

**RANCANG BANGUN APLIKASI PENGELOLAAN
GEDUNG SERBAGUNA BERBASIS WEB (STUDI
KASUS : MAKASSAU *BALLROOM*)**

SKRIPSI



**DZUL RAHMAT YUNUS
H131 16 314**

PROGRAM STUDI SISTEM INFORMASI

DEPARTEMEN MATEMATIKA

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM

UNIVERSITAS HASANUDDIN

MAKASSAR

AGUSTUS 2023

**RANCANG BANGUN APLIKASI PENGELOLAAN
GEDUNG SERBAGUNA BERBASIS WEB (STUDI
KASUS : MAKKASAU *BALLROOM*)**

SKRIPSI

Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana Sains pada
Program Studi Sistem Informasi Departemen Matematika Fakultas Matematika
dan Ilmu Pengetahuan Alam Universitas Hasanuddin Makassar

DZUL RAHMAT YUNUS

H13116314

PROGRAM STUDI SISTEM INFORMASI

DEPARTEMEN MATEMATIKA

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM

UNIVERSITAS HASANUDDIN

MAKASSAR

AGUSTUS 2023

HALAMAN PERNYATAAN KEONTETIKAN

Yang bertanda tangan dibawah ini :

Nama : DZUL RAHMAT YUNUS
NIM : H131 16 314
Program Studi : Sistem Informasi
Jenjang : S1

Menyatakan dengan ini bahwa karya tulisan saya berjudul

**RANCANG BANGUN APLIKASI PENGELOLAAN GEDUNG
SERBAGUNA BERBASIS WEB (STUDI KASUS : MAKKASAU
BALLROOM)**

adalah benar hasil karya saya sendiri bukan merupakan pengambil alihan tulisan orang lain dan belum pernah dipublikasikan dalam bentuk apapun.

Apabila dikemudian hari terbukti atau dapat dibuktikan bahwa sebagian atau keseluruhan skripsi ini merupakan hasil karya orang lain, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Makassar, 3 Agustus 2023



DZUL RAHMAT YUNUS
NIM. H131 16 314

**RANCANG BANGUN APLIKASI PENGELOLAAN GEDUNG
SERBAGUNA BERBASIS WEB (STUDI KASUS : MAKKASAU
BALLROOM)**

Disusun dan diajukan oleh


**DZUL RAHMAT YUNUS
H131 16 314**

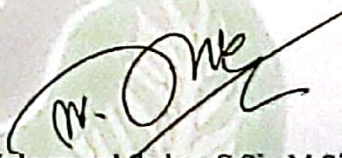
Telah dipertabankan di hadapan Panitia Ujian yang dibentuk dalam rangka
Penyelesaian Studi Program Sarjana pada Program Studi Sistem Informasi
Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Hasanuddin dan
dinyatakan telah memenuhi syarat kelulusan

Menyetujui,


Pembimbing Utama

Pembimbing Pendamping


Dr. Hendra, S.Si., M.Kom.
NIP. 197601022002121001


Muhammad Sadno, S.Si., M.Si
NIP. 199008162022043001

Ketua Program Studi


Dr. Hendra, S.Si., M.Kom.
NIP. 197601022002121001



HALAMAN PENGESAHAN

Skripsi ini diajukan oleh :

Nama : DZUL RAHMAT YUNUS
NIM : H131 16 314
Program Studi : Sistem Informasi
Judul Skripsi : RANCANG BANGUN APLIKASI
PENGELOLAAN GEDUNG SERBAGUNA
BERBASIS WEB (STUDI KASUS :
MAKKASAU BALLROOM)

Telah dipertahankan di depan Tim Penguji dan diterima sebagai bagian persyaratan untuk memperoleh gelar Sarjana Komputer pada Program Studi Sistem Informasi Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Hasanuddin.

DEWAN PENGUJI

		Tanda Tangan
1. Ketua	: Dr. Hendra, S.Si., M.Kom.	(.....)
2. Sekretaris	: Muhammad Sadno, S.Si., M.Si	(.....)
3. Anggota	: Dr. Khaeruddin, M.Sc	(.....)
4. Anggota	: A. Muh. Amil Sidik, S.Si., M.Si	(.....)

Ditetapkan di : Makassar

Tanggal : 3 Agustus 2023

KATA PENGANTAR

Puji syukur tanpa hentinya penulis panjatkan kehadiran Allah SWT atas limpahan nikmat rahmat dan hidayah-Nya yang tanpa henti sehingga penulis dapat menyelesaikan penyusunan skripsi yang berjudul “*RANCANG BANGUN APLIKASI PENGELOLAAN GEDUNG SERBAGUNA BERBASIS WEB (STUDI KASUS : MAKKASAU BALLROOM)*” yang disusun sebagai salah satu syarat untuk mencapai gelar Sarjana pada Program Studi Sistem Informasi Fakultas MIPA Universitas Hasanuddin. Shalawat serta salam tak lupa pula senantiasa penulis haturkan kepada junjungan Nabi Besar Muhammad SAW, sang suri tauladan terbaik yang telah membimbing kita dari zaman kegelapan menuju zaman yang terang benderang hingga saat ini.

Sembah sujud serta ucapan terima kasih tanpa henti kepada kedua orang tua kami yang tanpa pamrih senantiasa memberikan kasih sayang tulusnya. Karya ini kami persembahkan kepada keduanya sebagai wujud bakti kepada mereka. Dalam proses penyelesaian penelitian ini, penulis menyadari bahwa bantuan, bimbingan serta dukungan dari berbagai pihak juga mempunyai andil besar dalam penyelesaiannya. Oleh karena itu, penulis juga mengucapkan banyak terima kasih kepada :

1. Kedua Orang Tua dan Seluruh Keluarga Besar yang selalu memberikan motivasi selama proses penyelesaian studi di Fakultas MIPA Universitas Hasanuddin.
2. Rektor Universitas Hasanuddin, Bapak **Prof. Dr. Ir. Jamaluddin Jompa, M.Sc.** beserta jajarannya.
3. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam, Bapak **Dr. Eng. Amiruddin** beserta jajarannya.
4. Ketua Departemen Matematika FMIPA, Bapak **Prof. Dr. Nurdin, S.Si., M.Si.** beserta jajarannya.
5. Ketua Program Studi Sistem Informasi FMIPA Unhas, Bapak **Dr. Hendra, S.Si., M.Kom.** sekaligus menjadi dosen pembimbing akademik dan pembimbing utama yang senantiasa memberi motivasi dorongan dan

masukan dalam hal akademik serta ide, gagasan dan solusi dalam proses penyelesaian skripsi ini.

