

**RANCANG BANGUN APLIKASI RESERVASI GEDUNG UNTUK
KEGIATAN EKSTRAKURIKULER BERBASIS ANDROID (STUDI
KASUS : SMAN 1 TANJUNG BINTANG)**

(Skripsi)

Oleh

**LESY NUGRAHA
NPM 1915061011**



**FAKULTAS TEKNIK
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2025**

**RANCANG BANGUN APLIKASI RESERVASI GEDUNG UNTUK
KEGIATAN EKSTRAKURIKULER BERBASIS ANDROID (STUDI
KASUS : SMAN 1 TANJUNG BINTANG)**

Oleh

Lesy Nugraha

Skripsi

Sebagai Salah Satu Syarat untuk Mencapai Gelar

SARJANA TEKNIK

Pada

Jurusan Teknik Elektro

Fakultas Teknik Universitas Lampung



**FAKULTAS TEKNIK
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG**

2025

ABSTRAK

RANCANG BANGUN APLIKASI RESERVASI GEDUNG UNTUK KEGIATAN EKSTRAKURIKULER BERBASIS ANDROID (STUDI KASUS : SMAN 1 TANJUNG BINTANG)

Oleh

Lesy Nugraha

Pengelolaan reservasi gedung untuk kegiatan ekstrakurikuler di SMAN 1 Tanjung Bintang yang masih dilakukan secara manual kerap menimbulkan kendala, seperti inefisiensi proses, kesalahan pencatatan, dan konflik jadwal antar pengguna. Untuk mengatasi hal tersebut, penelitian ini merancang dan membangun sebuah aplikasi sistem informasi manajemen reservasi berbasis Android bernama “RESERVASIKUY”. Aplikasi dikembangkan menggunakan *framework Flutter*, layanan *backend Firebase*, serta metode pengembangan *Kanban*, dengan dukungan tiga peran utama: *Userektrakurikuler*, *Supervisor*, dan *Superadmin*. Pengujian dilakukan melalui dua tahap: pertama, pengujian fungsionalitas menggunakan teknik *black-box testing* berbasis *decision table* dengan 37 skenario yang seluruhnya dinyatakan lolos; kedua, pengujian pengalaman pengguna melalui *User Experience Questionnaire* (UEQ) yang melibatkan 34 responden. Hasil UEQ menunjukkan skor rata-rata pada keenam dimensi (Daya Tarik, Kejelasan, Efisiensi, Ketepatan, Stimulasi, dan Kebaruan) berada pada kategori *Excellent*. Berdasarkan hasil tersebut, dapat disimpulkan bahwa RESERVASIKUY mampu memberikan solusi yang efektif, efisien, dan mudah diterima oleh pengguna dalam mengatasi permasalahan manajemen reservasi gedung di sekolah.

Kata kunci — Sistem Informasi, Reservasi Gedung, Android, Flutter, Firebase, Kanban, User Experience Questionnaire.

ABSTRACT

DESIGN ANDROID BASED BUILDING RESERVATION APPLICATION FOR EXTRACURRICULAR ACTIVITIES (CASE STUDY: SMAN 1 TANJUNG BINTANG)

By

Lesy Nugraha

The manual management of extracurricular venue reservations at SMAN 1 Tanjung Bintang often leads to inefficiency, data errors, and scheduling conflicts. To address these issues, this research designed and developed an Android-based information system named “RESERVASIKUY”. The application was built using the Flutter framework and Firebase as the backend, following the Kanban development methodology. It provides support for three distinct user roles: extracurricular user, supervisor, and superadmin. System validation was carried out in two stages. First, functional testing was performed through black-box testing with a decision table approach, covering 37 scenarios, all of which successfully passed. Second, user experience evaluation was conducted using the User Experience Questionnaire (UEQ) involving 34 respondents. The results indicated excellent ratings across all six dimensions (Attractiveness, Clarity, Efficiency, Dependability, Stimulation, and Novelty). These findings confirm that RESERVASIKUY delivers an effective, efficient, and user-friendly solution for managing venue reservations, and can serve as a practical alternative to replace the manual process previously used at the school.

Kata kunci — Information System, Venue Reservation, Android, Flutter, Firebase, Kanban, User Experience Questionnaire.

Judul Skripsi

: RANCANG BANGUN APLIKASI
RESERVASI GEDUNG UNTUK
KEGIATAN EKSTRAKURIKULER
BERBASIS *ANDROID* (STUDI
KASUS : SMAN 1 TANJUNG
BINTANG)

Nama Mahasiswa

: *Iesy Nugraha*

Nomor Pokok Mahasiswa : 1915061011

Jurusan

: Teknik Elektro

Fakultas

: Teknik

MENYETUJUI

1. Komisi Pembimbing

Pembimbing Utama

Pembimbing Pendamping

Wahyu Eko S, S.T., M.Sc.
NIP. 197412012001121001

Mahendra Pratama, S.T., M.Eng.
NIP. 199112152019031013

2. Mengetahui

Ketua Jurusan
Teknik Elektro

Ketua Program Studi
Teknik Informatika

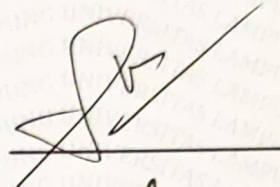
Herlinawati, S.T., M.T.
NIP. 197103141999032001

Yessi Mulyani, S.T., M.T.
NIP. 197312262000122001

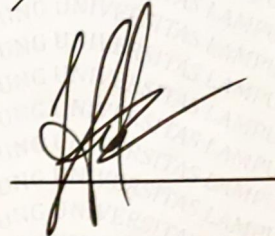
MENGESAHKAN

1. Tim Penguji

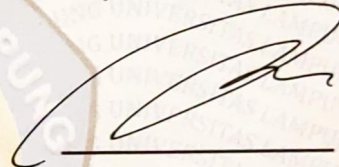
Ketua : **Wahyu Eko S, S.T., M.Sc.**



Sekretaris : **Mahendra Pratama, S.T., M.eng.**



Penguji : **Mona Arif Muda, S.T., M.T.**



2. Dekan Fakultas Teknik



Dr. Hi. Ahmad Herison, S.T, M.T.)

NIP. 196910302000031001

Tanggal Lulus Ujian Skripsi : 15 Desember 2025

SURAT PERNYATAAN

Saya yang bertandatangan di bawah ini, menyatakan bahwa skripsi saya dengan judul “Rancang Bangun Aplikasi Reservasi Gedung Untuk kegiatan Ekstrakurikuler Berbasis *Android* (Studi Kasus : SMAN 1 Tanjung Bintang) ” dibuat oleh saya sendiri. Semua hasil yang tercantum dalam skripsi ini telah sesuai dengan aturan penulisan karya ilmiah Universitas Lampung. Jika suatu saat terbukti bahwa skripsi ini merupakan salinan atau dibuat oleh orang lain, maka saya bersedia menerima sanksi sesuai dengan ketentuan hukum atau akademik yang berlaku.

Bandar Lampung, 15 Desember 2025

Pembuat pernyataan,



Lesy Nugraha

NPM. 1915061011

RIWAYAT HIDUP



Penulis dilahirkan di Jatibaru, pada tanggal 22 Oktober 2001. Penulis adalah anak keempat dari lima bersaudara dari pasangan Bapak Dwiatmo dan Ibu Aida Wati.

Penulis menyelesaikan pendidikannya di SDN 2 Jatibaru pada tahun 2013, kemudian melanjutkan di SMPN 1 Tanjung Bintang pada tahun 2016, kemudian melanjutkan di SMAN 1 Tanjung Bintang pada tahun 2019. Pada tahun 2019 penulis terdaftar sebagai Mahasiswa program studi Teknik Informatika, Jurusan Teknik Elektro, Fakultas Teknik, Universitas Lampung melalui jalur SNMPTN. Selama menjadi mahasiswa penulis melakukan beberapa kegiatan, antara lain :

1. Menjadi anggota biasa Himpunan Mahasiswa Teknik Elektro Universitas Lampung, Departemen Pendidikan dan Pengembangan Diri, Divisi Pendidikan Periode 2020/2021 dan periode 2021/2022.
2. Mengikuti program Studi Independen Kampus Merdeka dari Kementerian Pendidikan dan Budaya dengan mengambil Kelas *frontend Basic* di Ruang Guru *Camp* pada tahun 2022.
3. Melaksanakan Kuliah Kerja Nyata pada bulan Januari sampai dengan Februari 2022 di Desa Kemukus, Ketapang, Lampung Selatan.
4. Melakukan Kerja Praktik di Queen Network Nusantara (QNN) sebagai UI/UX *Designer* pada tahun ajaran 2022/2023.
5. Mengikuti program Studi Independen Kampus Merdeka dari Kementerian Pendidikan dan Budaya program magang merdeka sebagai SEO *Specialist* di *Kapanlagi Youniverse (PT. Liputan Enam Dot Com)* pada bulan Februari sampai dengan Juni 2023.

MOTTO

“Sesungguhnya beserta kesulitan ada kemudahan”
(Q.S Al-Insyirah : 6)

“Allah tidak membebani seseorang melainkan sesuai dengan
kesanggupannya”
(Q.S Al-Baqarah : 286)

“It always seems impossible until it’s done.”
(Nelson Mandela)

“Educating the mind without educating the heart is no education at all.”
(Aristotle)

PERSEMBAHAN

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Segala puji dan syukur kehadiran Allah SWT. atas segala Rahmat dan karunia-Nya sehingga saya dapat menyelesaikan skripsi ini. *Shalawat* serta salam kepada Nabi Muhammad SAW sebagai suri *tauladan akhlakul karimah*.

KUPERSEMBAHKAN KARYA INI KEPADA :

“Ayah saya Dwiatmo dan Ibu saya Aida Wati, yang senantiasa selalu memberikan yang terbaik, dan selalu memanjatkan doa yang tiada henti untuk saya. Saya ucapkan terima kasih sebesar-besarnya karena telah mendidik dan membesarkan dengan penuh kasih sayang sehingga menjadi seorang yang kuat dalam menghadapi banyak hal, kuat dalam menghadapi berbagai keadaan, serta mengajarkan untuk selalu menjadi pribadi yang baik hati. Semoga dengan ilmu yang didapatkan ini dapat menjadi amal *jariyah* bagi kalian”

“Seluruh teman-teman jurusan elektro angkatan 2019 yang telah menemani dan berjuang bersama di dunia perkuliahan. Banyak cerita dan momen yang telah kita lewati bersama yang kemudian nanti akan kita ceritakan dan kenang kembali. Sampai saat itu tiba, semoga kita telah menjadi orang-orang sukses yang selalu kita impikan.”

“Diri sendiri yang telah berjuang keras dan bertahan untuk menyelesaikan penelitian ini. Terima kasih telah bertahan dan berjuang. Ingatlah agar senantiasa berjuang sampai kamu mendapatkan apa yang selalu kamu impikan. Semoga kamu senantiasa sukses dan selalu bahagia.”

“Almamater tercinta, Universitas Lampung dan Jurusan Teknik Elektro”

SANWACANA

Puji syukur penulis panjatkan kepada *kehadirat* Allah SWT. yang telah memberikan rahmat dan *ridho-Nya* sehingga penulis dapat menyelesaikan penelitian yang berjudul “Rancang Bangun Aplikasi Reservasi Gedung Untuk Kegiatan Ekstrakurikuler Berbasis *Android* (Studi Kasus : SMAN 1 Tanjung Bintang)”. Selama masa penelitian penulis mendapatkan banyak bantuan, bimbingan dan dukungan dari berbagai pihak. Oleh karena itu penulis mengucapkan terima kasih sebanyak- banyaknya kepada:

1. Kedua orang tua dan keluarga tercinta penulis yang selalu memberikan dukungan, doa, rasa cinta tiada akhir dan mengingatkan penulis untuk terus berjuang dalam menyelesaikan penelitian ini;
2. Bapak Dr. Eng. Helmy Fitriawan, S.T., M.Sc., selaku Dekan Fakultas Teknik Universitas Lampung;
3. Ibu Herlinawati, S.T., M.T. selaku Ketua Jurusan Teknik Elektro Universitas Lampung;
4. Ibu Yessi Mulyani, S.T., M.T. selaku Ketua Program Studi Teknik Informatika Universitas Lampung;
5. Ibu Yessi Mulyani, S.T., M.T. selaku pembimbing akademik yang telah membantu kelancaran proses perkuliahan;
6. Bapak Wahyu Eko S, S.T., M.Sc. selaku pembimbing utama yang selalu memberikan bimbingan dan dukungan kepada penulis dalam menyelesaikan penelitian ini;
7. Bapak Mahendra Pratama, S.T., M.Eng. selaku pembimbing pendamping yang selalu meluangkan waktunya untuk memberikan bimbingan dalam menyelesaikan penelitian ini;
8. Bapak Mona Arif Muda, S.T., M.T. selaku penguji yang telah memberikan banyak kemudahan, masukan, serta saran kepada penulis

terhadap penelitian ini;

9. Mbak Rika selaku admin Program Studi Teknik Informatika yang telah banyak membantu penulis dalam segala urusan administrasi selama perkuliahan;
10. Bapak Haris Tri Okfianto, S.Pd., M.Pd. selaku kepala sekolah SMAN 1 Tanjung Bintang serta guru-guru, admin sekolah, dan para siswa yang telah membantu penulis dalam pelaksanaan penelitian ini.
11. Dwi lilis lestari yang telah bersama menemani penulis dalam menamatkan *chapter* perkuliahan ini.
12. Segenap teman-teman kelas TI A 2019 dan teman-teman jurusan teknik elektro angkatan 2019 yang telah bersama menemani penulis dalam menamatkan *chapter* perkuliahan ini.

Penulis berharap agar laporan ini dapat menjadi referensi bagi pengembangan keilmuan di bidang Teknik Informatika. Oleh karena itu, semoga penelitian ini bermanfaat bagi yang membacanya.

Bandar Lampung, 15 Desember 2025
Penulis,

Lesy Nugraha

DAFTAR ISI

DAFTAR ISI	i
DAFTAR GAMBAR	iii
DAFTAR TABEL	vi
I. PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah.....	4
1.3. Tujuan Penelitian	4
1.4. Manfaat Penelitian	5
1.5. Batasan Masalah	5
1.6. Sistematika Penulisan	6
II. TINJAUAN PUSTAKA	7
2.1 Reservasi Gedung.....	7
2.2 Sistem Informasi	8
2.3 Android	9
2.4 <i>Flutter</i>	11
2.5 <i>Dart</i>	12
2.6 <i>Kanban</i>	13
2.7 <i>Trello</i>	15
2.8 UML (<i>Unified Modeling Language</i>).....	16
2.9 <i>Firebase</i>	19
2.10 Testing (Pengujian)	20
1. <i>Blackbox testing</i>	20
2. <i>User Experience Questionnaire (UEQ)</i>	22

2.11	Penelitian Terdahulu	24
III.	METODE PENELITIAN	27
3.1.	Waktu dan Tempat	27
3.2.	Jadwal Penelitian	27
3.3.	Alat Penelitian	28
3.4.	Tahapan Penelitian	29
3.4.1.	<i>Analysis</i>	31
3.4.2.	<i>Design</i>	37
3.4.3.	Perancangan papan <i>kanban</i>	63
3.4.4.	Pengembangan dengan menggunakan <i>Kanban (Develop)</i>	68
3.4.5.	<i>Deploy</i>	69
3.4.6.	Penulisan Laporan	70
IV.	HASIL DAN PEMBAHASAN	71
4.1	Hasil.....	71
4.1.1	Implementasi Antarmuka Pengguna (<i>User Interface</i>).....	73
4.1.2	Implementasi Logika Inti Sistem (<i>Coding</i>).....	89
4.1.3	Hasil Pengujian Aplikasi (<i>Testing</i>).....	104
4.2	Pembahasan	118
4.2.1	Analisis Fungsionalitas Aplikasi	118
4.2.2	Analisis Pengalaman Pengguna	120
4.2.3	Jawaban Atas Rumusan Masalah	122
4.2.4	Analisis Biaya Implementasi Sistem.....	123
V.	KESIMPULAN & SARAN.....	125
5.1	Kesimpulan	125
5.2	Saran	126
	DAFTAR PUSTAKA	127

DAFTAR GAMBAR

Gambar 2.1 Diagram sistem informasi	8
Gambar 2.2 Android.....	9
Gambar 2.3 Android.....	10
Gambar 2.4 Flutter	11
Gambar 2.5 kanban dalam proses manufaktur	13
Gambar 2.6 Kanban Board.....	14
Gambar 2.7 Tampilan <i>Trello</i> pada halaman beranda	15
Gambar 2.8 Pertanyaan yang digunakan dalam digunakan metode <i>User Experience Questionnaire</i> (UEQ).....	23
Gambar 3.1 Diagram alir penelitian.....	30
Gambar 3.2 Usecase diagram aplikasi Reservasi Gedung untuk Kegiatan Ekstrakurikuler.....	38
Gambar 3.3 Class diagram aplikasi Reservasi Gedung untuk Kegiatan Ekstrakurikuler studi kasus SMAN1 Tanjung Bintang.....	39
Gambar 3.4 Activity diagram masuk aplikasi.....	41
Gambar 3.5 Activity diagram reservasi gedung.....	43
Gambar 3.6 Activity diagram supervisor mengelola data gedung	45
Gambar 3.7 Activity diagram supervisor mengelola akun <i>userekstrakurikuler</i> ...	47
Gambar 3.8 Activity diagram mengubah informasi profil.....	49
Gambar 3.9 Activity diagram supervisor mengakses menu laporan.....	50
Gambar 3.10 Activity diagram supervisor mengkonfirmasi reservasi.....	52
Gambar 3.11 Activity diagram user melihat reservasi.....	54
Gambar 3.12 Activity diagram <i>userekstrakurikuler</i> membatalkan reservasi di beranda	55
Gambar 3.13 Activity diagram <i>userekstrakurikuler</i> menyelesaikan reservasi di beranda	56

Gambar 3.14 Activity diagram <i>Userekstrakurikuler</i> Melihat Informasi Gedung dan Jadwal.....	57
Gambar 3.15 Activity diagram pengguna mengubah informasi profil	59
Gambar 3.16 Arsitektur sistem reservasi gedung studi kasus SMAN 1 Tanjung Bintang.....	60
Gambar 3.17 <i>wireframe design</i> aplikasi ‘RESERVASIKUY’	61
Gambar 4.1 Tampilan <i>splash screen</i> aplikasi <i>Reservasikuy</i>	73
Gambar 4.2 Tampilan masuk aplikasi <i>Reservasikuy</i>	74
Gambar 4.3 Tampilan antar muka <i>superadmin</i>	75
Gambar 4.4 Tampilan halaman beranda supervisor.....	78
Gambar 4.5 Tampilan halaman ‘gedung’ supervisor	79
Gambar 4.6 Tampilan halaman ‘laporan’ supervisor.....	81
Gambar 4.7 Tampilan halaman ‘saya’ supervisor.....	82
Gambar 4.8 Tampilan halaman ‘beranda’ <i>userekstrakurikuler</i>	84
Gambar 4.9 Tampilan halaman ‘gedung’ <i>userekstrakurikuler</i>	85
Gambar 4.10 Tampilan halaman ‘reservasi’ <i>userekstrakurikuler</i>	86
Gambar 4.11 Tampilan halaman ‘riwayat’ <i>userekstrakurikuler</i>	87
Gambar 4.12 Tampilan halaman ‘saya’ <i>userekstrakurikuler</i>	88
Gambar 4.13 <i>code snippet</i> model reservasi aplikasi	89
Gambar 4.14 <i>code snippet</i> akses data (<i>repositories</i>) reservasi aplikasi	92
Gambar 4.15 <i>code snippet</i> akses data (<i>repositories</i>) fungsi pembuatan reservasi aplikasi	93
Gambar 4.16 <i>code snippet</i> logika bisnis (<i>bloc</i>) fungsi daftar gedung yang bisa dipilih reservasi aplikasi.....	95
Gambar 4.17 <i>code snippet</i> logika bisnis (<i>bloc</i>) fungsi pengecekan reservasi aplikasi	96
Gambar 4.18 <i>code snippet</i> logika bisnis (<i>bloc</i>) fungsi pembuatan reservasi aplikasi	97
Gambar 4.19 <i>code snippet</i> halaman pemilihan jadwal dan gedung reservasi aplikasi	99
Gambar 4.20 <i>code snippet</i> halaman gedung yang tersedia aplikasi.....	100
Gambar 4.21 <i>code snippet</i> halaman <i>confirm_reservation.dart</i> aplikasi	101

