SMP Muhammadiyah 7 Semarang”. Hasil dari penelitian terkait adalah sebuah sistem informasi perpustakaan berbasis web yang digunakan oleh perpustakaan SMP Muhammadiyah 7 Semarang. Aplikasi tersebut memberikan manfaat dalam pengelolaan berbagai data untuk perpustakaan. Alasan penelitian ini sebagai referensi adalah terdapat kesamaan dalam penggunaan framework aplikasi yang di bangun. Selain itu aplikasi juga di tunjukan untuk melakukan booking suatu ruangan. Pada penelitian ini,

aplikasi yang di bangun bertujuan untuk sebagai sistem informasi perpustakaan. Sedangkan penelitian yang dilakukan peneliti adalah perancangan aplikasi manajemen ruangan.

4. Penelitian yang dilakukan oleh Fuad Fadlila Surenggana, Fitri Bimantoro dan Rival Biasrori (2021) yang berjudul “ Sistem Informasi Manajemen Ruangan Program Studi Teknik Informatika Menggunakan Framework Laravel”. Tujuan penelitian ini adalah membuat sistem informasi manajemen ruangan berbasis web yang dapat membantu staf prodi dalam memberikan pelayanan untuk mengatur manajemen ruangan sehingga lebih efektif dan efisien. Sistem informasi manajemen ruangan ini juga memberikan manfaat untuk dosen dan mahasiswa dalam proses peminjaman ruangan untuk menjalankan proses akademik maupun non-akademik. Alasan peneliti menjadikan penelitian ini sebagai referensi adalah terdapat kesamaan dalam penggunaan framework aplikasi yang di bangun.
5. Penelitian yang dilakukan oleh Mochammad Alif Pratama (2022) yang berjudul “Rancang Bangun Sistem Pemesanan Meeting Room Berbasis Web Menggunakan Framewok Laravel”. Pada penelitian ini bertujuan membuat aplikasi Booking Meeting Room di PT. Vivo Mobile Indonesia guna membantu pemesanan ruang meeting secara efektif dan efisien. Penelitian ini memiliki kesamaan dengan penelitian yang peneliti lakukan yaitu penggunaan framework laravel dan aplikasi yang bertujuan untuk booking meeting room.

Secara keseluruhan, penelitian terkait di atas memiliki kesamaan dengan penelitian yang peneliti lakukan adalah penggunaan laravel *framework* dalam aplikasi atau sistem yang dikembangkan. Sedangkan hal yang menjadi perbedaan adalah beberapa penelitian terkait di atas lebih berfokus pada peminjaman dan *booking* ruangan di lingkungan penelitian terkait. Sedangkan pada penelitian ini tidak hanya

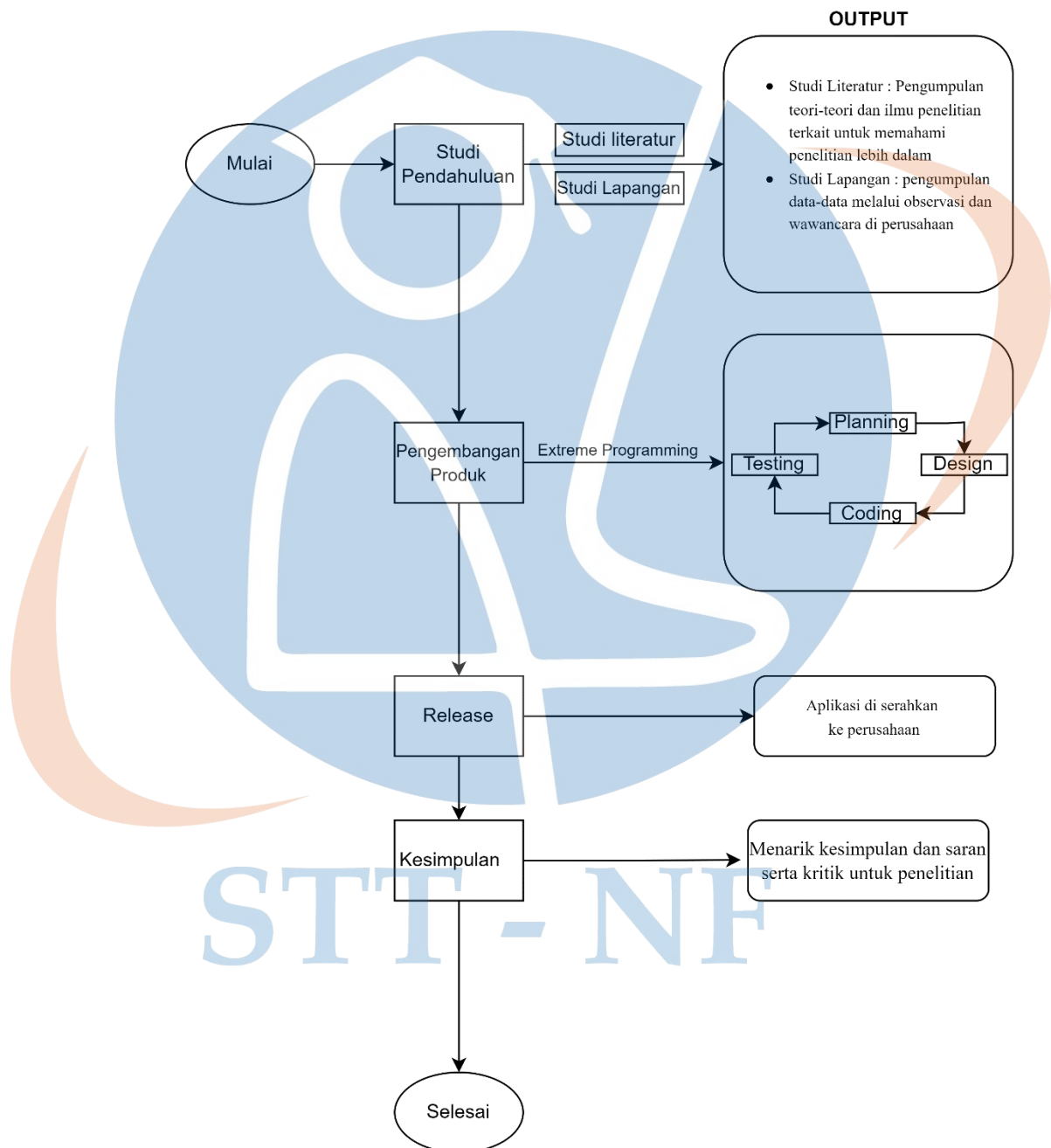
berfokus pada peminjaman akan tetapi memiliki jangkauan fokus yang lebih luas yaitu terkait manajemen ruangan yang lebih efektif dan efisien.



STT - NF

# BAB III METODOLOGI PENELITIAN

## 3.1 Tahapan Penelitian



Gambar 3. 1 Tahapan Penelitian

Berdasarkan gambar di atas, berikut adalah tahapan-tahapan dalam penelitian ini :

1. Studi Pendahuluan

Pada tahapan ini terdiri dari dua bagian yaitu studi literatur dan studi lapangan. Pada studi literatur mengumpulkan dan mempelajari teori-teori dari berbagai sumber seperti penelitian terkait, buku dan literatur lainnya. Sedangkan studi lapangan adalah pengumpulan data melalui observasi dan wawancara di PT. Tirta Asasta.

2. Pengembangan Produk/Aplikasi

Pengembangan produk atau aplikasi selanjutnya adalah memulai langkah-langkah pengembangan dengan menggunakan metode XP (*Extreme Programming*) yang menekankan pada pengembangan cepat dan adaptasi terhadap perubahan selama proses pengembangan. Tahapan ini terdiri dari empat langkah utama, yaitu:

a. *Planning* (Perencanaan)

Tahap ini melibatkan perencanaan kebutuhan aplikasi berdasarkan analisis masalah yang ada. Kegiatan yang dilakukan mencakup pembuatan *user stories*, penentuan kebutuhan sistem, dan perencanaan iterasi.

b. *Design* (Desain): Tahap ini bertujuan menerjemahkan hasil perencanaan ke dalam bentuk diagram UML, seperti diagram aktivitas dan diagram ERD, sebagai gambaran kebutuhan aplikasi. Selain itu, prototipe halaman aplikasi juga dibuat.

c. *Coding*

Proses penulisan kode aplikasi dimulai berdasarkan kebutuhan dan diagram yang telah dibuat pada tahap sebelumnya.

d. *Testing* (Pengujian)

Tahap ini melibatkan pengujian fitur-fitur aplikasi yang telah dikembangkan. Pengujian dilakukan pada setiap iterasi untuk memastikan iterasi tersebut berhasil dan dapat dilanjutkan ke tahap iterasi berikutnya.

### 3. *Release*

Tahapan selanjutnya adalah merilis aplikasi dengan mengirimkan keseluruhan kode aplikasi ke dalam web *hosting* yang dimiliki divisi IT. Tahapan ini dapat diartikan aplikasi sudah selesai di buat dan siap untuk digunakan.

### 4. Kesimpulan

Pada tahap ini diambil penarikan kesimpulan berdasarkan hasil dari pengujian dan implementasi aplikasi di perusahaan. Selain itu memberikan saran untuk penelitian yang akan datang berdasarkan kekurangan dari penelitian yang dilakukan.

## 3.2 Rancangan Penelitian

Untuk menjadi acuan dalam proses penelitian, dibuat rancangan penelitian sebagai berikut :



Gambar 3. 2 Rancangan Penelitian

### 3.2.1 Jenis Penelitian

Ditinjau berdasarkan tujuan penelitian, penelitian ini termasuk ke dalam penelitian pengembangan atau dalam bahasa inggris disebut *research & development*. Penelitian pengembangan merupakan salah satu dari penelitian terapan. Penelitian pengembangan bertujuan untuk mengembangkan suatu produk baru atau memperbaiki produk lama di tempat penelitian agar menyelesaikan yang terkait dengan perangkat atau sistem yang digunakan kurang tepat. Penelitian ini melibatkan proses *research* dan setelahnya proses *development*. Proses *research* meliputi pengumpulan informasi tentang permasalahan dan tentang kebutuhan pengguna. Sementara *development* dilakukan sebagai bentuk menyelesaikan permasalahan tersebut serta memenuhi kebutuhan pengguna melalui produk yang dikembangkan.



Dalam proses analisis data, metode yang digunakan adalah metode kualitatif yaitu metode yang mengumpulkan data melalui studi lapangan. Dalam metode ini, pendekatan yang digunakan adalah prosedur penelitian yang didapatkan berupa data-data deskriptif dari hasil wawancara dan observasi langsung di PT Tirta Asasta Depok.

### **3.2.2 Metode Analisis Data**

Berdasarkan metode penelitiannya, maka analisis data yang digunakan pada penelitian ini bersifat deskriptif kualitatif. Analisis deskriptif adalah suatu metode pengolahan data untuk mendapatkan gambaran terkait suatu informasi. Gambaran tersebut disajikan secara teratur, ringkas dan jelas mengenai suatu keadaan atau peristiwa di objek penelitian sehingga dapat ditarik pengertian atau makna tertentu. Pada proses analisis data kualitatif, data yang muncul berwujud kata-kata dan bukan rangkaian angka. Pada penelitian ini, data dikumpulkan dengan wawancara dan observasi langsung. Setelah data didapatkan maka akan dilakukan analisis untuk menghasilkan apa saja kebutuhan yang ada di aplikasi. Analisis data juga membantu peneliti dalam membuat *diagram activity* serta gambaran mengenai aplikasi. Selain itu, data yang dianalisis juga agar aplikasi berjalan sudah sesuai dengan standar operasional perusahaan dan infrastruktur perusahaan.

### **3.2.3 Metode Pengumpulan Data**

Dalam proses pengumpulan data, penelitian ini menggunakan metode sebagai berikut :

#### **1.1 Wawancara**

Melakukan kegiatan wawancara dengan calon pengguna aplikasi untuk mendapatkan data mengenai permasalahan yang ingin diselesaikan dan kebutuhan apa saja yang perlu disediakan oleh aplikasi. Wawancara dilakukan kepada dua pegawai PT. Tirta Asasta yang masing-masing mewakili pengguna ruangan dan pengelola ruangan. Tujuan dari wawancara untuk mengumpulkan data-data terkait kebutuhan aplikasi serta dokumen yang diperlukan

untuk pembangunan program aplikasi seperti dokumen Standar Operasi dan Prosedur (SOP) dan aturan lainnya yang berhubungan dengan kegiatan proses bisnis PT. Tirta Asasta agar selaras dengan Infrastruktur dan platform yang ada di perusahaan.

## 2.1 Observasi

Mengumpulkan data dengan observasi atau pengamatan secara langsung di lokasi objek penelitian untuk mendapatkan data-data secara langsung. Metode ini dilakukan dengan mendatangi lokasi PT. Tirta Asasta Depok.

## 3.1 Studi Pustaka

Metode studi pustaka adalah mengumpulkan data-data dari beberapa kajian literatur yang disadur dari buku, artikel ilmiah dan penelitian terkait sebelumnya.

### 3.2.4 Metode Pengujian

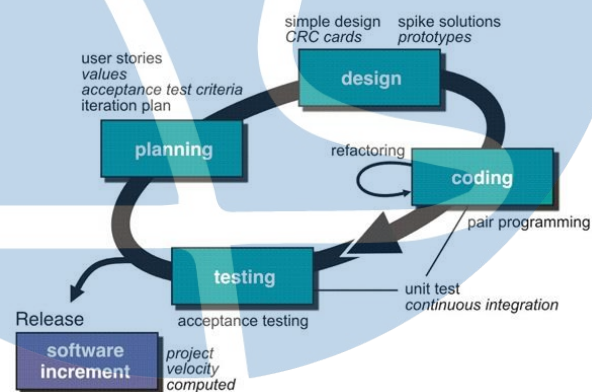
Metode pengujian pada penelitian ini menggunakan *black-box testing*. *Black-box testing* atau bisa disebut *behavioral testing* adalah pengujian yang menguji fungsionalitas dari aplikasi. Pengujian ini digunakan di tahap akhir pengembangan aplikasi untuk mengetahui apakah fungsionalitas aplikasi sudah berjalan dengan baik. Pengujian *black-box* fokus pada evaluasi fungsi aplikasi untuk memastikan bahwa setiap proses bekerja sesuai dengan ekspektasi. Metode ini menekankan pada pengujian berdasarkan kondisi input dan output tanpa perlu memahami kode atau struktur internal dari aplikasi. Pengujian *black-box* efektif untuk mendeteksi kesalahan atau kegagalan fungsi sehingga memungkinkan perbaikan yang diperlukan, menjadikan aplikasi lebih layak digunakan. [24].

*Black-box testing* yang digunakan pada penelitian ini adalah *Functional testing* dan *Non Functional testing*. *Functional testing* meliputi pengujian pada fitur-fitur pada aplikasi untuk memastikan fitur tersebut telah berjalan dengan baik. Sedangkan *Non Functional testing* merupakan pengujian yang berfokus terhadap hal-hal diluar fungsionalitas dari aplikasi. Jenis pengujian ini bertujuan

memastikan kualitas dan kinerja aplikasi yang dikembangkan, seperti pengujian aplikasi di berbagai perangkat yang berbeda.

### 3.2.5 Metode Implementasi dan Evaluasi

Metode implementasi yang digunakan dalam pengembangan adalah XP atau *extreme programming*. *Extreme programming* merupakan salah satu pendekatan dalam mengembangkan sebuah aplikasi secara cepat. Metode ini memudahkan proses pengembangan aplikasi menjadi lebih efektif, responsif, dan mudah diadaptasi. Pendekatan *Extreme Programming* (XP) cocok digunakan untuk mengembangkan aplikasi dengan kemampuan menyesuaikan diri secara cepat terhadap perubahan yang mungkin terjadi selama proses pengembangan aplikasi masih berlangsung. [25]. Berikut adalah gambaran mengenai tahapan atau alur kerja metode *Extreme Programming* (XP) :



Gambar 3. 3 alur kerja XP

( Sumber : Ilyas Kurnia Ramadhan, 2023 )

#### 1. *Planning*

Tahap pertama adalah melakukan *planning*. Pada tahap ini meliputi kegiatan perencanaan, identifikasi masalah dan mengumpulkan kebutuhan *user* dan kebutuhan aplikasi.

#### 2. *Design*

Tahap berikutnya adalah melakukan *design*. Tahap ini meliputi pembuatan prototype aplikasi, pemodelan sistem seperti pembuatan UML dan ERD.

### 3. *Coding*

Tahap selanjutnya adalah menerapkan *design* ke dalam aplikasi yang dirancang melalui proses *coding* [26]. Proses *coding* dilakukan menggunakan aplikasi editor bernama visual studio.

### 4. *Testing*

Tahapan terakhir yang dilakukan adalah melakukan pengujian fitur dan fungsionalitas dari aplikasi yang sedang dibangun. Tahapan ini memastikan fungsionalitas aplikasi sudah berjalan sesuai yang diharapkan.

#### **3.2.6 Lingkungan Pengembangan**

Lokasi penelitian dilakukan di PT. Tirta Asasta Depok yang beralamat Jl. Legong Raya No.1, Mekar Jaya, Kec. Sukmajaya, Kota Depok, Jawa Barat. PT. Tirta Asasta Depok merupakan perusahaan perseroan daerah atau perseroda yang bergerak di bidang pelayanan air bersih di wilayah kota Depok.

##### 1. Lingkungan perangkat keras :

- a. Laptop Lenovo Yoga *slim* 7
- b. Intel i5-11320H
- c. Layar 14''
- d. 16gb Ram
- e. 500gb *HDD Storage*

##### 2. Lingkungan Perangkat lunak :

- a. Visual studio code editor
- b. XAMPP 8.0
- c. Laravel 8.

STT - NF

## BAB IV IMPLEMENTASI DAN EVALUASI

### 4.1 Studi Pendahuluan

Pada tahapan awal pelaksanaan penelitian, peneliti melakukan studi lapangan dan studi pustaka. Studi lapangan yang dilakukan berupa wawancara dan observasi langsung yang memiliki tujuan untuk menganalisis masalah dan mengumpulkan data kebutuhan untuk aplikasi yang dapat menyelesaikan masalah tersebut. Sementara studi pustaka dilakukan guna menambah wawasan peneliti untuk sebagai referensi dan data terkait penelitian terkait yang dilakukan.

#### 4.1.1 Studi Lapangan

Studi lapangan yang dilakukan adalah mengumpulkan data Menggunakan metode wawancara dan observasi ke perusahaan.

a. Wawancara

Peneliti melakukan wawancara dengan dua responden dari perusahaan yang masing-masing mewakili sebagai calon *user* dan calon admin dari aplikasi. Kegiatan wawancara di lakukan pada hari Rabu tanggal 28 Mei 2024 di PT. Tirta Asasta Depok. Berikut adalah hasil wawancara yang dilakukan

*Tabel 4.1 Wawancara*

<b>Identitas Responden</b>	
Nama	: Muhammad Faris, S.Kom
NIP	: 20220710401
Jabatan	: Staf IT bagian pengembangan <i>software</i>
<b>Pertanyaan dan Jawaban Wawancara</b>	

**1. Bagaimana prosedur jika pegawai ingin meminjam ruangan yang berlaku saat ini?**

Saat ini, prosedur peminjaman ruangan dilakukan dengan bertemu atau menghubungi admin ruangan untuk menanyakan ketersediaan ruangan. Jika ruangan tersedia, admin akan mencatat peminjaman tersebut di catatan bagian ruangan. Dulu, memakai surat nota dinas, tetapi karena dianggap memakan waktu dan ribet jadi sekarang lebih langsung menghubungi sekretaris ruangan.

**2. Apakah ada kendala dalam prosedur peminjaman ruangan? Jika iya, apa saja?**

Banyak kendala yang sering terjadi, di antaranya:

1. Jadwal yang sudah dicatat sering kali digeser karena ada prioritas kegiatan lain.
2. Respons admin yang lambat atau kadang ketika ditemui langsung, admin sedang tidak ada di tempat.
3. Jadwal bentrok.
4. Pegawai merasa malas untuk bolak-balik dan prosesnya yang membingungkan.

**3. Seberapa sering kendala tersebut terjadi?**

Kendala tersebut terjadi cukup sering, menyebabkan tidak nyaman dalam proses peminjaman ruangan.

**4. Bagaimana pendapat Anda apabila sistem manajemen ruangan berganti menjadi berbasis *website*? Apakah akan ada manfaatnya?**

Menurut saya, sistem manajemen ruangan berbasis *website* akan sangat bermanfaat. Pertama, proses peminjaman akan lebih mudah tanpa harus bolak-balik ke ruangan admin, cukup melihat di sistem saja. Admin juga akan lebih terbantu

dengan adanya data riwayat peminjaman dan absensi yang dapat dilakukan secara digital.

**5. Apa saran atau rekomendasi yang diberikan untuk meningkatkan sistem peminjaman ruangan melalui web?**

Saya menyarankan agar *user* dapat melihat jadwal peminjaman ruangan secara *real-time* di aplikasi. Selain itu, aplikasi harus *responsive* di perangkat *mobile* karena kebanyakan pegawai mengakses aplikasi melalui perangkat tersebut. Juga, ada baiknya jika aplikasi menyediakan notifikasi *pop-up* atau notifikasi ke ponsel *user* dan admin. Selain itu ada fitur absen juga untuk peserta kegiatan, karena selama ini absen masih dilakukan dengan kertas.

**6. Bagaimana infrastruktur aplikasi yang ada saat ini di perusahaan? Apakah semua berbasis web atau android?**

Saat ini, aplikasi yang digunakan di perusahaan masih berbasis *website* semua. Aplikasi-aplikasi tersebut di kelola di server perusahaan agar lebih mudah dalam proses *maintance* aplikasi.