6. Bapak **Muhammad Sadno, S.SI., M.Si.** sebagai pembimbing pendamping yang senantiasa memberikan saran dan masukan kepada penulis selama proses penyelesaian skripsi ini.
7. Bapak **Dr. Khaeruddin, M.Sc.** sebagai anggota tim penguji yang senantiasa memberikan saran dan kritik yang membangun pada penelitian yang telah dilakukan oleh penulis.
8. Bapak **A. Muh. Amil Sidik, S.Si., M.Si.** juga sebagai anggota tim penguji atas saran dan kritik yang membangun pada penelitian yang telah dilakukan oleh penulis.
9. Seluruh Bapak dan Ibu dosen FMIPA Universitas Hasanuddin yang telah mendidik dan memberikan ilmunya sehingga penulis mampu menyelesaikan program sarjana. Serta para staff yang telah membantu dalam pengurusan berkas administrasi.
10. Pihak **Makkasau Hotel dan Ballroom** yang telah memberikan bantuan dalam kelancaran proses penyelesaian Tugas Akhir Skripsi ini.
11. Keluarga besar **Ilmu Komputer 2016** yang senantiasa membantu penulis selama menjalani pendidikan.
12. Saudara-saudara sepenanggungan, seperjuangan **Zero Café (Piki, Panji, Putri, Piang)** yang telah memberikan dukungan serta meluangkan waktu berbagi suka-duka perkuliahan selama menuntut ilmu. Sampai bertemu di puncak teman-teman.
13. Keluarga besar **La Tea Ri Duni 2016** yang senantiasa menjadi rumah kembali penulis ketika resah dengan dunia rantau. Jangan lupa palekko-palekkonya tiap minggu.
14. Kanda-kanda dan dinda-dinda **PMB-UH Latenritatta** yang telah membersamai penulis selama menuntut ilmu. **“Getteng Lempu Ada Tongeng, Warani Temappasilaingeng”**.
15. Semua pihak yang telah banyak berpartisipasi dan membantu, baik secara langsung maupun tidak langsung dalam penyusunan skripsi yang tidak bisa penulis sebutkan satu per satu. Kalian luar biasa.

Kami sangat menyadari masih terdapat banyak kekurangan dalam tugas akhir ini yang terjadi diluar kesengajaan kami sehingga sangat mengharapkan adanya kritikan dan saran yang bersifat membangun dari semua pihak. Akhir kata, semoga tugas akhir ini dapat bermanfaat bagi semua pihak yang membutuhkan.

Makassar, 3 Agustus 2023

DZUL RAHMAT YUNUS
NIM. H131 16 314

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

Sebagai civitas akademik Universitas Hasanuddin, saya yang bertanda tangan dibawah ini :

Nama : DZUL RAHMAT YUNUS
NIM : H131 16 314
Program Studi : Sistem Informasi
Departemen : Matematika
Fakultas : Matematika dan Ilmu Pengetahuan Alam
Jenis Karya : Skripsi

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Hasanuddin **Hak Predikator Royalti Noneksklusif** (*Non-exclusive Royalty-Free Right*) atas tugas akhir saya yang berjudul :

**“RANCANG BANGUN APLIKASI PENGELOLAAN GEDUNG
SERBAGUNA BERBASIS WEB (STUDI KASUS : MAKASSAU
BALLROOM)”**

beserta perangkat yang ada (jika diperlukan). Terkait dengan hal diatas, maka pihak Universitas Hasanuddin berhak menyimpan, mengalih-media/format-kan, mengolah 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 surat pernyataan ini saya buat dengan sebenarnya.

Dibuat di Makassar pada 3 Agustus 2023

Yang menyatakan

(DZUL RAHMAT YUNUS)

ABSTRAK

Pengelolaan gedung yang dilakukan di Makkasau *Ballroom* yang masih menggunakan cara manual dengan pencatatan kedalam buku besar dapat memungkinkan terjadinya kesalahan dalam pencatatan dan menyulitkan melakukan pencarian data. Hal ini tentu dianggap kurang efisien sehingga membutuhkan sebuah aplikasi yang dapat membantu mengatasi masalah tersebut. Sistem ini dikembangkan menggunakan metode *waterfall* dengan menerapkan MVC (*Model-View-Controller*) yang kemudian diimplementasikan ke dalam *framework* Laravel. Pengujian sistem dilakukan dengan menggunakan teknik pengujian *Blackbox* untuk mengetahui *output* dari sistem yang telah dikembangkan serta pengujian skala Likert untuk mengetahui aspek *usability* dari sistem yang ada. Hasil pengujian dengan metode pengujian *Blackbox* menunjukkan hasil yang sesuai dengan yang diharapkan. Sedangkan untuk pengujian aspek *usability* menggunakan metode pengujian skala Likert menunjukkan nilai persentasi *usability* yang sangat baik. Dengan dikembangkannya aplikasi pengelolaan ini dapat membantu dan meningkatkan efisiensi pengelolaan gedung serbaguna Makkasau *Ballroom*.

Kata Kunci : Rancang Bangun, Aplikasi, Gedung Serbaguna, Metode *Waterfall*, Laravel, MVC, *Blackbox*, *Usability*

ABSTRACT

The management of Makkasau Ballroom, which currently relies on manual methods of recording in large ledgers, may lead to errors in documentation and pose challenges in data retrieval. This inefficiency necessitates the development of an application to address these issues. The system was developed using the waterfall method and implemented within the Laravel framework, employing the Model-View-Controller (MVC) approach. The system was tested using the Blackbox testing technique to evaluate the system's output, as well as Likert scale testing to assess its usability. The Blackbox testing results aligned with expectations, while the usability testing using the Likert scale demonstrated a high percentage of usability. The development of this management application is expected to greatly assist and enhance the efficiency of managing the multipurpose Makkasau Ballroom facility.

Keyword : *Design and Development, Application, Multipurpose Building, Waterfall Method, Laravel, MVC, Blackbox, Usability.*

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PERNYATAAN KEOTENTIKAN	ii
HALAMAN PERSETUJUAN PEMBIMBING	iii
HALAMAN PENGESAHAN.....	iv
KATA PENGANTAR.....	v
PERSETUJUAN PUBLIKASI KARYA ILMIAH	viii
ABSTRAK	ix
<i>ABSTRACT</i>	x
DAFTAR ISI.....	xi
DAFTAR TABEL	xiv
DAFTAR GAMBAR.....	xv
DAFTAR LAMPIRAN	xvii
BAB I PENDAHULUAN.....	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	2
1.3. Batasan Masalah.....	2
1.4. Tujuan Penelitian.....	3
1.5. Manfaat Penelitian	3
BAB II TINJAUAN PUSTAKA.....	4
2.1. Landasan Teori	4
2.1.1 Pengertian Aplikasi	4
2.1.2 Aplikasi Berbasis Web	4
2.1.3 UML (<i>Unified Modeling Language</i>) Diagram.....	5
2.1.4 <i>Entity Relational Diagram</i> (ERD)	9