Gambar 4.22 <i>code snippet</i> halaman <i>confirm_reservation.dart</i> aplikasi	102
Gambar 4.23 <i>code snippet</i> halaman <i>confirm_reservation.dart</i> aplikasi	103
Gambar 4.24 Nilai rata-rata dan variasi UEQ.....	120
Gambar 4.25 Grafik hasil perbandingan dengan <i>benchmark</i>	120
Gambar 4.26 Data Konsistensi <i>Cronbach's Alpha</i>	121
Gambar L.1 gambar contoh formulir kuesioner.....	L-II
Gambar L.2 gambar kode QR link kode sumber	L-IV
Gambar L.3 gambar kode QR link kode sumber	L-IV

DAFTAR TABEL

Tabel 2.1 Simbol dan keterangan komponen use case diagram.....	17
Tabel 2.2 Simbol dan keterangan komponen Class diagram	18
Tabel 2.3 Aturan input decision table	22
Tabel 2.4 Tabel Penelitian Terdahulu	24
Tabel 3.1 Jadwal penelitian.....	27
Tabel 3.2 Alat yang digunakan dalam penelitian.....	28
Tabel 3.3 User story pada perancangan aplikasi reservasi gedung.....	34
Tabel 3.4 Kebutuhan fungsional pada aplikasi reservasi gedung	36
Tabel 3.5 Kebutuhan non-fungsional pada perancangan aplikasi reservasi gedung	37
Tabel 3.6 daftar kartu tugas (<i>task card</i>)	64
Tabel 4.1. Pengujian fitur <i>login</i> dilakukan menggunakan metode <i>blackbox</i> testing dengan teknik <i>decision table</i>	105
Tabel 4.2. Pengujian fitur mengelola akun supervisor oleh superadmin	106
Tabel 4.3. Pengujian fitur Manajemen profil <i>superadmin</i>	107
Tabel 4.4. Pengujian fitur manajemen reservasi aktor supervisor	108
Tabel 4.5. Pengujian fitur manajemen gedung aktor supervisor.....	108
Tabel 4.6. Pengujian fitur manajemen jadwal ekstrakurikuler oleh supervisor..	109
Tabel 4.7. Pengujian fitur manajemen Laporan aktor supervisor	110
Tabel 4.8. Pengujian fitur manajemen <i>userekstrakurikuler</i> oleh supervisor	111
Tabel 4.9. Pengujian fitur manajemen profil supervisor.....	112
Tabel 4.10. Pengujian fitur manajemen Reservasi dalam aktor <i>userekstrakurikuler</i>	113
Tabel 4.11. Pengujian fitur manajemen Riwayat oleh <i>userekstrakurikuler</i>	114
Tabel 4.12. Pengujian fitur manajemen profil <i>userekstrakurikuler</i>	114
Tabel 4.13. Data responden <i>User Experience Questionnaire</i>	115
Tabel 4.14. Data pengujian <i>User Experience Questionnaire</i>	115
Tabel 4.15. Konversi nilai jawaban UEQ	116

I. PENDAHULUAN

1.1. Latar Belakang

Ekstrakurikuler merupakan suatu aktivitas atau kegiatan yang dilakukan di luar jam pelajaran yang dapat dilakukan di lingkungan sekolah maupun di luar lingkungan sekolah yang bertujuan untuk mengembangkan pengetahuan, menambah wawasan, dan keterampilan. SMAN 1 Tanjung Bintang adalah sebuah sekolah menengah atas yang memiliki beragam kegiatan ekstrakurikuler. Kegiatan ekstrakurikuler di SMAN 1 Tanjung Bintang menjadi sarana penting dalam pengembangan potensi siswa di luar kurikulum akademik. Organisasi ekstrakurikuler di SMAN 1 Tanjung Bintang memiliki peran signifikan dalam membentuk karakter dan keterampilan siswa serta memberikan pengalaman penting selama masa pendidikan mereka.

Dalam melaksanakan kegiatan ekstrakurikuler diperlukan gedung yang memadai karena dapat memberikan tempat yang nyaman sesuai dengan kebutuhan siswa dalam menjalani kegiatan ekstrakurikuler. Gedung dalam kegiatan ekstrakurikuler sangat penting mengingat gedung dapat menjadi tempat kegiatan terlaksana dengan nyaman, aman dan sesuai dengan apa yang dibutuhkan oleh siswa. Gedung yang memadai dapat menjadi lingkungan yang membawa dampak baik terhadap siswa dalam meningkatkan keterampilan dan bakat mereka sehingga mereka dapat mengoptimalkan kemampuannya.

Meskipun kegiatan ekstrakurikuler aktif, pengelolaan pemesanan atau reservasi gedung untuk kegiatan ekstrakurikuler sering kali mengalami sejumlah kendala. Antara lain adalah ketersediaan gedung yang sering menjadi suatu masalah ketika akan dijadwalkan untuk kegiatan organisasi ekstrakurikuler. Proses

pemesanan yang masih menggunakan metode manual, seperti buku catatan atau formulir fisik, dinilai tidak efisien dan rentan terhadap kesalahan.

Dalam menangani hal tersebut, dibutuhkan sebuah sistem informasi yang dapat memudahkan proses reservasi gedung untuk kegiatan ekstrakurikuler di SMAN 1 Tanjung Bintang, sekaligus memiliki fleksibilitas untuk digunakan oleh sekolah-sekolah lain yang memiliki kebutuhan serupa. Dengan menekankan SMAN 1 Tanjung Bintang sebagai sampel dalam penelitian ini, aplikasi yang dikembangkan diharapkan dapat diterapkan secara lebih luas di sekolah-sekolah lain yang memiliki kebutuhan yang sama, sehingga memberikan manfaat yang lebih luas dalam pengelolaan kegiatan ekstrakurikuler.

Sistem atau prosedur yang bersifat manual yang digunakan saat ini dirasa kurang memadai untuk memenuhi kebutuhan manajemen reservasi gedung yang dirasa efisien dan transparan. Penggunaan buku catatan atau formulir fisik maupun dengan cara berbicara langsung dengan pihak sekolah yang berhubungan dengan manajemen reservasi gedung cenderung meningkatkan risiko kesalahan dalam informasi. Kurangnya integrasi antara berbagai aspek manajemen, seperti informasi ketersediaan gedung, penjadwalan, dan konfirmasi reservasi, juga menjadi tantangan tersendiri. Dalam menangani hal tersebut, dibutuhkan sebuah sistem informasi yang dapat memudahkan proses reservasi gedung untuk kegiatan ekstrakurikuler di SMAN 1 Tanjung Bintang. Oleh karena itu, sistem informasi yang dibangun harus dapat diakses oleh guru dan siswa yang terlibat dalam kegiatan ekstrakurikuler di SMAN 1 Tanjung Bintang. Dengan hadirnya sistem informasi yang terintegrasi, diharapkan bisa mengefisienkan proses manajemen reservasi gedung untuk kegiatan ekstrakurikuler sehingga dapat memberikan solusi terhadap masalah tersebut.

Urgensi pengembangan aplikasi sistem informasi manajemen reservasi gedung di SMAN 1 Tanjung Bintang sangat mendesak. Dengan semakin banyaknya kegiatan ekstrakurikuler yang dilakukan, kebutuhan akan sistem yang lebih efisien dan terintegrasi menjadi semakin nyata. Aplikasi ini diharapkan dapat memberikan solusi yang tidak hanya efisien tetapi juga mampu mengakomodasi peningkatan jumlah kegiatan dan pengguna di masa depan. Lebih lanjut, sistem ini harus bisa

dikonfigurasi secara mudah untuk digunakan oleh sekolah-sekolah lain, sehingga dapat menjadi standar manajemen reservasi gedung di lingkungan pendidikan yang lebih luas.

Sampai saat ini SMAN 1 Tanjung Bintang belum memiliki aplikasi *mobile* yang dapat mengelola proses reservasi gedung untuk kegiatan ekstrakurikuler. Untuk mengatasi permasalahan tersebut maka diperlukan suatu aplikasi yang dapat memudahkan pihak sekolah serta *User* untuk dapat mencari data gedung yang dapat di reservasi maupun gedung yang sedang dalam kondisi belum bisa dilakukan reservasi dan melakukan reservasi gedung untuk kegiatan ekstrakurikuler. Oleh karena itu pada penelitian ini akan dikembangkan sebuah aplikasi reservasi gedung untuk kegiatan ekstrakurikuler di SMAN 1 Tanjung Bintang berbasis Android. Hal ini juga berdasarkan jumlah pengguna ponsel di Indonesia yang tercatat dari awal tahun 2024 yang di mana Indonesia memiliki total populasi 278,7 juta jiwa dan 353,3 juta pengguna ponsel yang berarti banyak masyarakat Indonesia yang memiliki ponsel bahkan lebih dari satu [1]. Oleh karena itu, besar kemungkinan Android akan digunakan sebagai sistem reservasi gedung untuk kegiatan ekstrakurikuler di SMAN 1 Tanjung Bintang.

Ada beberapa hal yang perlu diperhatikan dalam penerapan sistem informasi ini, antara lain adalah penggunaan teknologi yang tepat guna, kemudian penyajian tampilan aplikasi yang ramah bagi pengguna, serta menjaga keamanan data dan menempatkan pengguna sesuai dengan fungsinya masing-masing. Selain itu, sistem informasi reservasi gedung untuk kegiatan ekstrakurikuler di SMAN 1 Tanjung Bintang juga harus dilengkapi dengan fitur-fitur yang sesuai seperti menemukan jadwal ekstrakurikuler, melakukan reservasi, dan lainnya. Sistem informasi reservasi gedung kegiatan ekstrakurikuler yang terintegrasi dengan baik di SMAN 1 Tanjung Bintang diharapkan dapat mempermudah proses pengelolaan kegiatan ekstrakurikuler di SMAN 1 Tanjung Bintang dan meningkatkan proses penjadwalan dan reservasi gedung.

1.2. Rumusan Masalah

Berdasarkan permasalahan yang telah diuraikan pada latar belakang mengenai sistem reservasi yang terdapat di SMAN 1 Tanjung Bintang, maka dapat dirumuskan beberapa pertanyaan penelitian sebagai berikut:

1. Bagaimana merancang dan membangun sebuah aplikasi sistem informasi berbasis Android yang mampu menggantikan proses manual yang tidak efisien dan rentan terhadap kesalahan data?
2. Bagaimana mengimplementasikan sebuah fitur yang dapat menyajikan informasi ketersediaan jadwal gedung secara *real-time* dan transparan untuk mencegah terjadinya konflik jadwal?
3. Bagaimana merancang sebuah sistem yang mampu menyediakan rekam jejak digital dari seluruh aktivitas reservasi untuk mempermudah proses pengawasan oleh pihak sekolah?
4. Sejauh mana efektivitas dan tingkat penerimaan aplikasi yang dibangun oleh pengguna akhir setelah dilakukan pengujian fungsionalitas dan pengalaman pengguna?

1.3. Tujuan Penelitian

Berdasarkan penjabaran rumusan masalah, maka tujuan yang ingin dicapai dalam penelitian ini adalah sebagai berikut:

1. Merancang dan membangun sebuah aplikasi sistem informasi berbasis Android yang mampu menggantikan proses manual yang tidak efisien dan rentan terhadap kesalahan data.
2. Mengimplementasikan sebuah fitur yang dapat menyajikan informasi ketersediaan jadwal gedung secara *real-time* dan transparan untuk mencegah terjadinya konflik jadwal.
3. Mengimplementasikan sebuah sistem yang mampu menyediakan rekam jejak digital dari seluruh aktivitas reservasi untuk mempermudah proses pengawasan oleh pihak sekolah.

4. Mengukur efektivitas dan tingkat penerimaan aplikasi yang dibangun oleh pengguna akhir melalui pengujian fungsionalitas dan pengalaman pengguna.

1.4. Manfaat Penelitian

Adapun manfaat yang didapatkan dari penelitian ini adalah sebagai berikut:

1. Membantu pengelolaan sistem reservasi gedung di SMAN 1 Tanjung Bintang, sehingga dapat mengurangi kesalahan sistem reservasi gedung yang selama ini dilakukan secara manual.
2. Memberikan transparansi serta akuntabilitas dalam sistem reservasi gedung di SMAN 1 Tanjung Bintang yang dapat memberikan informasi yang lebih akurat ke pengguna.
3. Menghasilkan suatu aplikasi *mobile* berbasis android yang dapat digunakan untuk reservasi gedung yang akan digunakan untuk kegiatan ekstrakurikuler di SMAN 1 Tanjung Bintang.
4. Mengembangkan aplikasi yang fleksibel dan dapat dikonfigurasi dengan mudah untuk digunakan oleh sekolah-sekolah lain di masa depan, sehingga aplikasi ini dapat menjadi solusi manajemen reservasi gedung yang lebih luas.

1.5. Batasan Masalah

Dalam perancangan dan implementasi aplikasi ini, terdapat beberapa batasan masalah agar penelitian ini lebih terarah dan memudahkan dalam penyelesaian penelitian ini, yaitu :

1. Penelitian ini berfokus hanya pada pengembangan aplikasi berbasis android.
2. Sistem dibangun dengan menggunakan *framework flutter* dengan *firebase*.
3. Sistem berfungsi hanya ketika pengguna terhubung dengan jaringan internet.
4. Sistem ini memiliki fokus hanya pada manajemen reservasi gedung SMAN 1 Tanjung Bintang untuk kegiatan ekstrakurikuler.

1.6. Sistematika Penulisan

Sistematika penulisan memuat lima bab yang di mana inti dari tiap-tiap bab bertujuan untuk membuat penulisan ini menjadi penulisan yang baik dan benar. Adapun sistematika penulisan dari penelitian ini sebagai berikut:

- I. PENDAHULUAN: Berisi latar belakang, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan masalah, manfaat penelitian dan sistematika penulisan.
- II. TINJAUAN PUSTAKA: Membahas mengenai dasar teori, beberapa istilah bertujuan untuk memahami penelitian mengenai Rancang Bangun Aplikasi Sistem Informasi Manajemen Reservasi Gedung untuk Kegiatan Organisasi ekstrakurikuler pada SMAN 1 Tanjung Bintang Berbasis Android. Bab ini berisi tentang pengertian Reservasi Gedung, Sistem Informasi, *Android*, *Flutter*, *Dart*, *Kanban*, *Trello*, *UML*, *Firebase*, dan *Testing*.
- III. METODE PENELITIAN: Memuat waktu dan tempat penelitian, jadwal penelitian, alat penelitian kemudian tahapan penelitian yang terdiri dari *Analysis*, *Design*, Perancangan papan *kanban*, Pengembangan dengan menggunakan metode *kanban*, *Deploy* dan penulisan laporan.
- IV. HASIL DAN PEMBAHASAN: Pada bab ini memuat tentang hasil serta pembahasan yang diperoleh dalam penelitian.
- V. KESIMPULAN: Memuat tentang kesimpulan dari hasil penelitian dan saran dari hasil penelitian untuk pengembangan penelitian selanjutnya.

DAFTAR PUSTAKA: Memuat berbagai sumber referensi yang digunakan dalam penelitian.

LAMPIRAN: Memuat lampiran-lampiran penelitian.

II. TINJAUAN PUSTAKA

2.1 Reservasi Gedung

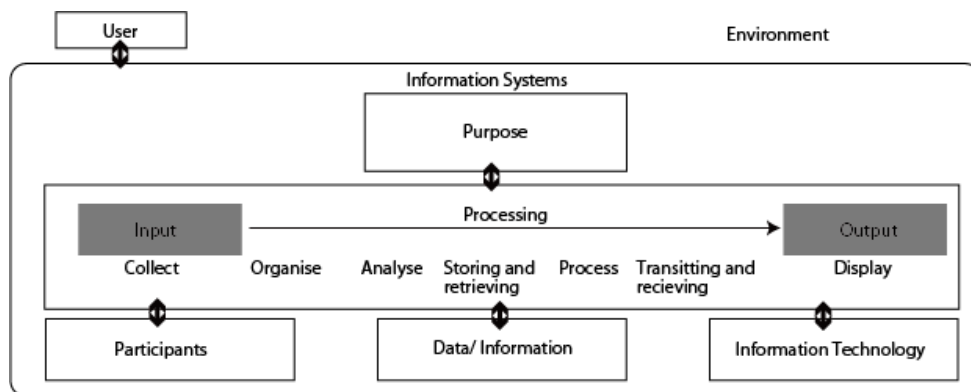
Reservasi adalah proses membuat kesepakatan untuk memesan barang atau jasa, di mana sudah ada kesepakatan antara konsumen dan penyedia barang atau jasa, tetapi belum ada transaksi sebenarnya. Saat melakukan reservasi, biasanya terjadi pertukaran informasi antara kedua belah pihak agar kesepakatan tersebut bisa terwujud. Secara umum, reservasi adalah perjanjian atau pemesanan tempat antara dua pihak atau lebih, seperti pemesanan ruangan, kamar, tempat duduk, atau lainnya pada waktu atau periode tertentu, serta dilengkapi dengan pelayanan jasanya [2].

Reservasi gedung yang akan digunakan dalam penelitian ini yaitu reservasi gedung untuk melaksanakan kegiatan ekstrakurikuler. Dikarenakan bahwa dalam kegiatan ekstrakurikuler diperlukan gedung yang memadai karena dapat memberikan tempat yang nyaman sesuai dengan kebutuhan siswa dalam menjalani kegiatan ekstrakurikulernya. Gedung dalam kegiatan ekstrakurikuler sangat penting mengingat gedung dapat menjadi tempat kegiatan terlaksana dengan nyaman, aman dan sesuai dengan apa yang dibutuhkan oleh siswa. Gedung yang memadai dapat menjadi lingkungan yang membawa dampak baik terhadap siswa dalam meningkatkan keterampilan dan bakat mereka sehingga mereka dapat mengoptimalkan kemampuannya.

2.2 Sistem Informasi

Pengertian Sistem informasi menurut Jonny Seah adalah gabungan dari berbagai komponen teknologi informasi yang bekerja sama untuk menghasilkan informasi. Informasi ini berguna untuk membangun komunikasi dalam sebuah organisasi atau kelompok. Sistem ini bisa memenuhi kebutuhan dalam memproses transaksi sehari-hari, mendukung operasional, manajemen, serta kegiatan strategis organisasi. Selain itu, sistem ini juga bisa memberikan laporan yang diperlukan kepada pihak luar tertentu [3].