**Identitas Responden**

Nama : Fadiah Addini  
NIP : 20220720322  
Jabatan : Sekretaris Dirut

**Pertanyaan dan Jawaban Wawancara**

**1. Bagaimana prosedur peminjaman ruangan yang berlaku saat ini?**

Untuk saat ini, prosedur peminjaman ruangan dilakukan via *chat*. Pegawai yang ingin meminjam menghubungi sekretaris ruangan terkait untuk menanyakan perihal ketersediaan ruangan. Jika waktu sesuai, sekretaris akan menanyakan kegiatan apa yang akan diadakan dan berapa orang yang hadir untuk menyediakan konsumsi. Secara prosedur sebenarnya harus menggunakan surat nota dinas, tetapi terkadang surat tersebut harus melewati atasan setiap divisi terlebih dahulu sehingga dirasa

memakan waktu cukup lama, sehingga saat ini lebih banyak yang langsung menghubungi via *chat* pribadi.

**2. Apakah ada kendala dalam prosedur peminjaman ruangan dari pihak penanggung jawab ruangan? Jika iya, apa saja?**

Khusus di ruangan saya, sering terjadi beberapa kendala seperti:

1. Pembatalan sepihak tanpa ada konfirmasi ketika peminjaman batal.
2. Terjadi *double booking*, karena peminjam menghubungi sekretaris yang berbeda.
3. Terjadinya pergeseran peminjam karena terkait prioritas kegiatan yang dilakukan.
4. Perubahan jumlah peserta yang hadir, biasanya karena ada tamu dari luar. menyebabkan kebingungan dalam menyediakan konsumsi.

**3: Bagaimana pendapat Anda apabila sistem manajemen ruangan berganti menjadi berbasis *website*? Apakah akan ada manfaatnya?**

Pasti bermanfaat, karena prosedur yang sekarang memerlukan konfirmasi ulang dan pertanyaan tentang ketersediaan ruangan. Sementara itu, jika menggunakan aplikasi, peminjam bisa langsung melihat ketersediaan ruangan.

**4: Apa saran atau rekomendasi yang Anda berikan untuk meningkatkan sistem manajemen ruangan melalui web?**

Pertama, *user* harus dapat melihat jadwal peminjaman ruangan agar mereka tidak perlu bertanya lagi. Fitur peminjaman yang memudahkan admin untuk menerima ajukan, adanya riwayat peminjaman sebagai catatan rekap, dan fitur untuk mendata peserta acara juga diperlukan karena untuk kebutuhan pendataan ulang untuk konsumsi. Hal ini penting karena terkadang terdapat peserta kegiatan dari luar perusahaan yang tidak tercatat sehingga menyebabkan kesalahan komunikasi dalam



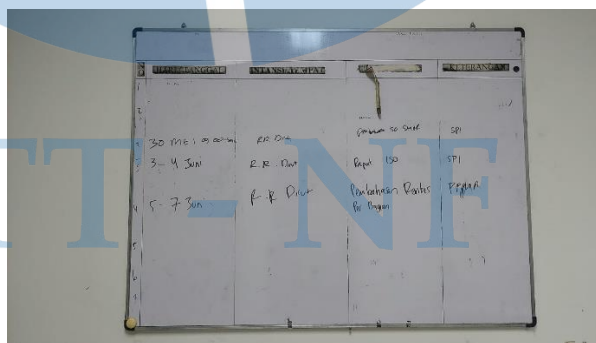
menyediakan ruangan Yang pasti, kendala-kendala yang sebelumnya saya sebutkan bisa diselesaikan dengan aplikasi ini.

**5: Apakah ada waktu-waktu tertentu di mana permintaan peminjaman ruangan sangat tinggi? Bagaimana cara mengatasi situasi tersebut?**

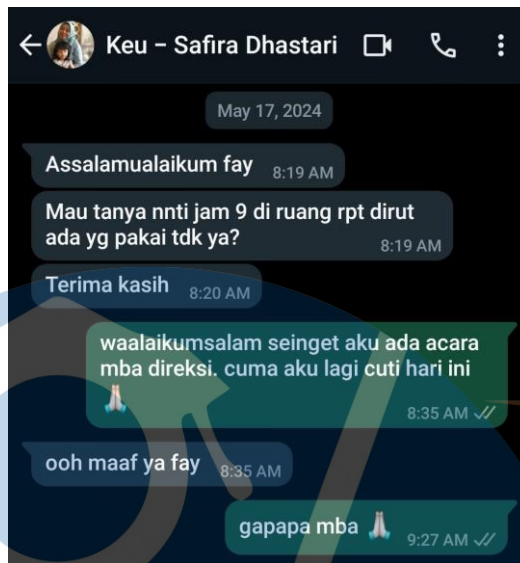
Biasanya, permintaan peminjaman ruangan sangat tinggi di pertengahan tahun dan akhir tahun karena waktu tersebut digunakan untuk membahas berbagai rapat tahunan. Untuk solusinya, biasanya kami antar sekretaris melakukan koordinasi lagi. Jadi semisal ruangan saya penuh, bisa di alihkan ke ruangan lain untuk digunakan.

b. Observasi

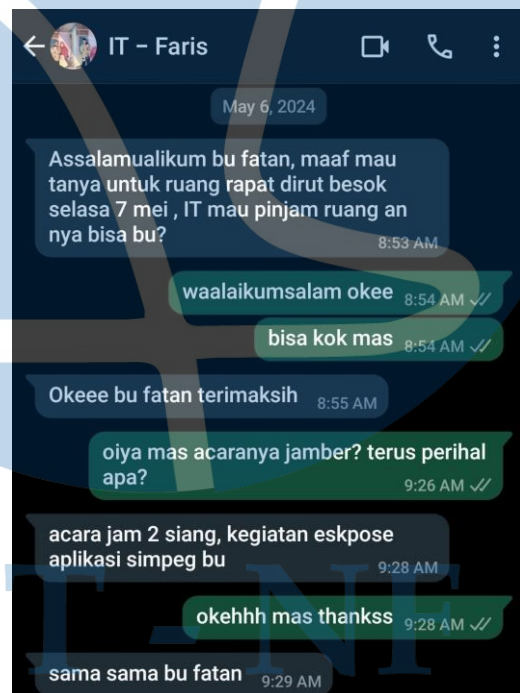
Observasi di lakukan pada hari Selasa tanggal 28 Mei 2024 dengan mendatangi lokasi perusahaan secara langsung. Tujuan observasi ini dilakukan agar peneliti dapat melihat langsung situasi manajemen ruangan yang ada di perusahaan. Observasi dilakukan dengan melihat ke bagian ruangan sekretaris serta melihat berbagai ruangan yang biasa digunakan untuk berbagai kegiatan. Selain itu penulis juga melihat *chat* yang di lakukan oleh pegawai yang ingin melakukan peminjaman ruangan. Berikut adalah beberapa gambar hasil observasi yang dilakukan oleh peneliti



Gambar 4. 1 papan tulis untuk mencatat jadwal peminjaman



Gambar 4. 2 chat pegawai menanyakan ketersediaan ruangan



Gambar 4. 3 chat pegawai menanyakan ketersediaan ruangan

Berdasarkan hasil observasi yang dilakukan peneliti berpendapat masih ada kekurangan dalam penulisan jadwal peminjaman di papan tulis sekretaris. Hal ini menjadi salah satu penyebab terjadinya kendala berupa kesalahan dalam mencatat jadwal peminjaman. Selain itu, respons sekretaris dapat dikatakan cukup cepat dalam menjawab percakapan mengenai

peminjaman ruangan sehingga pegawai yang bertanya bisa mengetahui ketersediaan tanpa waktu yang lama. Akan tetapi perlu beberapa percakapan untuk mengetahui detail peminjaman seperti untuk kegiatan apa dan pukul berapa peminjaman di lakukan.

#### **4.1.2 Studi Pustaka**

Studi pustaka dilakukan dengan mencari beberapa penelitian yang memiliki keterkaitan dengan peneliti yang dilakukan. Tujuan studi pustaka adalah untuk menambah referensi dan beberapa data untuk peneliti kaji. Hasil studi pustaka mendapatkan 5 penelitian terkait yang sudah dijelaskan pada bab sebelumnya.

#### **4.2 Pengembangan Sistem:**

Setelah melakukan studi lapangan, tahapan penelitian yang dilakukan selanjutnya adalah melakukan pengembangan sistem aplikasi. Dalam tahapan ini, peneliti menggunakan metode *extreme programming*. *Extreme programming* merupakan salah satu pendekatan dalam mengembangkan sebuah aplikasi secara cepat. Pendekatan *Extreme Programming* (XP) cocok digunakan untuk mengembangkan aplikasi dengan kemampuan menyesuaikan diri secara cepat terhadap perubahan yang mungkin terjadi selama proses pengembangan aplikasi masih berlangsung. Dalam metode ini, terdapat empat tahapan yang di lalui, yaitu *planing, design, coding dan testing*. Tahapan ini bersifat secara siklus yang disebut sebagai iterasi. Ketika iterasi selesai dan tidak ada perubahan atau penambahan pada aplikasi mencapai di tahapan akhir yaitu *release*.

#### 4.2.1 Planning

Pada tahap ini meliputi kegiatan identifikasi masalah, mengumpulkan kebutuhan *user* dan membuat rencana iterasi yang akan dilalui. Identifikasi masalah dan kebutuhan user di kumpulkan melalui metode wawancara dan observasi yang sudah dilakukan pada tahap sebelumnya. Berdasarkan wawancara tersebut, Prosedur peminjaman dianggap kurang efektif karena pegawai harus menghubungi bagian sekretaris ruang terkait untuk mengetahui jadwal peminjaman.

Masalah utama manajemen ruangan saat ini adalah ketidaktahuan pegawai dalam jadwal peminjaman di suatu ruangan yang menyebabkan harus menghubungi bagian admin terlebih dahulu. Sementara pada bagian sekretaris memiliki kendala terkait pencatatan jadwal peminjaman suatu ruangan yang masih kurang efektif dan menyebabkan kendala berupa *double booking*. Selain itu, pencatatan jumlah peserta menggunakan kertas masih tidak efektif untuk kebutuhan penyediaan konsumsi. Setelah mengidentifikasi masalah tersebut, maka dibuat *user stories* sebagai bentuk perencanaan kebutuhan apa saja yang dibutuhkan aplikasi.

tabel 4.2 User stories

No.	User	Role	Kebutuhan sistem
1	Pegawai	User	<ol style="list-style-type: none"><li>1. Pegawai dapat melihat jadwal pinjam ruangan</li><li>2. Pegawai dapat mengajukan peminjaman ruangan</li><li>3. Pegawai dapat melakukan absen di kegiatan yang di ikuti</li></ol>
2	Sekretaris Ruang	Admin Ruangan	<ol style="list-style-type: none"><li>1. Sekretaris Ruang dapat menerima atau menolak ajuan peminjaman ruangan</li><li>2. Hanya Sekretaris ruangan terkait yang bisa menerima atau menolak ajuan</li><li>3. Sekretaris dapat melihat rekap absen kegiatan yang berlangsung untuk menyediakan konsumsi</li></ol>
3	Pegawai Ruma	Super Admin	<ol style="list-style-type: none"><li>1. Super admin dapat melakukan konfirmasi di semua pengajuan ruangan</li></ol>

			<ol style="list-style-type: none"> <li>2. Super admin dapat melakukan registrasi pegawai baru</li> <li>3. Super admin dapat melakukan modifikasi terhadap data master di aplikasi</li> <li>4. Super admin dapat melihat riwayat peminjaman semua ruangan</li> </ol>
4	Tamu	Tamu	Tamu dapat melakukan absensi di kegiatan yang tamu ikuti.

Berdasarkan *user stories* yang sudah dibuat, terdapat 4 *role* yang akan menggunakan aplikasi yaitu pegawai sebagai *user*, sekretaris ruangan sebagai admin, pegawai rumba sebagai super admin dan tamu sebagai tamu. Dari data tersebut peneliti dapat membuat analisis kebutuhan sistem dengan membagi berdasarkan empat *role* tersebut.

a. Kebutuhan *User*

tabel 4.3 analisis kebutuhan user

No	Halaman <i>User</i>	Fungsi yang disediakan / keterangan
1	Halaman Muka	<ol style="list-style-type: none"> <li>1. Menampilkan jadwal peminjaman semua ruangan menggunakan tampilan kalender</li> <li>2. Menampilkan detail kegiatan peminjaman ruangan</li> <li>3. Kemudahan melihat detail ruangan yang digunakan pada tanggal tertentu</li> <li>4. Menampilkan jadwal peminjaman di hari aplikasi di akses</li> </ol>
2	Halaman Pengajuan Peminjaman	<ol style="list-style-type: none"> <li>1. Dapat memilih ruangan yang ingin di pinjam</li> <li>2. Melihat detail ruangan yang dipilih</li> <li>3. Mengisi <i>form</i> pengajuan</li> <li>4. Mengajukan pengajuan peminjaman</li> <li>5. Melihat jadwal kegiatan hari ini</li> <li>6. Melihat jadwal kegiatan lusa</li> </ol>

		7. Melihat detail pengajuan yang sudah di ajukan
3	Halaman Daftar Hadir	<ol style="list-style-type: none"> <li>1. Melihat daftar kegiatan yang di ikuti</li> <li>2. Memilih kegiatan yang di ikuti untuk melakukan absen</li> </ol>
4	Halaman Absen	Menampilkan <i>form</i> absen
5	<i>Sidebar</i>	<ol style="list-style-type: none"> <li>1. Menampilkan menu <i>button</i> halaman muka</li> <li>2. Menampilkan menu <i>button</i> pengajuan</li> <li>3. Menampilkan menu <i>button</i> daftar kegiatan</li> <li>4. Menampilkan menu <i>button</i> logout</li> </ol>

Pada kebutuhan *user*, menu yang terdapat di *sidebar* antara lain adalah menu halaman muka, halaman pengajuan untuk melakukan pengajuan peminjaman ruangan, halaman daftar kegiatan yang berisi kegiatan yang diikuti oleh user, dan menu logout untuk keluar dari aplikasi. Menu halaman muka memberikan informasi terkait jadwal peminjaman dalam bentuk kalender. Menu halaman pengajuan terdapat *form* pengajuan memungkinkan pengguna untuk mengajukan peminjaman ruangan. Halaman daftar kegiatan membantu *user* untuk memilih kegiatan yang ingin di lakukan absensi. Menu *logout* memberikan opsi untuk keluar dari aplikasi dengan aman setelah selesai digunakan.

STT - NF

b. Kebutuhan admin

tabel 4. 4 kebutuhan admin

No	Halaman Admin	Fungsi yang disediakan / keterangan
1	Halaman Muka	<ol style="list-style-type: none"> <li>1. Menampilkan jadwal peminjaman semua ruangan dalam bentuk kalender</li> <li>2. Menampilkan detail kegiatan peminjaman ruangan</li> <li>3. Kemudahan melihat detail ruangan yang digunakan pada tanggal tertentu</li> <li>4. Menampilkan jadwal peminjaman di hari aplikasi di akses</li> </ol>
2	Halaman Pengajuan Peminjaman	<ol style="list-style-type: none"> <li>1. Dapat memilih ruangan yang ingin di pinjam</li> <li>2. Melihat detail ruangan yang dipilih</li> <li>3. Mengisi <i>form</i> pengajuan</li> <li>4. Mengajukan pengajuan peminjaman</li> <li>5. Menampilkan jadwal kegiatan hari ini</li> <li>6. Menampilkan jadwal kegiatan lusa</li> <li>7. Melihat detail pengajuan yang sudah di ajukan</li> </ol>
3	Halaman Daftar Hadir	<ol style="list-style-type: none"> <li>1. Menampilkan daftar hadir peserta sesuai dengan ruangan sekretaris. Jika sekretaris bertugas di ruangan Direktur Utama, maka hanya akan muncul daftar hadir untuk ruangan tersebut.</li> <li>2. Menampilkan daftar peserta yang sudah melakukan absensi</li> </ol>
4	Halaman Agenda	<ol style="list-style-type: none"> <li>1. Menampilkan daftar pengajuan ruangan yang terkait dengan sekretaris ruangan terkait saja</li> <li>2. Menampilkan daftar pengajuan yang diterima</li> <li>3. Menerima notifikasi ketika ada pengajuan yang masuk</li> </ol>

5	Halaman Riwayat	Menampilkan daftar riwayat kegiatan peminjaman yang sudah berlalu
6	<i>Sidebar</i>	<ol style="list-style-type: none"> <li>1. Menampilkan menu <i>button</i> halaman muka</li> <li>2. Menampilkan menu <i>button</i> Pengajuan</li> <li>3. Menampilkan menu <i>button</i> Daftar hadir</li> <li>4. Menampilkan menu <i>button</i> Riwayat</li> <li>5. Menampilkan menu <i>button</i> logout</li> </ol>

Pada kebutuhan admin, terdapat perbedaan pilihan menu sidebar dengan user. Admin mendapatkan akses untuk memilih menu halaman riwayat yang berisi daftar pengajuan yang masuk ke dalam sistem. Selain itu, pada halaman agenda, hanya menampilkan daftar pengajuan yang terkait dengan admin sekretaris ruangan tertentu. Sebagai contoh, jika admin sekretaris ruang direktur utama mengakses halaman agenda, maka hanya daftar pengajuan yang mengajukan ruangan direktur utama saja yang tampil. Hal ini memastikan bahwa setiap admin dapat fokus pada pengelolaan dan pemantauan pengajuan yang relevan dengan ruang yang mereka tangani, meningkatkan efisiensi dan akurasi dalam pengelolaan ruangan.

#### c. Kebutuhan Super Admin

*tabel 4. 5 kebutuhan super admin*

No	Halaman Admin	Fungsi yang disediakan / keterangan
1	Halaman Muka	<ol style="list-style-type: none"> <li>1. Menampilkan jadwal peminjaman semua ruangan dalam bentuk kalender</li> <li>2. Menampilkan detail kegiatan peminjaman ruangan</li> <li>3. Kemudahan melihat detail ruangan yang digunakan pada tanggal tertentu</li> <li>4. Menampilkan jadwal peminjaman di hari aplikasi di akses</li> </ol>
2	Halaman Pengajuan Peminjaman	<ol style="list-style-type: none"> <li>1. Dapat memilih ruangan yang ingin di pinjam</li> <li>2. Melihat detail ruangan yang dipilih</li> <li>3. Mengisi <i>form</i> pengajuan</li> </ol>



		<ol style="list-style-type: none"> <li>4. Mengajukan pengajuan peminjaman</li> <li>5. Menampilkan jadwal kegiatan hari ini</li> <li>6. Menampilkan jadwal kegiatan lusa</li> <li>7. Melihat detail pengajuan yang sudah di ajukan</li> <li>8. Menerima notifikasi apabila jadwal yang dipilih sudah ada</li> </ol>
3	Halaman Daftar Hadir	<ol style="list-style-type: none"> <li>1. Menampilkan Seluruh daftar hadir kegiatan</li> <li>2. Menampilkan daftar peserta yang sudah melakukan absensi</li> </ol>
4	Halaman Agenda	<ol style="list-style-type: none"> <li>1. Menampilkan seluruh daftar pengajuan ruangan</li> <li>2. Menampilkan daftar pengajuan yang sudah diterima Terdapat notifikasi <i>pop-up</i> ketika ada pengajuan yang masuk</li> </ol>
5	Halaman Riwayat	Menampilkan daftar riwayat kegiatan peminjaman yang sudah berlalu
6	Menu <i>Sidebar</i>	<ol style="list-style-type: none"> <li>1. Menampilkan menu <i>button</i> halaman muka</li> <li>2. Menampilkan menu <i>button</i> data master yang berisikan : <ul style="list-style-type: none"> <li>- Halaman Ruangan</li> <li>- Halaman Fasilitas</li> <li>- Halaman Divisi</li> </ul> </li> <li>3. Menampilkan menu <i>button</i> Pengajuan</li> <li>4. Menampilkan menu <i>button</i> Daftar hadir</li> <li>5. Menampilkan menu <i>button</i> Agenda</li> <li>6. Menampilkan menu <i>button</i> Riwayat</li> <li>7. Menampilkan menu <i>button</i> Register</li> <li>8. Menampilkan menu <i>button</i> <i>logout</i></li> </ol>

Berdasarkan tabel di atas, perbedaan kebutuhan super admin dan admin terletak pada menu akses *sidebar*, yaitu terdapat menu data master yang berisikan halaman ruangan, halaman fasilitas, dan halaman divisi. Halaman data master digunakan untuk mengelola data aplikasi seperti data ruangan, fasilitas, dan divisi. Selain itu, pada halaman agenda, super admin akan dapat menampilkan seluruh daftar pengajuan tanpa terkecuali. Singkatnya, perbedaan dengan admin adalah super admin dapat meninjau seluruh pengajuan peminjaman, sehingga memiliki kontrol penuh terkait pengajuan dalam aplikasi.

d. Kebutuhan tamu

No	Halaman Tamu	Fungsi yang disediakan / keterangan
1	Halaman Tamu	Menampilkan daftar kegiatan yang dapat dipilih untuk melakukan absen
2	Halaman Absen Tamu	Menampilkan <i>form</i> absen untuk tamu

Berdasarkan wawancara, beberapa kegiatan rapat mengundang pihak dari luar atau disebut sebagai tamu. Setiap tamu yang hadir dapat mengisi absensi sebagai pendataan untuk kebutuhan konsumsi. Oleh karena itu, berdasarkan tabel di atas, kebutuhan utama tamu adalah menyediakan halaman daftar kegiatan untuk dipilih dan melakukan absensi di aplikasi. Hal ini memastikan bahwa setiap tamu yang hadir tercatat dengan baik, dan kebutuhan logistik seperti konsumsi dapat disiapkan dengan lebih akurat.

e. Kebutuhan non fungsional

No	Non Fungsional	Fungsi yang disediakan / keterangan
1	<i>Responsive</i>	Aplikasi dapat menyesuaikan sesuai perangkat yang digunakan
2	<i>Pagination</i>	Menambahkan <i>pagination</i> di tabel aplikasi agar membuat tata letak lebih ramah dan meningkatkan navigasi
3	<i>Dark mode</i>	Mengubah tampilan tema aplikasi menjadi lebih gelap

Tabel di atas memberikan gambaran terkait apa saja kebutuhan non-fungsional untuk aplikasi. *Responsivitas* merupakan kebutuhan yang penting agar aplikasi dapat diakses menggunakan berbagai perangkat, mulai dari desktop hingga perangkat *mobile*, sehingga memastikan pengalaman pengguna yang konsisten. *Pagination* diperlukan untuk menampilkan data dalam jumlah besar secara efisien, memudahkan pengguna untuk melihat dan menavigasi data. Selain itu, fitur *dark mode* juga menjadi salah satu kebutuhan yang memberikan opsi tampilan yang lebih gelap untuk pengguna yang memiliki preferensi tersebut.