2.1.5	<i>Model-View-Controller (MVC)</i>	9
2.1.6	<i>Basis Data (Database)</i>	11
2.1.7	<i>Framework Laravel</i>	12
2.1.8	<i>User Interface dan User Experience</i>	13
2.1.9	<i>Metode Pengujian</i>	14
2.2.	<i>Penelitian Terkait</i>	17
2.2.1	<i>Perancangan Sistem Informasi Penyewaan Gedung Pada Gedung Balai Komando Kopassus Berbasis Web</i>	17
2.2.2	<i>Sistem Informasi Penyewaan Gedung Serbaguna Berbasis Web Di HKBP Bandung Barat</i>	17
2.2.3	<i>Aplikasi Pengelolaan Sewa Gedung Kantor Berbasis Web Pada PT. SARINAH</i>	18
BAB III METODE PENELITIAN		19
3.1	<i>Waktu dan Lokasi Penelitian</i>	19
3.2	<i>Metode Pengumpulan Data</i>	19
3.3	<i>Metode Pengembangan Sistem</i>	19
3.4	<i>Metode Pengembangan User Interface dan User Experience</i>	21
3.5	<i>Rancangan Sistem</i>	22
3.5.1	<i>Use Case Diagram</i>	22
3.5.2	<i>Tampilan Sistem</i>	23
3.6	<i>Instrumen Penelitian</i>	27
BAB IV HASIL DAN PEMBAHASAN		29
4.1.	<i>Hasil Pengembangan Sistem</i>	29
4.1.1	<i>Requirement Analysis</i>	29
4.1.2	<i>System and Software Design</i>	30
4.1.3	<i>Implementation</i>	44
4.1.4	<i>Integration and System Testing</i>	49
4.1.5	<i>Operation and Maintenance</i>	51
BAB V KESIMPULAN DAN SARAN		52

5.1.	Kesimpulan	52
5.2.	Saran.....	53
DAFTAR PUSTAKA.....		54
LAMPIRAN.....		57
7.1.	Pengujian <i>Blackbox</i> Pada Fungsi <i>Login</i> dan <i>Logout</i>	57
7.2.	Pengujian <i>Blackbox</i> Fungsi Pemesanan.....	57
7.3.	Pengujian <i>Blackbox</i> Fungsi Inventaris.....	58

DAFTAR TABEL

Tabel 2.1 Hubungan Antar <i>Class</i>	7
Tabel 2.2 Simbol-Simbol <i>Activity Diagram</i>	8
Tabel 2. 3 Interpretasi Skor	16
Tabel 4. 1 Tabel <i>Users</i>	35
Tabel 4. 2 Tabel <i>Orders</i>	36
Tabel 4. 3 Tabel Kategori	37
Tabel 4. 4 Tabel Inventaris	37
Tabel 4. 5 Permasalahan yang dihadapi oleh pengguna.....	38
Tabel 4. 6 Hasil Pengujian <i>Usability Testing</i>	50
Tabel 7. 1 Pengujian <i>Blackbox</i> Fungsi <i>Login</i> dan <i>Logout</i>	57
Tabel 7. 2 Pengujian <i>Blackbox</i> Fungsi Pemesanan	57
Tabel 7. 3 Pengujian <i>Blackbox</i> Fungsi Inventaris	58

DAFTAR GAMBAR

Gambar 2.1 Simbol-Simbol <i>Use Case Diagram</i>	5
Gambar 2.2 Komponen Penyusun <i>Class Diagram</i>	7
Gambar 2.3 Ilustrasi <i>Model View Controller</i>	10
Gambar 2.4 Ilustrasi Pengujian <i>Black Box</i>	14
Gambar 3.1 Metode <i>Waterfall</i>	21
Gambar 3.2 <i>Use Case Diagram</i> Admin	22
Gambar 3.3 <i>Use Case Diagram</i> Pelanggan	23
Gambar 3.4 Halaman Depan Admin	24
Gambar 3.5 Halaman Pendaftaran Akun Baru.....	24
Gambar 3.6 Halaman Reservasi Gedung	25
Gambar 3.7 Halaman Jadwal Reservasi Gedung	25
Gambar 3.8 Halaman Daftar Fasilitas Gedung	26
Gambar 3.9 Halaman Pembayaran Pelanggan.....	26
Gambar 3.10 Halaman Pengelolaan Pembayaran	27
Gambar 4. 1 <i>Use Case Diagram</i> Admin	30
Gambar 4. 2 <i>Use Case Diagram</i> Pelanggan	31
Gambar 4. 3 <i>Activity Diagram</i> Proses Login	32
Gambar 4. 4 <i>Activity Diagram</i> Penerimaan Pesanan	33
Gambar 4. 5 <i>Activity Diagram</i> Pengelolaan Data Pesanan	34
Gambar 4. 6 <i>Activity Diagram</i> Pengelolaan Data Inventaris	35
Gambar 4. 7 <i>Wireframe</i> Halaman Login	40
Gambar 4. 8 <i>Wireframe</i> Halaman <i>Dashboard</i>	40
Gambar 4. 9 <i>Wireframe</i> Halaman Data Pesanan	41
Gambar 4. 10 <i>Wireframe</i> Halaman Data Inventaris	41
Gambar 4. 11 <i>Wireframe</i> Halaman <i>Home</i>	42
Gambar 4. 12 <i>Wireframe</i> Halaman Galeri.....	42
Gambar 4. 13 <i>Wireframe</i> Halaman Jadwal Acara	43
Gambar 4. 14 <i>Wireframe</i> Halaman Paket Acara	43
Gambar 4. 15 <i>Wireframe</i> Halaman Tentang Kami	44
Gambar 4. 16 Halaman <i>Home</i>	45
Gambar 4. 17 Halaman Galeri	45

Gambar 4. 18 Halaman Jadwal	46
Gambar 4. 19 Halaman Paket Acara	47
Gambar 4. 20 Halaman Tentang Kami.....	47
Gambar 4. 21 Halaman <i>Login</i>	48
Gambar 4. 22 Halaman <i>Dashboard</i>	48
Gambar 4. 23 Halaman Pesanan	49
Gambar 4. 24 Halaman Data Inventaris.....	49

DAFTAR LAMPIRAN

Pengujian <i>Blackbox</i> Fungsi <i>Login</i> dan <i>Logout</i>	57
Pengujian <i>Blackbox</i> Fungsi Pemesanan	57
Pengujian <i>Blackbox</i> Fungsi Inventaris	58

BAB I

PENDAHULUAN

1.1. Latar Belakang

Perkembangan teknologi yang melaju dengan sangat pesat memberikan dampak yang signifikan di berbagai bidang, termasuk bidang teknologi dan informasi. Semakin cepat perkembangan teknologi, maka akan semakin adaptif pengelolaan sistem informasi yang digunakan. Teknologi yang berkembang ini dapat digunakan untuk manajemen data dan pencarian informasi di bidang pendidikan, industri, kesehatan dan bisnis serta menyederhanakan proses manajemen (Paimasrul, 2019).

Semakin cepat perkembangan teknologi yang terjadi maka semakin dibutuhkan pula penyesuaian dalam pemanfaatannya tidak terkecuali dalam bidang pengelolaan sebuah gedung serbaguna. Dengan perkembangan teknologi yang sedemikian rupa, sayangnya masih banyak ditemukan sistem pengelolaan gedung dengan cara manual. Hal ini dapat menyebabkan berbagai masalah seperti kesalahan dalam pendataan penyewaan gedung, kekeliruan dalam pendataan pembayaran gedung hingga tercecernya berbagai data penting berkaitan dengan pengelolaan gedung. Salah satu contoh sistem pengelolaan gedung yang masih dilakukan secara manual adalah yang terjadi di Makkasau *Ballroom*.