Dari pernyataan di atas dapat disimpulkan bahwa sistem informasi sangat penting dalam suatu proses sistem. Dari proses sistem tersebut, kita bisa mengetahui permasalahan yang sedang dihadapi, sehingga bisa menilai apakah sistem informasi tersebut masih bisa beroperasi dengan baik atau tidak, baik sistem yang manual maupun sistem yang sudah canggih. Jika sistem informasi tersebut sudah tidak layak digunakan lagi, maka perlu dilakukan pembaharuan atau upgrade sistem agar sistem dapat berjalan lebih baik dan dalam proses pengolahan data bisa lebih akurat [4].



Gambar 2.1 Diagram Sistem Informasi

Sistem informasi ialah bentuk dari sebuah komunikasi sistem yang dapat mewakili data serta diproses sebagai bentuk dari memori sosial. Sebelum ada sistem informasi, sebagian pekerjaan dikerjakan secara manual dan memerlukan waktu yang lumayan lama sehingga memungkinkan terjadi sebuah kesalahan yang bisa berakibat fatal. Dengan adanya sistem informasi akan mempermudah pekerjaan secara akurat dan cepat [5].

2.3 Android

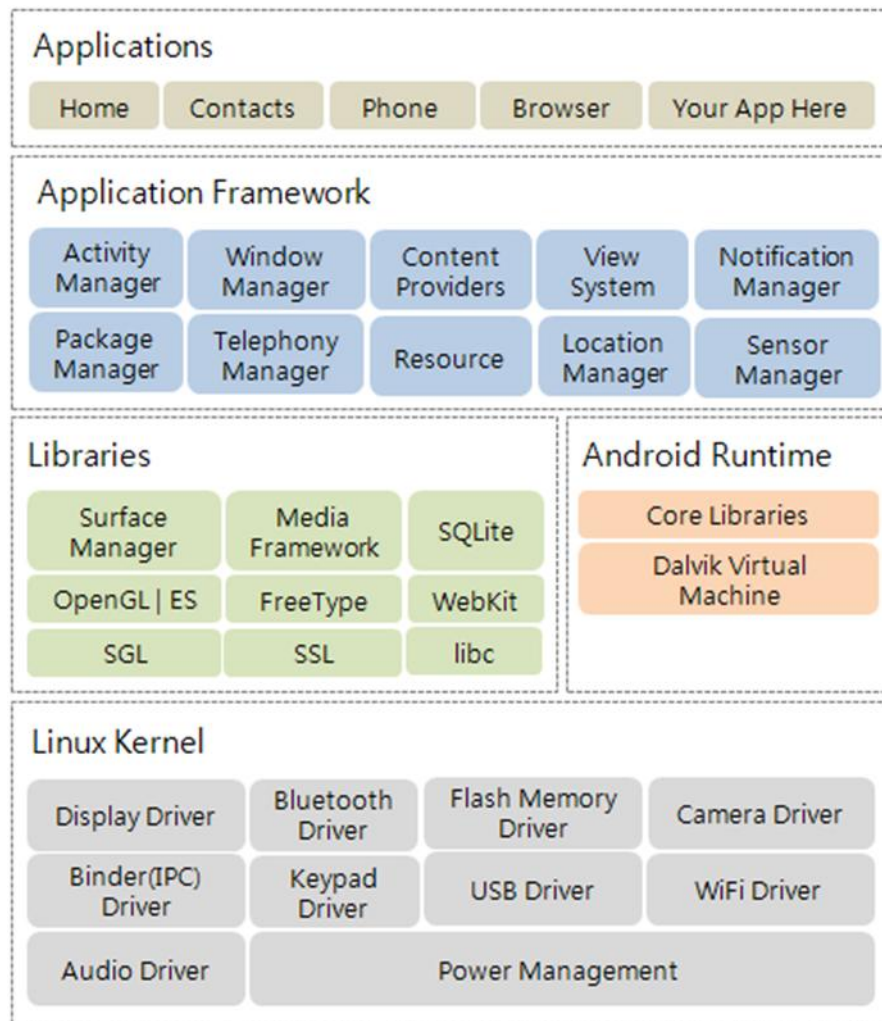
Sejarah pencipta android seperti Andy Rubin, Chris White, dan Nick Sears awalnya ingin membuat perangkat kamera digital. Namun, nanti pengguna mereka beralih menjadi *smartphone*. Tahun 2007, Android dirilis secara resmi oleh Google. Sebelumnya, Google membeli Android pada tahun 2005 dengan dana 50 juta dolar. Setelah mendapat dukungan dari Google, para pendiri Android bergabung dengan perusahaan tersebut dan membuat sebuah *prototype*. Pada tahun 2006, *prototype* yang dihasilkan adalah Blackberry, yang memiliki *keyboard*. Lalu, perusahaan Apple merilis *smartphone* pertama dengan *keyboard* layar sentuh, yaitu iPhone. Karena itu, Google mulai mengembangkan *smartphone* dengan *keyboard* layar sentuh agar bisa bersaing dengan iPhone [6].

Menurut Naszrudin Safaat android adalah sebuah sistem operasi yang diperuntukan perangkat *mobile* yang berbasis *linux* yang mencakup *middleware*, sistem operasi dan aplikasi. Android menyajikan platform terbuka bagi para *developer* untuk membuat aplikasi mereka sendiri [7]. Dari penjelasan tersebut, dapat disimpulkan bahwa Android adalah sistem operasi berbasis Linux yang dibuat oleh perusahaan Android Inc. Sistem ini digunakan pada *smartphone* dan gadget, serta bersifat sumber terbuka, yang memungkinkan para pengembang membuat dan membagikan berbagai aplikasi sesuai kebutuhan masyarakat.



Gambar 2.2 Android

Android adalah sistem operasi yang menggunakan *kernel* Linux dan dikembangkan oleh Google. Tujuannya adalah untuk mendukung perangkat elektronik berbasis layar sentuh. Android merupakan sistem sumber terbuka, sehingga bisa diubah, diperbaiki, dan didistribusikan oleh pengembang perangkat lunak. Gambar 3 berikut merupakan arsitektur android:



Gambar 2.3 Android

Pembuatan aplikasi Android bisa dilakukan dengan berbagai platform, tetapi saat ini yang paling sering digunakan adalah Android Studio. Android Studio adalah program yang memiliki berbagai fasilitas yang dibutuhkan untuk membuat aplikasi Android, yang biasanya disebut juga sebagai *Integrated Development Environment* atau disingkat dengan IDE.

Banyak orang dari bermacam kalangan usia yang sudah mengenal dan menggunakan Android sebagai alat komunikasi serta media hiburan yang praktis, karena bisa dibawa ke mana saja tanpa dibatasi oleh ruang dan waktu. Android mampu menarik pengguna karena memiliki berbagai fungsi yang terus berkembang sesuai dengan perkembangan teknologi. Saat ini, Android telah memiliki berbagai fungsi yang terus berkembang, antara lain :

- a. Diperuntukan untuk menyimpan data
- b. Membuat catatan pekerjaan ataupun perencanaan pekerjaan
- c. Mencatat janji pertemuan (*appointment*) disertai pengingat waktu (*reminder*)
- d. Menghitung perhitungan dasar sederhana kalkulator
- e. Mengirim serta menerima pesan.
- f. Mencari informasi (berita, hiburan, dan informasi lain) dari internet
- g. Memainkan berbagai permainan-permainan sederhana
- h. Integrasi ke bermacam peralatan lain seperti PDA, *MP3 player*, dan GPS (*Global Positioning System*) [8].

Berdasarkan kegunaan android yang telah dijabarkan, android mempunyai kegunaan yang baik dalam bidang pendidikan. Seperti membuat catatan pekerjaan ataupun perencanaan pekerjaan sekolah yang membutuhkan gedung untuk reservasi acara kegiatan ekstrakurikuler. Dengan lewat suatu aplikasi yang ada di android dapat memudahkan sekolah dan *User* untuk mencari data gedung yang dapat di reservasi dan memesan gedung untuk kegiatan ekstrakurikuler.

2.4 Flutter

Flutter adalah sebuah SDK (*Software Development Kit*) yang digunakan untuk membuat aplikasi *mobile*, dan dikembangkan langsung oleh Google. *Flutter* memiliki performa yang baik dan bisa digunakan untuk membuat aplikasi yang berjalan di Android dan iOS menggunakan satu *codebase* saja. *Flutter* memakai bahasa pemrograman *Dart*, yang dianggap mudah dipelajari jika sudah terbiasa dengan bahasa seperti Java atau *JavaScript*. Selain itu, *Flutter* juga dilengkapi dengan kerangka kerja berbasis *reactive-functional*, mesin *render* 2D, *widget* siap pakai, serta berbagai *tools* yang membantu dalam pengembangan aplikasi [9].



Gambar 2.4 Flutter

Dengan *Flutter*, pengembang bisa membuat aplikasi untuk Android dan iOS menggunakan bahasa yang serupa, yakni *Dart*, yang dibuat oleh Google pada tahun 2011. Sebelum ada *Flutter*, aplikasi untuk Android biasanya dibuat menggunakan Java atau *Kotlin*, sedangkan aplikasi iOS dibuat dengan *Objective-C* atau *Swift*. *Flutter* dirancang untuk mempermudah serta mempercepat proses pembuatan aplikasi *mobile* yang bisa berjalan di Android dan iOS tanpa harus menguasai dua bahasa pemrograman secara terpisah. [10].

2.5 Dart

Dart adalah bahasa pemrograman yang diciptakan oleh Google, dan dirancang oleh Lars Bak serta Kasper Lund. Bahasa ini bisa digunakan untuk membuat aplikasi server, antarmuka perintah baris (*command line interface*), web, atau *mobile* (Android dan iOS). *Dart* mendukung pendeklarasian fungsi di luar kelas, yang disebut juga dengan top-level *function*. Pada *Dart*, kode utama disimpan di dalam fungsi *main*, sama bak di C/C++ [9].

Dart adalah bahasa yang bersifat *open source*. *Dart* memakai konsep berorientasi objek dengan *sintaks* yang mirip dengan C. Hal ini mendukung konsep seperti antarmuka dan kelas. Berbeda dengan bahasa pemrograman yang lainnya, *Dart* tidak mendukung *array* secara langsung, tetapi bisa *mereplikasi* seperti struktur data *array*, *generic*, serta pengetikan opsional [11].

Dart menggunakan tipe statis, yang artinya sebelum digunakan, *variable* harus didefinisikan terlebih dahulu. *Dart* dapat berjalan di berbagai perangkat. Untuk web, *Dart* menggunakan *dart2js* yang mengubah kode *Dart* menjadi *JavaScript* agar bisa diakses oleh browser. Dalam perangkat desktop, *Dart* menggunakan *dart2aot* yang mengubah *Dart* menjadi bahasa mesin. Untuk perangkat *mobile*, *Dart* menggunakan *Flutter*. *Sintaks* dalam *Dart* mudah dipelajari karena memiliki kesamaan dengan *sintaks* bahasa pemrograman lainnya, seperti *JavaScript* dan Java [12].

2.6 Kanban

Kata *Kanban* bermula dari Bahasa Jepang yang artinya secara harfiah "lihat dan tulis" atau "bagian yang terlihat". Secara umum, istilah ini merujuk pada suatu sinyal tertentu dalam bidang manufaktur, yaitu kartu *Kanban*. *Kanban* adalah sebuah metode dalam tata laksana proyek yang digunakan untuk menyederhanakan dan mengelola proses produksi. Metode ini pertama kali dikembangkan oleh Taiichi Ohno, yang merupakan insinyur industri di Toyota, dengan intensi meningkatkan efisiensi dalam proses manufaktur [13].



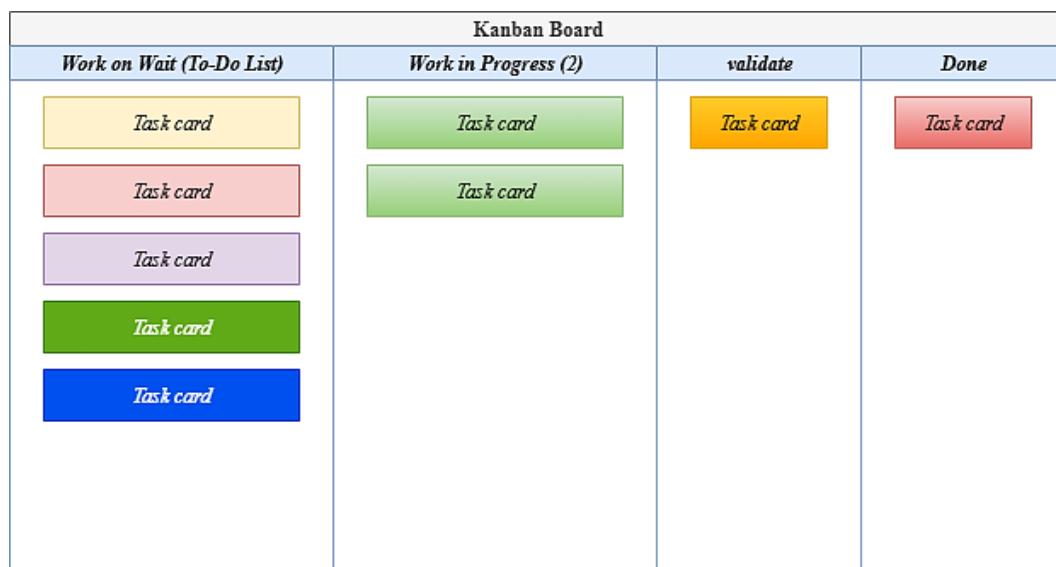
Gambar 2.5 kanban dalam proses manufaktur

Kanban merupakan metode manajemen alur kerja yang membantu organisasi mengelola dan meningkatkan layanan yang mengasung pekerjaan pengetahuan untuk membantu memvisualisasikan pekerjaan, memaksimalkan efisiensi, dan meningkatkan secara berkelanjutan. Pekerjaan diwakilkan di papan *kanban*, untuk memungkinkan mengoptimalkan penyampaian kerja di beberapa tim. Dengan fokus yang lebih besar pada efisiensi serta pemanfaatan kemajuan teknologi komputasi, metode *Kanban* mulai meninggalkan bidang industri otomotif dan berhasil diterapkan di berbagai sektor komersial yang kompleks. lunak, R&D, dan lain-lain [14].

Papan *Kanban* dasar mencakup tiga kolom utama termasuk:

- *To do* : semua tugas yang belum dimulai dicantumkan di kolom ini. Umumnya, tugas diurutkan berdasarkan waktu kedatangannya, namun terkadang hal ini dilakukan sesuai dengan prioritas.
- *In progress* : kolom ini dialokasikan untuk tugas-tugas yang sedang dikerjakan di momen.
- *Done* : menunjukkan tugas yang telah selesai.

Pendekatan sederhana ini dapat memberikan kejelasan besar tentang apa yang harus dilakukan dan apa yang harus dilakukan menunjukkan hambatan yang ada, jika ada. Tentu, papan dapat diperluas untuk mencakup sebagai banyak kolom sesuai kebutuhan untuk kerangka kerja [15].



Gambar 2.6 Kanban Board.

Metode *kanban* berupa *work flow* yang memvisualisasikan pekerjaan serta memaksimalkan efisiensi. Pada *Kanban* yang akan digunakan dalam penelitian ini, terdapat kolom-kolom yang termasuk ke dalam *flow*, seperti *To do* atau *Work On Wait (WOW)*, *In progress* atau *Work In Progress (WIP)*, *validate*, dan *Done* sesuai dengan kebutuhan kerangka kerja.

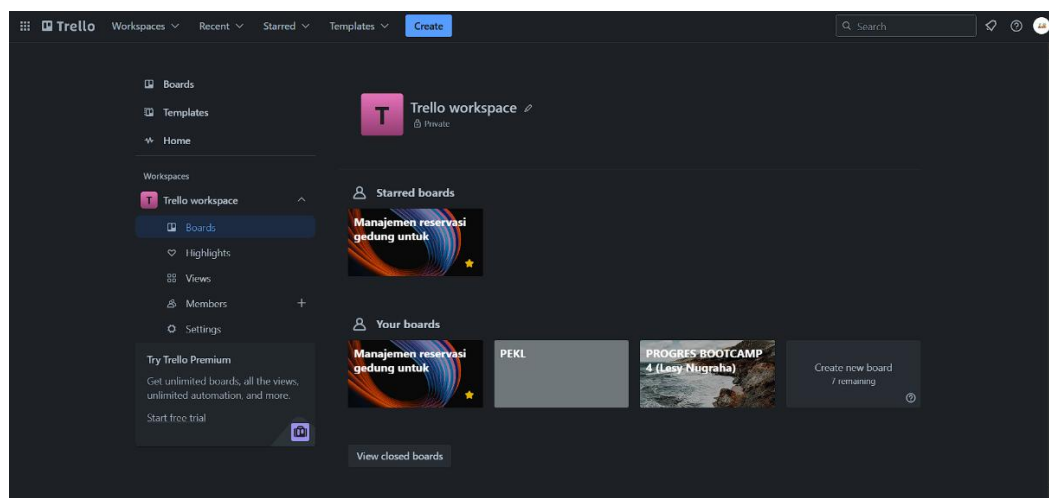
Batasan pada *WiP* akan mengubah sistem “*push*”. Menjadi sistem “*pull*”, di mana *task card* baru tidak dimulai sampai pekerjaan selesai (atau pada kesempatan yang lebih jarang, dibatalkan). Memiliki terlalu banyak pekerjaan yang selesai sebagian adalah pemborosan dan mahal dan, yang terpenting, hal ini memperpanjang waktu tunggu, sehingga menghambat kelangsungan organisasi

responsif terhadap pelanggan mereka dan terhadap perubahan keadaan dan peluang. Mengamati, membatasi, dan kemudian mengoptimalkan jumlah pekerjaan yang masuk kemajuan sangat penting untuk kesuksesan dengan *Kanban* [13].

2.7 Trello

Trello adalah aplikasi yang digunakan untuk mengelola tugas dalam sebuah tim yang terdiri dari banyak anggota. Aplikasi ini bisa memberi tugas kepada anggota tim dan memantau apakah tugas tersebut sudah selesai atau belum. Hal ini karena *Trello* memiliki fitur *action* yang bisa digunakan. Dalam *Trello*, pembagian tugas dilakukan di *board*. Di *board* tersebut, pengguna bisa melihat kemajuan kerja tugas. *Trello* juga memungkinkan untuk mengunggah berbagai jenis dokumen, seperti RAR, dokumen pengolah kata, dokumen angka, gambar, dan lainnya [16].

Trello adalah alat kolaborasi yang mengatur proyek atau tugas dalam bentuk papan digital, sehingga bisa dibawa ke mana saja dengan mudah melalui ponsel atau browser. *Trello* dirancang dengan prinsip manajemen kerja yang menekankan pada visualisasi dan kerja sama. Alat ini memungkinkan pengguna membuat daftar tugas atau pekerjaan yang kemudian diperwakili oleh kartu-kartu yang bisa ditempatkan di berbagai kolom atau daftar yang menunjukkan tahap atau status proyek [17].