Setiap kebutuhan dirancang dan dikembangkan secara berulang (iterasi) agar kebutuhan yang telah diidentifikasi dapat terpenuhi semuanya. Tahapan setelah pembuatan *user stories* adalah melakukan perencanaan pembagian iterasi yang akan diimplementasikan ke dalam pengembangan. Pembagian Iterasi ini memiliki tujuan untuk memecah fokus menjadi lebih kecil agar pengembangan dapat mengurangi kesalahan dari sistem maupun koordinasi dengan divisi IT perusahaan. Setiap tahapan iterasi akan dilakukan pengujian atau *testing* untuk memastikan tidak ada kesalahan sehingga dapat berpindah ke tahapan selanjutnya. Berdasarkan *user stories* yang sudah dibuat, maka iterasi dibagi menjadi 5 bagian :

**a. Iterasi *Databases***

Pada iterasi pertama, fokus utama adalah membuat *database* yang akan digunakan untuk penyimpanan data selama proses pengembangan dan peluncuran aplikasi. Langkah-langkah yang akan dilakukan mencakup pembuatan diagram ERD sebagai acuan struktur *database*. *Database* yang digunakan adalah MySQL, yang diakses melalui phpMyAdmin yang disediakan oleh XAMPP.

**b. Iterasi Integrasi Template dan Data Master**

Iterasi selanjutnya berfokus pada dua tugas utama. Pertama, integrasi template *Bootstrap*. dilakukan untuk memastikan bahwa template yang dipilih berhasil diintegrasikan ke dalam Project. Hal ini mencakup penyesuaian seperti *sidebar*, *dark theme*, *pagination* dan desain responsif agar sesuai dengan kebutuhan aplikasi. Kedua, pekerjaan pada data master mencakup pembuatan fungsi CRUD (*Create, Read, Update,*

*Delete*) untuk halaman-halaman data utama seperti ruangan, pegawai, divisi, dan fasilitas. Iterasi ini merupakan yang harus dikerjakan sebelum iterasi yang lain karena data yang dihasilkan akan digunakan oleh fitur lainnya seperti pengajuan dan absen. Dengan demikian, iterasi ini menetapkan dasar yang kuat untuk pengembangan selanjutnya, dengan memiliki tampilan yang baik dan fungsionalitas dasar CRUD yang sudah terimplementasi.

**c. Iterasi Pengembangan Fitur Pengajuan**

Iterasi ketiga, berfokus pada penambahan Modul Pengajuan, meliputi penambahan halaman yang terkait dengan pengajuan dan kegiatan ruangan. Pertama, halaman pengajuan rapat yang memungkinkan pengguna untuk mengajukan permintaan rapat baru dengan mengisi formulir yang disediakan. Halaman agenda rapat akan digunakan oleh admin dan menampilkan daftar peminjaman yang sudah di ajukan oleh *user*. Halaman riwayat akan mencatat riwayat peminjaman yang telah berlangsung. Halaman absen akan memungkinkan peserta kegiatan untuk melakukan absensi melalui aplikasi. Terakhir, halaman daftar hadir akan menampilkan daftar kegiatan beserta daftar hadir peserta yang hadir dalam kegiatan tertentu. Iterasi ini akan menggunakan data master yang sebelumnya sudah dibuat.

**d. Iterasi Modul Halaman Utama**

Iterasi selanjutnya, akan mengerjakan halaman muka aplikasi. Fitur utama dalam iterasi ini adalah menampilkan jadwal peminjaman seluruh ruangan dalam tampilan kalender. Pengguna dapat dengan mudah melihat ketersediaan ruangan dan jadwal peminjaman. Setiap *event* di kalender dapat pilih untuk memunculkan modal yang menampilkan detail dari kegiatan, sehingga pengguna dapat melihat informasi secara lebih rinci tentang acara yang dijadwalkan. Selain itu, akan ada modal detail ruangan yang akan memberikan informasi lengkap tentang ruangan tertentu, termasuk fasilitas, kapasitas, dan foto. Oleh karena itu

iterasi ini dikerjakan ditahap ke empat agar data pengajuan dapat ditampilkan di halaman utama.

e. Iterasi *Role*

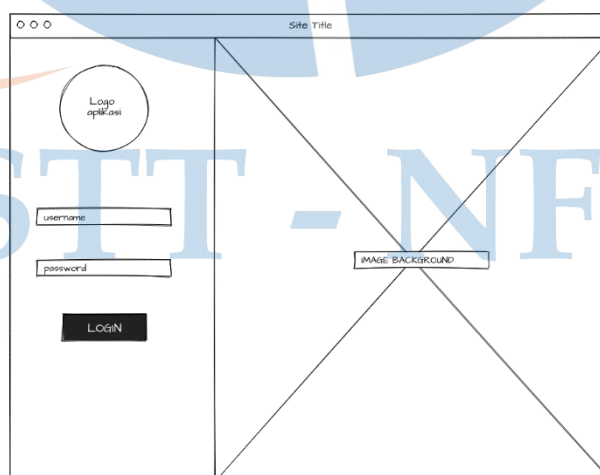
Iterasi terakhir, yang melibatkan penambahan *role*, yang mengatur hal manajemen pengguna dan hak akses. *Role* akan terdiri dari admin, *user*, tamu, dan super admin. Tujuan utama dari iterasi ini adalah memastikan bahwa setiap *role* memiliki akses yang sesuai dengan hak akses yang diberikan kepada mereka. Hak akses tersebut akan dibatasi dengan memunculkan *button* menu *sidebar* sesuai dengan *role* yang sedang masuk. Iterasi ini juga akan memastikan bahwa fungsi *login* dan registrasi berfungsi dengan baik.

#### 4.2.2 Design

Pada tahap ini, dilakukan *design* terkait aplikasi yang berdasarkan *planing* sudah dibuat di tahapan sebelumnya. Design tersebut meliputi *prototype*, diagram ERD dan diagram *activity* aplikasi. Tujuan *design* ini adalah memberikan gambaran terkait aplikasi yang akan di bangun sehingga tetap sesuai dengan kebutuhan yang sudah direncanakan.

a. *Prototipe*

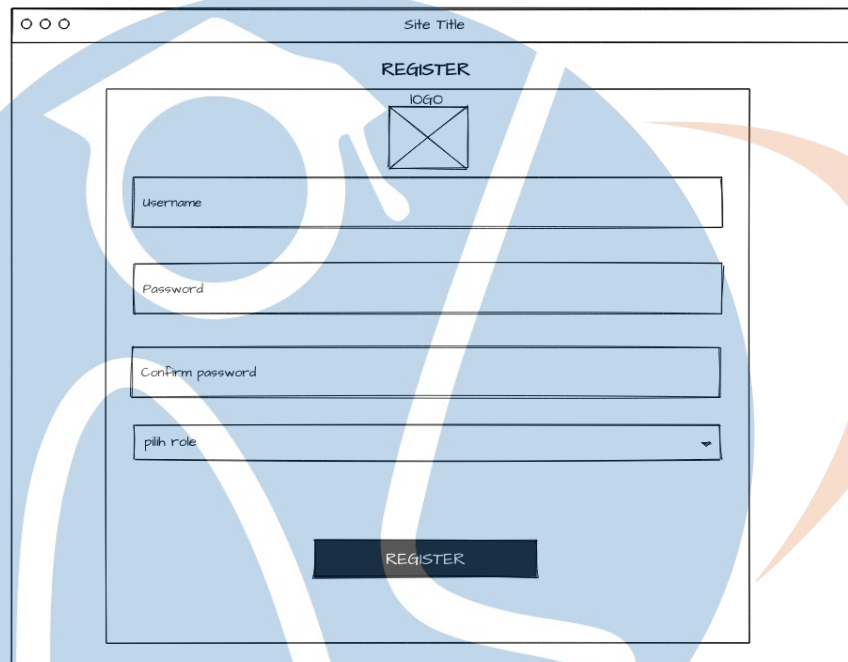
1. Halaman *Login*



Gambar 4. 4 halaman login

Halaman *login* terdapat dua *form* yaitu *username* dan *password* yang harus di isi sesuai data yang dimiliki. Terdapat satu *button login* untuk memerintahkan sistem memproses data yang di *input*.

## 2. Halaman Register

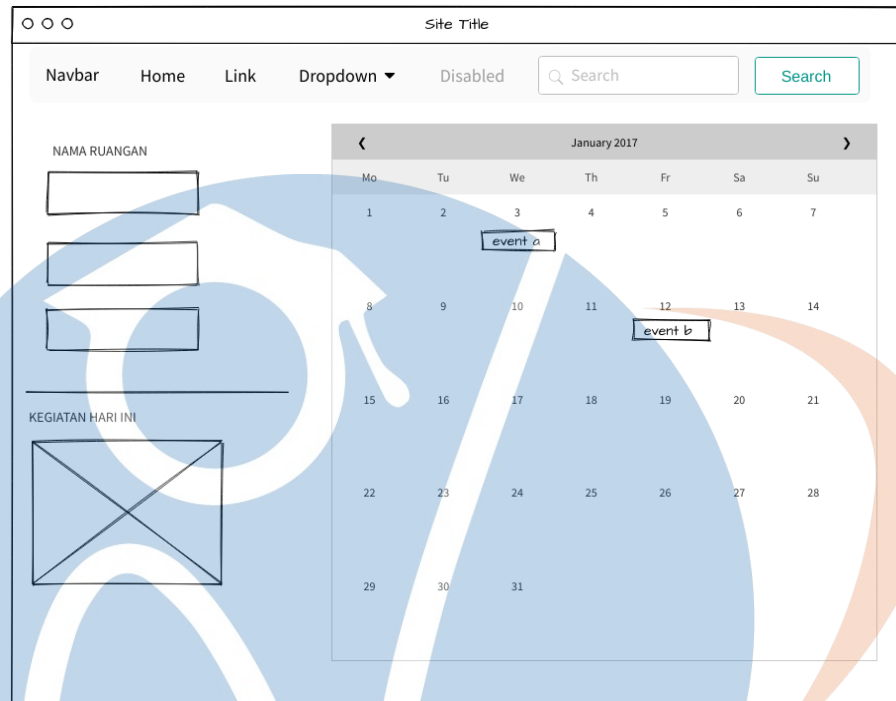
The image shows a web browser window with a registration form. The browser's title bar says "Site Title". The page title is "REGISTER". There is a "LOGO" placeholder with an 'X' inside a box. Below the logo are four input fields: "Username", "Password", "Confirm password", and a dropdown menu labeled "pilih role". At the bottom of the form is a dark "REGISTER" button.

Gambar 4. 5 halaman register

Pada halaman register terdapat tiga kolom *form* yaitu *username*, *password* dan konfirmasi *password* untuk registrasi akun baru. *Select box* pilih *role* digunakan untuk memilih *role* yang diberikan kepada akun yang dibuat. Terdapat satu *button* register untuk memproses pembuatan akun baru.

STT - NF

### 3. Halaman Muka



Gambar 4. 6 halaman muka

Halaman aplikasi menampilkan jadwal peminjaman ruangan dalam tampilan kalender. Pada sisi kiri terdapat daftar nama ruangan. Masing-masing nama ruangan dapat dipilih untuk menampilkan detail dari ruangan tersebut. Di bawah bagian daftar ruangan, terdapat data kegiatan hari ini yang menampilkan apa saja kegiatan yang terdapat di hari aplikasi di akses

STT - NF

#### 4. Halaman Data Master

AMARR

Navbar Home Link Dropdown Disabled Search Search

Tambah Ruangan

NO	Nama Ruangan	Kapasitas	Fasilitas	PIC	Foto Ruangan	Action
1	Ruang Dirut	14	LCD SCREEN, AC	M. Faris		<a href="#">Edit</a> <a href="#">Hapus</a>
2	Aula	50	AC, Papan Tulis	M. Rizal		<a href="#">Edit</a> <a href="#">Hapus</a>

« 1 2 3 4 5 »

Gambar 4. 7 halaman data master

Halaman data master menampilkan tabel dari data halaman tersebut. Contoh pada gambar adalah halaman ruangan, yang memberikan detail dari ruangan tersebut. Terdapat tiga *button* yaitu tambah, untuk menambahkan data baru, edit untuk mengubah data dari ruangan dan hapus untuk menghapus data ruangan.

STT - NF



## 5. Halaman Pengajuan

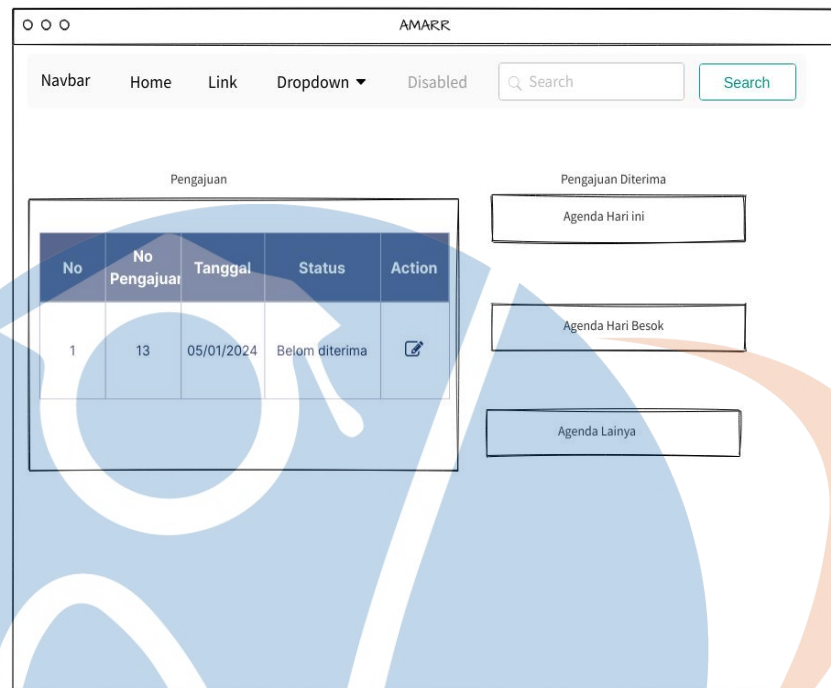
The screenshot displays the AMARR application interface. At the top, there is a navigation bar with 'Navbar', 'Home', 'Link', 'Dropdown', and 'Disabled' options, along with a search bar and a 'Search' button. The main content area is divided into two sections. On the left, titled 'Form pilih ruangan', there is a 'Select' dropdown menu and a button with an 'X' icon. Below this are three buttons labeled 'Jadwal Kegiatan hari ini', 'Jadwal Kegiatan lusa', and 'Jadwal Kegiatan lainnya'. On the right, titled 'FORM PEMINJAMAN RUANG', there are input fields for 'Nama', 'Divisi', 'Waktu', and 'Nama kegiatan'. To the right of these fields is a 'pilih peserta' box. A 'Tambah' button is located below the input fields. At the bottom of the right section, there is a 'Data Pengajuan' box.

Gambar 4. 8 halaman pengajuan

Halaman pengajuan berisikan *form* untuk melakukan pengajuan peminjaman ruangan. Terdapat dua *form* yang pertama *form* memilih ruangan, pengguna dapat memilih ruangan melalui *select box* yang ada, lalu sistem akan menampilkan data ruangan yang di pilih. *Form* kedua merupakan *form* data pengajuan seperti nama pemohon, divisi, waktu, nama kegiatan dan *box* untuk memilih nama peserta. Pada *form* kedua juga terdapat *button* tambah untuk mengirimkan data pengajuan. Setelah *button* tambah dipilih, maka data pengajuan akan ditampilkan di bagian data pengajuan.

STT - NF

## 6. Halaman Agenda



Gambar 4. 9 halaman agenda

Halaman Agenda akan digunakan oleh admin untuk memproses pengajuan yang masuk ke sistem. Halaman ini menampilkan tabel berisi pengajuan yang sudah masuk, dan admin dapat memilih untuk menerima atau menolak pengajuan tersebut melalui tab *action*. Selain itu, terdapat tiga *box* yang menampilkan data pengajuan yang sudah diterima, dibagi berdasarkan waktu peminjamannya:

1. Agenda Hari Ini: Menampilkan peminjaman yang terjadi pada hari aplikasi diakses.
2. Agenda Hari Besok: Menampilkan peminjaman yang terjadi pada hari setelah aplikasi diakses.
3. Agenda Lainnya: Menampilkan data pengajuan yang akan datang dalam 3 hari atau lebih ke depan.

## 7. Halaman Absen

The screenshot shows a web browser window titled 'AMARR'. The navigation bar contains 'Navbar', 'Home', 'Link', 'Dropdown', and 'Disabled' menus, along with a search bar. The main content area is titled 'Absen Peserta Rapat' and contains the following form elements: a text input field for 'Nama Kegiatan', a text input field for 'Waktu Kegiatan', a signature field labeled 'Tanda tangan', and a dark blue button labeled 'ABSEN'.

Gambar 4. 10 halaman absen

Halaman absen terdapat dua kotak yang berisikan nama kegiatan dan waktu kegiatan. Data tersebut akan terisi secara otomatis oleh sistem sesuai dengan kegiatan yang dipilih oleh *user*. *User* hanya perlu mengisi kotak tandan tangan untuk melakukan absen. Ketika tombol absen di pilih maka absensi akan dicatat oleh sistem.

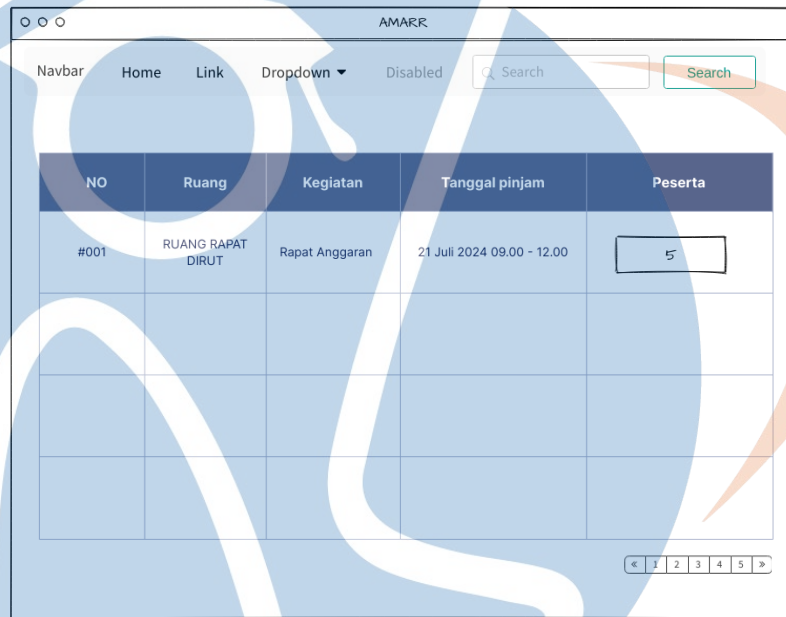
## 8. Halaman Absen Tamu

The screenshot shows a web browser window titled 'AMARR'. The main content area is titled 'Absen Peserta Rapat' and contains the following form elements: a text input field for 'Nama', a text input field for 'Asal Instansi', a text input field for 'Nama Kegiatan', a text input field for 'Waktu Kegiatan', a signature field labeled 'Tanda tangan', and a dark blue button labeled 'ABSEN'.

Gambar 4. 11 halaman absen tamu

Halaman absen tamu digunakan oleh tamu kegiatan untuk mengisi absensi. Perbedaan dengan halaman tamu sebelumnya adalah terdapat kolom nama dan asal instansi yang perlu di isi oleh tamu kegiatan.

## 9. Halaman Daftar Hadir



NO	Ruang	Kegiatan	Tanggal pinjam	Peserta
#001	RUANG RAPAT DIRUT	Rapat Anggaran	21 Juli 2024 09.00 - 12.00	5

Gambar 4. 12 halaman daftar hadir

Halaman daftar hadir menampilkan tabel daftar kegiatan yang sudah disetujui. Pada tabel tersebut terdapat kotak menu jumlah peserta di kolom jumlah peserta yang akan menampilkan modal detail kehadiran peserta.

STT - NF

## 10. Halaman Riwayat

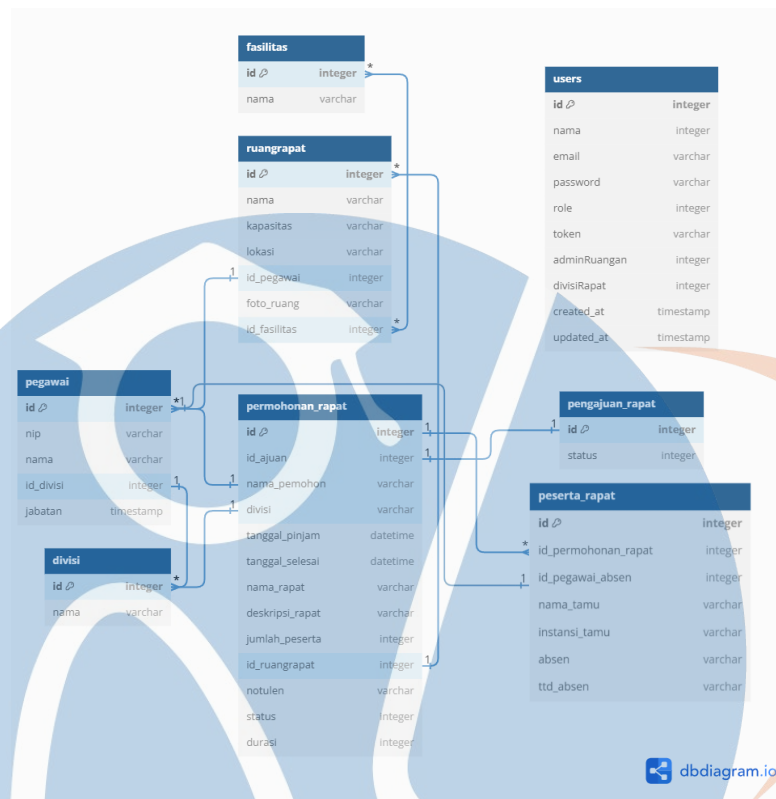
NO	Ruangan	Kegiatan	Tanggal Pinjam	Peserta	Status	Notulen
1	Ruang Dirut	Rapat Anggaran	3 Juni 2024 15.00 - 16.30	10	Ditolak	
2	Aula	Rapat Evaluasi	3 Juni 2024 15.00 - 16.30	10	Selesai	

Gambar 4. 13 halaman riwayat

Halaman riwayat terdapat tabel yang menampilkan seluruh daftar pengajuan yang tercatat di sistem, baik yang diterima dan yang ditolak dan yang sudah selesai. Hal ini digunakan sebagai data pencatatan peminjaman ruangan. Pada kolom peserta dapat melihat rekap dari absen peserta. Di kolom notulen, admin dapat mengunggah file notulen kegiatan tertentu ke sistem.