Makkasau *Ballroom* merupakan sebuah gedung serbaguna yang berlokasi di Jalan MT. Haryono Nomor 38, Kecamatan Tanete Riattang Barat, Kabupaten Bone. Selama ini sistem pengelolaan Makkasau *Ballroom* masih dilakukan secara manual dengan cara pencatatan ke dalam buku. Sistem yang masih manual tersebut dapat memungkinkan terjadinya kesalahan dalam pencatatan yang berkaitan dengan penyewaan gedung dan dapat menyulitkan dalam melakukan pencarian jika suatu waktu diperlukan.

Untuk mengatasi masalah yang telah diuraikan diatas, maka penulis menganggap dibutuhkan suatu aplikasi berbasis manajemen pengelolaan yang lebih terintegrasi. Hal ini menjadi faktor pendukung bagi penulis untuk melakukan

penelitian yang berjudul “**Rancang Bangun Aplikasi Pengelolaan Gedung Serbaguna Berbasis Web (Studi Kasus : Makkasau *Ballroom*)**”.

Aplikasi ini menggunakan sistem komputerisasi yang meliputi pendataan penyewaan gedung, penjadwalan penyewaan gedung serta pembayaran yang nantinya dapat memudahkan dalam proses pengelolaan gedung. Aplikasi ini diharapkan dapat memperbaiki sistem pengelolaan gedung serta dapat meningkatkan efisiensi kerja dan pelayanan dari Makkasau *Ballroom*.

1.2. Rumusan Masalah

Berdasarkan yang telah diuraikan pada latar belakang di atas, maka dapat dirumuskan bahwa masalah yang melatar belakangi penelitian ini adalah kurang efisiennya pengelolaan penyewaan gedung karena masih dilakukan secara manual melalui pencatatan pada buku. Berdasarkan penjelasan diatas, maka rumusan masalah pada penelitian ini yaitu :

1. Bagaimana merancang dan membangun aplikasi pengelolaan yang dapat membantu proses pengelolaan gedung serbaguna Makkasau *Ballroom* ?
2. Implementasi seperti apa yang diperlukan dalam merancang dan membangun aplikasi pengelolaan gedung serbaguna Makkasau *Ballroom* ?
3. Apa saja tahapan pengujian yang dapat dilakukan untuk mengetahui efektifitas dari aplikasi yang telah dibangun?
4. Bagaimana hasil implementasi aplikasi ini dalam upaya mengoptimalkan pelayanan di gedung serbaguna Makkasau *Ballroom* ?

1.3. Batasan Masalah

Batasan masalah berfungsi agar penelitian ini dapat memberikan pemahaman yang jelas dan terarah terhadap penelitian yang akan dilakukan serta agar pembahasan yang ada tidak melenceng dari pokok permasalahan. Adapun batasan masalah pada penelitian ini yaitu :

1. Pembuatan *database* Aplikasi Pengelolaan Gedung Serbaguna Makkasau *Ballroom* menggunakan bahasa pemrograman PHP versi 7.4.29, *framework* Laravel versi 8 dan pengolah *database MySQL* versi 5.2.0.

2. Fitur pengelolaan yang ada pada aplikasi hanya bisa dilakukan oleh admin, yaitu pihak Makkasau *Ballroom*.
3. Penyewa memiliki akses untuk melihat jadwal gedung, memesan gedung dan fasilitas yang dibutuhkan.

1.4. Tujuan Penelitian

Berdasarkan latar belakang serta rumusan masalah yang telah diuraikan diatas, maka penelitian ini bertujuan untuk :

1. Merancang dan membangun aplikasi pengelolaan gedung serbaguna Makkasau *Ballroom*.
2. Mengetahui hasil implementasi dari aplikasi pengelolaan gedung serbaguna Makkasau *Ballroom*.
3. Mengetahui hasil pengujian aplikasi yang telah dibuat.
4. Mengoptimalkan pengelolaan gedung serbaguna Makkasau *Ballroom* agar lebih efisien dan cepat melalui aplikasi berbasis web.

1.5. Manfaat Penelitian

Adapun manfaat dari penelitian ini yaitu :

1. Menambah wawasan pengetahuan dalam pembuatan aplikasi berbasis web.
2. Mengetahui tingkat efektivitas dari aplikasi yang telah dibuat.
3. Mempermudah alur informasi gedung dan pemesanan gedung bagi penyewa.
4. Mempermudah dan mempercepat pelayanan dari pihak Makkasau *Ballroom* sehingga dapat meningkatkan efisiensi kerja dan kualitas layanan.

BAB II

TINJAUAN PUSTAKA

2.1. Landasan Teori

2.1.1 Pengertian Aplikasi

Aplikasi adalah program siap pakai yang dapat digunakan untuk menjalankan perintah dari pengguna dengan tujuan mendapatkan hasil yang lebih akurat tergantung pada tujuan pembuatan aplikasi tersebut. Aplikasi juga dapat diartikan penggunaan salah satu teknik pengolahan data untuk memecahkan suatu masalah berdasarkan perhitungan yang diinginkan atau diharapkan. (Hasan dkk., 2014).

Menurut Asropudin yang dikutip oleh Nursyanti menjelaskan bahwa aplikasi adalah perangkat lunak yang dibuat oleh individu maupun kelompok dengan tujuan melaksanakan tugas-tugas tertentu, misalnya Microsoft Word, Microsoft Excel (Nursyanti dkk., 2019).

2.1.2 Aplikasi Berbasis Web

Website adalah suatu kumpulan-kumpulan halaman yang menampilkan berbagai macam informasi teks, data, gambar, video maupun gabungan dari semuanya bersifat statis dan dinamis.

Web merupakan suatu sistem yang berkaitan dengan pengolahan dokumen dan digunakan sebagai media untuk menampilkan teks, gambar, multimedia dan lainnya pada jaringan internet (Nursyanti dkk., 2019).

Aplikasi berbasis web adalah aplikasi yang dapat dijalankan melalui web browser. Aplikasi berbasis web ini juga bisa menggunakan internet maupun intranet dan tidak tergantung pada sistem operasi yang digunakan (Rizky, 2010). Aplikasi berbasis web ini mempunyai beberapa kelebihan yaitu :

- a. Aplikasi dapat dijalankan di berbagai sistem operasi, seperti : Windows, Linux dan Mac Os.
- b. Untuk menjalankan aplikasi di banyak komputer tidak memerlukan pemasangan aplikasi di setiap komputer, cukup dengan menyalin kode

programnya ke server atau salah satu komputer. Untuk komputer lain yang ingin menjalankan program cukup membuka alamat host server dimana program disimpan melalui browser.

- c. Aplikasi dapat dijalankan dari jarak jauh dengan menggunakan koneksi internet (Tonggiroh dkk., 2017).