Gambar 2.7 Tampilan *Trello* pada halaman beranda.

Trello bisa dibayangkan seperti papan tulis yang berisi beberapa catatan, dimana setiap catatan berisi tugas yang diberikan kepada suatu tim. Di dalam catatan tersebut, pengguna bisa menempelkan foto atau sumber data lainnya. Artinya, *Trello* bisa memberi tahu pengguna apa yang sedang dikerjakan, siapa yang bertanggung jawab, serta sampai di mana tahapan suatu proyek yang sedang berlangsung [18].

2.8 UML (*Unified Modeling Language*)

Unified Modeling Language atau sering disingkat sebagai UML adalah sebuah bahasa yang digunakan untuk menentukan, menggambarkan, membuat, dan mendokumentasikan berbagai bagian dari sistem perangkat lunak, baik yang sedang direncanakan maupun sedang dikembangkan. Dengan pendekatan ini, sistem dianggap sebagai suatu objek yang mencakup data dan proses, atau dapat beroperasi secara independen dalam suatu set sistem [19].




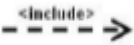
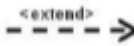
UML sangat penting karena berfungsi sebagai alat untuk menyampaikan atau menerjemahkan apa yang diinginkan pengguna ke dalam bentuk yang bisa dipahami oleh pengembang sistem. UML merupakan sebuah bahasa pemodelan perangkat lunak yang telah ditetapkan secara resmi dan digunakan sebagai alat untuk merancang dan memperjelas arsitektur perangkat lunak. UML digunakan untuk memahami, menentukan, membangun, dan mendokumentasikan segala hal yang terjadi dalam lingkungan yang rumit [20]. Dalam konsep dasar tersebut terdapat beberapa diagram yang mendefinisikan tiap-tiap konsep dasar *UML*. Di antaranya sebagai berikut:

1. Rancangan Diagram (*Use case*)

Dalam diagram *use case* admin maupun kasir menggambarkan *actor* dalam *use case* ini yang menjalankan tugas sesuai dengan hak akses di dalam sistem. Diagram *use case* digunakan untuk menunjukkan bagian-bagian fungsional dari sebuah sistem. Setiap *use case* menunjukkan sebuah aktivitas yang dilakukan oleh aktor dengan sistem. *Use case* merupakan tugas yang spesifik, seperti *login* ke dalam sistem, membuat daftar belanja, dan sebagainya. Aktor adalah entitas

seperti manusia atau mesin yang berinteraksi dengan teknik prosedur untuk melakukan tugas-tugas tertentu.. Adapun komponen *use case* sebagai berikut:

Tabel 2.1 Simbol dan keterangan komponen *use case diagram*.

Nama	Simbol	Keterangan
<i>Actor</i>		Merepresentasikan seseorang atau entitas yang berkorelasi dengan sistem tetapi tidak memegang kendali atas <i>use case</i> .
<i>Use case</i>		Merepresentasikan gambaran fungsional sebuah sistem.
Relasi <i>Asosiasi</i>		Merepresentasikan sebuah interaksi antara aktor dengan <i>use case</i> tertentu.
Relasi <i>Include</i>		Merepresentasikan hubungan <i>use case</i> tambahan di mana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya. Arah panah mengarah kepada <i>use case</i> tambahan.
Relasi <i>Extend</i>		Merepresentasikan <i>use case</i> tambahan di mana <i>use case</i> yang ditambahkan dapat berdiri sendiri. Orientasi panah mengarah pada <i>use case</i> yang ditambahkan.

2. Diagram Aktivitas (*Activity Diagram*)

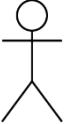



Diagram aktivitas menunjukkan alur atau proses dalam sistem atau perangkat lunak. Misalnya, alur aktivitas di kasir yang terhubung dengan sistem. Diagram aktivitas menjelaskan bagaimana berbagai aktivitas berlangsung dalam sistem yang sedang dirancang, mulai dari bagaimana aktivitas dimulai, keputusan yang mungkin muncul, hingga bagaimana aktivitas tersebut berakhir. Diagram aktivitas juga bisa menunjukkan proses yang terjadi secara bersamaan dalam beberapa eksekusi. Diagram aktivitas adalah jenis diagram *state* khusus di mana sebagian besar *state* berupa

tindakan, dan sebagian besar transisi diaktifkan setelah *state* sebelumnya selesai.

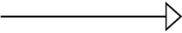
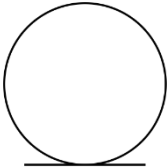
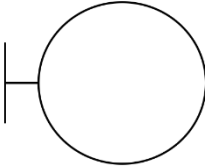
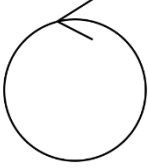
3. *Class diagram*

Class diagram merupakan jalinan antara *class* sudah terdapat nama *class*, *attribute* dan *method*. *Class* merupakan spesifikasi yang jika diubah menjadi objek akan menghasilkan suatu objek, dan merupakan inti dari pengembangan dan desain berbasis objek. *Class* menjelaskan kondisi atau sifat-sifat dari suatu sistem, serta memberikan layanan untuk mengubah atau mengatur kondisi tersebut. Diagram *class* mengilustrasikan struktur dan penjelasan tentang *class*, *package*, dan objek beserta hubungan antara satu sama lain ibarat *containment*, pewarisan, asosiasi, dan lainnya.

Tabel 2.2 simbol dan keterangan *Class diagram*.

Nama	Simbol	Keterangan
<i>Actor</i>		Merepresentasikan entitas yang ada di luar sistem dan berinteraksi dengan sistem tersebut. Bisa berbentuk orang, proses, atau sistem lain yang berkomunikasi dengan sistem.
Objek		Memberitahukan objek yang berinteraksi dengan pesan.
<i>Life Line</i>		Memberitahukan kehidupan suatu objek.
Waktu Aktif		Memberitahukan objek dalam keadaan aktif dan berinteraksi.

Tabel 2.2 simbol dan keterangan *Class diagram* (lanjutan).

Nama	Simbol	Keterangan
<i>Message</i>		Menerangkan pengiriman pesan. Ini adalah komunikasi antar objek yang menunjukkan tindakan yang akan dilakukan.
<i>Entity Class</i>		Menunjukkan suatu penyimpanan data atau informasi
<i>Boundary Class</i>		Merupakan bagian dari sistem, bisa berupa antarmuka atau alat yang berinteraksi dengan sistem lain.
<i>Control Class</i>		Elemen yang mengatur aliran informasi dalam sebuah skenario. Objek ini biasanya mengatur sikap dan proses bisnis.

2.9 Firebase

Firebase merupakan sebuah layanan yang dipunyai oleh *google* dan diperuntukan untuk mempermudah para developer aplikasi dalam membangun aplikasinya. *Firebase Cloud Service Provider* dan *Backend as a Service* merupakan penyelesaian yang ditawarkan oleh *google* untuk memudahkan pekerjaan pengembang dalam pengembangan aplikasi *mobile* maupun web. *Firebase* mempunyai bermacam SDK yang memungkinkan untuk mengintegrasikan layanan ini dengan beberapa platform seperti *Android*, *Ios*, *JavaScript*, *C++*, hingga *unity*. Sebenarnya *firebase* pertama kali didirikan oleh Andrew Lee dan James Tamplin pada tahun 2011. Produk pertama kali yang dikembangkan oleh mereka adalah *Realtime database* [21].

Firebase mempunyai layanan fundamental, yaitu menyediakan *database realtime* dan *backend* sebagai layanan (*Backend as a Service*). *Backend* merupakan suatu bagian dalam *code* aplikasi yang di mana berhubungan langsung dengan isi *database*. Dengan adanya *firebase*, developer aplikasi nantinya tidak perlu membuat *backend* secara sendiri melainkan menggunakan API yang telah disediakan *firebase* sehingga proses pengembangan aplikasi dapat dipersingkat.

Firebase dibangun dengan memakai *database Mon-goDB* sehingga *firebase* memakai tipe *database NoSQL*. Dikarenakan *firebase* menggunakan tipe *database NoSQL*, oleh karenanya struktur *database* dari *firebase* bersifat fleksibel sehingga pas untuk digunakan pada aplikasi berbasis *mobile*. Layanan *firebase* ini menyediakan pengembangan aplikasi API yang memungkinkan aplikasi data yang selanjutnya disinkronisasi di klien dan disimpan di *cloud firebase* ini. *Firebase* menyediakan *library* untuk berbagai *client* platform yang memungkinkan integrasi dengan *Android*, *Ios*, *Javascript*, *Java*, *Objective-C*, dan *Node* aplikasi Js dengan konsep *realtime* [22].

2.10 Testing (Pengujian)

Pengujian perangkat lunak merupakan suatu proses memverifikasi dan memvalidasi apakah aplikasi maupun program perangkat lunak telah memenuhi persyaratan bisnis serta teknis yang memandu desain dan pengembangannya, dan apakah berperilaku sesuai yang diharapkan. Pengujian perangkat lunak juga merupakan teknik yang umum digunakan untuk memeriksa dan memverifikasi kualitas perangkat lunak [23]. Pada sistem informasi reservasi gedung ini dijalankan dua pengujian yaitu pengujian pertama dengan pengujian menggunakan *blackbox* testing. Kemudian pengujian kedua dilakukan ketika sistem informasi telah selesai dibuat yaitu dengan menggunakan pengujian pengalaman pengguna yaitu *user experience questionnaire*.

1. *Blackbox* testing

Blackbox testing disebut sebagai pengujian fungsional, adalah metode pengujian perangkat lunak yang berfokus pada fungsionalitas eksternal suatu sistem. Penguji memperlakukan sistem sebagai “kotak hitam”, hanya melihat

input dan *output* dan tidak dapat mengakses kode sumber internal. Metode pengujian ini didasarkan pada spesifikasi sistem dan bertujuan untuk memastikan bahwa perangkat lunak menghasilkan *output* yang sesuai. Dengan menggunakan metode ini, penguji dapat menilai sistem dari perspektif pengguna dengan fokus pada pengalaman pengguna tanpa membutuhkan pengetahuan mendalam tentang cara aplikasi bekerja di dalamnya [24].

Black box Testing, yang juga dikenal sebagai pengujian fungsional, adalah metodologi pengujian perangkat lunak yang berfokus pada fungsionalitas eksternal suatu sistem. Dalam pendekatan ini, penguji tidak memiliki akses ke kode sumber internal dan memperlakukan perangkat lunak sebagai “kotak hitam,” hanya mengamati *input* dan *output*. Proses pengujian didasarkan pada spesifikasi dan persyaratan sistem, bertujuan untuk memastikan bahwa perangkat lunak menghasilkan *output* sesuai dengan harapan saat diberikan *input* tertentu dalam kondisi eksekusi tertentu. Metode ini memungkinkan penguji untuk menilai sistem dari sudut pandang pengguna, menekankan pengalaman pengguna tanpa memerlukan pengetahuan mendalam tentang cara kerja internal aplikasi [25].

Salah satu metode pengujian *black box* yang paling terstruktur dan efektif adalah *Decision Table Testing* atau pengujian tabel keputusan. Teknik ini digunakan untuk menguji bagaimana sistem merespons berbagai kombinasi *input* yang rumit, dengan hasilnya disusun dalam bentuk tabel. Pendekatan ini juga disebut tabel sebab-akibat karena kemampuannya untuk menunjukkan hubungan logis antara kondisi (sebab) dan tindakan (akibat) yang dihasilkan oleh program [26].

Penerapan *decision table* dianggap penting dalam merancang pengujian karena beberapa alasan. Pertama, teknik ini membantu penguji untuk menganalisis dampak dari berbagai kombinasi *input* secara teratur. Kedua, tabel keputusan memberikan cara yang jelas untuk menjelaskan aturan bisnis yang kompleks, sehingga berguna bukan hanya bagi penguji tetapi juga bagi pengembang (developer). Selain itu, metode ini juga menjadi latihan yang terorganisir untuk menyusun skenario pengujian ketika menghadapi logika sistem yang rumit [26].

Struktur dasar dari *decision table* dapat divisualisasikan seperti pada Tabel berikut ini [27].

Tabel 2.3 Aturan *input decision table*.

Kondisi	R1	R2	R3	R4
<i>Input (T/F)</i>	F	T	F	F
<i>Input (T/F)</i>	F	F	T	F
<i>Output (E/S)</i>	E	E	E	S

Keterangan : T = *True*

E = *Error*

F = *False*

S = *Success*

R = *Rule* pengujian

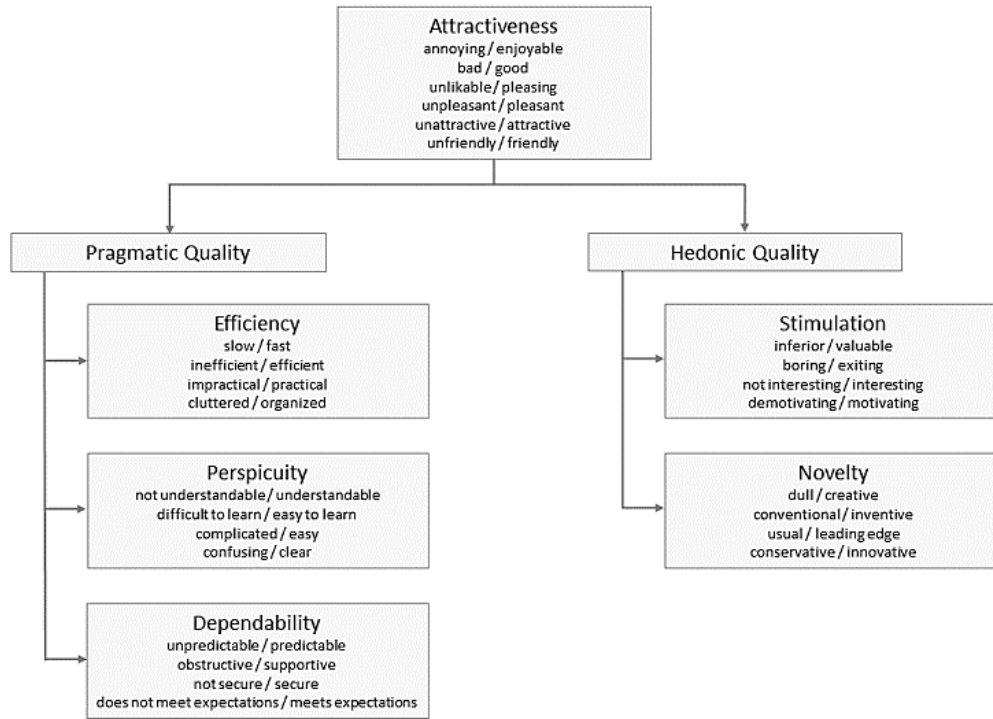
R1 = *Rule* pengujian 1,dst.

2. *User Experience Questionnaire (UEQ)*

User Experience Questionnaire merupakan survei yang menilai pengalaman subjektif pengguna terhadap suatu produk. Minimal dibutuhkan 20-30 responden untuk memberikan hasil yang stabil berdasarkan buku panduan penggunaan UEQ. *User Experience Questionnaire* memiliki 26 item yang dikelompokkan menjadi enam proporsi [28]:

1. *Attractiveness* (daya tarik): kesan menyeluruh pada produk?
2. *Perspiciuity* (kejelasan): tentang kemudahan saat menggunakan produk dan kemudahan saat mengenali produk tersebut?
3. *Efficiency* (efisiensi): tentang seberapa efisien serta cepatnya reaksi produk terhadap masukan dari pengguna?
4. *Dependability* (ketepatan) : perasaan pengguna saat bekerja dengan produk dan pengguna dapat mengendalikan interaksi?
5. *Stimulation* (stimulasi) : kesenangan saat melakukan interaksi dengan produk.

6. *Novelty* (kebaruan) : seberapa inovatif serta kreatifnya sebuah produk sehingga bisa menarik perhatian pengguna?



Gambar 2.8 Pertanyaan yang digunakan dalam metode *User Experience Questionnaire (UEQ)* [28].

Pada tahapan ini item pertanyaan diskalakan dari -3 sampai dengan +3 merupakan jawaban paling negatif -3, 0 menandakan netral, dan +3 menandakan paling positif. Dalam evaluasi jika nilai rata-rata $> 0,8$ maka akan mendapatkan evaluasi positif, jika nilai terdapat pada rentang $-0,8 - 0,8$ maka mendapatkan evaluasi normal dan $< -0,8$ merupakan evaluasi negatif [28].

2.11 Penelitian Terdahulu

Adapun penelitian terdahulu yang memiliki topik yang relevan dengan topik penelitian ini adalah sebagai berikut.

Tabel 2.4 Tabel Penelitian Terdahulu.

No.	Penulis (Tahun)	Judul	Isi
1	Rendy Rian Chrisna Putra dan Eza Budi Perkasa (2019).	Aplikasi Peminjaman Ruangan Kantor Gubernur Provinsi Kepulauan Bangka Belitung Berbasis Android.	Penelitian berisi tentang pengembangan aplikasi Android untuk mempermudah peminjaman ruangan rapat di Kantor Gubernur, menggantikan sistem manual yang kurang efisien. Menggunakan model <i>Prototype</i> , mereka mengumpulkan kebutuhan melalui wawancara, observasi, dan studi pustaka, lalu mengembangkan dan menguji <i>prototype</i> dengan Java di Android Studio serta <i>backend</i> PHP dan <i>Mysql</i> . Aplikasi ini juga memanfaatkan <i>Unified Modelling Language</i> (UML) untuk pemodelan sistem. Pengujian <i>blackbox</i> menunjukkan aplikasi berfungsi dengan baik, memudahkan peminjaman ruangan, dan membantu pengelolaan peminjaman secara efektif, sehingga proses menjadi lebih efisien dan terorganisir [29].
2	Kadek Dika Sukmana, Ketut Queena Fredlina dan Putu Trisna Hady Permana (2022).	Model Aplikasi Peminjaman Ruangan Berbasis Web Pada Tingkat Fakultas di Perguruan Tinggi.	Penelitian berisi tentang pengembangan aplikasi web untuk peminjaman ruangan di tingkat fakultas guna meningkatkan efisiensi dan efektivitas proses manual sebelumnya. Menggunakan metode SDLC, aplikasi ini menyediakan antarmuka peminjaman <i>online</i> yang mudah digunakan, statistik peminjaman, dan laporan untuk manajemen fakultas. Studi kasus dilakukan di Fakultas Ekonomi dan Bisnis Universitas Udayana, dan aplikasi diuji dengan <i>Blackbox</i> Testing untuk memastikan semua fitur berfungsi dengan baik. Fitur utama mencakup kemampuan pemesanan ruangan <i>online</i> , melihat ketersediaan, dan konfirmasi peminjaman. Hasil penelitian menunjukkan aplikasi ini berhasil meningkatkan efisiensi peminjaman, mengurangi beban administratif, dan mempermudah pengguna dalam melakukan peminjaman tanpa harus berurusan langsung dengan petugas, sehingga proses peminjaman menjadi lebih cepat dan terorganisir [30].

Tabel 2.4 Tabel Penelitian Terdahulu (lanjutan).