STT - NF

## b. Diagram ERD



Gambar 4. 14 Diagram ERD

*Entity-Relationship* Diagram (ERD) di atas ini merupakan struktur basis data untuk aplikasi manajemen ruangan rapat. Diagram ini terdiri dari beberapa tabel utama dan relasinya, yaitu *users*, pegawai, divisi, ruang rapat, fasilitas, permohonan rapat, pengajuan rapat, dan peserta rapat. Berikut adalah penjelasan relasi di dalam ERD:

1. Relasi antara *users* dan permohonan rapat. Relasi ini bersifat *one-to-many*, di mana satu pengguna dapat mengajukan beberapa permohonan rapat
2. Relasi antara pegawai dan divisi. Setiap pegawai termasuk dalam satu divisi. Relasi ini bersifat *one-to-many*, di mana satu divisi dapat memiliki banyak pegawai dan satu pegawai hanya memiliki satu divisi.

3. Relasi antara ruang rapat dan pegawai. Setiap ruangan rapat dikelola oleh satu pegawai. Relasi ini bersifat *one-to-many*, di mana satu pegawai dapat mengelola lebih dari satu ruangan.

4. Relasi antara ruang rapat dan fasilitas. Setiap ruangan rapat dapat memiliki banyak fasilitas. Relasi ini bersifat *many-to-many*, di mana satu fasilitas dapat dimiliki oleh banyak ruangan rapat dan satu ruangan dapat memiliki banyak fasilitas.

5. Relasi antara permohonan rapat dan pengajuan rapat. Setiap permohonan rapat terkait dengan satu pengajuan rapat. Relasi ini bersifat *one-to-one*, di mana satu pengajuan rapat hanya dapat terkait dengan satu permohonan rapat.

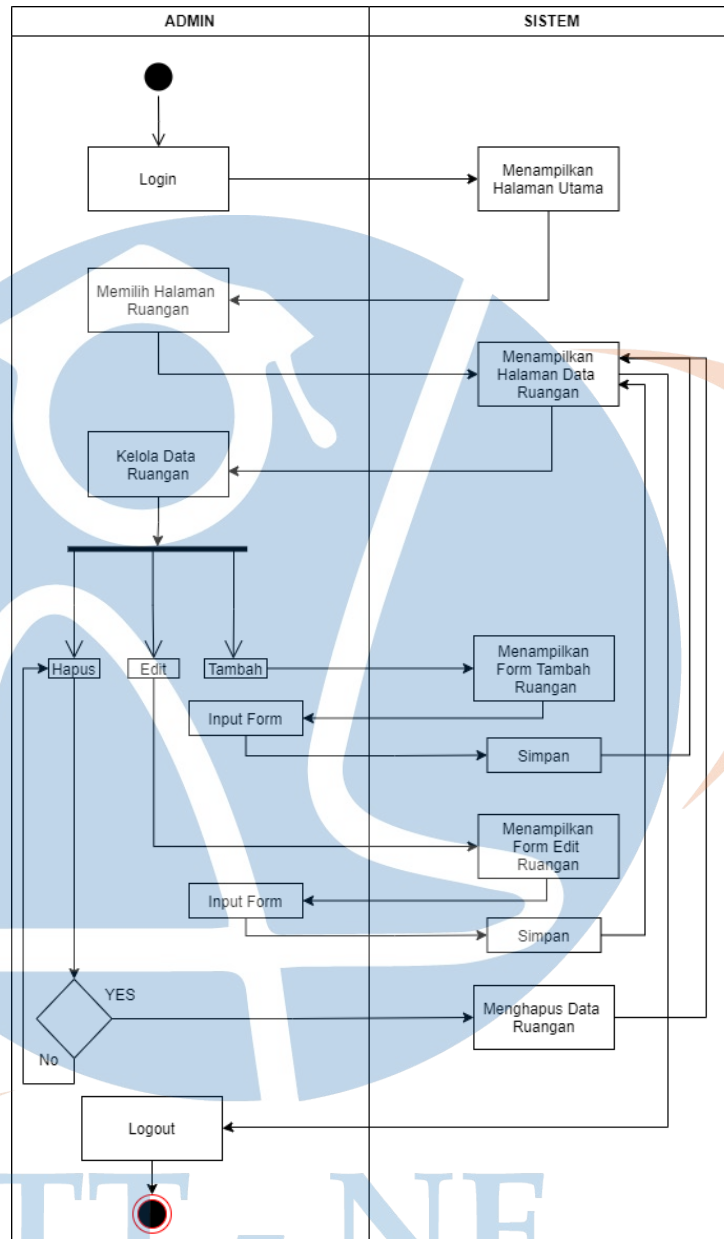
6. Relasi antara peserta\_rapat dan permohonan rapat. Setiap peserta\_rapat terkait dengan satu permohonan rapat. Relasi ini bersifat *one-to-many*, di mana satu permohonan rapat dapat memiliki banyak catatan peserta\_rapat sedangkan satu peserta\_rapat hanya dapat terikat dengan satu permohonan rapat.

Diagram ERD yang dibuat digunakan dalam tahapan iterasi pertama, yaitu perancangan *databases*.

c. Diagram *activity*

Diagram *activity* merupakan diagram yang menggambarkan alur aktivitas pengguna sistem. Diagram *activity* memberikan gambaran bagaimana masing-masing fungsionalitas berakhir dan bagaimana suatu fungsionalitas berakhir [27]. Diagram *activity* yang dibuat menjadi acuan di beberapa iterasi yang dikerjakan.

1. Diagram *activity* pengelolaan data master



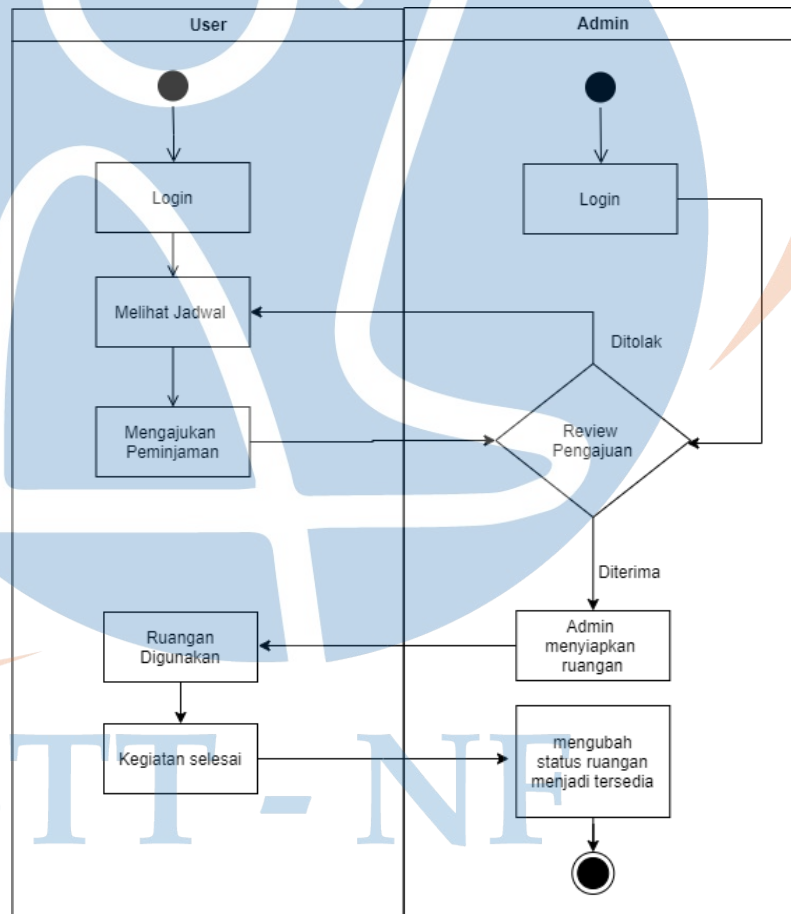
Gambar 4. 15 diagram Pengajuan

Diagram *activity* diatas menunjukkan alur proses pengelolaan data ruangan oleh admin dalam sistem. Proses dimulai dengan admin melakukan *login*, diikuti dengan sistem menampilkan halaman utama. Setelah *login*, admin memilih halaman data ruangan, dan sistem menampilkan daftar ruangan yang ada. Di halaman tersebut, admin memiliki opsi untuk menambah, mengedit, atau menghapus data ruangan. Jika admin memilih



untuk menambah ruangan, sistem akan menampilkan *form* tambah data ruangan, sedangkan jika memilih untuk mengedit, sistem akan menampilkan *form* edit ruangan. Untuk menghapus data, sistem meminta konfirmasi sebelum menghapus data ruangan yang dipilih. Setelah selesai mengelola data, admin dapat melakukan keluar dari sistem. Diagram ini menggambarkan interaksi antara admin dan sistem pada setiap tahap pengelolaan data ruangan. Diagram *activity* pengelolaan ruangan digunakan sebagai referensi ketika proses pengerjaan iterasi data master.

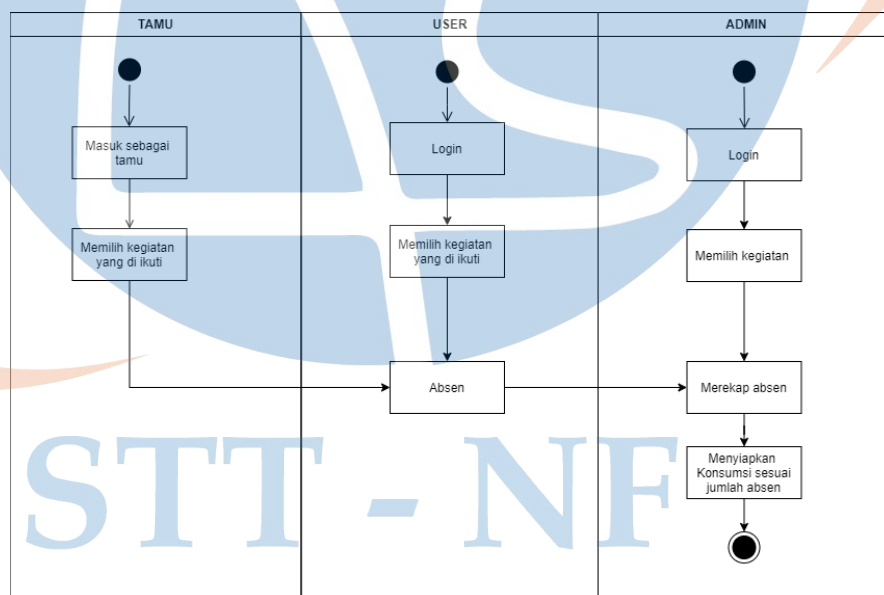
## 2. Diagram *activity* Pengajuan



Gambar 4. 16 diagram *activity* pengajuan kegiatan

Diagram aktivitas ini di atas menggambarkan proses peminjaman ruangan yang melibatkan dua peran utama: *user* dan *admin*. *User* memulai dengan *login* ke dalam sistem, melihat jadwal ruangan, dan mengajukan peminjaman ruangan. *Admin* kemudian melakukan *login* dan meninjau pengajuan peminjaman. Jika pengajuan ditolak, *user* kembali melihat jadwal dan dapat mengajukan peminjaman kembali. Jika pengajuan diterima, *admin* menyiapkan ruangan dan mengubah status ruangan menjadi digunakan. Setelah kegiatan selesai, *admin* kembali mengubah status ruangan menjadi tersedia. Diagram *activity* ini membantu menggambarkan alur kerja dan tanggung jawab masing-masing peran dalam proses pengajuan peminjaman ruangan. Diagram *activity* ini digunakan sebagai referensi ketika di tahap iterasi pengembangan fitur pengajuan dan iterasi halaman muka

### 3. Diagram *activity* absen



Gambar 4. 17 diagram *activity* absen

Diagram aktivitas ini menggambarkan proses absen dalam sistem manajemen kegiatan yang melibatkan tiga peran utama: tamu, *user*, dan *admin*. Tamu dapat masuk ke aplikasi tanpa *login*, memilih kegiatan yang diikuti, dan melakukan absen. *User* harus melakukan *login* terlebih dahulu,

kemudian memilih kegiatan dan melakukan absen. Admin juga memulai dengan *login*, memilih kegiatan yang akan dipantau, lalu merekap data absen dari para peserta, dan menyiapkan konsumsi berdasarkan jumlah absen. Diagram ini membantu dalam proses pengerjaan pada iterasi Pengembangan Fitur Pengajuan dan iterasi.

#### 4.2.3 Coding

Pada tahapan ini, proses mengubah kebutuhan sistem dan rancangan yang sudah dibuat dilakukan melalui proses *coding*. Proses yang dilalui pada tahapan ini adalah sebagai berikut :

1. Persiapan Lingkungan Pengembangan
  - a. Menggunakan *code editor* VSCode untuk menulis kode.
  - b. Menggunakan XAMPP sebagai server lokal.
2. *Pair Programming*
  - a. Melibatkan dua anggota tim yaitu peneliti dan satu pegawai divisi IT
  - b. Satu orang bertugas menulis kode (*driver*), Sedangkan yang lainnya mengawasi dan memberikan masukan (*observer/navigator*). Anggota tim bergantian peran sebagai *driver* dan *observer/navigator* di setiap iterasi.
  - c. Kode yang ditulis diunggah ke *workstation* untuk dilakukan uji coba dan *monitoring* oleh anggota tim yang lain.
  - d. Workstation yang digunakan adalah web *hosting* yang dimiliki oleh divisi IT.

STT - NF

### 3. Konfigurasi Database

#### a. Menyiapkan database



Gambar 4. 18 nama database

Membuat database untuk pengembangan dengan nama ruangrapat

Tabel	Tindakan	Baris	Jenis	Penyortiran	Ukuran	Beban
divisi	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	16	InnoDB	utf8mb4_unicode_ci	16.0 KB	-
failed_jobs	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	0	InnoDB	utf8mb4_unicode_ci	32.0 KB	-
fasilitas	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	3	InnoDB	utf8mb4_unicode_ci	16.0 KB	-
migrations	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	6	InnoDB	utf8mb4_unicode_ci	16.0 KB	-
organisasi	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	122	InnoDB	latin1_swedish_ci	16.0 KB	-
password_resets	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	0	InnoDB	utf8mb4_unicode_ci	16.0 KB	-
pegawai	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	423	InnoDB	latin1_swedish_ci	160.0 KB	-
pengajuan	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	5	InnoDB	utf8mb4_unicode_ci	16.0 KB	-
permohonan_rapat	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	22	InnoDB	utf8mb4_unicode_ci	16.0 KB	-
personal_access_tokens	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	0	InnoDB	utf8mb4_unicode_ci	16.0 KB	-
peserta_rapat	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	0	InnoDB	utf8mb4_general_ci	16.0 KB	-
ruangrapat	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	5	InnoDB	utf8mb4_unicode_ci	16.0 KB	-
users	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	428	InnoDB	utf8mb4_unicode_ci	96.0 KB	-

Gambar 4. 19 struktur tabel database

Gambar di atas merupakan struktur tabel yang terdapat di dalam database ruangrapat. Tabel tersebut dibuat berdasarkan rancangan ERD yang sebelumnya sudah dibuat.

#### b. Menghubungkan Database dengan proyek aplikasi

```
AMMAR
├── app
│   ├── Console
│   ├── Exceptions
│   ├── Exports
│   ├── Http
│   ├── Mail
│   ├── Models
│   ├── Providers
│   ├── bootstrap
│   ├── config
│   ├── database
│   ├── public
│   ├── resources
│   ├── routes
│   ├── ruangRapat
│   ├── storage
│   ├── stubs
│   ├── tests
│   └── vendor
├── .editorconfig
└── .env
    1 APP_NAME=Laravel
    2 APP_ENV=local
    3 APP_KEY=base64://z/q6XiQPgE-j0SPxotQewDr9hDu0+gQMpCVQaRBKSU=
    4 APP_DEBUG=true
    5 APP_URL=http://localhost
    6
    7 LOG_CHANNEL=stack
    8 LOG_DEPRECATIONS_CHANNEL=null
    9 LOG_LEVEL=debug
    10
    11 DB_CONNECTION=mysql
    12 DB_HOST=127.0.0.1
    13 DB_PORT=3306
    14 DB_DATABASE=ruangRapat
    15 DB_USERNAME=root
    16 DB_PASSWORD=
    17
    18 BROADCAST_DRIVER=log
    19 CACHE_DRIVER=file
    20 FILESYSTEM_DRIVER=local
    21 QUEUE_CONNECTION=sync
    22 SESSION_DRIVER=file
    23 SESSION_LIFETIME=120
```

Gambar 4. 20 struktur tabel database

Untuk menghubungkan *database* dengan proyek aplikasi, peneliti membuka *file .env* lalu mengubah informasi pada bagian *DB\_DATABASE* menjadi nama *database* yang dihubungkan yaitu *ruangRapat*. Setelah berhasil terhubung maka tahapan *coding* dikerjakan sesuai dengan ketentuan yang sudah dibuat.

#### 4.2.4 Testing

Tahapan *testing* dalam Extreme Programming mencakup pengujian yang dilakukan secara terus-menerus sepanjang siklus pengembangan untuk memastikan kualitas dan fungsionalitas perangkat lunak. Pada tahapan ini metode pengujian yang digunakan adalah black-box testing. Black-box testing adalah pengujian yang menguji fungsional dan non fungsional dari aplikasi yang sedang dikembangkan. pengujian atau testing dilakukan disetiap iterasi pengembangan selesai dilakukan. Berikut adalah testing yang terjadi di setiap iterasi

##### 1. Pengujian Iterasi *Databases*

Tabel 4. 6 Testing *Databases*

No	Test Case	Skenario	Hasil Yang Diharapkan	Hasil Pengujian
1	Struktur <i>Database</i>	Memeriksa setiap tabel, kolom, tipe data dan relasi agar sesuai dengan rancangan ERD	Struktur data sudah sesuai	Berhasil
2	Pengujian Fungsi CRUD	Menguji kemampuan <i>Create, Read, Update, Delete</i> pada setiap tabel.	Setiap tabel dapat melakukan <i>Create, Read, Update, Delete</i> .	Berhasil

Pada pengujian iterasi pertama, dilakukan pengujian kemampuan dan kepastian struktur *databases* yang dibuat. Pengujian dilakukan dengan pengujian langsung pada database memalui antarmuka phpMyAdmin yang disediakan oleh *software* xampp. Hasil dari pengujian tersebut adalah berhasil.

## 2. Pengujian iterasi Integrasi *Template* dan Data Master

tabel 4. 7 *Testing Integrasi template dan Data master*

Halaman Ruangan				
No	Test Case	Skenario	Hasil Yang Diharapkan	Hasil Pengujian
1	Menambah Ruangan	Klik menu tambah ruangan, muncul modal form, isi form lalu klik tambah	Data Ruangan berhasil ditambah	Berhasil
2	Mengedit ruangan	Klik menu edit, muncul modal form data ruangan, edit form lalu klik edit	Data ruangan berubah	Berhasil
4	Menghapus ruangan	Klik menu hapus, muncul modal konfirmasi dan pilih yes	Data ruangan terhapus	Berhasil
5	Melihat foto ruangan	Klik menu foto, muncul modal berisikan foto ruangan	Menampilkan foto ruangan	Berhasil

6	Pagination	Muncul kotak pencarian di table dan terdapat navigasi jumlah table	Menu pencarian data muncul dan terdapat navigasi jumlah tabel	Berhasil
<b>Halaman Fasilitas</b>				
No	Test Case	Skenario	Hasil Yang Diharapkan	Hasil Pengujian
1	Menambah Fasilitas	Klik menu tambah fasilitas, muncul modal form, isi form lalu klik tambah	Data fasilitas berhasil ditambah	Berhasil
2	Mengedit ruangan	Klik menu edit, muncul modal form data fasilitas, edit form lalu klik edit	Data fasilitas berubah	Berhasil
4	Menghapus ruangan	Klik menu hapus, muncul modal konfirmasi dan pilih yes	Data fasilitas terhapus	Berhasil
6	Pagination	Muncul kotak pencarian di table dan terdapat navigasi jumlah table	Menu pencarian data muncul dan terdapat navigasi jumlah tabel	Berhasil
<b>Halaman Divisi</b>				

No	Test Case	Skenario	Hasil Yang Diharapkan	Hasil Pengujian
1	Menambah Divisi	Klik menu tambah divisi, muncul modal form, isi form lalu klik tambah	Data divisi berhasil ditambah	Berhasil
2	Mengedit Divisi	Klik menu edit, muncul modal form data divisi, edit form lalu klik edit	Data divisi berubah	Berhasil
4	Menghapus Divisi	Klik menu hapus, muncul modal konfirmasi dan pilih yes	Data divisi terhapus	Berhasil
6	Pagination	Muncul kotak pencarian di table dan terdapat navigasi jumlah table	Menu pencarian data muncul dan terdapat navigasi jumlah tabel	Berhasil

#### Halaman Pegawai

No	Test Case	Skenario	Hasil Yang Diharapkan	Hasil Pengujian
1	Menambah Pegawai	Klik menu tambah pegawai, muncul modal form, isi form lalu klik tambah	Data pegawai berhasil ditambah	Berhasil



2	Mengedit Pegawai	Klik menu edit, muncul modal form data pegawai, edit form lalu klik edit	Data pegawai berubah	Berhasil
4	Menghapus Pegawai	Klik menu hapus, muncul modal konfirmasi dan pilih yes	Data pegawai terhapus	Berhasil
6	Pagination	Muncul kotak pencarian di table dan terdapat navigasi jumlah table	Menu pencarian data muncul dan terdapat navigasi jumlah tabel	Berhasil

Pada tahap iterasi selanjutnya, dilakukan pengujian terhadap halaman data master yang meliputi halaman pegawai, divisi, ruangan, dan fasilitas. Pengujian ini berfokus pada fungsionalitas CRUD (*Create, Read, Update, Delete*) dari setiap halaman untuk memastikan tidak terjadi kesalahan dalam kegiatan pengelolaan data. Selain itu, pengujian non-fungsional pada fitur *pagination* juga dilakukan untuk memastikan halaman terlihat lebih rapi dan mudah digunakan, serta memudahkan navigasi bagi pengguna, terutama ketika data yang ditampilkan sangat banyak. Hasil pengujian pada tahap iterasi ini menunjukkan bahwa semua pengujian berhasil dengan baik.