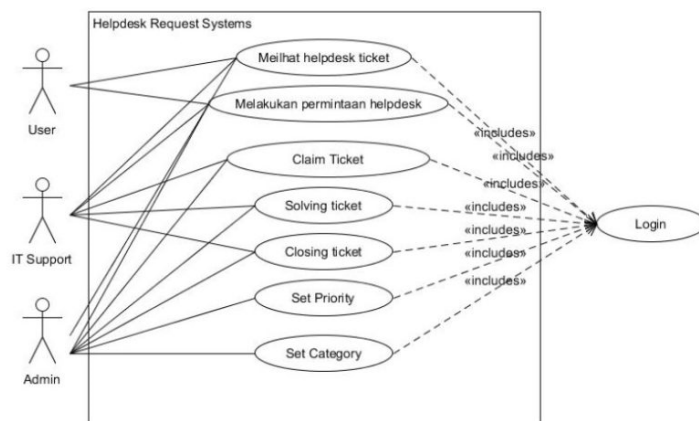
2.1.3 UML (*Unified Modeling Language*) Diagram

Unified Modeling Language (UML) adalah suatu bahasa yang digunakan untuk menentukan, memvisualisasikan, membangun dan mendokumentasikan suatu sistem informasi. UML dikembangkan sebagai suatu alat untuk analisis dan desain berorientasi objek oleh Grady Booch, Jim Rumbaugh dan Ivar Jacobson (Rama, 2008).

Dalam UML terdapat beberapa macam diagram yang digunakan untuk memodelkan aplikasi perangkat lunak berorientasi objek, yaitu :

a. *Use Case Diagram*

Use case diagram adalah salah satu jenis diagram dari *Unified Modeling Language* (UML) yang menggambarkan bagaimana interaksi yang terjadi antara sistem dan aktor. Diagram ini juga dapat menjadi gambaran tentang bagaimana *interface* dari sebuah sistem bekerja dan dapat mengidentifikasi tipe interaksi yang terjadi antara pengguna dan sistem. Dalam *use case* diagram ini ada beberapa simbol yang digunakan, simbol ini dapat dilihat pada gambar 2.1.



Gambar 2.1 Simbol-Simbol Use Case Diagram

Adapun penjelasan dari setiap simbol pada *use case* diagram pada gambar 2.1 adalah sebagai berikut :

- 1) **Actor**, menjelaskan siapa yang berinteraksi dengan sistem. Actor akan memberikan informasi kepada sistem, serta menerima informasi dari sistem.
- 2) **System**, menyatakan sistem dalam relasi dengan aktor-aktor yang menggunakannya dan fitur-fitur yang harus disediakan.
- 3) **Use Case**, merupakan gambaran fungsional dari sebuah sistem. *Use case* juga menjelaskan tentang tindakan/aksi yang dilakukan oleh aktor.
- 4) **Include**, menunjukkan bahwa suatu *use case* seluruhnya merupakan fungsionalitas dari *use case* lainnya.
- 5) **Extend**, menunjukkan bahwa suatu *use case* merupakan tambahan fungsional dari *use case* lainnya jika suatu kondisi terpenuhi.

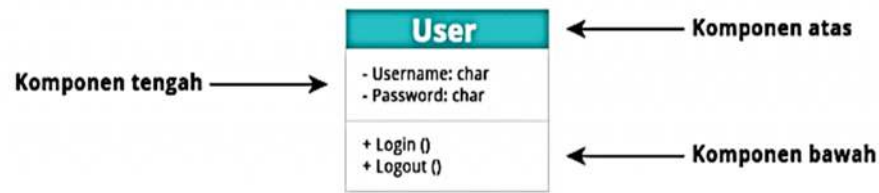
b. **Class Diagram**

Class diagram merupakan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain suatu sistem, juga memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem (Hendini, 2016).

Menurut Ambler (2005), *class diagram* adalah sebuah diagram yang menampilkan hubungan antar *class* yang di dalamnya terdapat atribut dan fungsi dari suatu objek.

Class diagram mempunyai mempunyai tiga komponen penyusun yaitu :

- 1) **Komponen atas**, berisikan nama *class*. Setiap *class* pasti memiliki nama yang berbeda-beda.
- 2) **Komponen tengah**, berisikan atribut *class*. Komponen ini digunakan untuk menjelaskan nilai dari suatu kelas. Atribut dalam kelas ini dituliskan dengan memasukkan tipe nilai.
- 3) **Komponen bawah**, menyertakan operasi yang ditampilkan dalam bentuk gambar. Operasi ini menggambarkan bagaimana suatu *class* berinteraksi dengan data.



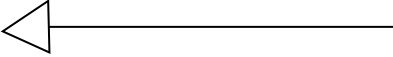
Gambar 2.2 Komponen Penyusun Class Diagram

Sumber: (Ambler, 2005)

Dalam *class diagram* ini terdapat pula hubungan antar *class*. Hubungan tersebut dapat dilihat pada tabel 2.1 berikut :

Tabel 2.1 Hubungan Antar Class

Penjelasan	Lambang
Asosiasi , yaitu hubungan statis kelas. Umumnya menggambarkan <i>class</i> yang memiliki atribut berupa <i>class</i> lain atau <i>class</i> yang harus mengetahui adanya <i>class</i> lain. Panah <i>navigability</i> menunjukkan arah <i>query</i> antar <i>class</i> .	
Agregasi , yaitu hubungan yang menyatakan bagian (“ terdiri atas..”) atau biasa disebut relasi mempunyai sebuah	
Composition , yaitu sebuah kelas tidak bisa berdiri sendiri dan harus merupakan bagian dari kelas yang lain, maka kelas tersebut mempunyai relasi komposisi.	
Dependency , yaitu hubungan yang menunjukkan operasi pada suatu kelas yang lain	




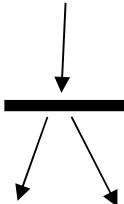
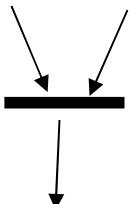
<p>Generalisasi, yaitu hubungan hierarki antar kelas. Kelas dapat diturunkan dari kelas lain dan mewarisi semua atribut dan metode kelas asalnya dan menambahkan fungsionalitas baru, sehingga baru disebut anak dari kelas yang diwarisinya.</p>	
--	--

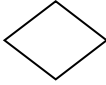
Sumber : (Booch dkk., 1999)

c. Activity Diagram

Activity diagram menggambarkan *workflow* (aliran kerja) atau aktivitas yang terjadi dalam sebuah sistem (Hendini, 2016). Simbol-simbol yang digunakan dalam *activity diagram* yaitu :

Tabel 2.2 Simbol-Simbol Activity Diagram

Gambar	Keterangan
	<p>Start point, merupakan penanda awal aktivitas.</p>
	<p>End point, merupakan penanda untuk akhir dari aktivitas.</p>
	<p>Activities, menggambarkan suatu proses yang terjadi.</p>
	<p>Fork/percabangan, digunakan untuk menunjukkan kegiatan yang dilakukan secara parallel atau untuk menggabungkan dua kegiatan paralel menjadi satu.</p>
	<p>Joint, digunakan untuk menunjukkan adanya dekomposisi</p>

	Decision point , menggambarkan pilihan untuk pengambilan keputusan
---	---

Sumber : (Hendini, 2016)

2.1.4 Entity Relational Diagram (ERD)

Entity Relational Diagram (ERD) adalah diagram yang berfungsi untuk menampilkan relasi logika dan interaksi di antara para entitas. Sebuah model *Entity Relational Diagram* (ERD) menunjukkan tampilan sistem secara keseluruhan untuk menciptakan struktur hubungan fisik data (Yonathan dkk., 2018).