No.	Penulis (Tahun)	Judul	Isi
3	Andi Ahmad Prasetya (2018).	Sistem Informasi Reservasi Gedung Serbaguna di Kota Palembang Berbasis Android.	Penelitian berisi tentang pengembangan sistem reservasi gedung serbaguna berbasis Android untuk membantu penyewa mengakses informasi jadwal, harga, fasilitas, dan paket gedung. Sistem ini dibangun dengan metode <i>Prototype</i> menggunakan UML, Java untuk Android, PHP untuk web, dan <i>Mysql</i> . Aplikasi mencakup halaman utama, fasilitas, paket, status pemesanan, kelola gedung, kelola acara, dan jadwal. Pengujian <i>Blackbox</i> memastikan minim kesalahan. Hasilnya, sistem memudahkan penyewa melihat jadwal dan memesan gedung tanpa datang langsung, serta mencatat transaksi dan menyediakan laporan grafik untuk admin. Kesimpulannya, sistem ini efektif dan efisien dalam mengelola informasi gedung dan mempermudah reservasi [31].
4	Yudi Novianto, Nadila Aulia, Dodo Zaenal Abidin (2024).	Perancangan Sistem Informasi Penyewaan Lapangan Badminton Pada Hall Gembira Kota Jambi Berbasis Web.	Penelitian ini bertujuan untuk membuat sistem informasi berbasis <i>website</i> yang digunakan untuk menyewakan lapangan badminton di <i>Hall Gembira Kota Jambi</i> . Dengan sistem yang masih menggunakan cara manual, yaitu pemesanan melalui <i>telpon</i> , pembayaran dilakukan langsung di tempat, dan pencatatan menggunakan buku, sehingga prosesnya kurang efisien dan rentan kesalahan. Dengan menggunakan metode pengembangan <i>waterfall</i> , penelitian ini berhasil membuat aplikasi berupa <i>website</i> . Hasil dari penelitian ini memberikan dua manfaat utama, yaitu memudahkan pelanggan untuk menyewa dan membayar secara <i>online</i> , serta memudahkan admin dan pimpinan dalam mengelola transaksi dan membuat laporan secara digital [32].

Tabel 2.4 Tabel Penelitian Terdahulu (lanjutan).

No.	Penulis (Tahun)	Judul	Isi
5	Davina Kurnia Hiuredhy dan Yos Richard Beeh, S.T., M.Cs. (2022).	Aplikasi Reservasi Ibadah Mawar Sharon Salatiga Menggunakan <i>Flutter</i> .	Penelitian ini dirancang untuk mengatasi keterbatasan kursi akibat pandemi Covid-19. Aplikasi ini menggunakan <i>framework Flutter</i> dan <i>Firebase</i> untuk menciptakan solusi agar jemaat melakukan reservasi kursi secara <i>online</i> dengan efisien. Sebelum pandemi, gereja mampu menampung lebih banyak jemaat, namun dengan pembatasan saat pandemi, diperlukan sistem yang lebih terstruktur untuk mengatur kehadiran jemaat. Aplikasi ini membantu pengelola gereja dalam mendata kehadiran jemaat dan memenuhi persyaratan protokol kesehatan. Dengan <i>Flutter</i> , aplikasi dapat berjalan pada berbagai perangkat seperti <i>smartphone</i> dan komputer, memberikan fleksibilitas akses bagi pengguna. <i>Firebase</i> menyediakan <i>database real-time</i> yang efisien untuk pencatatan data, memungkinkan pengelolaan data yang akurat. Kombinasi teknologi ini menghasilkan aplikasi yang tidak hanya <i>user-friendly</i> dan cepat dikembangkan, tetapi juga efektif dalam mendukung administrasi gereja dan memastikan bahwa semua prosedur kesehatan dipatuhi selama masa pandemi, sehingga jemaat dapat beribadah dengan aman dan nyaman [33].

III. METODE PENELITIAN

3.1. Waktu dan Tempat

Waktu dan tempat penelitian dilaksanakan pada:

Waktu Penelitian : Juli 2024 sampai dengan Agustus 2025.

Tempat Penelitian : SMAN 1 Tanjung Bintang.

3.2. Jadwal Penelitian

Penelitian dilakukan dari bulan Juli 2024 hingga Agustus 2025 dengan uraian pada tabel 3.1.

Tabel 3.1 Jadwal Penelitian

No.	Kegiatan	Bulan					
		Juli 2024	Agustus 2024	September 2024	Oktober 2024	November 2024.... Juli 2025	Agustus 2025
1.	<i>Analysis</i>						
2.	Design						
3.	Perancangan Papan <i>Kanban</i>						

Tabel 3.1 Jadwal Penelitian (lanjutan)

No.	Kegiatan	Bulan					
		Juli 2024	Agustus 2024	September 2024	Oktober 2024	November 2024.... Juli 2025	Agustus 2025
4.	Pengembangan dengan menggunakan <i>Kanban</i>						
5.	<i>Deploy</i>						
6.	Penulisan Laporan						

3.3. Alat Penelitian

Alat-alat yang digunakan dalam penelitian rancang bangun aplikasi sistem informasi reservasi gedung untuk kegiatan ekstrakurikuler di SMAN 1 Tanjung Bintang terdapat pada tabel 3.2.

Tabel 3.2 Alat yang digunakan dalam penelitian

No.	Nama	Spesifikasi
1	Laptop Acer Nitro AN515-45	<i>AMD RYZEN 5 5600H 3.3 Ghz, RAM 16GB, Sistem Operasi Windows 11</i>
2	<i>Figma</i>	<i>Desktop Version</i>
3	<i>Android Studio</i>	Versi 2023.2.1
4	<i>Flutter</i>	Versi 3.22.2
5	<i>Dart</i>	Versi 3.4.3
6	<i>Microsoft Edge</i>	Versi 126.0.2592.56
7	<i>Firebase</i>	Versi 13.11.2
8	Android Emulator	Android 10.0

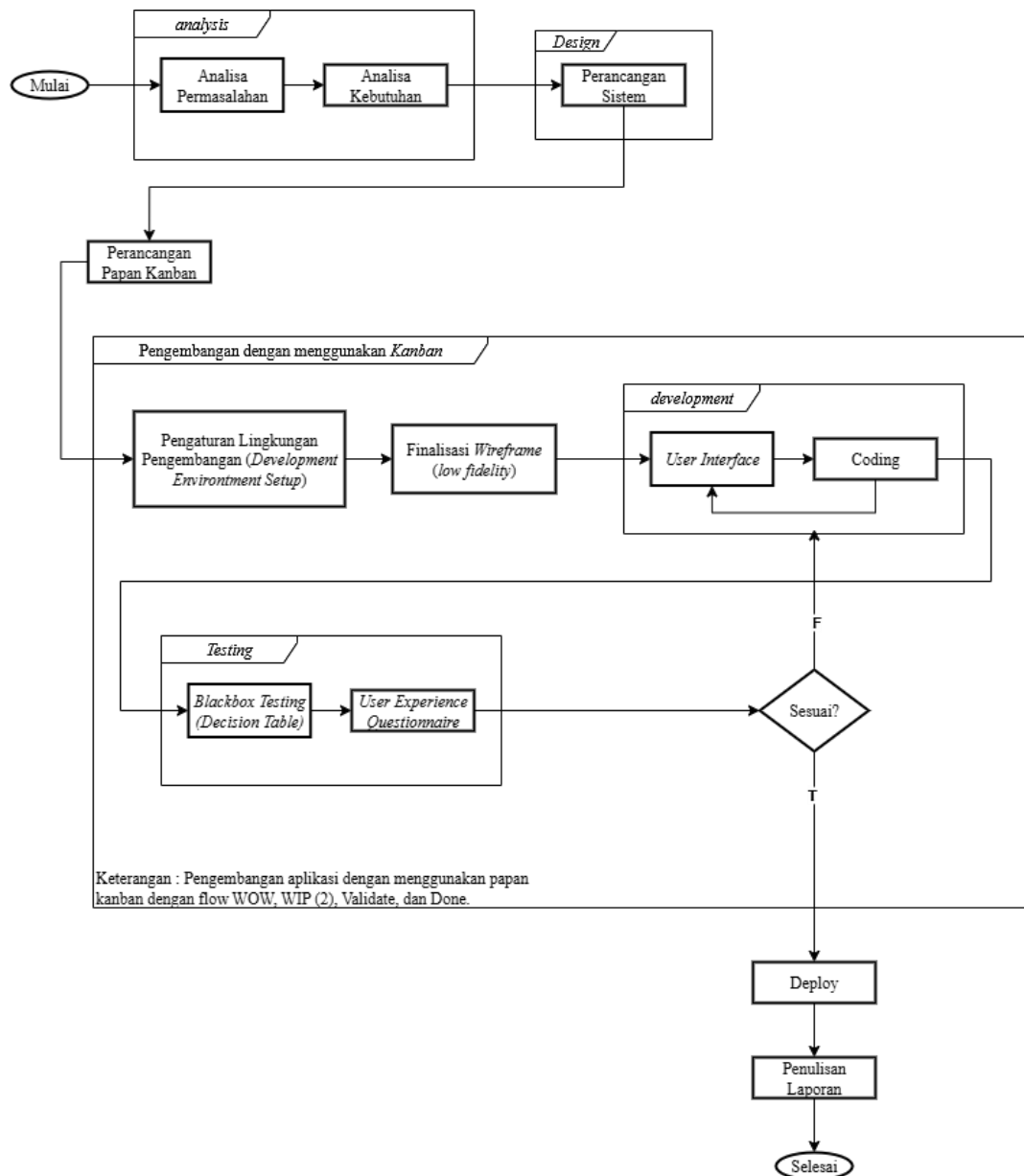
3.4. Tahapan Penelitian

Menurut *Pressman* dalam bukunya yang berjudul *Software Engineering: A Practitioner's Approach*, pada setiap proses pengembangan perangkat lunak yang terstruktur, terlepas dari model spesifik yang digunakan, akan selalu mencakup lima aktivitas kerangka kerja yang bersifat generik. Kelima aktivitas fundamental tersebut adalah:

1. *Communication* (Komunikasi): Tahap awal untuk berkomunikasi intensif dengan pemangku kepentingan (*stakeholder*) guna memahami tujuan dan mengumpulkan kebutuhan. Tahap ini direpresentasikan dalam penelitian ini melalui Analisa Permasalahan dan Analisa Kebutuhan.
2. *Planning* (Perencanaan): Mendefinisikan rencana kerja, estimasi jadwal, dan pelacakan progres.
3. *Modeling* (Permodelan): Membuat model atau representasi untuk lebih memahami kebutuhan dan merancang solusi perangkat lunak. Tahap ini diwujudkan melalui Perancangan Sistem menggunakan UML dan *wireframe*.
4. *Construction* (Konstruksi): Aktivitas yang menggabungkan pembuatan kode (*code generation*) dan pengujian (*testing*) untuk menemukan kesalahan. Ini sesuai dengan fase *developing* dan Pengujian *Black box* dalam penelitian ini.
5. *Deployment* (Penyebaran): Tahap penyerahan produk kepada pengguna dan mendapatkan umpan balik. Dalam konteks penelitian ini, tahap ini diwakili oleh Pengujian Akhir (UEQ) pada produk yang sudah jadi.

Dengan demikian, tahapan penelitian yang diadopsi dalam skripsi ini merupakan implementasi praktis dari kerangka kerja fundamental yang telah diakui secara luas dalam disiplin rekayasa perangkat lunak tersebut [34].

Terdapat beberapa tahapan yang dilakukan dalam penelitian ini, dapat dilihat pada gambar di bawah ini.



Gambar 3.1 Diagram alir penelitian.

Pada gambar 3.1. di atas merupakan tahapan yang dilakukan ketika melaksanakan penelitian, dimulai dengan melakukan tahapan analisa yang terdiri dari analisa permasalahan dan juga analisa kebutuhan. Dalam analisa permasalahan

dilakukan analisis permasalahan mengenai reservasi gedung untuk kegiatan ekstrakurikuler dengan studi kasus SMAN 1 Tanjung Bintang. Selanjutnya dalam analisa kebutuhan meliputi beberapa aspek yang di mana bertujuan untuk menunjang kebutuhan apa saja yang diperlukan untuk penelitian ini seperti *user story*, target pengguna, kebutuhan fungsional dan kebutuhan non-fungsional. Kemudian selanjutnya adalah tahap *design* yang terdiri dari perancangan sistem yang berisi *unified modeling language* (UML), arsitektur sistem, dan *wireframe design* awal. Lalu beralih ke dalam tahapan perancangan papan *kanban* yang merupakan tahap awal pembuatan *backlog* ataupun *task card* dengan acuan dari *user story* dalam pembuatan *backlog* tersebut.

Tahapan selanjutnya adalah pengembangan dengan menggunakan *kanban* pada tahap ini menerapkan alur kerja (*workflow*) yang dibagi dalam beberapa kolom seperti *Work on Wait* (WOW), *Work in Progress* (WIP), Testing, serta *Done* sesuai dengan konsep yang sudah dijelaskan dalam tinjauan pustaka. Pada tahap ini terdapat beberapa alur kerja dari semua *task card* yang terbagi menjadi beberapa tahapan yaitu pengaturan lingkungan pengembangan (*Development Environment setup*), finalisasi *wireframe* (*low fidelity*), kemudian tahapan *development* yang terdiri dari *user interface* dan *coding*. Kemudian dilanjutkan dengan tahapan Testing yang terdiri dari *Blackbox testing* (*decision table*) dan *User experience questionnaire* (UEQ) selanjutnya di periksa apakah sudah sesuai, jika sesuai maka akan langsung diteruskan ke tahap selanjutnya yaitu *deploy*, jika tidak maka akan dikembalikan untuk dikerjakan kembali agar sesuai di tahapan *developing*. Selanjutnya merupakan tahapan *deploy* yang dilanjutkan dengan tahapan penulisan laporan dan penelitian dinyatakan telah sampai dalam tahap selesai.

3.4.1. Analysis

Tahap analisis adalah bagian awal dari seluruh proses penelitian, yang bertujuan agar kita memahami dengan baik permasalahan yang ada dan menentukan batasan serta solusi yang akan dibuat. Tahap ini terdiri dari dua kegiatan utama yang saling terkait, yaitu Analisis Permasalahan untuk menemukan

hambatan dan kelemahan pada sistem yang sudah berjalan, serta Analisis Kebutuhan untuk mengubah permasalahan tersebut menjadi beberapa kebutuhan fungsional dan non-fungsional yang harus dipenuhi oleh sistem baru.

1. Analisa Permasalahan

Pada tahap ini dilakukan analisa permasalahan mengenai reservasi gedung untuk kegiatan ekstrakurikuler di SMAN 1 Tanjung Bintang. Pada saat ini pengurus yang berkaitan dengan kegiatan ekstrakurikuler yang ingin melakukan reservasi gedung di SMAN 1 Tanjung Bintang dilakukan secara manual, hal ini menjadi bisa menjadi masalah karena bisa terjadinya kesalahan informasi dalam reservasi gedung untuk kegiatan ekstrakurikuler serta menjadi tidak efisien.

Dalam pengembangan sistem informasi manajemen reservasi gedung ini, muncul beberapa permasalahan pada tahap perancangan, implementasi, dan pengujian sistem. Permasalahan ini tidak hanya relevan bagi Sekolah SMAN 1 Tanjung Bintang, namun juga upaya untuk menjadikan sistem ini lebih fleksibel, efektif, dan dapat diterapkan secara luas di berbagai sekolah. Salah satu tantangan utamanya adalah menjadikan sistem ini tidak terlalu bergantung pada kebutuhan khusus sekolah dan lebih bersifat umum. Sumbangan dosen ini menyoroti pentingnya fleksibilitas ini sehingga sekolah lain yang memiliki kebutuhan serupa dapat menerapkan sistem ini. Tantangan utamanya adalah merancang *database* dan logika sistem yang dapat menampung data dari berbagai sekolah tanpa membatasi fungsionalitas utama.

Selain itu, ada tiga aktor utama yaitu *superadmin*, supervisor, dan pengguna ekstrakurikuler, yang masing-masing memiliki hak akses berbeda. Caranya adalah bagaimana membangun sistem kontrol akses berbasis peran sehingga setiap aktor hanya memiliki akses ke fungsionalitas yang sesuai dengan perannya. Misalnya, *superadmin* bertanggung jawab mengelola akun supervisor, namun supervisor memiliki wewenang yang lebih luas atas data gedung CRUD, kegiatan ekstrakurikuler, konfirmasi reservasi dan akun pengguna, begitu juga dengan *user* ekstrakurikuler yang memiliki peran yang berbeda. Hal ini memerlukan desain

yang cermat untuk menghindari tumpang tindih hak akses di antara ketiga aktor tersebut.

Selain itu, Sistem harus dapat menjamin bahwa konflik penjadwalan tidak terjadi ketika beberapa pengguna mencoba memesan gedung yang sama pada waktu yang bersamaan. Masalah ini memerlukan algoritma yang efisien untuk memeriksa ketersediaan jadwal dan memberikan umpan balik yang jelas dan cepat kepada pengguna. Selain itu, beberapa kegagalan terkait validasi *input* data teridentifikasi pada tahap pengujian fungsional dengan teknik pengujian *black box* menggunakan teknik *decision table*. Metodologi pengukuran pengalaman pengguna *User Experience Questionnaire* (UEQ) juga menjadi fokus analisis ini. Salah satu masalahnya adalah semua pengguna yang mengikuti kegiatan ekstrakurikuler, harus dapat memahami semua fitur yang ditawarkan dan memberikan kesan yang positif.

Berdasarkan analisis ini, permasalahan utama yang dihadapi dalam pengembangan aplikasi meliputi penyesuaian kebutuhan generalisasi sistem, manajemen hak akses *multi-aktor*, validasi konflik jadwal reservasi, dan pengukuran pengalaman pengguna. Setiap permasalahan ini diatasi melalui pendekatan yang sistematis selama proses perancangan, pengujian, dan evaluasi, dengan harapan sistem ini dapat berjalan optimal dan diterima dengan baik oleh pengguna di berbagai sekolah.

2. Analisa Kebutuhan

Pada tahap ini dilakukan analisis kebutuhan dari aplikasi sistem informasi Manajemen Reservasi Gedung untuk kegiatan ekstrakurikuler di SMAN 1 Tanjung Bintang. Analisa kebutuhan dilakukan dengan tujuan untuk memastikan bahwa aplikasi yang dikembangkan atau diterapkan dapat memenuhi ekspektasi dan tujuan yang diinginkan oleh pengguna. Kebutuhan-kebutuhan. Kebutuhan-kebutuhan tersebut akan dituangkan dalam bentuk sebagai berikut :

a) *User Story*

User story adalah beberapa kalimat dalam bahasa sederhana yang menguraikan hasil yang diinginkan. Mereka tidak merincinya. Persyaratan ditambahkan kemudian, setelah disepakati oleh tim. *User story* sangat cocok dengan kerangka kerja seperti *scrum* dan *kanban*. Dalam *scrum*, cerita pengguna ditambahkan ke sprint dan “dibakar habis” selama durasi sprint. Sedangkan *kanban* memasukkan cerita pengguna ke dalam *backlog* mereka dan menjalankannya melalui alur kerjanya [35].

Tahapan ini dilakukan supaya menghasilkan serangkaian *user story* yang akan menjadi pedoman dalam pembuatan *backlog* yang digunakan dalam pengembangan aplikasi. *User story* penting dilakukan untuk menjadi acuan terhadap *backlog* yang dibutuhkan dalam pengembangan aplikasi yang akan dibuat. Format dari *user story* adalah sebagai berikut:

Sebagai [persona] saya [ingin] [sehingga]

Persona adalah target dari penelitian ini yang telah ditentukan, kemudian ingin adalah hal yang dilakukan oleh pengguna sehingga mencapai hasil yang diharapkan.