### 3. Iterasi Pengembangan Fitur Pengajuan

tabel 4. 8 Testing Iterasi Pengembangan fitur pengajuan

<b>Halaman Permohonan</b>				
<b>No</b>	<b>Test Case</b>	<b>Skenario</b>	<b>Hasil Yang Diharapkan</b>	<b>Hasil Pengujian</b>
1	Memilih ruangan	Memilih ruangan di <i>select box</i>	Menampilkan detail ruangan yaitu kapasitas, lokasi dan penanggung jawab.	Berhasil
			Menampilkan Jadwal ruangan di hari terkait	Berhasil
2	Validasi Sistem	Waktu yang dipilih sudah ada di jadwal	Sistem menolak dan memberikan notifikasi jadwal sudah terisi	Berhasil
		Jumlah peserta melebihi kapasitas ruangan	Sistem menolak dan memberikan notifikasi jumlah peserta melebihi kapasitas	Berhasil
3	Mengajukan permohonan Ruang	Mengisi formulir peminjaman ruangan dengan data yang benar	Sistem menyimpan pengajuan ke tabel pengajuan dan memberikan notifikasi data disimpan	Berhasil
<b>Tabel Pengajuan</b>				
1	Hapus data pengajuan	Memilih menu hapus pengajuan	Klik menu hapus, muncul modal konfirmasi dan pilih <i>yes</i>	Berhasil
2	Melihat detail pengajuan	Memilih menu detail pengajuan	Menampilkan modal detail ajuan	Berhasil
3	Pagination	Memuat data lebih dari 10	Terdapat search box di tabel dan navigasi	Berhasil

			memuncil data yang lain	
<b>Halaman Agenda</b>				
<b>No</b>	<b>Test Case</b>	<b>Skenario</b>	<b>Hasil Yang Diharapkan</b>	<b>Hasil Pengujian</b>
1	Melakukan Konfirmasi Pengajuan	Memilih menu konfirmasi	Menampilkan modal berisikan detail ajuan dan opsi terima atau tolak	Berhasil
		Memilih menu terima di modal konfirmasi	Muncul modal konfirmasi menerima, pilih yes lalu ajuan akan diterima dan disimpan di bagian pengajuan diterima	Berhasil
		Memilih menu tolak di modal konfirmasi	Muncul modal konfirmasi tolak, pilih yes lalu ajuan akan ditolak dan disimpan di bagian riwayat pengajuan	Berhasil
<b>Tab pengajuan diterima</b>				
1	Melihat data peserta kegiatan	Memilih menu peserta	Menampilkan modal berisi data peserta kegiatan beserta tanda absen	Berhasil

2	Mengubah status kegiatan menjadi selesai	Memilih menu selesai	Muncul modal konfirmasi selesai, pilih <i>yes</i> lalu kegiatan dianggap selesai dan disimpan di riwayat pengajuan	Berhasil
<b>Halaman Riwayat</b>				
<b>No</b>	<b>Test Case</b>	<b>Skenario</b>	<b>Hasil Yang Diharapkan</b>	<b>Hasil Pengujian</b>
1	Melihat daftar peserta	Memilih menu peserta	Menampilkan modal berisi data peserta kegiatan beserta tanda absen	Berhasil
2	Mengunduh rekap absen	Memilih menu rekap rapat	Mengunduh file rekap rapat dalam format pdf	berhasil
3	Mengunggah file notulen kegiatan	Memilih menu unggah notulen	Mengunggah file notule	Berhasil

4	Melihat file notulen kegiatan	Memilih menu lihat file notulen	Mengunduh file notulen dalam bentuk pdf	Berhasil
5	Pagination	Memuat data lebih dari 10	Terdapat search box di tabel dan navigasi memunculkan data yang lain	Berhasil
<b>Halaman Daftar Hadir</b>				
<b>No</b>	<b>Test Case</b>	<b>Skenario</b>	<b>Hasil Yang Diharapkan</b>	<b>Hasil Pengujian</b>
1	Melihat data peserta kegiatan	Memilih menu peserta	Menampilkan modal berisi data peserta kegiatan beserta tanda absen	Berhasil
2	<i>Pagination</i>	Memuat data lebih dari 10	Terdapat search box di tabel dan navigasi memunculkan data yang lain	Berhasil
<b>Halaman Absen</b>				

No	Test Case	Skenario	Hasil Yang Diharapkan	Hasil Pengujian
1	Melakukan Absen	Mengisi <i>form</i> , lalu memilih menu absen dan mendapatkan notifikasi absen berhasil	Absen tercatat ke sistem dan menampilkan notifikasi berhasil absen	Berhasil

Iterasi ketiga, adalah pengembangan fitur pengajuan peminjaman. Hal ini meliputi pengajuan pinjaman, penerimaan peminjaman, riwayat pengajuan, daftar hadir dan absensi. Dalam tahap ini dilakukan pengujian terhadap semua halaman dan fitur yang di kembangkan, seperti memastikan user dapat mengajukan peminjaman dan melakukan absen. Sedangkan admin dapat meninjau peminjaman yang masuk dan melihat daftar hadir peserta kegiatan. Secara keseluruhan Hasil pengujian iterasi tersebut berhasil semua.

#### 4. Iterasi Modul Halaman Utama

tabel 4. 9 Iterasi Modul Halaman Utama

Halaman Dashboard/Home				
No	Test Case	Skenario	Hasil Yang Diharapkan	Hasil Pengujian
1	Melihat detail kegiatan di kalender	Memilih tanggal tertentu yang terdapat peminjaman ruangan	Menampilkan Modal detail peminjaman	Berhasil
2	Melihat Detail data ruangan	Memilih nama ruangan yang ingin dilihat	Menampilkan modal berisi detail ruangan	Berhasil

3	Melihat kegiatan yang ada di hari ini	Memilih menu <i>dropdown</i> kegiatan hari ini	Menampilkan daftar kegiatan yang ada di hari ini	Berhasil
---	---------------------------------------	--	--	----------

Pada tahap iterasi modul halaman utama, dilakukan pengujian terhadap fitur-fitur yang terdapat pada halaman utama. Pengujian meliputi modal detail kegiatan, modal yang berisikan data ruangan yang dipilih serta fitur yang menampilkan kegiatan yang terdapat di hari ini. Hasil pengujian pada tahap ini mendapatkan keberhasilan di semua pengujian yang dilakukan.

#### 5. Iterasi Role

tabel 4. 10 Iterasi role

<b>Halaman Login</b>				
<b>No</b>	<b>Test Case</b>	<b>Skenario</b>	<b>Hasil Yang Diharapkan</b>	<b>Hasil Pengujian</b>
<b>1</b>	Login Berdasarkan <i>Role</i>	Login sebagai Admin	Menampilkan halaman yang dapat di akses sesuai dengan role	Berhasil
		Login sebagai Super Admin	Menampilkan halaman yang dapat di akses sesuai dengan role	Berhasil
		Login sebagai User	Menampilkan halaman yang dapat di akses sesuai dengan role	Berhasil
		Login sebagai tamu	Menampilkan halaman daftar kegiatan	Berhasil

<b>Halaman Registrasi</b>				
<b>No</b>	<b>Test Case</b>	<b>Skenario</b>	<b>Hasil Yang Diharapkan</b>	<b>Hasil Pengujian</b>
<b>1</b>	Registrasi	Registrasi akun dengan <i>role</i> user	Akun baru berhasil di registrasi sesuai dengan <i>role</i> yang dipilih	Berhasil
		Registrasi akun dengan <i>role</i> admin		Berhasil
		Registrasi akun dengan <i>role</i> super admin		Berhasil
<b>2</b>	Validasi	<i>Form</i> tidak di isi	Muncul notifikasi form harus di isi	Berhasil
		Konfirmasi password berbeda	Muncul notifikasi <i>password</i> tidak cocok	Berhasil
<b>Halaman Tamu</b>				
<b>No</b>	<b>Test Case</b>	<b>Skenario</b>	<b>Hasil Yang Diharapkan</b>	<b>Hasil Pengujian</b>
<b>1</b>	Melihat halaman untuk tamu	Memili menu masuk sebagai tamu	Menampilkan halaman tamu berisikan daftar kegiatan	Berhasil



		di halaman <i>login</i>		
2	Melakukan absen	Memilih menu absen di kegiatan yang dipilih	Mengarahkan ke halaman absen, mengisi form absen lalu kembali ke halaman tamu	Berhasil

Pada pengujian tahap ini, meliputi pengujian terkait akses dan tampilan yang sesuai dengan role yang masuk ke dalam sistem. Meliputi isi menu *sidebar* yang menampilkan sesuai dengan *role* yang masuk ke sistem. Pada halaman registrasi dilakukan pengujian yang memastikan dapat membuat akun baru beserta *role* yang dipilih. Selain itu dilakukan pengujian validasi terhadap halaman *login* dan registrasi. Pada halaman *login* Validasi yang dilakukan adalah dengan menguji ketika *password* atau *username* yang dimasukkan salah, maka sistem menolak masuk dan memberikan notifikasi alasan ditolak. Sedangkan pada halaman registrasi dilakukan pengujian ketika *form* kosong dan kotak konfirmasi *password* tidak sesuai maka sistem akan memberikan notifikasi *error*. Hasil Pengujian yang didapatkan secara keseluruhan merupakan berhasil.

## 6. Testing non fungsional

tabel 4. 11 Testing Non-fungsional

No	Test Case	Skenario	Hasil Yang Diharapkan	Hasil Pengujian
1	<i>Responsive</i>	Menguji coba aplikasi di buka di berbagai perangkat	Tampilan aplikasi menyesuaikan dengan perangkat yang digunakan	Berhasil
2	<i>Sidebar</i>	Menguji <i>button sidebar</i>	<i>Sidebar</i> dapat dibuka dan ditutup tanpa ada masalah	Berhasil
3	Tema Gelap	Memilih menu ubah tema	Tampilan tema aplikasi menjadi tema gelap	Berhasil
5	Nama <i>user</i> yang sedang <i>login</i>	Menampilkan nama sesuai dengan yang	Menampilkan nama sesuai dengan yang	Berhasil

		sedang <i>login</i> di sistem	sedang <i>login</i> di sistem	
6	Notifikasi jumlah pengajuan yang masuk di menu agenda	Mengajukan pengajuan peminjaman	Muncul <i>pop up badge</i> berupa jumlah di samping menu <i>button</i> halaman agenda	Berhasil

Setelah memastikan pengujian setiap iterasi selesai, dilakukan pengujian non-fungsional untuk menguji kebutuhan non-fungsional aplikasi. Pengujian ini meliputi *pagination*, *responsive*, *sidebar*, tema gelap, dan menampilkan nama pengguna yang sedang *login* di *sidebar*. Pengujian ini bertujuan untuk memastikan aplikasi memberikan pengalaman pengguna yang optimal dan nyaman saat digunakan. Hasil dari pengujian ini menunjukkan bahwa semua aspek berhasil dengan baik.

Secara keseluruhan, hasil pengujian setiap terasi mendapatkan predikat berhasil. Meskipun ada beberapa perubahan terhadap fitur yang direncanakan. Seperti fitur notifikasi, rencana awal adalah pengguna mendapatkan notifikasi ketika terdapat pengajuan, karena beberapa kendala, notifikasi yang di gambarkan berupa *pop up* pada menu *sidebar* untuk *role* admin. Berdasarkan hal tersebut, pengujian sudah sesuai dengan skenario yang diharapkan dan aplikasi dapat di teruskan di tahap selanjutnya.

#### 4.2.5 Release

Pada tahap akhir pengembangan aplikasi ini, dilakukan *release* dengan mengunggah keseluruhan *file* proyek aplikasi ke dalam web *hosting* yang dimiliki divisi IT. Aplikasi ini akan dirilis dengan nama AMARR yang merupakan singkatan dari Aplikasi Manajemen Ruang Rapat. Proses *upload* dilakukan ke web *hosting* yang dimiliki oleh divisi IT perusahaan, dengan domain <https://amarr.asastatech.com/>.

#### 4.3 Hasil

##### 1. Logo Aplikasi

The logo for AMARR (Aplikasi Manajemen Ruang Rapat) features a stylized blue and white graphic above the text "AMARR" in a bold, blue, sans-serif font.

Gambar 4. 21 logo AMARR

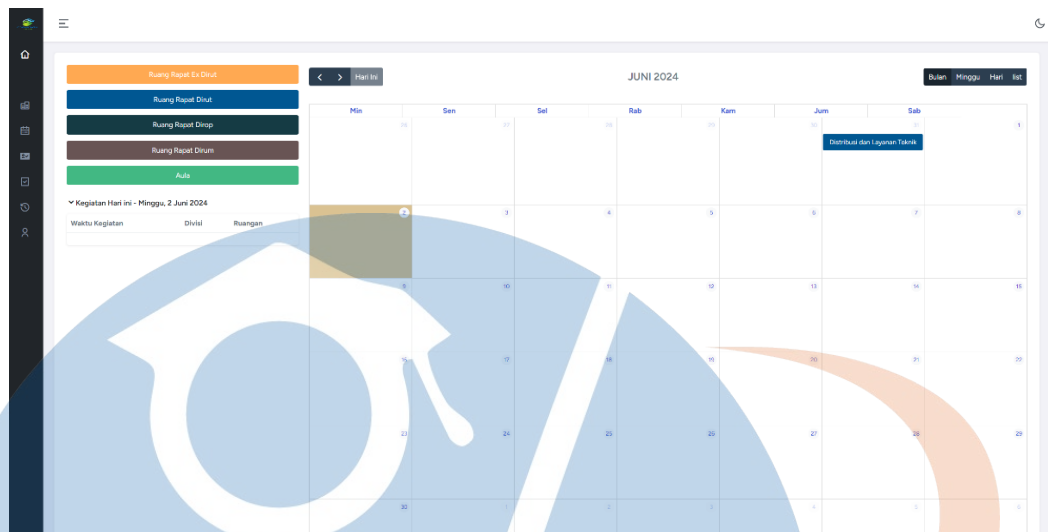
##### 2. Halaman Login



Gambar 4. 22 halaman login

Pada halaman *login*, diberikan tambahan menu absen sebagai tamu untuk pihak tamu yang ingin melakukan absensi. Selain itu terdapat informasi jadwal kegiatan di hari aplikasi di akses.

### 3. Halaman Muka



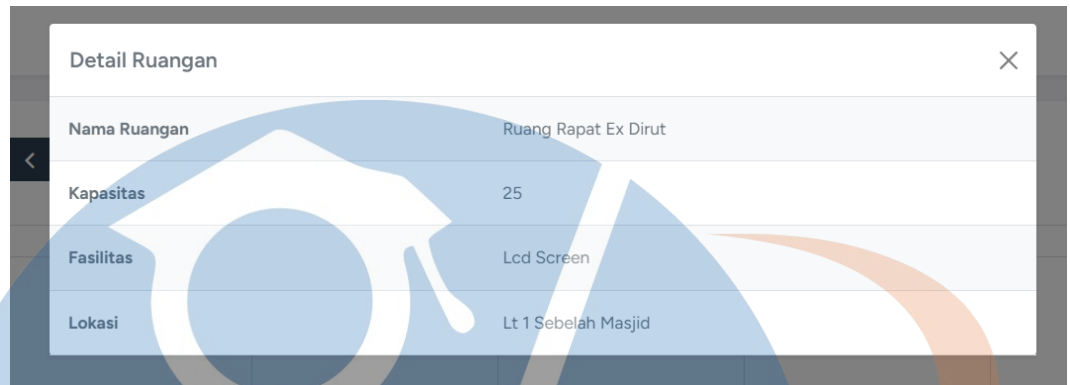
Gambar 4. 23 halaman muka

Halaman muka memberikan tampilan jadwal peminjaman dengan format kalender. Setiap *event* yang terdapat di kalender dapat dipilih untuk menampilkan detail *event*. Daftar nama ruangan pada sebelah kiri juga dapat dipilih untuk menampilkan detail ruangan. Di bawah daftar ruangan terdapat informasi kegiatan apa saja yang terdapat pada hari terkait.



Gambar 4. 24 modal keterangan kegiatan

Gambar di atas adalah tampilan modal yang berisikan detail dari *event*. Detail tersebut terdiri dari nama kegiatan, tanggal dan waktu, ruangan yang digunakan dan nama-nama peserta kegiatan.



Gambar 4. 25 modal detail ruangan

Gambar di atas menampilkan modal yang memberikan detail suatu ruangan. Detail tersebut terdiri dari nama, kapasitas, fasilitas dan lokasi ruangan.

STT - NF

#### 4. Halaman data master (ruang rapat)

NO	Nama	Kapasitas	Penanggung Jawab	Fasilitas	Lokasi	Foto Ruangan	Opsi
1	Ruang Rapat Ex Dint	25	Agus Handi	Lot Screen	Lt 1 Sebelah Masjid		<span>Edit</span> <span>Hapus</span>
2	Ruang Rapat Dint	50	Adin Shabila Mulya, S.Hum.	Lot Screen	Lt 3		<span>Edit</span> <span>Hapus</span>
3	Ruang Rapat Dlepp	60	Adnan Zulfizar	TV	Lt 3		<span>Edit</span> <span>Hapus</span>
4	Ruang Rapat Drum	30	Affiah Azahra	TV	Lt 2		<span>Edit</span> <span>Hapus</span>
5	Aula	38	Amriyati, S.Pd.		Lt 3		<span>Edit</span> <span>Hapus</span>