Menurut Santoso (2004), diperlukan adanya pengelompokan pada unsur-unsur dasar pembangun untuk membuat sebuah ERD. Berikut adalah unsur - unsur dasar yang dibutuhkan untuk membuat sebuah ERD yaitu :

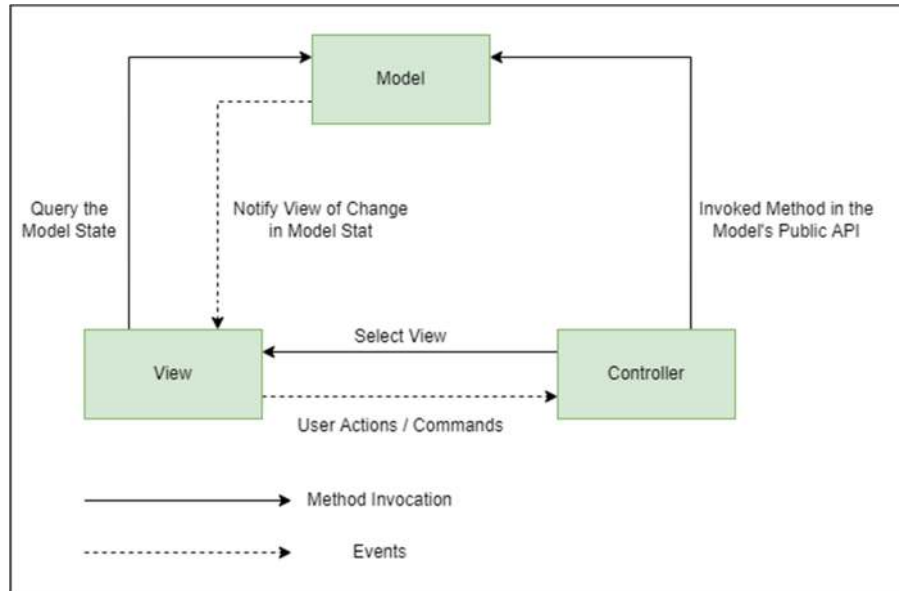
- 1) Entitas yaitu objek yang berasal dari suatu yang nyata maupun abstrak. Objek ini disimbolkan dengan persegi panjang.
- 2) Atribut yaitu elemen dari sebuah entitas yang berfungsi untuk mendefinisikan karakter dari entitas dan bertindak sebagai *primary key*. Atribut ini disimbolkan dengan *ellipse*.
- 3) Relasi yang mempunyai hubungan *one-to-one* (1:1) dan *one-to-many* (1:N).
- 4) Relasi yang mempunyai hubungan *many-to-many* (M:N).

2.1.5 Model-View-Controller (MVC)

Pola *Model-View-Controller* (MVC) membagi aplikasi dalam tiga bentuk yaitu *model*, *view* dan *controller*. *Model* adalah bagian komponen yang bertindak sebagai logika inti dari sebuah aplikasi. *View* adalah *user interface* dari *controller* dan merupakan tampilan umum dari aplikasi yang menjadi tempat interaksi dari pengguna. Sedangkan komponen *controller* mengimplementasikan *flow* yang menjadi pengontrol antara *view* dan *model* (Simanjuntak dkk., 2016).

Menurut Hidayati (2020), *model*, *view* dan *controller* saling berkaitan antara satu dengan yang lain sehingga ketiga komponen ini harus tersinkronisasi satu sama lain. Secara garis besar, *model* menggambarkan struktur data, *view* menggambarkan tampilan yang menjadi tempat pengguna berinteraksi dan *controller* merupakan

penghubung antara *model* dan *view*. Ilustrasi dari MVC dapat dilihat pada gambar 2.3 dibawah.



Gambar 2.3 Ilustrasi Model View Controller

Sumber : (Hidayati, 2020)

Selanjutnya menurut Burbeck yang dikutip oleh Hidayati (2020), dijelaskan bahwa arsitektur MVC dibagi menjadi tiga lapisan, yaitu :

- 1) **Model**, digunakan untuk mengelola informasi dan memberitahu pengamat ketika ada perubahan informasi. Model mengandung data dan fungsi yang berhubungan dengan pemrosesan data.
- 2) **View**, bertanggung jawab untuk pemetaan grafis ke sebuah perangkat.
- 3) **Controller**, menerima *input* dari pengguna dan mendistribusikan model dan *view* untuk melakukan aksi berdasarkan masukan tersebut. Sehingga *controller* bertanggung jawab untuk pemetaan aksi pengguna akhir terhadap respon aplikasi.

Pola pengembangan dengan MVC adalah pola yang memungkinkan pengembangan web yang lebih efisien jika ditinjau dari sudut pandang pengembangan, pengklasifikasian tugas, dokumentasi, pengujian dan pemeliharaan. Penggunaan MVC dapat membantu pengembangan aplikasi menjadi bagian – bagian kecil sehingga memungkinkan pengembangan dan pemecahan masalah yang lebih tepat dan terstruktur (Riana dkk., 2018).

2.1.6 Basis Data (*Database*)

2.1.6.1 Penjelasan Basis Data (*Database*)

Basis data adalah kumpulan data deskriptif yang terhubung secara *logical* dan dirancang untuk memenuhi kebutuhan informasi pengguna. Basis data adalah kumpulan data dalam jumlah besar yang digunakan secara bersama – sama oleh berbagai pihak (Hamidy, 2017).

Basis data merupakan komponen penting dalam pengembangan sistem informasi, karena merupakan tempat semua data sistem diterima dan diatur, sehingga dapat dieksplorasi untuk mengumpulkan informasi dalam berbagai bentuk yang berbeda. Basis data adalah kumpulan dari kelompok data yang terkait. Data diatur sedemikian rupa untuk menghindari duplikasi yang tidak perlu, sehingga dapat diproses dengan tepat dan dengan mudah menghasilkan informasi yang dibutuhkan (Tonggiroh, 2017).

2.1.6.2 *Relational Database Management System (RDBMS)*

Database Management System (DBMS) adalah aplikasi yang dirancang khusus sebagai aplikasi yang dapat berinteraksi dengan pengguna, aplikasi lain maupun basis data itu sendiri yang bertujuan untuk mengumpulkan dan menganalisis data. DBMS adalah perangkat lunak yang bertujuan khusus untuk membuat, mengontrol dan mengelola basis data (Hamidy, 2017). Beberapa contoh DBMS yang sering digunakan adalah MySQL dan Microsoft Access.

Relational Database Management System (RDBMS) adalah program basis data yang terdiri dari tabel-tabel yang saling memiliki hubungan. Sebuah basis data terdiri dari banyak tabel yang selanjutnya terdiri dari banyak kolom *field* yang di dalamnya terdapat data. Setiap tabel dari basis data ini mempunyai *private key* yang berbeda dan kemudian dihubungkan dengan tabel yang lain menjadi *foreign key*.

Dengan RBMS ini maka data akan secara konsisten disimpan di suatu tabel, kemudian tabel lain yang membutuhkan data serupa tinggal menghubungkannya dengan menggunakan *foreign key*.