Berikut merupakan *user story* dari aplikasi rancang bangun aplikasi manajemen reservasi gedung berbasis android dengan studi kasus SMAN 1 Tanjung Bintang sesuai dengan tabel 3.3.

Tabel 3.3 *User story* pada perancangan aplikasi reservasi gedung.

Aktor	Ingin	Sehingga
<i>Userekstrakurikuler</i>	Melakukan <i>login</i> menggunakan akun yang disediakan oleh Supervisor.	Bisa mengakses aplikasi dengan kontrol yang dikelola oleh supervisor.
<i>Userekstrakurikuler</i>	Melihat jadwal ketersediaan gedung.	Bisa memilih waktu yang tepat untuk melakukan reservasi tanpa konflik.
<i>Userekstrakurikuler</i>	Membatalkan reservasi yang sudah saya buat.	Bisa menyesuaikan jadwal jika ada perubahan rencana.

Tabel 3.3 *User story* pada perancangan aplikasi reservasi gedung. (lanjutan)

Aktor	Ingin	Sehingga
<i>Userekstrakurikuler</i>	Melihat riwayat reservasi saya.	Bisa melacak penggunaan gedung untuk kegiatan ekstrakurikuler saya.
Supervisor	Melakukan <i>login</i> menggunakan akun yang disediakan oleh super admin.	Bisa mengelola reservasi dan <i>user</i> di sekolah saya dengan kontrol yang terpusat.
Supervisor	Mengkonfirmasi atau menolak reservasi yang diajukan oleh <i>user</i> .	Bisa mengelola penggunaan gedung secara efisien dan sesuai dengan kebijakan.
Supervisor	Mengelola informasi gedung dan jadwal yang tersedia.	Bisa menyediakan informasi yang akurat dan terkini kepada <i>user</i> .
Supervisor	Mengakses laporan reservasi.	Bisa memantau penggunaan gedung dan membuat keputusan berdasarkan data.
Supervisor	Mengelola akun <i>user</i> ekstrakurikuler.	Bisa memastikan bahwa hanya pengguna yang sah yang dapat melakukan reservasi.
<i>Superadmin</i>	Membuat dan mengelola akun supervisor.	Bisa memastikan bahwa setiap sekolah memiliki supervisor yang sah, dan dapat mengonfigurasi dengan mudah apabila ingin menambahkan sekolah dengan kebutuhan yang sama.

b) Target Pengguna

Pada tahap ini untuk menentukan target pengguna dari aplikasi Manajemen reservasi gedung untuk kegiatan ekstrakurikuler di SMAN 1 Tanjung Bintang dilakukan berdasarkan dari analisa permasalahan. Permasalahan dilatar belakangi dengan masalah pengurus yang berkaitan dengan kegiatan ekstrakurikuler yaitu dari siswa maupun pengurus sekolah, dan guru yang berkaitan dengan kegiatan ekstrakurikuler di SMAN 1 Tanjung Bintang. Dengan demikian target pengguna dari penelitian merupakan pengurus yang berkaitan dengan kegiatan ekstrakurikuler di SMAN 1 Tanjung Bintang.

c) Kebutuhan *Fungsional*

Kebutuhan fungsional didapatkan berdasarkan hasil dari analisa permasalahan yang kemudian dilakukan analisa terkait hal apa saja yang harus dilakukan oleh sistem. Kebutuhan fungsional dari aplikasi reservasi gedung dengan studi kasus di SMAN 1 Tanjung Bintang yang akan dibuat yaitu melakukan *login* , reservasi gedung, konfirmasi reservasi, pembatalan reservasi, lihat informasi gedung dan jadwal ekstrakurikuler, riwayat/laporan reservasi, dan kelola akun. Adapun kebutuhan fungsional dari aplikasi manajemen reservasi gedung berbasis android dengan studi kasus SMAN 1 Tanjung Bintang sesuai dengan tabel 3.4.

Tabel 3.4 Kebutuhan fungsional pada aplikasi reservasi gedung.

No.	fitur	Kebutuhan Fungsional
1	Melakukan <i>Login</i>	<i>Userekstrakurikuler</i> , Supervisor, dan <i>Superadmin</i> harus dapat melakukan <i>login</i> untuk dapat menggunakan fitur untuk masing-masing aktor tersebut.
2	Reservasi Gedung	<i>Userekstrakurikuler</i> harus dapat melakukan reservasi gedung dengan memasukkan detail kegiatan, tanggal, dan waktu.
3	Konfirmasi Reservasi	Supervisor harus dapat mengonfirmasi atau menolak reservasi yang diajukan oleh pengguna.
4	Pembatalan Reservasi	<i>Userekstrakurikuler</i> harus dapat membatalkan reservasi gedung yang telah dibuat.
5	Lihat Informasi Gedung dan Jadwal Ekstrakurikuler	<i>Userekstrakurikuler</i> harus dapat melihat informasi gedung dan juga jadwal ekstrakurikuler.
6	Riwayat/Laporan Reservasi	<i>Userekstrakurikuler</i> dan supervisor harus dapat melihat riwayat reservasi yang pernah dilakukan dengan detail setiap reservasi.
7	Kelola Akun	Supervisor harus dapat mengelola akun <i>Userekstrakurikuler</i> kemudian <i>superadmin</i> harus dapat mengelola akun supervisor.

d) Kebutuhan *Non-Fungsional*

Kebutuhan ini merupakan kebutuhan pendukung/pelengkap kebutuhan fungsional. Pada kebutuhan ini berisi fitur-fitur yang mendukung berjalannya fitur utama. Kebutuhan non-fungsional dari aplikasi reservasi gedung dengan

studi kasus di SMAN 1 Tanjung Bintang meliputi beberapa kategori yaitu kinerja, keamanan, dan ketersediaan. Adapun kebutuhan non-fungsional dari aplikasi manajemen reservasi gedung berbasis android dengan studi kasus SMAN 1 Tanjung Bintang sesuai dengan tabel 3.5.

Tabel 3.5 Kebutuhan non-fungsional pada perancangan aplikasi reservasi gedung.

No.	Kategori	Kebutuhan Non-Fungsional
1	Kinerja	Waktu respons maksimum untuk setiap operasi tidak boleh lebih dari 3 detik.
2	Keamanan	Diperlukan <i>autentikasi</i> menggunakan akun yang telah disediakan oleh admin agar pengguna bisa menggunakan aplikasi.
3	Ketersediaan	Sistem berjalan pada <i>smartphone</i> dengan sistem operasi Android, serta koneksi internet agar aplikasi dapat berjalan.

3.4.2. Design

Setelah semua kebutuhan sistem didefinisikan pada tahap analisis, penelitian dilanjutkan ke tahap *design* (perancangan). Tahap ini bertujuan untuk menerjemahkan daftar kebutuhan tersebut menjadi sebuah cetak biru (*blueprint*) teknis yang siap untuk diimplementasikan. Seluruh aktivitas dalam tahap ini dirangkum dalam Perancangan Sistem, yang mencakup permodelan proses bisnis menggunakan UML, perancangan arsitektur, serta perancangan antarmuka pengguna melalui *wireframe* (awal).

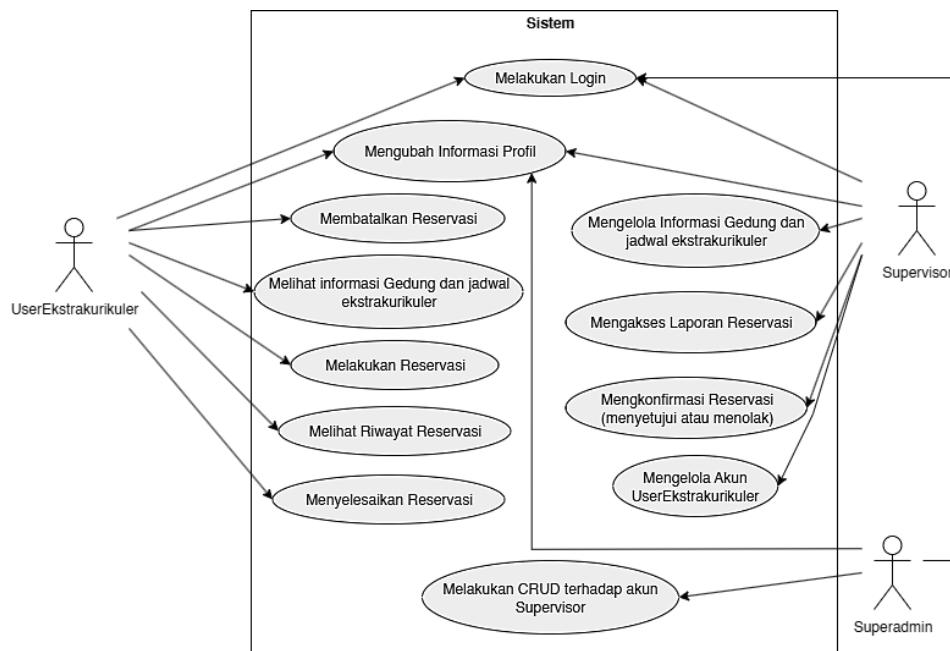
1. Unified Modeling Language (UML)

Unified Modeling Language (UML) merupakan bahasa visual yang digunakan untuk permodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung. Dalam penelitian ini *Unified Modeling Language* yang diusulkan adalah *usecase diagram*, *Class diagram*, dan *Activity diagram*. *Unified Modeling Language* tersebut nantinya akan

memberikan sebuah gambaran bagaimana interaksi antara aktor dengan sistem, mempresentasikan struktur obyek dalam sistem, alur kerja dalam sistem, dan interaksi antar obyek dari sistem.

1) *Usecase Diagram*

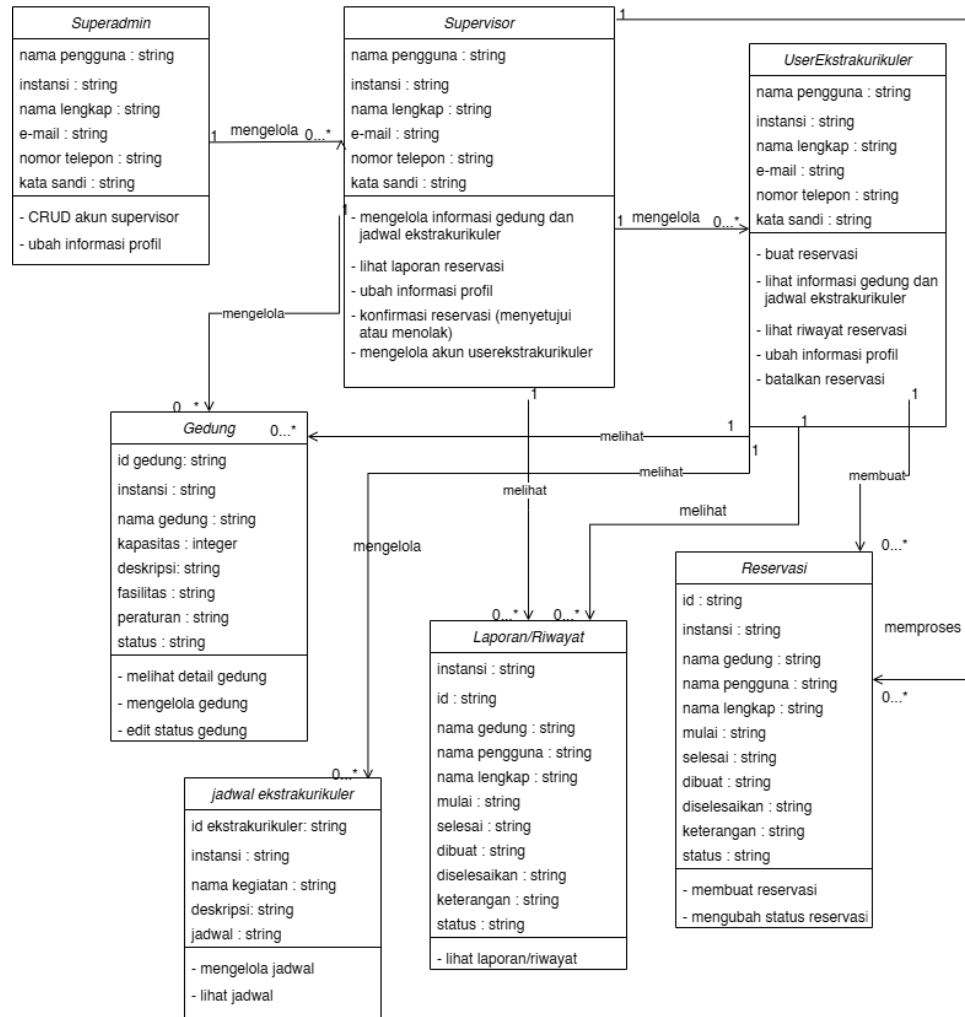
Pada *use case diagram* sistem, terdapat 3 aktor yaitu *Userekstrakurikuler*, Supervisor dan *Superadmin* serta memiliki beberapa aktivitas. Aktor *Userekstrakurikuler* dapat melakukan aktivitas seperti melakukan *login* (masuk aplikasi), mengubah informasi profil, membatalkan reservasi, melihat informasi gedung dan jadwal ekstrakurikuler, melakukan reservasi, melihat riwayat reservasi dan menyelesaikan reservasi. Aktor Supervisor dapat melakukan aktivitas melakukan *login* (masuk aplikasi), mengubah informasi profil, mengelola informasi gedung dan jadwal ekstrakurikuler, mengakses laporan reservasi, mengkonfirmasi reservasi (menyetujui atau menolak), dan mengelola akun *Userekstrakurikuler*. Kemudian pada aktor *Superadmin* dapat melakukan melakukan *login* (masuk aplikasi), mengubah informasi profil, dan CRUD terhadap akun supervisor. Hasil dari penentuan *usecase diagram* aplikasi Reservasi Gedung untuk Kegiatan Ekstrakurikuler dapat dilihat pada Gambar 3.2



Gambar 3. 2 *Usecase diagram* aplikasi Reservasi Gedung untuk kegiatan Ekstrakurikuler.

2) Class Diagram

Adapun *Class diagram* kebutuhan sistem adalah sebagai berikut:



Gambar 3. 3 *Class diagram* aplikasi Reservasi Gedung untuk Kegiatan Ekstrakurikuler studi kasus SMAN1 Tanjung Bintang.

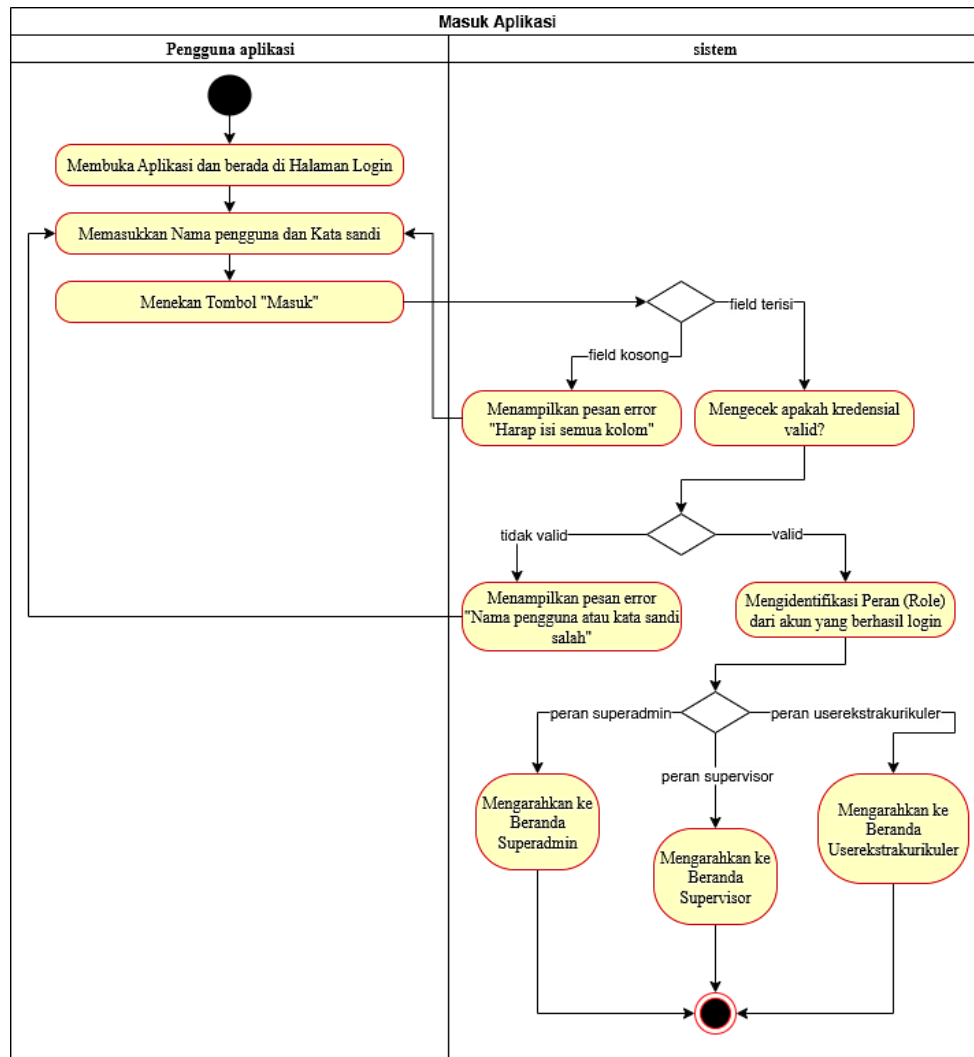
Gambar 3.3 merupakan *Class diagram* dari aplikasi Reservasi Gedung untuk Kegiatan Ekstrakurikuler studi kasus SMAN1 Tanjung Bintang. Pada gambar terdapat tujuh *class* yang saling berhubungan, di mana pengguna *superadmin* dapat mengelola akun supervisor dan juga informasi pribadinya. Kemudian supervisor dapat mengelola akun *Userekstrakurikuler*, mengelola informasi gedung dan juga jadwal ekstrakurikuler, melihat laporan, konfirmasi reservasi dan juga mengubah informasi pribadinya. Pengguna *Userekstrakurikuler* dapat membuat

reservasi, membatalkan reservasi, melihat informasi gedung dan jadwal ekstrakurikuler, melihat riwayat reservasi, dan mengubah informasi pribadinya. Kaitan antara *class superadmin* dengan supervisor adalah *superadmin* dapat mengelola banyak pengguna supervisor serta *class supervisor* juga dapat mengelola banyak pengguna *class Userekstrakurikuler*. Keterkaitan oleh masing-masing pengguna juga dipengaruhi oleh atribut instansi yang dimilikinya, dalam hal ini supervisor hanya dapat mengelola *class* yang instansinya sama. Hubungan ini mencerminkan bahwa *Superadmin*, Supervisor dan *Userekstrakurikuler* memiliki peran yang berbeda tetapi saling terkait dalam pengelolaan reservasi gedung.