Gambar 4. 26 halaman ruang rapat

**Tambah Ruangan** ✕

Nama Ruangan

Kapasitas ruangan

Penanggung Jawab  
Pilih Penanggung Jawab ▼

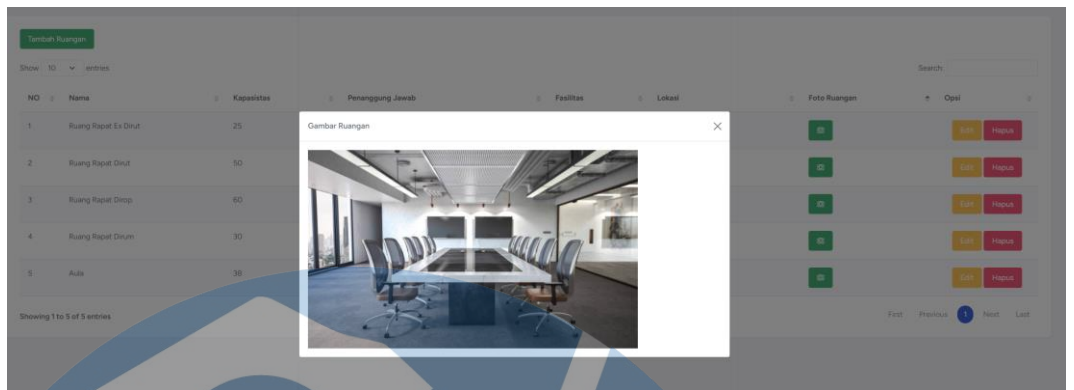
Gambar Ruangan  
 No file chosen

Fasilitas

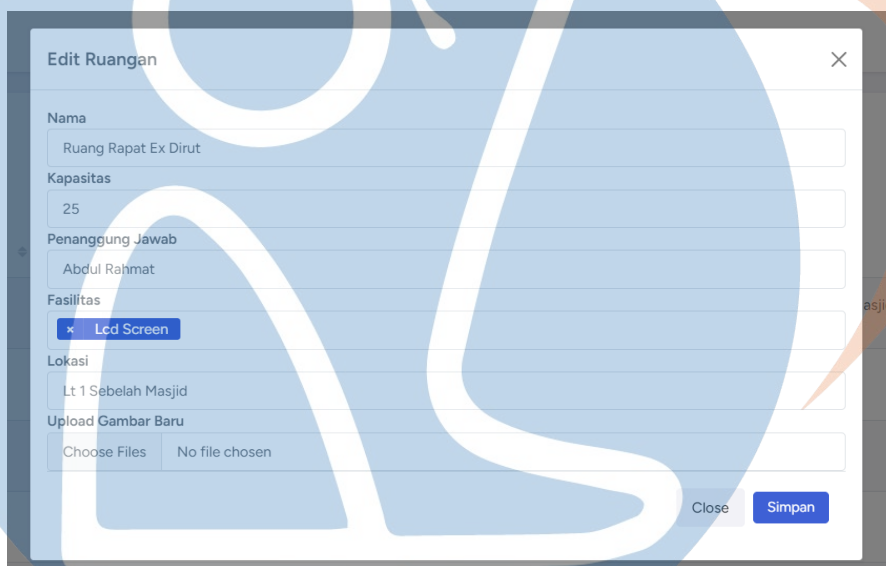
Lokasi

Close Simpan

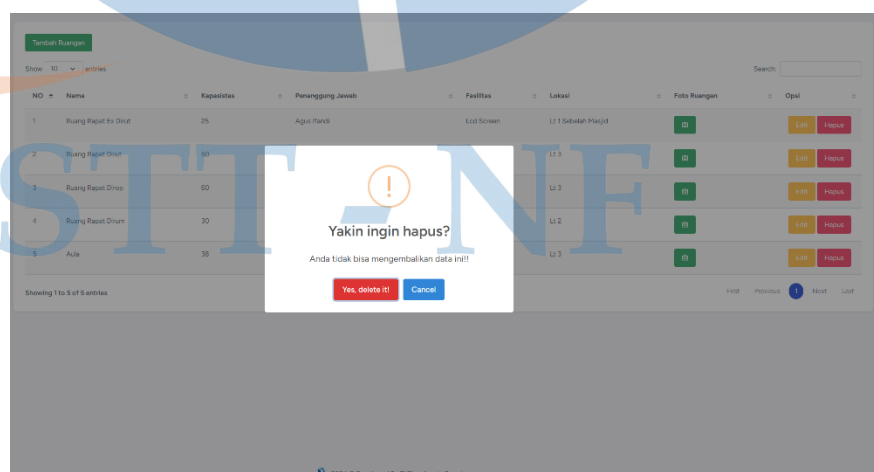
Gambar 4. 27 modal tambah ruangan



Gambar 4. 28 foto ruangan terkait



Gambar 4. 29 modal edit ruangan

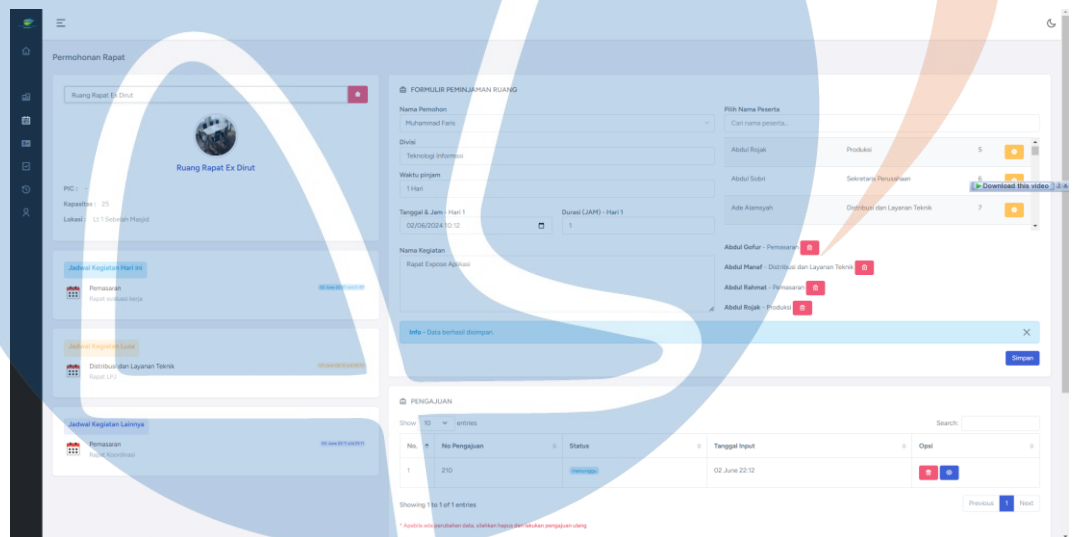


Gambar 4. 30 modal hapus



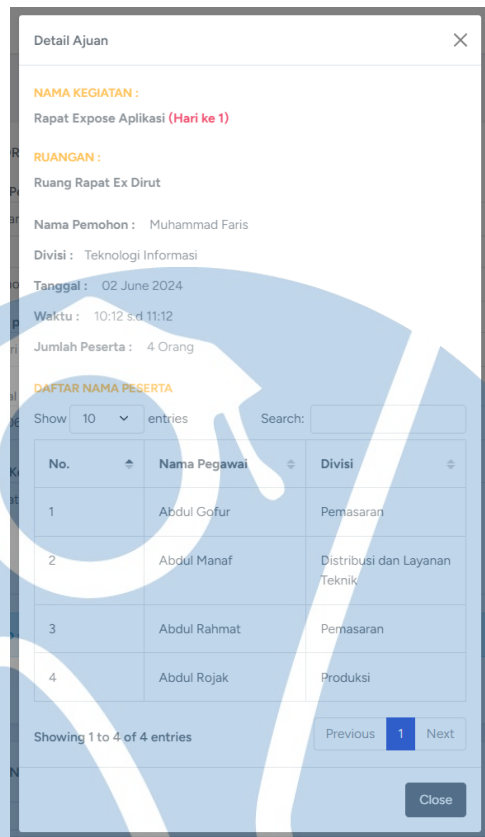
Pada hasil halaman data master, peneliti menggunakan halaman ruangan rapat sebagai salah satu data master. Pada halaman tersebut terdapat tabel yang menampilkan data ruangan beserta opsi untuk hapus dan edit. Menu tambah ruangan diletakan di atas tabel untuk admin menambah data ruangan. Setiap menu tersebut akan memunculkan modal terkait. Menu tambah ruangan akan menampilkan modal beserta *form* untuk mengisi data ruangan yang ingin ditambahkan. Menu edit akan menampilkan modal beserta *form* data ruangan yang ingin di ubah. Menu foto akan menampilkan foto dari ruangan terkait. Menu hapus akan menampilkan menu konfirmasi untuk menghapus data ruangan.

## 5. Halaman Permohonan



Gambar 4. 31 halaman permohonan

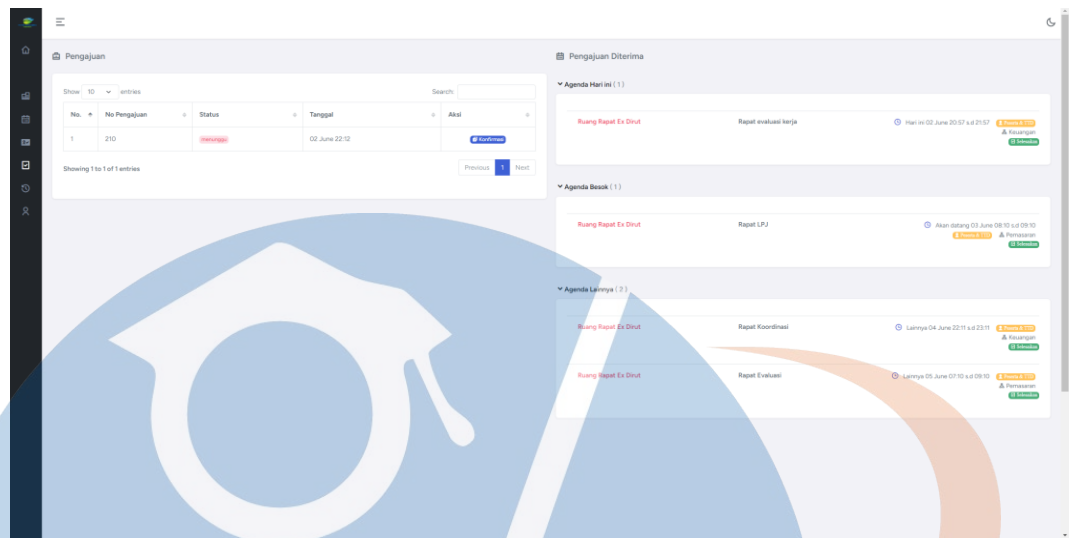
# STT - NF



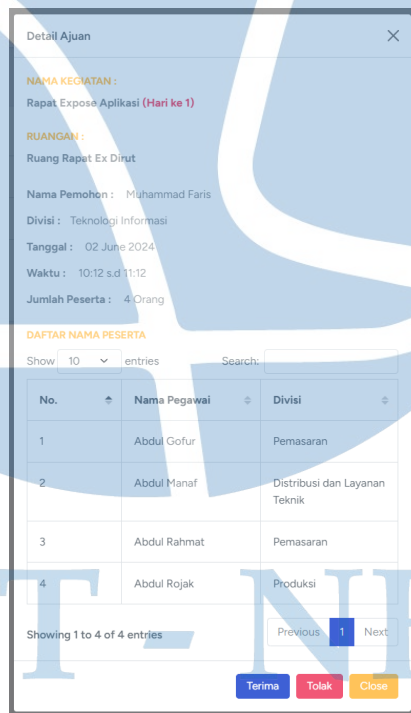
Gambar 4. 32 modal detail ajuan

Pada halaman permohonan terdapat *form* permohonan yang dapat di isi untuk melakukan pengajuan pinjaman. Di bawah bagian memilih ruangan terdapat jadwal ruangan yang dipilih berdasarkan tiga waktu yaitu hari ini, lusa dan 3 hari atau lebih dari hari ini. Tampilan tersebut untuk memberikan informasi kepada pengguna sehingga tidak perlu kembali ke halaman muka untuk melihat jadwal. Ajuan yang sudah tercatat sistem akan muncul pada tabel pengajuan beserta menu detail yang menampilkan modal detail ajuan.

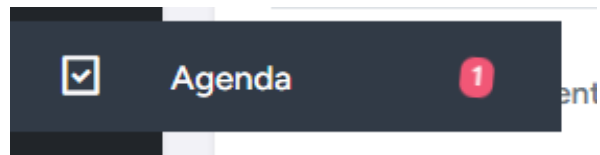
## 6. Halaman Agenda



Gambar 4. 33 halaman agenda



Gambar 4. 34 modal konfirmasi ajuan



Gambar 4. 35 notifikasi jumlah ajuan yang harus di review

Pada halaman agenda terdapat dua bagian, yaitu pengajuan dan pengajuan diterima. Pengajuan akan menampilkan daftar ajuan yang harus di konfirmasi oleh admin sedangkan pengajuan diterima adalah daftar pengajuan yang sudah diterima dan dibagi menjadi 3 berdasarkan waktu peminjamannya. Agenda hari ini, agenda besok dan agenda lainnya untuk peminjam 3 hari atau lebih dari hari ini. Pada *sidebar* aplikasi akan menampilkan jumlah pengajuan yang harus dikonfirmasi oleh admin.

## 7. Halaman Riwayat

No	Ruang	Kegiatan	Divisi	Tanggal Pinjam	Perserta	Notulen
1	Ruang Rapat Ex Druut	Rapat Expose Aplikasi	Perencanaan Teknik	2 Juni 2024 10:12 s.d 11:12	[Status]	[Upload]
2	Ruang Rapat Ex Druut	Rapat Koordinasi	Kuangan	4 Juni 2024 22:11 s.d 23:11	[Status]	[Upload]
3	Ruang Rapat Ex Druut	Rapat Evaluasi	Pemasaran	5 Juni 2024 07:10 s.d 09:10	[Status]	[Upload]
4	Ruang Rapat Ex Druut	Rapat LPTJ	Pemasaran	3 Juni 2024 08:19 s.d 09:10	[Status]	[Upload]
5	Ruang Rapat Ex Druut	Rapat evaluasi kerja	Kuangan	2 Juni 2024 10:57 s.d 21:57	[Status]	[Upload]
6	Ruang Rapat Druup	ABCD	Kuangan	3 Juni 2024 15:33 s.d 16:33	[Status]	[Upload]
7	Ruang Rapat Ex Druut	Rapat nyoba	Kuangan	31 Mei 2024 15:04 s.d 16:00	[Status]	[Upload]
8	Ruang Rapat Druut	Rapat nyoba	Pemasaran	31 Mei 2024 08:53 s.d 10:53	[Status]	[Upload]
9	Ruang Rapat Ex Druut	Bisnis	Kuangan	23 Mei 2024 21:14 s.d 22:14	[Status]	[Upload]
10	Ruang Rapat Ex Druut	Kapastan	Kuangan	6 Mei 2024 21:05 s.d 22:05	[Status]	[Upload]

Gambar 4. 36 halaman riwayat

Pada halaman riwayat akan menampilkan daftar pengajuan yang sudah tercatat oleh sistem, baik yang diterima, ditolak dan yang sudah selesai. Terdapat menu untuk mengunduh rekap absen berformat pdf. Selain itu terdapat menu unggah berkas notulensi dari kegiatan. Terdapat juga fitur untuk memfilter data pengajuan berdasarkan waktunya

## 8. Halaman Daftar Hadir

No.	Ruang	Kegiatan	Divisi	Tanggal Pinjam	Peserta	Absen
1	Ruang Rapat Ex Dirut	Rapat evaluasi kerja	Keuangan	2 Juni 2024 20:57 s.d 21:57	4	Absen
2	Ruang Rapat Ex Dirut	Kapasitas	Keuangan	6 Mei 2024 21:05 s.d 22:05	10	Absen
3	Ruang Rapat Dirut	Ada deh	Keuangan	3 Mei 2024 19:39 s.d 20:39	4	Absen
4	Ruang Rapat Dirut	Ada deh	Keuangan	2 Mei 2024 18:39 s.d 20:39	9	Absen

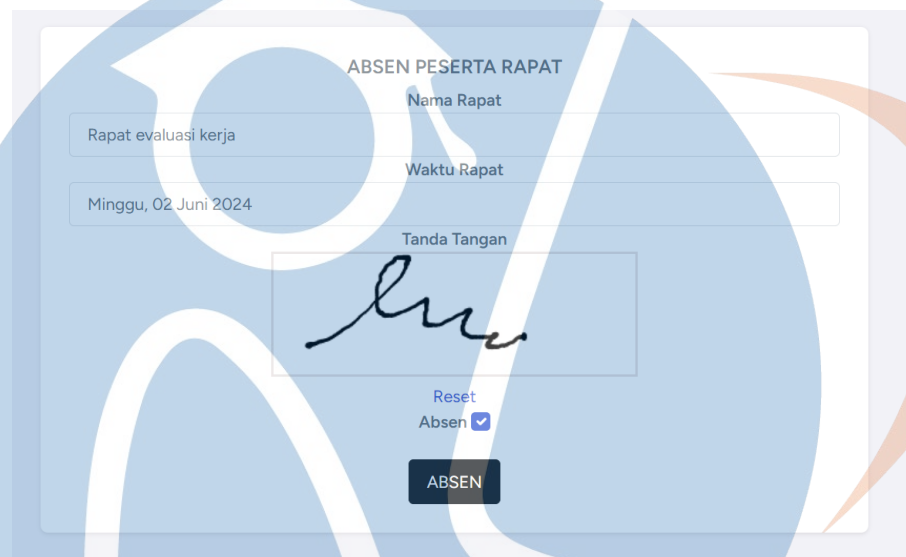
Gambar 4. 37 halaman daftar hadir

No.	Nama Pegawai	Divisi	Tanda Tangan	Waktu
1	Abdul Gofur	Pemasaran		22:56 WIB
2	Abdul Manaf	Distribusi dan Layanan Teknik		20:58 WIB
3	Abdul Rachman Sholeh	Distribusi dan Layanan Teknik		20:58 WIB
4	Abdul Rahmat	Pemasaran		20:58 WIB

Gambar 4. 38 modal daftar absen peserta

Pada halaman daftar hadir menampilkan daftar kegiatan dan menu yang menampilkan daftar absen peserta. Untuk user, dapat melakukan absen pada halaman ini. Daftar kegiatan yang tampil hanya kegiatan yang mereka ikuti saja. Sedangkan pada halaman admin akan menampilkan seluruh daftar kegiatan yang diterima.

#### 9. Halaman Absen



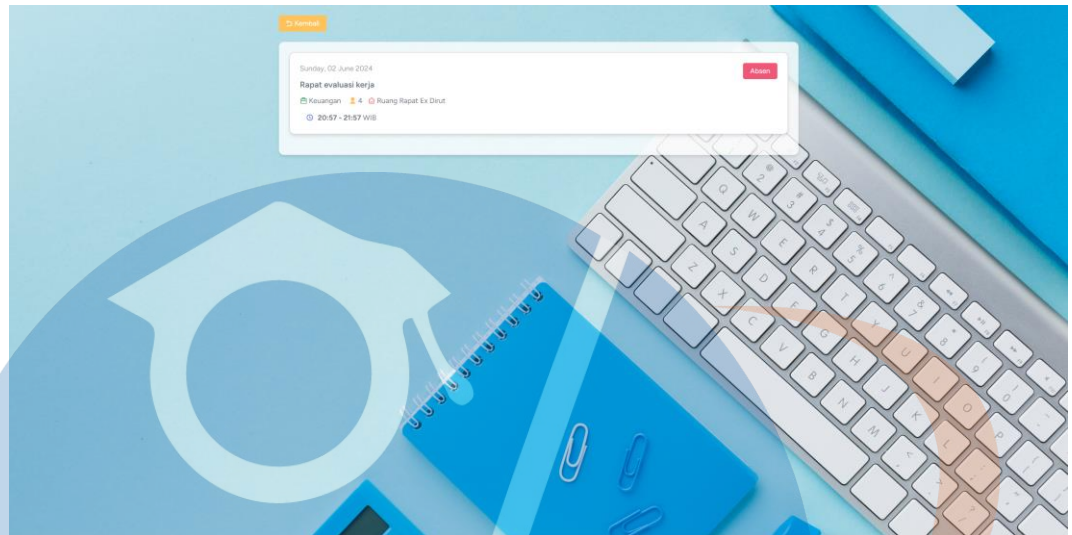
The screenshot shows a web form titled "ABSEN PESERTA RAPAT". It contains the following elements:

- Nama Rapat:** Input field with the value "Rapat evaluasi kerja".
- Waktu Rapat:** Input field with the value "Minggu, 02 Juni 2024".
- Tanda Tangan:** A rectangular box containing a handwritten signature in black ink.
- Reset:** A small blue text link.
- Absen:** A checkbox that is currently checked.
- ABSEN:** A dark blue button with white text.

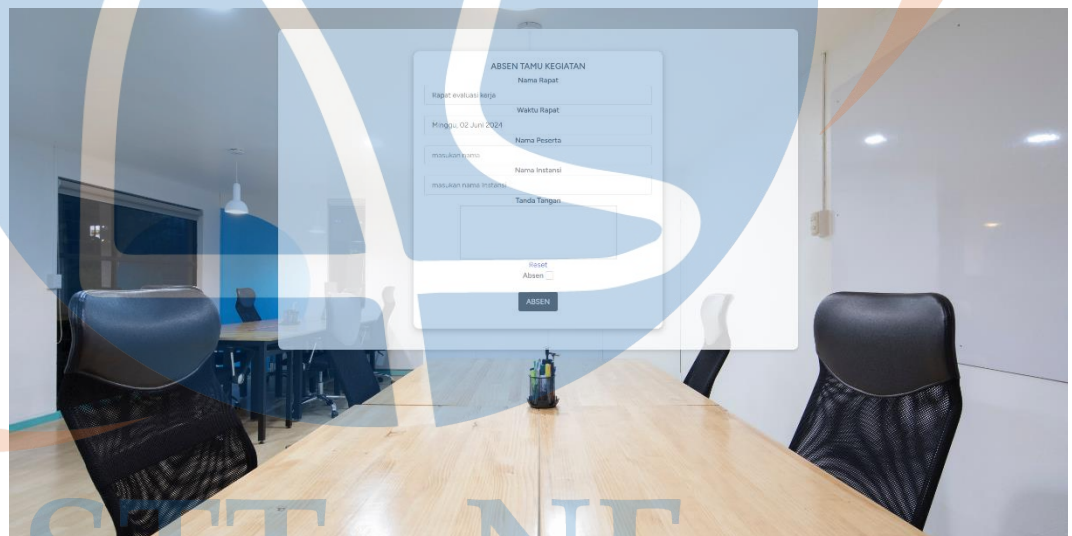
Gambar 4. 39 halaman absen

Dalam halaman absensi peserta, terdapat kotak tanda tangan yang digunakan untuk mencatat tanda tangan peserta. Jika terjadi kesalahan, pengguna dapat menggunakan tombol "Reset" untuk mengulang proses tanda tangan. Selanjutnya, sebelum tombol "Absen" dapat dipilih, pengguna harus mencentang *checkbox* absen. Dengan demikian, tombol "Absen" akan aktif dan memungkinkan peserta untuk melakukan absensi.

## 10. Halaman Tamu



Gambar 4. 40 halaman tamu



Gambar 4. 41 halaman absen tamu

Dalam halaman tamu, akan menampilkan daftar kegiatan di hari aplikasi di akses. Selanjutnya tamu dapat memilih tombol absen dan akan diarahkan ke halaman absen untuk tamu. Perbedaan dengan absen pegawai perusahaan adalah terdapat *form* yang perlu di isi, yaitu nama peserta dan nama kegiatan.

## 11. Akses aplikasi via ponsel



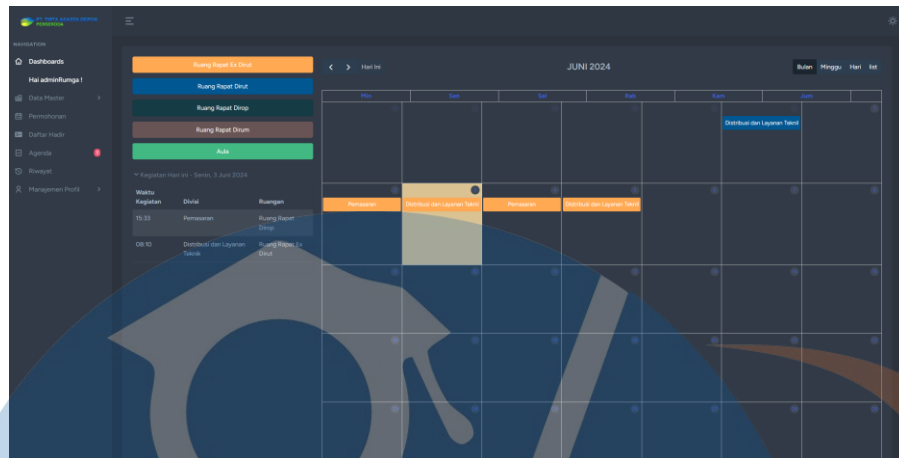
Gambar 4. 42 halaman muka di akses menggunakan ponsel

Gambar di atas menampilkan halaman muka yang di akses menggunakan ponsel. Ukuran aplikasi otomatis menyesuaikan dengan ukuran perangkat yang mengakses atau bisa disebut sebagai *responsive*.