2.1.6.3 MySQL

MySQL merupakan RDBMS (*Relational Database Management System*) atau server *database* yang mengelola *database* dengan cepat menampung dalam jumlah sangat besar dan dapat diakses oleh banyak *user* (Novianti, 2017).

Menurut Adi Nugroho yang dikutip oleh Amalia., dkk (2020), MySQL (*My Structured Query Language*) adalah: “ Suatu sistem basis data *relation* atau *Relational Database Management System* (RDBMS) yang mampu bekerja secara cepat dan mudah digunakan. MySQL juga merupakan program pengakses *database* yang bersifat jaringan, sehingga dapat digunakan untuk aplikasi *multi user* (banyak pengguna). MySQL didistribusikan gratis dibawah lisensi GPL (*General Public License*) di mana setiap program bebas menggunakan MySQL namun tidak bisa dijadikan produk turunan yang dijadikan *closed source* atau komersial ”.

2.1.6.4 PHP (*Hypertext Preprocessor*)

PHP (*PHP : Hyperlink Preprocessor*) merupakan bahasa pemrograman yang sudah sangat umum digunakan dalam pengembangan situs web. Bahasa ini awalnya dikembangkan oleh seorang programmer asal Denmark-Kanada bernama Rasmus Lerdorf pada tahun 1994 yang selanjutnya di produksi oleh The PHP Group hingga sekarang.

PHP merupakan bahasa pemrograman yang dapat dimasukkan kedalam HTML. PHP adalah sebuah bahasa *scripting server-side* yang berarti *server*-lah yang menerjemahkan skrip program yang kemudian hasil dari terjemahan tersebut dikirim kepada *user* sesuai perintah yang dimaksud (Firman, 2016).

2.1.7 *Framework Laravel*

2.1.7.1 *Framework*

Framework adalah bagian dari pemrograman yang siap digunakan ulang kapan saja, sehingga tidak harus membuat skrip yang sama dengan fungsi yang sama. Sebagai contoh jika ingin membuat sekumpulan halaman-halaman web yang menggunakan paginasi halaman, *framework* telah menyediakan fungsi yang

memungkinkan hal tersebut dan cukup digunakan fungsi tersebut dengan memperhatikan kaidah-kaidah yang ada pada masing-masing *framework* (Sari., dkk 2019).

2.1.7.2 Laravel

Laravel adalah kerangka kerja web *open source* dan gratis berdasarkan PHP, dibuat oleh Taylor Otwell dan digunakan untuk mengembangkan aplikasi web menggunakan pola MVC. Struktur pola MVC di Laravel sedikit berbeda dari struktur pola MVC umum. Pada Laravel terdapat perutean yang menghubungkan permintaan pengguna dan pengontrol. Oleh karena itu, pengontrol tidak segera menerima permintaan tersebut (Sari., dkk 2019).

Laravel memfokuskan pengembangan *website* pada pihak *end-user* yang menitikberatkan pada kesederhanaan dan tampilan elegan serta fungsionalitas *website* yang berjalan sebagaimana mestinya. Dengan hal tersebut, pihak pengembang yang menggunakan *framework* ini dapat menggunakannya untuk membangun proyek skala kecil maupun besar.

2.1.8 *User Interface dan User Experience*

A. User Interface

User Interface adalah elemen yang sangat penting karena *user interface* inilah nantinya yang akan berinteraksi dengan pengguna sebagai bagian dari pengalaman mereka dalam menggunakan aplikasi. *User Interface* bukan hanya berkaitan tentang aspek visual seperti warna dan bentuk, tetapi lebih jauh dari itu juga melibatkan penyajian yang tampilan dan fungsi yang tepat kepada pengguna.

User interface merupakan bagian yang menghubungkan antara pengguna dan pengalaman dalam menggunakan aplikasi. Desain *user interface* yang baik harus menghubungkan dengan tepat antara estetika dan interaktivitas yang intuitif tanpa memerlukan upaya tambahan yang dapat menyebabkan pengguna mengalami kesulitan dalam penggunaan aplikasi (Rochmawati, 2019).

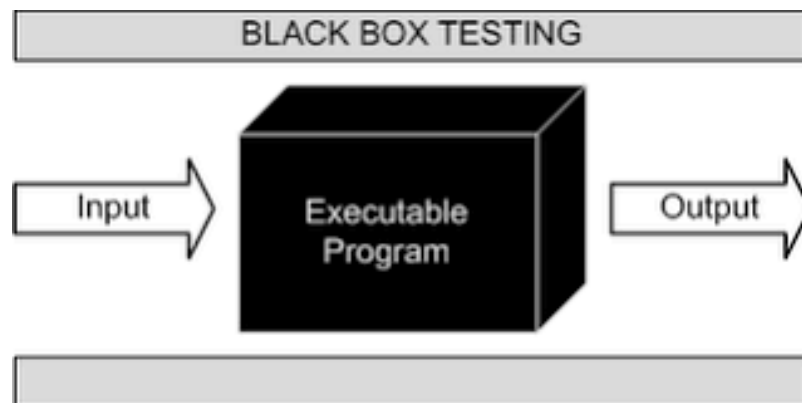
B. User Experience

User Experience menjadi salah satu faktor yang mempunyai peranan penting dalam proses pengembangan aplikasi. Tanpa perancangan yang matang akan berakibat aplikasi yang dikembangkan tidak dapat digunakan secara maksimal sebagaimana mestinya dan dapat menyebabkan pengguna tidak nyaman dalam menggunakan aplikasi sehingga beralih menggunakan aplikasi lain (Yolanovia., dkk 2021).

Menurut ISO 9241-11 yang dikutip oleh Maulana (2020) menjelaskan bahwa *user experience* adalah respon dan persepsi pengguna sebagai bentuk dari sebuah produk, jasa dan sistem. *User experience* merupakan bentuk dimana *user* dapat merasakan kesenangan dan kepuasan dari menggunakan sebuah produk, memegang atau melihat produk tersebut. UX tidak dapat dirancang oleh desainer tapi desainer dapat menerjemahkan kegiatan *user* menjadi UX yang baik.

2.1.9 Metode Pengujian

A. Pengujian Black Box



Gambar 2.4 Ilustrasi Pengujian Black Box

Sumber : (Febriyanti., dkk 2021)

Pengujian adalah serangkaian kegiatan yang direncanakan dan sistematis untuk pengujian atau mengevaluasi kebenaran yang diinginkan (Febriyanti., dkk 2021). Pengujian perangkat lunak sangat penting karena setiap orang membuat kesalahan pada proses pembuatan perangkat lunak. Kesalahan di setiap perangkat lunak akan berbeda untuk setiap perangkat lunak (Ningrum., dkk 2019).

Metode *Black Box* adalah salah satu metode pengujian yang mudah digunakan karena hanya memeriksa *output* berdasarkan *input* yang dilakukan. Jumlah data uji yang akan digunakan dapat dilihat melalui jumlah bidang input yang akan diuji dan aturan *entry* data yang harus dipenuhi. Dengan pendekatan ini kita dapat melihat jika fungsi dari program tersebut berjalan sesuai dengan kebutuhan (Febriyanti., dkk 2021).