3) Activity Diagram

Sistem mempunyai beberapa fungsi yang dapat dijelaskan pada *Activity diagram* berikut:

- *Activity diagram* masuk aplikasi



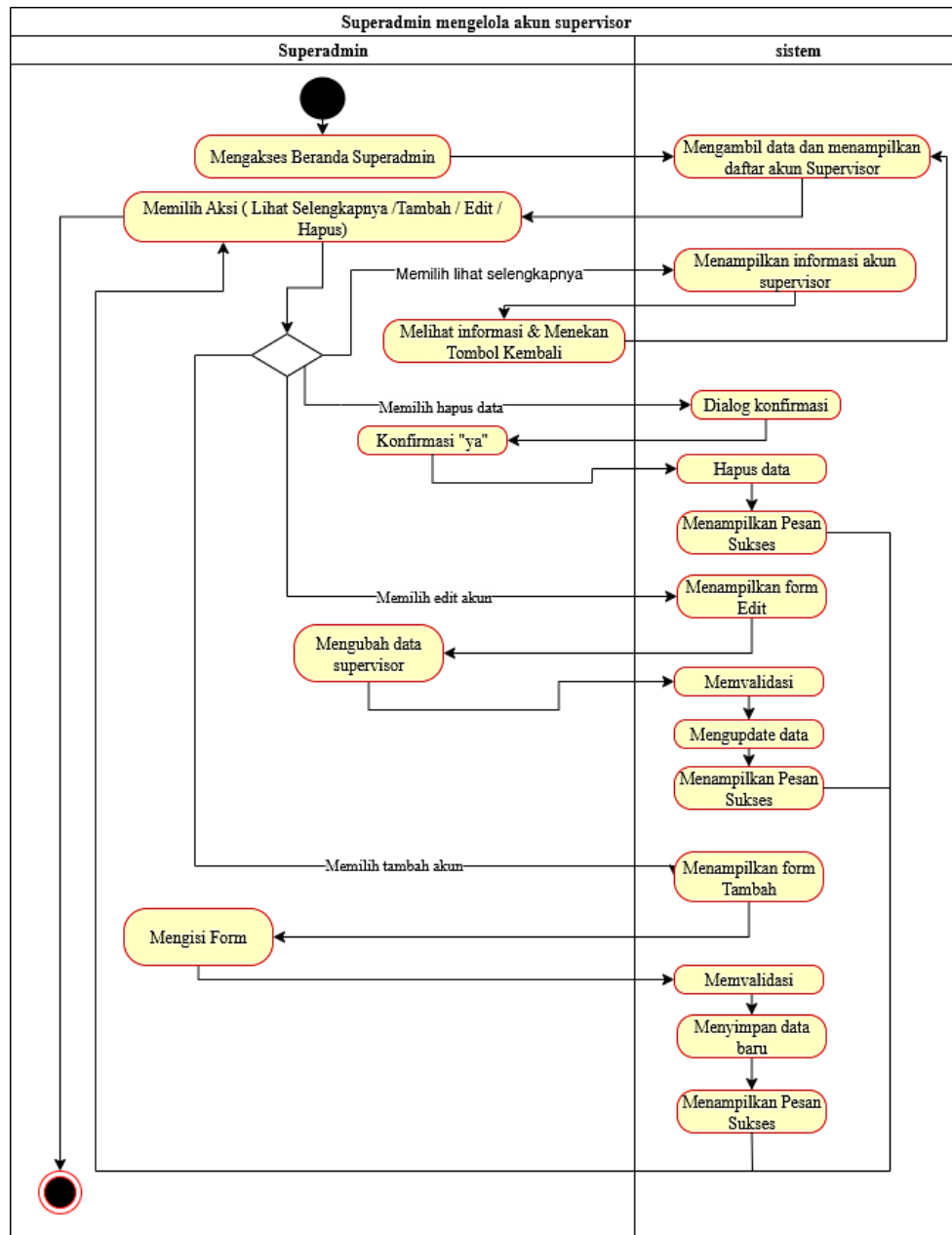
Gambar 3.4 *Activity diagram* masuk aplikasi.

Gambar 3.4 menjelaskan aktivitas alur kerja sistem saat pengguna masuk ke dalam aplikasi. Proses ini melibatkan interaksi antara Pengguna aplikasi (yang dapat berupa *Superadmin*, *Supervisor*, atau *Userekstrakurikuler*) dan Sistem, yang diawali dengan pengguna membuka aplikasi dan berada di halaman *Login*. Setelah pengguna memasukkan nama pengguna dan kata sandi lalu menekan tombol "Masuk", Sistem akan melakukan serangkaian validasi. Titik keputusan pertama adalah validasi *input*, di mana Sistem memeriksa apakah kedua *field* telah terisi. Jika kosong, Sistem akan

menampilkan pesan *error* dan meminta pengguna melengkapi data. Jika terisi, proses berlanjut ke tahap *autentikasi*, di mana kredensial pengguna dicocokkan dengan data di *database*.

Apabila kredensial tidak valid, Sistem akan kembali menampilkan pesan *error*. Jika *autentikasi* berhasil, Sistem akan melakukan tahap otorisasi dengan mengidentifikasi peran (*role*) dari akun tersebut. Berdasarkan peran yang teridentifikasi, Sistem akan mengarahkan pengguna ke halaman beranda yang sesuai: Beranda *Superadmin*, Beranda *Supervisor*, atau Beranda *Userekstrakurikuler*. Alur ini memastikan bahwa setiap pengguna hanya dapat mengakses antarmuka yang sesuai dengan hak aksesnya, yang menandai keberhasilan proses masuk aplikasi.

- Activity diagram reservasi gedung



Gambar 3.5 Activity diagram reservasi gedung.

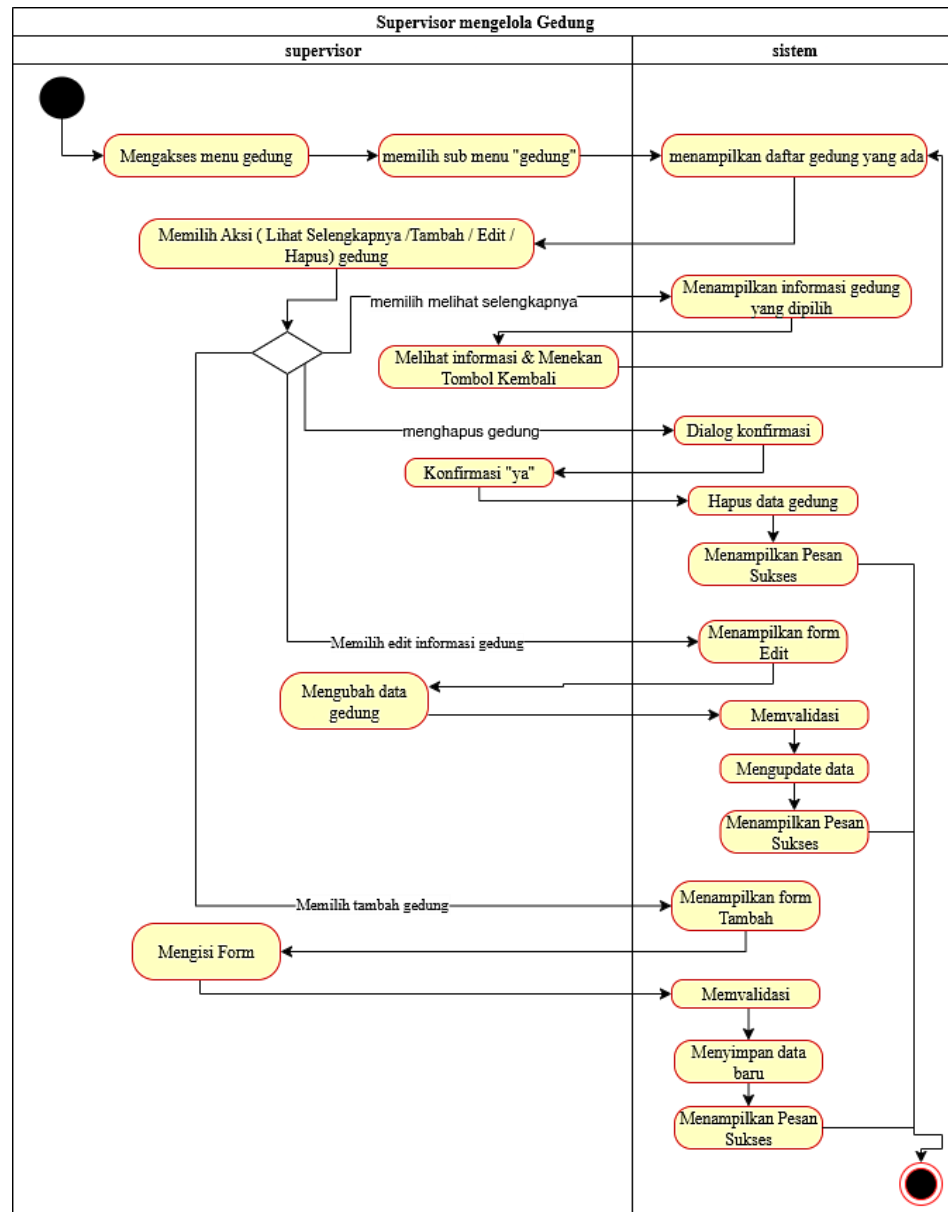
Gambar 3.5 menyajikan visualisasi alur kerja yang komprehensif untuk fitur utama *Superadmin*, yaitu bagaimana mengelola akun Supervisor. Diagram diatas memetakan setiap langkah dan interaksi yang terjadi antara aktor *Superadmin* dan Sistem. Aktivitas diawali ketika *Superadmin* mengakses halaman beranda. Sebagai respons langsung, Sistem akan mengambil seluruh data Supervisor dari *database* dan menampilkannya

dalam bentuk daftar yang informatif. Setiap item dalam daftar ini dilengkapi dengan serangkaian opsi aksi (Lihat Selengkapnya, Edit, Hapus), sementara tombol untuk menambah data baru juga tersedia. Titik ini menjadi gerbang keputusan (*decision node*) utama, di mana *Superadmin* dapat memilih alur kerja selanjutnya. Alur "Lihat Selengkapnya" berfungsi sebagai proses observasi non-destruktif, di mana sistem akan menampilkan informasi detail dari akun yang dipilih, dan setelahnya *Superadmin* akan kembali ke halaman daftar utama untuk dapat memilih aksi lainnya.

Untuk alur kerja yang bersifat modifikasi data—yaitu Tambah, Edit, dan Hapus—diagram ini menunjukkan adanya protokol validasi dan konfirmasi yang sistematis untuk menjaga keamanan dan keakuratan data. Pada proses penambahan dan pengeditan, Sistem akan terlebih dahulu menyajikan formulir yang sesuai. Setiap data yang dikirim oleh *Superadmin* akan melewati tahap validasi oleh Sistem untuk memastikan kelengkapan dan format data sudah benar. Hanya jika data dinyatakan valid, Sistem akan melanjutkan proses penyimpanan (untuk data baru) atau pembaruan (untuk data yang sudah ada) di dalam *database*. Sementara itu, untuk proses penghapusan data yang bersifat permanen, Sistem menerapkan langkah konfirmasi untuk mencegah kesalahan yang tidak disengaja, di mana *Superadmin* harus memberikan persetujuan akhir sebelum data benar-benar dihapus.

Setiap alur kerja yang berhasil dieksekusi akan diakhiri dengan pesan sukses yang informatif dari Sistem, yang memberikan umpan balik langsung kepada *Superadmin*. Secara keseluruhan, diagram aktivitas ini tidak hanya menggambarkan langkah-langkah prosedural, tetapi juga menyoroti bagaimana sistem dirancang secara cermat dengan adanya validasi *input*, konfirmasi aksi *kritikal*, dan alur interaksi yang jelas, memastikan bahwa *Superadmin* dapat menjalankan tugasnya secara efektif dan aman.

- *Activity* diagram supervisor mengelola data gedung



Gambar 3.6 *Activity* diagram supervisor mengelola data gedung.

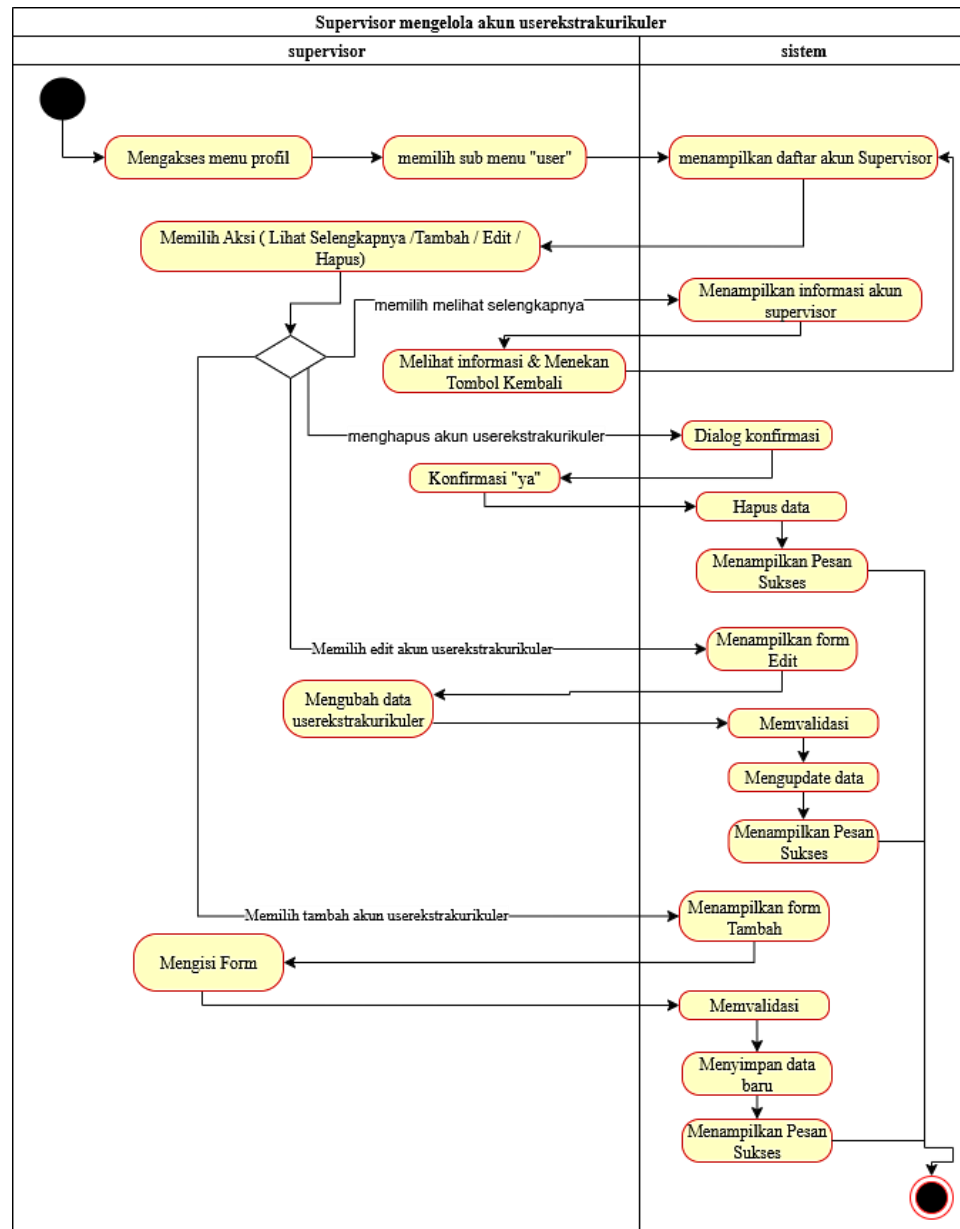
Gambar 3.6 menyajikan visualisasi detail mengenai alur kerja pada fitur supervisor mengelola data gedung. Diagram ini secara spesifik memetakan interaksi dan pembagian tanggung jawab antara aktor Supervisor sebagai pengelola data dan Sistem sebagai pemroses data. Fitur ini merupakan fondasi operasional dari keseluruhan sistem reservasi, karena data gedung yang akurat dan terkini menjadi prasyarat utama agar proses reservasi oleh

userekstrakurikuler dapat berjalan dengan valid. Alur kerja dimulai ketika Supervisor, setelah mengakses menu utama, memilih untuk masuk ke dalam sub-menu pengelolaan gedung. Sistem secara proaktif merespons dengan mengambil dan menyajikan seluruh data gedung yang tersimpan di *database* dalam bentuk sebuah daftar yang komprehensif. Halaman daftar ini berfungsi sebagai pusat kendali, di mana Supervisor dapat membuat keputusan berdasarkan informasi yang tersaji. Dari sini, alur dapat bercabang sesuai dengan pilihan aksi Supervisor: mereka dapat memilih untuk menambah gedung baru melalui tombol yang tersedia, atau berinteraksi langsung dengan data gedung yang sudah ada melalui ikon aksi untuk melihat detail, mengedit, ataupun menghapus.

Untuk setiap alur kerja yang melibatkan modifikasi data, diagram ini menyoroti adanya mekanisme kontrol dan validasi yang dirancang untuk menjaga integritas data. Pada proses penambahan atau pembaruan data, Sistem tidak langsung memproses *input* dari Supervisor; sebaliknya, Sistem akan menyediakan formulir khusus. Data yang dimasukkan kemudian akan melewati gerbang validasi untuk memastikan semua informasi yang diperlukan telah terisi dengan benar sebelum dieksekusi ke *database*. Demikian pula untuk aksi penghapusan, yang memiliki dampak permanen, Sistem tidak langsung menghapus data melainkan menampilkan sebuah dialog konfirmasi yang meminta persetujuan akhir dari Supervisor. Langkah ini berfungsi sebagai lapisan pengaman untuk mencegah kehilangan data yang tidak disengaja. Setiap operasi yang berhasil—baik itu menambah, mengubah, atau menghapus—selalu diakhiri dengan umpan balik berupa pesan sukses dan pembaruan otomatis pada daftar gedung. Hal ini menciptakan sebuah siklus kerja yang intuitif dan informatif bagi Supervisor.

Secara keseluruhan, diagram aktivitas ini menegaskan bahwa fitur manajemen gedung telah dirancang secara matang, memberikan Supervisor kendali penuh atas data fasilitas dengan alur yang aman, terstruktur, dan efisien.

- Activity diagram supervisor mengelola akun *userekstrakurikuler*



Gambar 3.7 Activity diagram supervisor mengelola akun *userekstrakurikuler*.