STT - NF



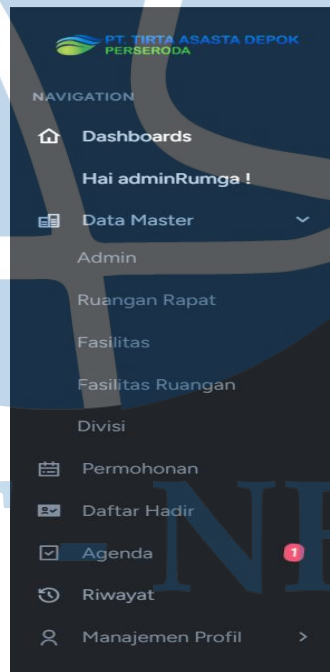
## 12. Tema Gelap Aplikasi



Gambar 4. 43 tema gelap aplikasi

Gambar di atas menampilkan tampilan aplikasi ketika diubah menjadi tema gelap.

## 13. Sidebar



Gambar 4. 44 sidebar

Tampilan bagian *sidebar* aplikasi yang menampilkan menu apa saja yang terdapat pada *sidebar*. Selain itu, terdapat *pop up* notifikasi berupa jumlah pengajuan yang masuk dan perlu di tinjau di sebelah menu agenda.

## BAB V

### KESIMPULAN DAN SARAN

#### 5.1 Kesimpulan

Berdasarkan penelitian yang telah dilaksanakan, dapat diambil kesimpulan bahwa pengembangan aplikasi manajemen ruang rapat berbasis web di PT. Tirta Asasta Depok diberikan nama “AMARR” yang merupakan singkatan dari manajemen ruangan rapat. Hasil dari perancangan AMARR telah menjawab rumusan masalah yang dibuat sebelumnya, yang dijelaskan seperti berikut :

1. Proses perancangan AMARR melewati beberapa tahapan. Pada tahap awal dilakukan studi pendahuluan, yang meliputi studi literatur, yaitu pengumpulan teori dan penelitian terkait sebagai referensi peneliti serta studi lapangan, yaitu mengumpulkan data-data terkait permasalahan manajemen ruangan yang saat ini berjalan dengan cara observasi dan wawancara di perusahaan. Tahap selanjutnya adalah pengembangan produk. Pada tahap ini digunakan metode *extreme programming*(XP) yaitu metode yang cepat dan dapat beradaptasi terhadap perubahan. Pada Metode XP terdapat 4 tahapan. Pertama adalah *planning*, membuat *user stories* dan kebutuhan sistem berdasarkan hasil wawancara dan observasi yang sudah dilakukan sebelumnya. Hasil dari *user stories* tersebut adalah terdapat empat *role* yang akan menggunakan aplikasi. Selain itu, tahap *planning* juga membuat perencanaan iterasi. Tahapan kedua adalah *design*. Yaitu pembuatan diagram UML dan *prototype* aplikasi berdasarkan *user stories* kebutuhan sistem sebelumnya. Setelah hal tersebut, dilakukan tahapan *coding*. Tahap ini merupakan proses menulis kode aplikasi berdasarkan tahapan sebelumnya. Penulisan kode menggunakan aplikasi VScode dan menggunakan metode *pair-programming*. Tahapan terakhir adalah dilakukan pengujian fungsionalitas dan non-fungsional menggunakan metode *black-box* testing. Pengujian dilakukan di setiap iterasi untuk memastikan iterasi tersebut sudah berhasil dan dapat dilanjutkan ke iterasi selanjutnya. Hasil dari pengujian ini sudah sesuai dengan yang diharapkan

walaupun terdapat beberapa penyesuaian fitur. Setelah memastikan aplikasi sudah selesai dikembangkan, maka melewati tahap *release*. Tahapan ini menyerahkan aplikasi kepada perusahaan. Aplikasi diserahkan dengan mengunggah keseluruhan file kode aplikasi di *web hosting* yang dimiliki perusahaan. Selanjutnya aplikasi diproses oleh perusahaan agar dapat di implementasikan di perusahaan.

2. Berdasarkan hasil dari pengembangan aplikasi Amarr didapatkan beberapa fitur yang menyesuaikan dengan kebutuhan dari *user*, fitur tersebut antara lain fitur pengajuan ruangan, yaitu fitur yang memberikan akses kepada pengguna/pegawai untuk mengajukan peminjaman ruangan. Fitur ini juga merangkap untuk admin mengelola peminjaman, baik menolak atau menerima ajuan yang masuk ke aplikasi. Pada bagian admin, akan terdapat notifikasi pop-up pada menu agenda yang menampilkan jumlah pengajuan yang harus di tinjau oleh admin. Selain itu, Fitur ini juga meliputi riwayat pengajuan yang dapat melihat data pengajuan yang masuk dan mengunggah berkas notulen kegiatan. Fitur lain yang dikembangkan adalah fitur absensi kegiatan. Fitur ini memudahkan pengguna untuk melakukan absen di kegiatan yang di ikuti. Selain itu, fitur absen juga memudahkan admin untuk melihat jumlah kehadiran di suatu kegiatan agar admin dapat menyediakan konsumsi sesuai jumlah peserta. Fitur ini juga diberikan untuk pihak tamu yang mengikuti kegiatan di perusahaan. Fitur terakhir, adalah pengelolaan data master untuk admin. Fitur ini memberikan kemudahan dalam melakukan pengolahan data yang dibutuhkan seperti, data ruangan, data fasilitas ruangan, data pegawai dan data divisi. Admin dapat melakukan perubahan data, menambahkan data atau menghapus data terkait.
3. Pengujian aplikasi AMARR menggunakan metode *black-box* testing dilakukan pada setiap iterasi pengembangan untuk memastikan fungsionalitas aplikasi sesuai dengan spesifikasi yang diharapkan dan bebas dari kesalahan. Setiap pengujian melibatkan skenario tertentu untuk memverifikasi hasil yang ingin diperoleh. Selain itu, pengujian non-fungsional juga dilakukan, mencakup aspek seperti *pagination* pada setiap

tabel, *responsivitas* halaman saat diakses menggunakan perangkat *mobile*, dan tema tampilan gelap. Hasil pengujian menunjukkan bahwa aplikasi berfungsi dengan baik dalam aspek fungsional maupun non-fungsional, sehingga dapat digunakan tanpa masalah dan memiliki tampilan yang diharapkan. Oleh karena itu, Hasil pengujian menunjukkan bahwa aplikasi AMARR tidak hanya berfungsi dengan baik secara fungsional tetapi juga memenuhi kebutuhan non-fungsional, seperti *pagination*, *responsivitas*, dan tema tampilan gelap. Ini menandakan bahwa aplikasi siap digunakan dengan performa dan tampilan yang optimal di berbagai perangkat.

## 5.2 Saran

Aplikasi manajemen ruangan rapat berbasis web di PT. Tirta Asasta menggunakan *framework* laravel atau AMARR sudah berhasil dikembangkan. Hasil dari pengujian yang dilakukan juga sesuai dengan yang diharapkan. Meskipun begitu, masih terdapat kekurangan aplikasi yang dapat di kembangkan untuk memperbaiki kualitas AMARR. Beberapa saran kekurangan yang dapat dikembangkan lebih lanjut :

- a. Untuk peneliti yang akan datang guna mengembangkan AMARR dapat lebih lanjut mengembangkan fitur notifikasi penerimaan ajuan peminjaman ruangan. Seperti notifikasi masuk melalui *whatsapp* atau notifikasi yang secara langsung muncul di ponsel admin. Notifikasi untuk pegawai yang mengajukan ajuan ketika diterima juga dapat dijadikan bahan pengembangan lebih lanjut.
- b. Aplikasi AMARR dapat di kembangkan lebih lanjut agar menjadi aplikasi berbasis android.

## DAFTAR PUSTAKA

- [1] M. A. d. Y. S. Alfa, "Booking Meeting Management System Berbasis Framework Laravel," vol. III, no. 12, pp. 1-20, 2019.
- [2] R. D. Yahya, "Perbandingan Query Builder dan Eloquent pada Laravel," Medium, 28 April 2023. [Online]. Available: <https://medium.com/dot-intern/perbandingan-query-builder-dan-eloquent-pada-laravel-5492ef64e2d4>. [Diakses 7 Maret 2024].
- [3] M. D. Sadiyah, A. Sarbini dan S. , "Implementasi Fungsi Manajemen dalam Meningkatkan Pelayanan Kepada Calon Jamaah Haji," *MABRUR: Academic Journal of Hajj and Umra*, vol. 1, no. 2, pp. 29-50, 2022.
- [4] Bakri, "Pengertian Manajemen Menurut Para Ahli," PDAI - BAKRI, 15 Januari 2022. [Online]. Available: <https://bakri.uma.ac.id/pengertian-manajemen-menurut-para-ahli/>. [Diakses 8 Maret 2024].
- [5] S. Mariko, "APLIKASI WEBSITE BERBASIS HTML DAN JAVASCRIPT UNTUK MENYELESAIKAN FUNGSI INTEGRAL PADA MATA KULIAH KALKULUS," *Jurnal Inovasi Teknologi Pendidikan*, vol. 1, no. 6, pp. 80-91, 2019.
- [6] N. Batubara, A. dan S. Y. Prayogi, "PERANCANGAN APLIKASI ELARNING BERBASIS WEB DI SMP NEGERI 1 SAIPAR DOLOK HOLE," *SEMNAS TEK UISU 2021*, vol. 1, no. 1, pp. 138-145, 2021.
- [7] F. P. Nasution, R. O. Batubara dan M. I. Maulana, "Dasar Pengenalan HTML pada Desain Web," *PUBLIDIMAS*, vol. 2, no. 1, pp. 86-91, 2022.
- [8] A. Sahi, "APLIKASI TEST POTENSI AKADEMIK SELEKSI SARINGAN MASUK LP3I BERBASIS WEB ONLINE MENGGUNAKAN FRAMEWORK CODEIGNITER," *Jurnal Teknologi Informasi Dan Komunikasi*, vol. 7, no. 1, pp. 120-129, 2020.

- [9] D. D. J. TJ Sitinjak, M. dan J. Suwita , “ANALISA DAN PERANCANGAN SISTEM INFORMASI ADMINISTRASI KURSUS BAHASA INGGRIS PADA INTENSIVE ENGLISH COURSE DI CILEDUG TANGERANG,” *JURNAL IPSIKOM*, vol. 8, no. 1, 2020.
- [10] F. Evan, “Jenis-Jenis Database: Relational, NoSQL, dan NewSQL,” BINUS Higher Education, 6 Febuari 2023. [Online]. Available: <https://sis.binus.ac.id/2023/02/06/jenis-jenis-database-relational-nosql-dan-newsql/>. [Diakses 7 Maret 2024].
- [11] “PEMBUATAN E-COMMERCE PADA RAJA KOMPUTER MENGGUNAKAN BAHASA PEMROGRAMAN PHP DAN DATABASE MYSQL,” *Jurnal Media Infotama*, vol. 7, no. 1, pp. 54-66, 2021.
- [12] Y. K, “Pengertian MySQL, Fungsi, dan Cara Kerjanya (Lengkap),” NiagaHosterblog, 23 April 2022. [Online]. Available: <https://www.niagahoster.co.id/blog/mysql-adalah/>. [Diakses 7 Maret 2024].
- [13] “trends.builtwith.com,” BuiltWith, 10 April 2024. [Online]. Available: <https://trends.builtwith.com/framework/Laravel>. [Diakses 11 April 2024].
- [14] A. M. Sari, “Pengertian PHP, Sejarah, Sintak Dasar dan Kelebihannya,” FIKTI UMSU, 31 Mai 2023. [Online]. Available: <https://fikti.umsu.ac.id/pengertian-php-sejarah-sintak-dasar-dan-kelebihannya/>. [Diakses 7 Maret 2024].
- [15] F. A, “Apa Itu Framework? Pengertian, Fungsi, dan Contohnya,” Hostinger, 19 Desember 2022. [Online]. Available: <https://www.hostinger.co.id/tutorial/framework-adalah>. [Diakses 7 Maret 2024].
- [16] D. Ambriani, “RANCANG BANGUN REPOSITORY PUBLIKASI ILMIAH DOSEN BERBASIS WEB MENGGUNAKAN FRAMEWORK LARAVEL,” *Jurnal Manajemen Informatika*, vol. 10, no. 1, pp. 58-66, 2020.

- [17] V. Halim, "Laravel Framework," BINUS Higher Education, 5 April 2019. [Online]. Available: <https://sis.binus.ac.id/2019/04/05/laravel-framework/>. [Diakses 7 Maret 2024].
- [18] "Laravel," Laravel, [Online]. Available: <https://laravel.com/>. [Diakses 7 Maret 2024].
- [19] D. Intern, "Apa itu Activity Diagram? Beserta Pengertian, Tujuan, Komponen," Dicoding, 10 Maret 2021. [Online]. Available: <https://www.dicoding.com/blog/apa-itu-activity-diagram/>. [Diakses 11 Maret 2024].
- [20] M. L. A. Latukolan, A. Arwan dan M. T. Ananta, "Pengembangan Sistem Pemetaan Otomatis Entity Relationship Diagram Ke Dalam Database," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 3, no. 4, pp. 4058-4065, 2019.
- [21] M. A. Maksum, "Pengertian XAMPP, Fungsi, dan Cara Menggunakannya," dewaweb, 23 Mai 2022. [Online]. Available: <https://www.dewaweb.com/blog/apa-itu-xampp/>. [Diakses 7 Maret 2024].
- [22] M. S. Novendri, A. Saputra dan C. E. Firman, "APLIKASI INVENTARIS BARANG PADA MTS NURUL ISLAM DUMAI MENGGUNAKAN PHP DAN MYSQL," *JURNAL MANAJENEN DAN TEKNOLOGI INFORMASI*, vol. 10, no. 2, pp. 46-57, 2019.
- [23] E. Santi, "Kelebihan Visual Studio Code dan Cara Menggunakannya," IDwebhost, 6 Desember 2022. [Online]. Available: <https://idwebhost.com/blog/kelebihan-visual-studio-code-untuk-coding-fitur-dan-cara-menggunakannya/>. [Diakses 7 Maret 2024].
- [24] Y. D. Wijaya dan M. W. Astuti, "PENGUJIAN BLACKBOX SISTEM INFORMASI PENILAIAN KINERJA KARYAWAN PT INKA (PERSERO) BERBASIS EQUIVALENCE PARTITIONS," *Jurnal Digital Teknologi Informasi*, vol. 4, no. 1, pp. 22-26, 2021.

- [25] G. Taufiq, Y. Handrianto dan . S. , “Model Extreme Programming Untuk Rancang Bangun Sistem Informasi Penjadwalan Kuliah,” *Jurnal Infortech* , vol. 4, no. 2, pp. 190-199, 2022.
- [26] I. Carolina dan A. Supriyatna, “PENERAPAN METODE EXTREME PROGRAMMING DALAM PERANCANGAN APLIKASI PERHITUNGAN KUOTA SKS MENGAJAR DOSEN,” *Jurnal IKRA-ITH Informatika*, vol. 3, no. 1, pp. 106-113, 2019.
- [27] R. Sari, F. Hamidy dan S. , “SISTEM INFORMASI AKUNTANSI PERHITUNGAN HARGA POKOK PRODUKSI PADA KONVEKSI SJM BANDAR LAMPUNG,” *Jurnal Teknologi dan Sistem Informasi (JTSI)*, vol. 2, no. 1, pp. 65-73, 2021.



STT - NF



## LAMPIRAN

### Lampiran 1 Panduan Penggunaan Aplikasi

Berikut adalah panduan penggunaan aplikasi AMARR yang bertujuan memberikan gambaran dan pengetahuan kepada pengguna aplikasi.

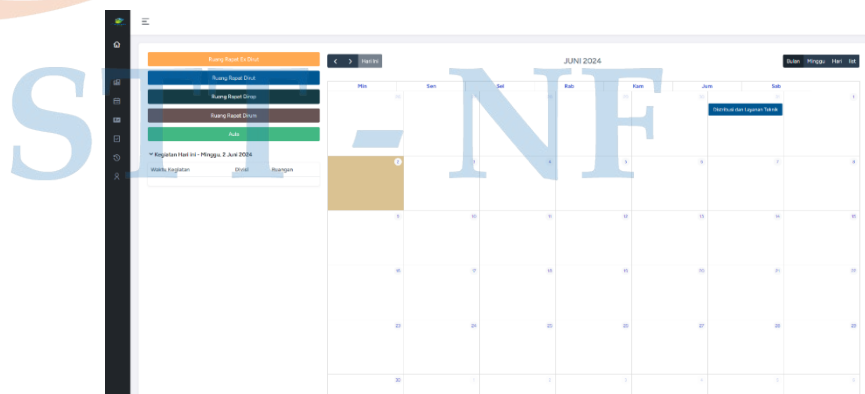
#### 1. Halaman *Login*



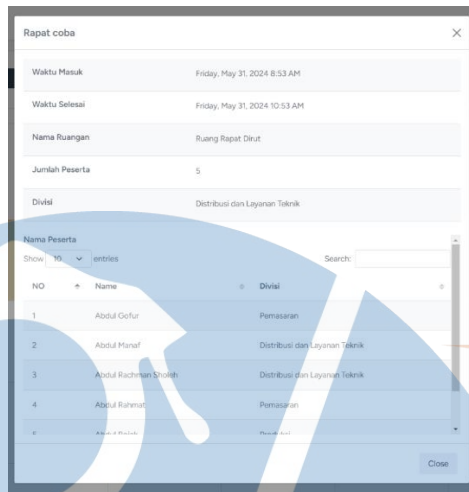
- Pada halaman *login* pengguna diharapkan mengisi *username* dan *password* yang sesuai dengan akun masing-masing
- Pengguna tamu dapat memilih halaman *login* sebagai tamu untuk absensi
- Terdapat informasi mengenai kegiatan hari ini pada gambar halaman *login*

#### 2. Halaman *Dashboard*

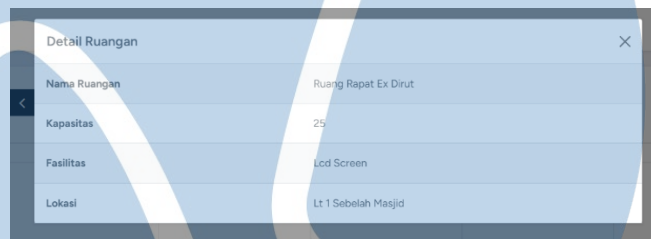
- Pada halaman *dashboard* pengguna dapat melihat jadwal peminjaman ruangan dengan format kalender



b. Pilih jadwal peminjaman tertentu untuk melihat detail peminjaman



c. Pilih nama ruangan tertentu untuk melihat detail suatu ruangan

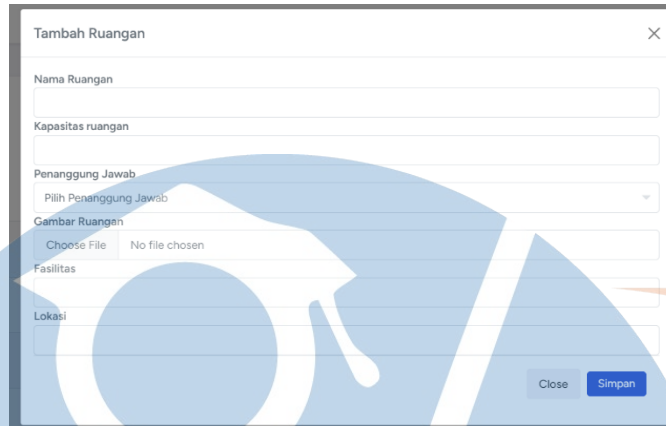


### 3. Halaman Data Master

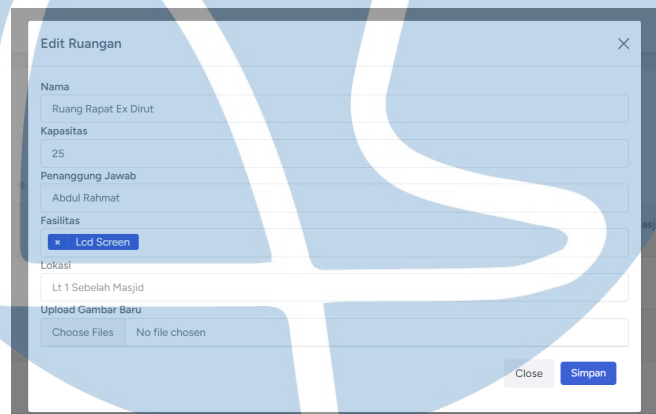


a. Halaman ini hanya dapat di akses oleh Super admin

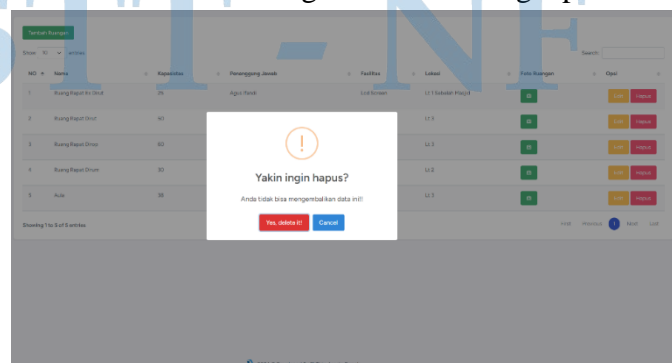
- b. Untuk menambahkan suatu data master, super admin dapat memilih “Tambah ruangan” dan akan muncul *form* menambah ruangan



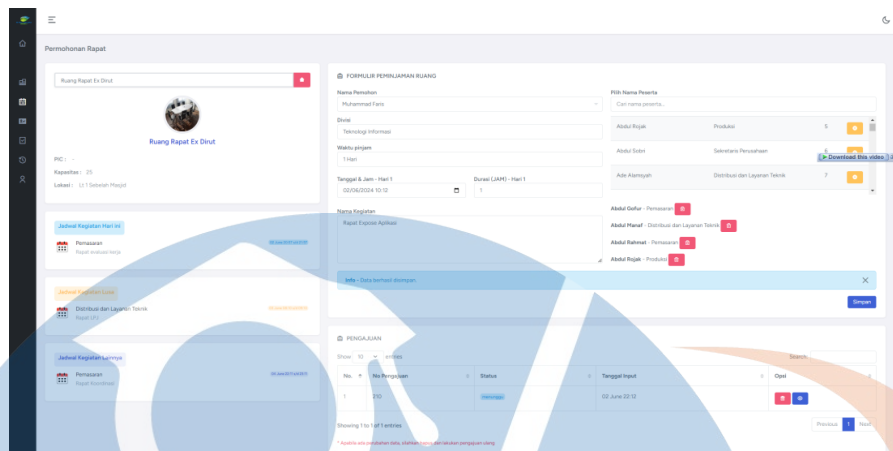
- c. Untuk mengubah data suatu ruangan super admin dapat memilih “edit” dan akan muncul *form* edit ruangan.