B. Pengujian Aspek *Usability* Skala *Likert*

Skala *likert* merupakan suatu skala psikometrik yang umumnya digunakan dalam angket dan juga sangat umum digunakan dalam riset berbentuk survei. Nama skala ini berasal dari nama Rensis Likert, yang menerbitkan suatu laporan yang menjelaskan cara penggunaannya. Dalam skala *likert* ini responden diminta untuk melengkapi kuesioner yang mengharuskan mereka melakukan penilaian dari serangkaian pertanyaan yang berkaitan dengan variabel penelitian yang dilakukan.

Skala *likert* adalah skala pengukuran yang dikembangkan oleh Likert. Skala *likert* memiliki empat atau lebih item yang digabungkan untuk membentuk skor yang menggambarkan karakteristik seseorang seperti pengetahuan, sikap, dan perilaku. Skor gabungan (biasanya rata-rata dari semua pertanyaan) dapat digunakan untuk analisis data (Maryuliana., dkk 2016).

Data yang diolah pada metode skala *likert* ini diperoleh dari kuesioner yang diberikan kepada pihak responden yang lalu diolah menggunakan metode skala *likert*. Menurut Mulyatiningsih dkk. (2013), metode skala *likert* mempunyai lima pilihan skala jawaban yaitu tidak setuju, kurang setuju, netral, setuju dan sangat setuju. Agar tanggapan responden lebih tegas, maka disarankan hanya menggunakan 4 (empat) skala jawaban dan menghilangkan pilihan jawaban “N” atau Netral. Jawaban dari skala *likert* dikategorikan menjadi :

- a. Tidak Setuju (TS) diberi skor 1
- b. Kurang Setuju (KS) diberi skor 2
- c. Setuju (S) diberi skor 3
- d. Sangat Setuju (SS) diberi skor 4

Data hasil pengujian ini dianalisis dengan menghitung rata-rata skor setiap jawaban dari responden. Skor yang ditetapkan dapat dihitung dengan metode sebagai berikut (Sugiyono, 2015):

$$Skor_{total} = (J_{SS} \times 4) + (J_S \times 3) + (J_{KS} \times 2) + (J_{TS} \times 1) \quad (3.1)$$

Keterangan :

J_{SS} = Jawaban Sangat Setuju

J_S = Jawaban Setuju

J_{KS} = Jawaban Kurang Setuju

J_{TS} = Jawaban Tidak Setuju

Setelah diperoleh skor total, selanjutnya perhitungan persentase skor untuk mendapatkan hasil pengujian *usability* dengan menggunakan rumus :

$$P_{skor} = \frac{Skor_{total}}{i \times r \times 4} \times 100\% \quad (3.2)$$

Keterangan :

$Skor_{total}$ = Skor total hasil responden menjawab

i = Jumlah pertanyaan

r = Jumlah responden

Setelah diperoleh hasil dari persentase skor, kemudian dibandingkan dengan tabel kriteria interpretasi skor seperti yang ada pada tabel berikut ini :

Tabel 2. 3 Interpretasi Skor

Persentase (%)	Interpretasi
0%-20%	Sangat kurang/Rendah
21%-40%	Kurang/Rendah
41%-60%	Cukup
61%-80%	Baik/Tinggi
81%-100%	Sangat baik/Tinggi

Sumber : (Riduwan, 2013)

Aspek *usability* dikatakan memenuhi standar apabila persentase yang diperoleh menunjukkan nilai yang tinggi. Apabila yang terjadi sebaliknya, maka diperlukan revisi dan evaluasi lebih lanjut terhadap sistem yang dikembangkan.

2.2. Penelitian Terkait

2.2.1 Perancangan Sistem Informasi Penyewaan Gedung Pada Gedung Balai Komando Kopassus Berbasis Web

Penelitian ini dilakukan dengan tujuan merancang sistem informasi penyewaan gedung balai komando. Penelitian ini dilakukan atas dasar banyaknya keluhan konsumen terkait informasi penyewaan gedung yang harus dilakukan secara manual dengan datang langsung di gedung tersebut. Selain itu kendala dalam melakukan pendaftaran juga terjadi, dimana sering ditemukan adanya duplikasi pesanan dalam waktu dan tempat yang sama sehingga terjadi ketidakakuratan data. Dalam penelitian ini penulis melakukan wawancara dan observasi yang selanjutnya dilakukan pengembangan sistem dengan metode *waterfall*.

Penelitian ini dilakukan dengan melalui beberapa tahap yaitu : pengumpulan data, analisa kebutuhan data, perancangan sistem, pengujian dan implementasi. Perancangan dari aplikasi ini menggunakan UML (*Unified Modified Language*) dengan menggunakan *software Codeigniter* dan bahasa pemrograman PHP.

Hasil yang dicapai dalam penelitian ini adalah mempublikasikan informasi dengan cepat sehingga para pelanggan tidak mengalami kendala dalam mencari informasi dan juga dalam aplikasi ini, pelanggan dapat langsung melakukan pemesanan gedung (Juniardi., dkk 2018).

2.2.2 Sistem Informasi Penyewaan Gedung Serbaguna Berbasis Web Di HKBP Bandung Barat

Kesulitan masyarakat dalam memperoleh informasi mengenai penyewaan gedung HKBP Bandung Barat menjadi latar belakang dari penelitian ini. Peneliti menilai bahwa di era perkembangan teknologi yang pesat ini, harusnya informasi mengenai penyewaan gedung dapat diperoleh masyarakat dengan mudah dan cepat oleh konsumen.

Pada penelitian ini penulis menggunakan metode pengumpulan data dengan melakukan wawancara dan observasi serta metode pengembangan berorientasi

objek dan pengembangan sistem menggunakan metode *prototype* dengan menggunakan bahasa pemrograman PHP.

Hasil dari penelitian ini adalah sistem informasi yang memungkinkan konsumen untuk memperoleh informasi penyewaan gedung dengan baik. Selain itu, pihak konsumen juga dapat memilih jadwal secara langsung sehingga tidak terjadi bentrok dengan jadwal dari konsumen lain serta pembayaran yang bisa dilakukan via bank (Paimasrul, 2019).

2.2.3 Aplikasi Pengelolaan Sewa Gedung Kantor Berbasis Web Pada PT. SARINAH

Penelitian ini membahas tentang pengelolaan gedung PT. Sarinah yang masih dilakukan secara manual. Penyimpanan dokumen data sewa yang masih dilakukan secara manual dengan cara pengarsipan fisik menyebabkan pencarian data sewa memakan banyak waktu. Selain itu peneliti menilai proses pemberitahuan tagihan sewa yang masih dilakukan secara pos masih kurang efektif karena memerlukan waktu yang relatif lama.

Penelitian ini menggunakan metode sistem *Rapid Application Development* (RAD), metode ini berorientasi pada waktu penyelesaian yang cepat tanpa mengurangi kualitas sistem yang dihasilkan. Hasil dari penelitian ini adalah terbuatnya aplikasi pengelolaan sewa gedung kantor berbasis web di PT Sarinah (Persero) Jakarta (Andjani dkk., 2019).