- d. Untuk menghapus suatu data, super admin dapat memilih “hapus” dan akan muncul dialog *box* untuk menghapus.





#### 4. Halaman Permohonan



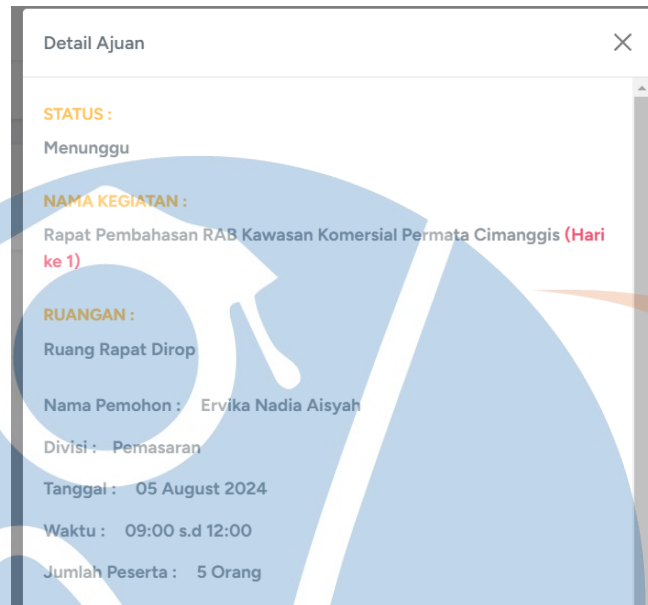
- Halaman permohonan dipergunakan oleh *user* pegawai untuk mengajukan peminjaman ruangan
- Pegawai memilih suatu ruangan yang ingin dipinjam



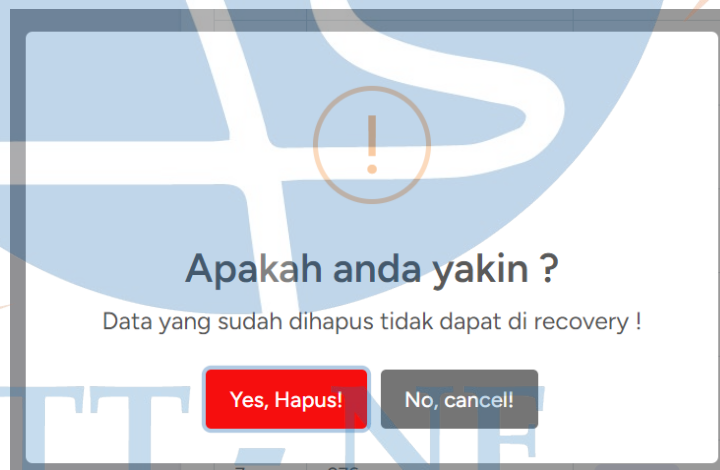
- Setelah itu mengisi formulir peminjaman ruang
- Setelah mengisi seluruh data, maka dapat memilih “ajukan” dan data pengajuan akan masuk di bagian pengajuan

No.	No Pengajuan	Status	Tanggal Input	Opsi
1	294	menunggu	31 July 09:15	 

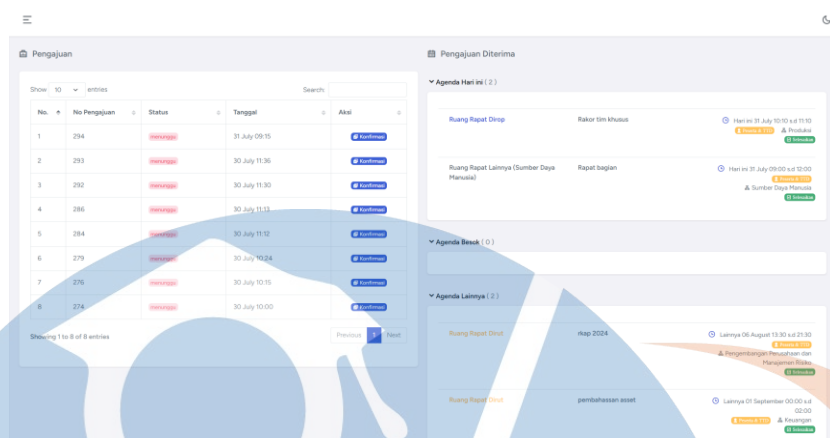
- e. Pengguna dapat melihat detail yang sudah di ajukan dengan memilih *icon* “mata” dan akan muncul modal detail ajuan



- f. Pengguna dapat membatalkan suatu ajuan dengan memilih *icon* “tempat sampah” dan akan muncul modal hapus.



## 5. Halaman Agenda

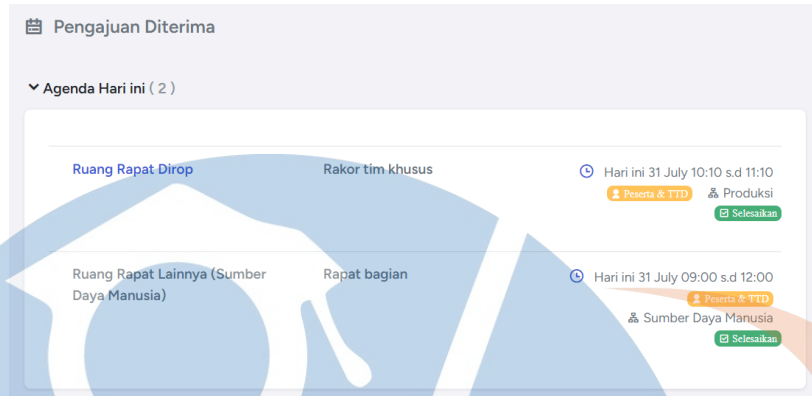


- Halaman agenda hanya dapat di akses oleh admin dan super admin
- Super Admin dapat meninjau seluruh pengajuan yang masuk
- Admin hanya dapat meninjau pengajuan yang terkait dengan ruangan admin.
- Untuk melakukan peninjauan, pengguna dapat memilih menu “konfirmasi” dan setelahnya akan muncul modal detail ajuan untuk di terima atau di tolak oleh admin

No.	No Pengajuan	Status	Tanggal	Aksi
1	294	menunggu	31 July 09:15	<a href="#">Konfirmasi</a>



- e. Jika di terima, maka pengajuan akan masuk ke bagian pengajuan diterima, dan admin mempersiapkan ruangan.



- f. Jika di tolak, admin akan melakukan konfirmasi kepada pegawai yang mengajukan.
- g. Kegiatan peminjaman yang sudah selesai, dapat admin selesaikan dengan memilih menu selesaikan yang terdapat di bagian pengajuan diterima



- h. Muncul modal selesaikan rapat dengan memilih jam mulai dan jam selesai yang sesuai.

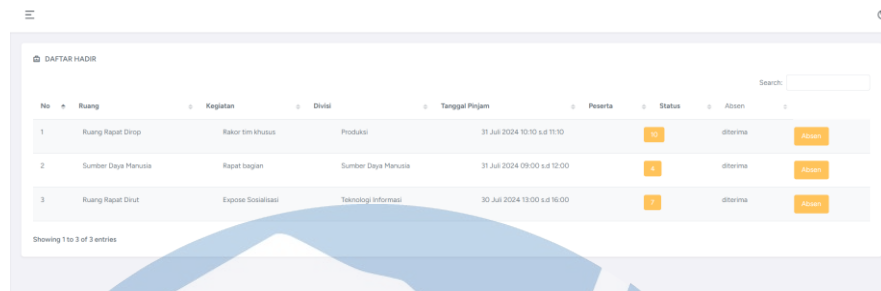
Selesaikan Rapat

Jam Mulai  
31/07/2024 10:10

Jam Selesai  
31/07/2024 11:10

Selesaikan Close

## 6. Halaman Daftar Hadir dan Absen



DAFTAR HADIR

No	Ruang	Kegiatan	Divisi	Tanggal Pinjam	Peserta	Status	Absen
1	Ruang Rapat Dipro	Rakor tim khusus	Produksi	31 Juli 2024 10:10 s.d 11:10	10	diterima	Absen
2	Sumber Daya Manusia	Rapat bagan	Sumber Daya Manusia	31 Juli 2024 09:00 s.d 12:00	4	diterima	Absen
3	Ruang Rapat Dipro	Expose Sosialisasi	Teknologi Informasi	30 Juli 2024 13:00 s.d 16:00	7	diterima	Absen

Showing 1 to 3 of 3 entries

- Halaman Daftar hadir berisikan rapat yang sudah diterima
- Halaman daftar hadir dapat di akses oleh pegawai dan admin
- Pegawai dapat memilih menu absen untuk berpindah ke halaman absen



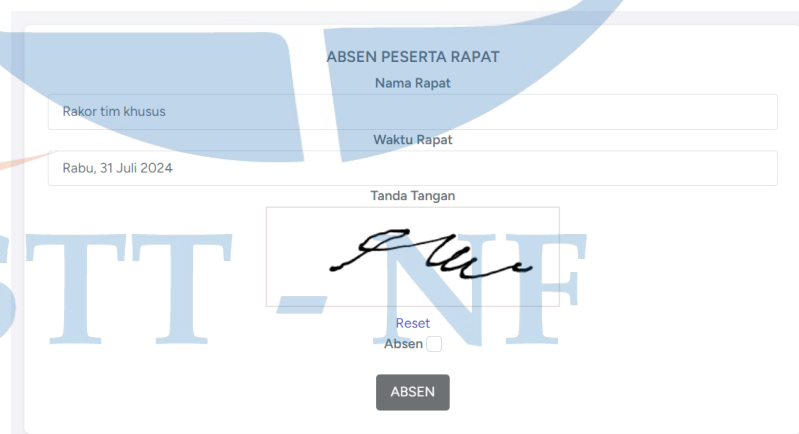
No	Ruang	Kegiatan	Divisi	Tanggal Pinjam	Peserta	Status	Absen
1	Ruang Rapat Dipro	Rakor tim khusus	Produksi	31 Juli 2024 10:10 s.d 11:10	10	diterima	Absen

- Pegawai memilih menu absen pada nama pegawai yang di inginkan



NO	Nama Peserta	Absen
1	Aimat, S.E	Absen


- Pegawai mengisi absen dengan memberikan tanda tangan di halaman absen



ABSEN PESERTA RAPAT

Nama Rapat  
Rakor tim khusus

Waktu Rapat  
Rabu, 31 Juli 2024

Tanda Tangan  


Reset Absen

ABSEN



- f. Pegawai harus melakukan centang pada kota absen agar menu absen dapat di pilih

Tanda Tangan



Reset

Absen

- g. Pegawai dapat membuat ulang tanda tangan dengan memilih menu “reset”
- h. Pegawai yang sudah absen tidak dapat melakukan absen dengan tulisan sudah absen pada bagian absen

NO	Nama Peserta	Absen
1	Aimat, S.E.	Sudah Absen

- i. Admin dapat melihat catatan absen dengan memilih menu jumlah peserta di bagian kolom peserta

Peserta

10

4

7

STT - NF

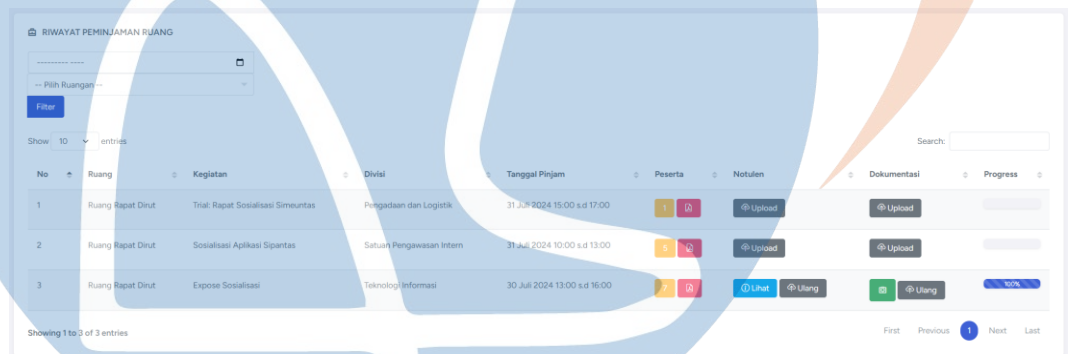
j. Muncul modal berisikan detail absensi peserta kegiatan



No.	Nama Pegawai	Divisi	Tanda Tangan	Waktu
1	Aimat, S.E.	Produksi		13:40 WIB
2	Ajie Prasetya Saputra	Distribusi dan Layanan Teknik		10:43 WIB
3	Akhmad Syafari	Teknologi Informasi		10:43 WIB

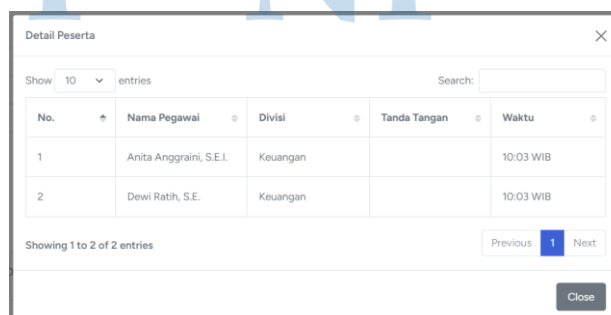
- k. Absensi dilakukan di awal setiap kegiatan untuk kebutuhan konsumsi apabila suatu rapat berjalan lebih dari 3 jam
- l. Apabila terdapat penambahan jumlah peserta di luar yang sudah didaftarkan, maka peserta tambahan dapat melakukan absensi di bagian absen tamu

## 7. Halaman Riwayat



No	Ruang	Kegiatan	Divisi	Tanggal Pinjam	Peserta	Notulen	Dokumentasi	Progress
1	Ruang Rapat Dirut	Trial Rapat Sosialisasi Simeuntas	Pengadaan dan Logistik	31 Juli 2024 15:00 s.d 17:00				
2	Ruang Rapat Dirut	Sosialisasi Aplikasi Sipantas	Satuan Pengawasan Intern	31 Juli 2024 10:00 s.d 13:00				
3	Ruang Rapat Dirut	Expose Sosialisasi	Teknologi Informasi	30 Juli 2024 13:00 s.d 16:00				

- a. Halaman riwayat berisikan semua data peminjaman ruangan
- b. Halaman riwayat dapat di akses oleh admin, super admin dan pegawai.
- c. Pengguna dapat melihat detail absen peserta dengan memilih menu angka jumlah peserta



No.	Nama Pegawai	Divisi	Tanda Tangan	Waktu
1	Anita Anggraini, S.E.I.	Keuangan		10:03 WIB
2	Dewi Rath, S.E.	Keuangan		10:03 WIB

- d. Pengguna dapat merekap data absensi dengan memilih menu berlogo “pdf” dan berkas akan terunduh

#### DAFTAR HADIR

Nama Kegiatan : Rapat Absen Online

Ruang Rapat Dirut

Kamis, 01 Agustus 2024 10:01 - Kamis, 01 Agustus 2024 14:01

NO	Nama	Divisi	Jabatan/ Instansi	Ttd	Waktu Absen
1	Hastori Prabu	Sumber Daya Manusia		Belum Absen	Rabu, 31 Juli 2024 23:08
2	Marisa Ghaisani	Sumber Daya Manusia		Belum Absen	Rabu, 31 Juli 2024 23:08
3	Sely Wulandary, S.E.	Sumber Daya Manusia		Belum Absen	Rabu, 31 Juli 2024 23:08
4	Try Sucl Amelia	Sumber Daya Manusia		Belum Absen	Rabu, 31 Juli 2024 23:08
5	Faradisa Athalla	Sumber Daya Manusia		Belum Absen	Rabu, 31 Juli 2024 23:08

Scan QR Rapat



- e. Pegawai yang sudah melakukan peminjaman ruangan dapat mengunggah dokumen notulen kegiatan dengan memilih menu “upload” pada bagian notulen

Upload Notulen

Silakan pilih file notulen yang akan diunggah.

Pilih File

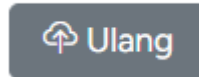
Choose File No file chosen

Close Upload

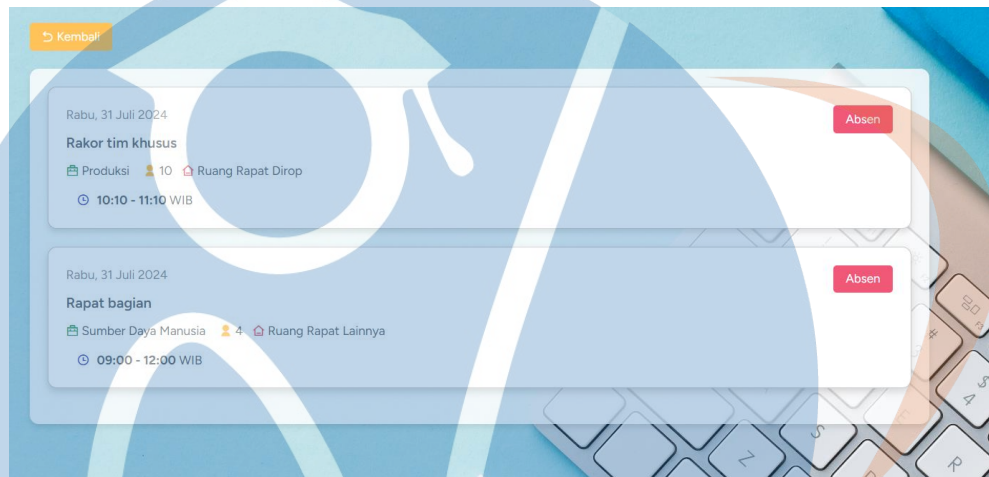
- f. Pengguna dapat melihat dokumen notulen yang sudah di unggah dengan memilih menu “lihat” pada bagian notulen

 Lihat

- g. Pengguna dapat mengunggah ulang dokumen dengan memilih menu “ulang” pada bagian notulen



## 8. Halaman Tamu

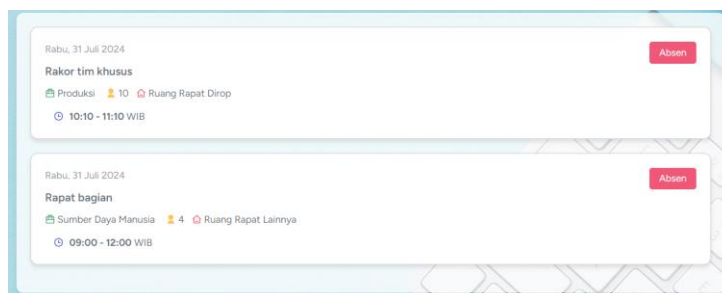


- Halaman tamu digunakan oleh tamu dari luar perusahaan yang mengikuti kegiatan di PT. Tirta Asasta Depok
- Halaman tamu juga digunakan jika terdapat tambahan peserta ketika kegiatan sedang berlangsung
- Halaman tamu dapat di akses dengan memilih menu “*login sebagai tamu*” di halaman *login*



Log in Sebagai Tamu

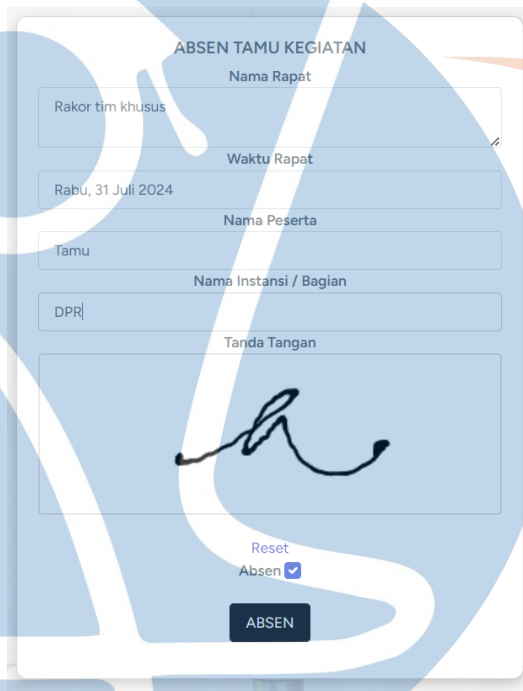
- Pada halaman tamu terdapat daftar kegiatan yang terdapat di hari aplikasi di akses



- e. Tamu memilih menu absen di kegiatan yang di ikuti

Absen

- f. Pada halaman absen tamu terdapat kolom nama dan divisi/instansi asal tamu untuk di isi



ABSEN TAMU KEGIATAN

Nama Rapat  
Rakor tim khusus

Waktu Rapat  
Rabu, 31 Juli 2024

Nama Peserta  
Tamu

Nama Instansi / Bagian  
DPR

Tanda Tangan

Reset  
Absen

ABSEN

- g. Dengan memilih menu absen maka jumlah peserta kegiatan akan bertambah.

STT - NF