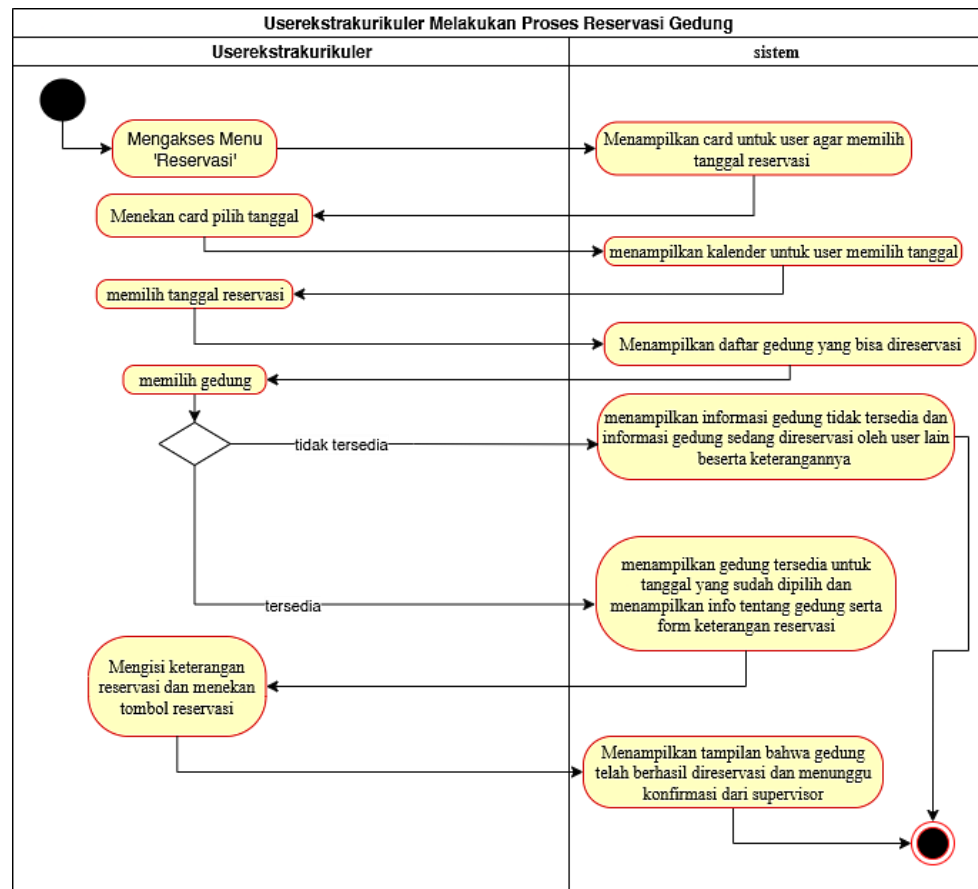
Gambar 3.7 merinci alur kerja Supervisor dalam menjalankan fungsi esensial lainnya, yaitu manajemen akun untuk *Userekstrakurikuler*. Fitur ini sangat krusial karena berkaitan langsung dengan kontrol akses pengguna akhir ke dalam sistem reservasi. Diagram ini memetakan interaksi antara Supervisor sebagai administrator dan Sistem dalam melakukan serangkaian operasi CRUD (*Create, Read, Update, Delete*) terhadap akun-akun

pengguna. Proses ini dimulai ketika Supervisor mengakses menu Manajemen *User*. Sebagai respons, Sistem akan menampilkan daftar seluruh akun *User* Ekstrakurikuler yang terdaftar di bawah wewenanganya. Halaman ini berfungsi sebagai dasbor kontrol utama, di mana Supervisor dapat melihat siapa saja yang memiliki akses dan dapat memilih untuk menambah akun *user* baru, atau memilih salah satu akun yang ada untuk melakukan aksi lebih lanjut seperti melihat detail informasi, mengedit data, atau menghapus akun.

Untuk setiap aksi modifikasi, sistem menerapkan protokol yang sistematis untuk menjaga integritas data. Saat menambah atau mengedit akun *User*, Supervisor akan dipandu melalui formulir khusus yang kemudian divalidasi oleh sistem untuk memastikan semua data yang diperlukan—seperti nama penanggung jawab dan asal ekstrakurikuler—terisi dengan benar sebelum disimpan atau diperbarui. Untuk aksi penghapusan akun *User*, yang merupakan tindakan *kritikal*, sebuah dialog konfirmasi ditampilkan sebagai langkah pengaman untuk mencegah penghapusan data pengguna secara tidak disengaja.

Setiap operasi yang sukses akan selalu diakhiri dengan umpan balik berupa pesan konfirmasi dari sistem dan diikuti dengan pembaruan otomatis pada daftar pengguna. Secara keseluruhan, diagram aktivitas ini menunjukkan bahwa Supervisor dibekali dengan perangkat administratif yang lengkap untuk mengelola siklus hidup akun pengguna, mulai dari pendaftaran hingga penghapusan. Hal ini menjamin bahwa hanya pihak yang sah dan *terverifikasi* yang dapat mengakses dan menggunakan sistem reservasi di sekolah tersebut.

- Activity diagram *userekstrakurikuler* melakukan proses reservasi gedung



Gambar 3.8 Activity diagram mengubah informasi profil.

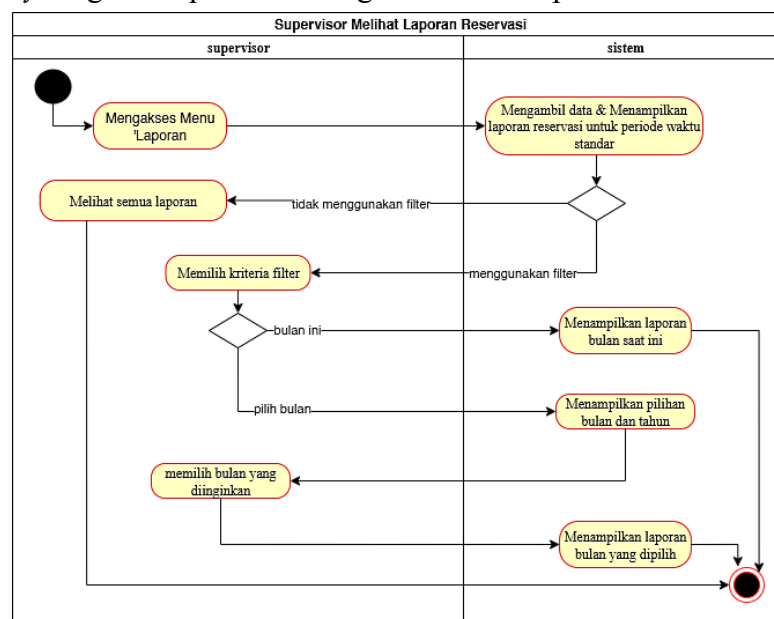
Gambar 3.8 menyajikan alur kerja untuk fungsionalitas paling inti dari perspektif pengguna akhir, yaitu proses pengajuan reservasi gedung oleh *Userekstrakurikuler*. Diagram ini memetakan setiap langkah interaksi antara *User* dan *Sistem*, mulai dari pemilihan jadwal hingga pengajuan reservasi berhasil dikirim. Proses ini dirancang untuk menggantikan alur manual yang tidak efisien dan rentan terhadap kesalahan. Proses dimulai ketika *User* mengakses menu untuk melakukan reservasi.

Untuk mempermudah pencarian jadwal yang valid, *Sistem* akan memandu pengguna melalui beberapa tahap penyaringan. Pertama, *Sistem* akan menampilkan antarmuka kalender agar *User* dapat memilih tanggal kegiatan yang diinginkan. Berdasarkan tanggal yang dipilih tersebut, *Sistem* kemudian akan menyaring dan menampilkan daftar gedung yang berpotensi tersedia pada hari itu, memberikan pengguna gambaran awal fasilitas yang dapat

dipesan. Setelah *User* memilih gedung yang spesifik, Sistem menjalankan validasi yang paling krusial dalam alur ini, yang direpresentasikan oleh *decision node* pada diagram. Sistem akan melakukan pengecekan secara *real-time* ke *database* untuk memastikan ketersediaan gedung pada tanggal dan slot waktu yang telah dipilih. Jika jadwal tersebut ternyata sudah dipesan oleh pengguna lain, Sistem akan menampilkan pesan informasi yang jelas bahwa gedung tidak tersedia, lengkap dengan keterangan siapa yang sedang *mereservasinya*, lalu mengarahkan *User* untuk memilih jadwal lain. Namun, jika jadwal dipastikan tersedia, alur akan berlanjut ke tahap pengisian detail.

Pada tahap akhir, Sistem akan menampilkan formulir reservasi untuk diisi oleh *User* dengan keterangan rinci mengenai kegiatan yang akan dilaksanakan. Setelah *User* memeriksa kembali datanya dan menekan tombol untuk mengajukan reservasi, Sistem akan menyimpan data tersebut ke dalam *database* dengan status awal "Menunggu Konfirmasi" dari Supervisor. Sebagai umpan balik langsung, sebuah pesan sukses akan ditampilkan kepada *User*, menandakan bahwa proses pengajuan telah selesai dan kini bola berada di tangan pihak sekolah untuk tahap validasi selanjutnya.

- *Activity diagram supervisor mengakses menu laporan*

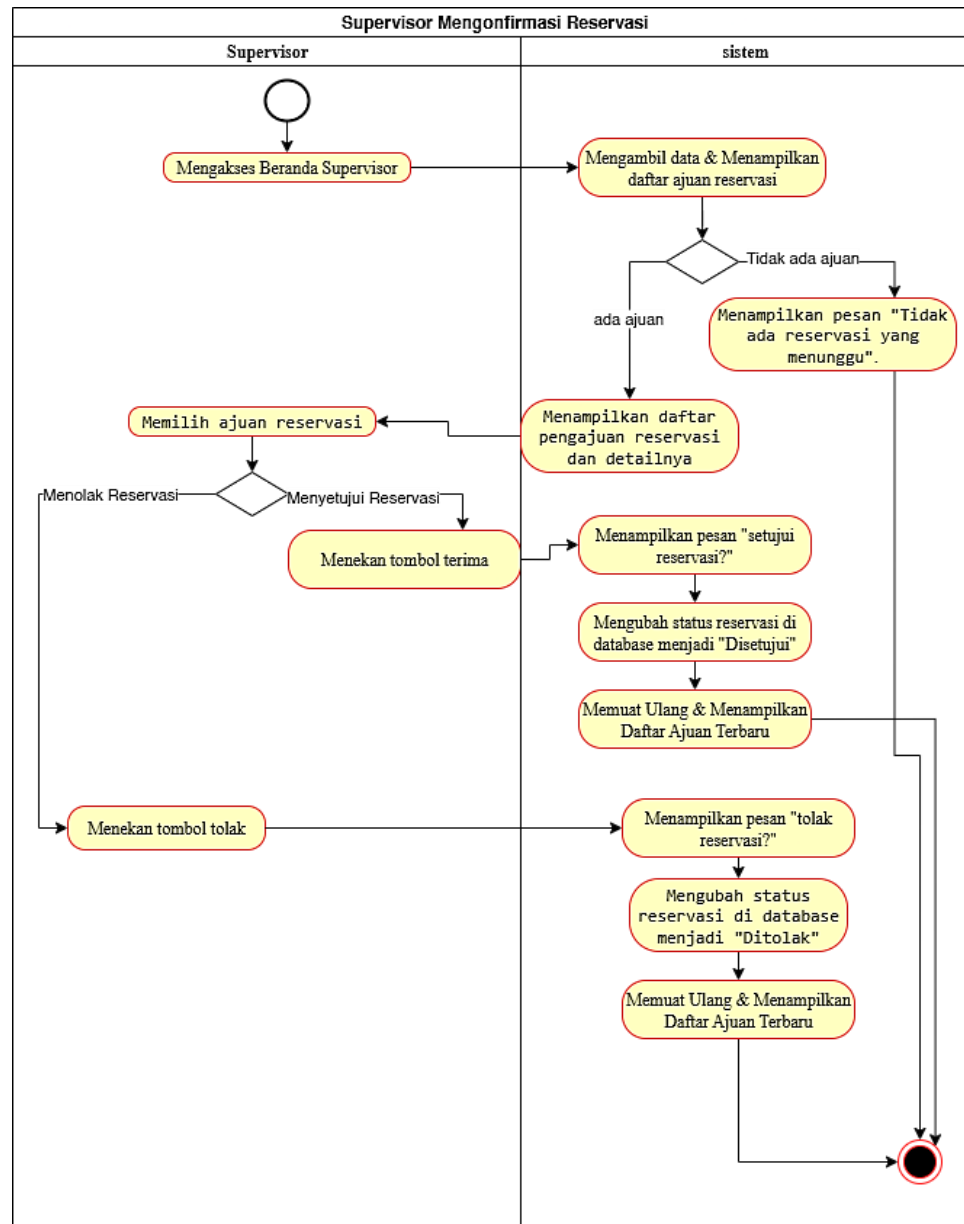


Gambar 3.9 *Activity diagram supervisor mengakses menu laporan.*

Gambar 3.9 menguraikan fungsionalitas pelaporan yang dirancang khusus untuk aktor Supervisor. Fitur ini memungkinkan Supervisor untuk memantau, merekapitulasi, dan mengevaluasi seluruh aktivitas reservasi gedung di lingkup sekolahnya. Alur kerja ini menunjukkan bagaimana data transaksi diolah menjadi informasi yang berguna untuk pengambilan keputusan.

Proses dimulai ketika Supervisor mengakses menu 'Laporan'. Alih-alih menyajikan halaman kosong, Sistem dirancang untuk proaktif dengan cara langsung mengambil data dan menampilkan laporan reservasi untuk periode waktu standar, misalnya untuk bulan berjalan. Tampilan *default* ini memberikan Supervisor gambaran cepat mengenai aktivitas terkini tanpa perlu melakukan langkah tambahan. Dari sini, alur bercabang sesuai kebutuhan Supervisor, yang direpresentasikan oleh *decision node* pertama: Supervisor dapat langsung menganalisis laporan standar yang tersaji, atau memilih untuk melakukan pencarian data yang lebih spesifik dengan menggunakan fitur filter. Jika Supervisor memilih untuk menggunakan filter, diagram ini menunjukkan adanya logika percabangan lebih lanjut untuk mengakomodasi berbagai kebutuhan analisis data. Supervisor dapat memilih kriteria filter yang telah disediakan, seperti melihat laporan berdasarkan bulan spesifik atau periode lainnya. Sebagai respons, Sistem akan menampilkan opsi yang relevan, memproses permintaan tersebut, dan kemudian menyajikan sebuah laporan baru yang telah disaring sesuai dengan kriteria yang diinginkan. Kemampuan untuk menghasilkan laporan yang spesifik ini sangat krusial untuk keperluan administrasi, seperti rekapitulasi bulanan atau evaluasi penggunaan fasilitas tertentu. Secara keseluruhan, alur kerja ini menunjukkan bagaimana sistem tidak hanya berfungsi sebagai pencatat transaksi reservasi, tetapi juga mampu mengubah data mentah tersebut menjadi informasi yang bermakna. Dengan menyediakan laporan standar secara otomatis serta fitur filter yang fleksibel, aplikasi ini memberdayakan Supervisor dengan data yang akurat untuk mendukung pengambilan keputusan, merencanakan alokasi sumber daya, dan meningkatkan efektivitas pengelolaan fasilitas sekolah secara keseluruhan.

- *Activity* diagram supervisor mengkonfirmasi reservasi



Gambar 3.10 *Activity* diagram supervisor mengkonfirmasi reservasi.

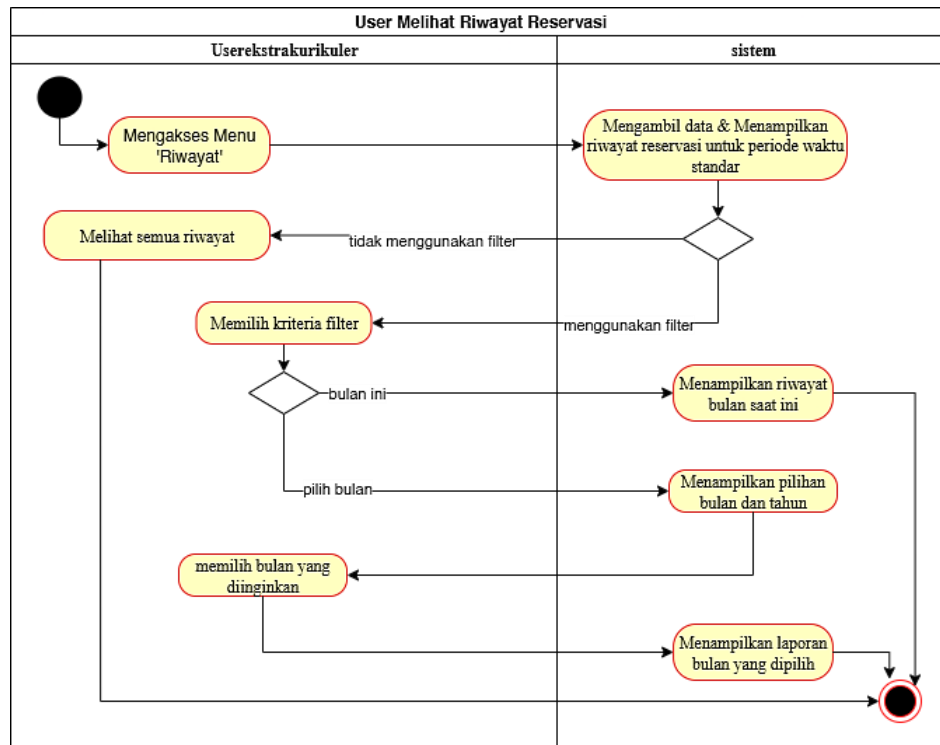
Gambar 3.10 menguraikan alur kerja yang merupakan tanggung jawab utama dan paling krusial dari aktor Supervisor, yaitu proses validasi ajuan reservasi. Fitur ini berfungsi sebagai gerbang kontrol digital pihak sekolah dalam mengelola alokasi dan penggunaan fasilitas. Alur ini memetakan secara rinci setiap interaksi antara Supervisor sebagai peninjau (*validator*) dan Sistem sebagai pemroses (*eksekutor*).

Proses ini secara efektif dimulai ketika Supervisor mengakses halaman beranda. Tanpa perlu tindakan lebih lanjut, Sistem secara proaktif mengambil dan menampilkan seluruh daftar reservasi yang berstatus 'Menunggu Konfirmasi', yang secara fungsional bertindak sebagai 'antrean kerja' (*task queue*) digital bagi Supervisor. Diagram ini juga secara cermat menangani kondisi di mana tidak ada ajuan, dengan menampilkan pesan informatif jika daftar tersebut kosong, sehingga Supervisor tahu tidak ada tindakan yang perlu diambil saat itu.

Ketika terdapat satu atau lebih ajuan di dalam antrean, alur berlanjut dengan Supervisor memilih salah satu ajuan spesifik untuk ditinjau lebih dalam. Sistem kemudian menyajikan halaman detail yang berisi semua informasi relevan—mulai dari identitas pemohon, nama kegiatan, hingga waktu pelaksanaan—yang memungkinkan Supervisor untuk membuat keputusan yang terinformasi. Setelah menganalisis detailnya, Supervisor dihadapkan pada titik keputusan utama yang direpresentasikan oleh *decision node*: menyetujui atau menolak reservasi tersebut berdasarkan kebijakan yang berlaku. Berdasarkan keputusan yang diambil, Supervisor akan menekan tombol aksi yang sesuai ('Terima' atau 'Tolak').

Sistem kemudian akan mengeksekusi perintah ini dengan mengubah status reservasi secara permanen di dalam *database* menjadi 'Disetujui' atau 'Ditolak'. Sebagai langkah akhir yang penting untuk menciptakan siklus kerja yang efisien, Sistem akan menampilkan pesan sukses sebagai umpan balik dan secara otomatis memuat ulang daftar ajuan. Pembaruan otomatis ini memastikan bahwa reservasi yang baru diproses langsung hilang dari antrean, memungkinkan Supervisor untuk secara efisien beralih ke tugas validasi berikutnya tanpa kebingungan.

- *Activity diagram user melihat reservasi*



Gambar 3.11 *Activity diagram user melihat reservasi.*

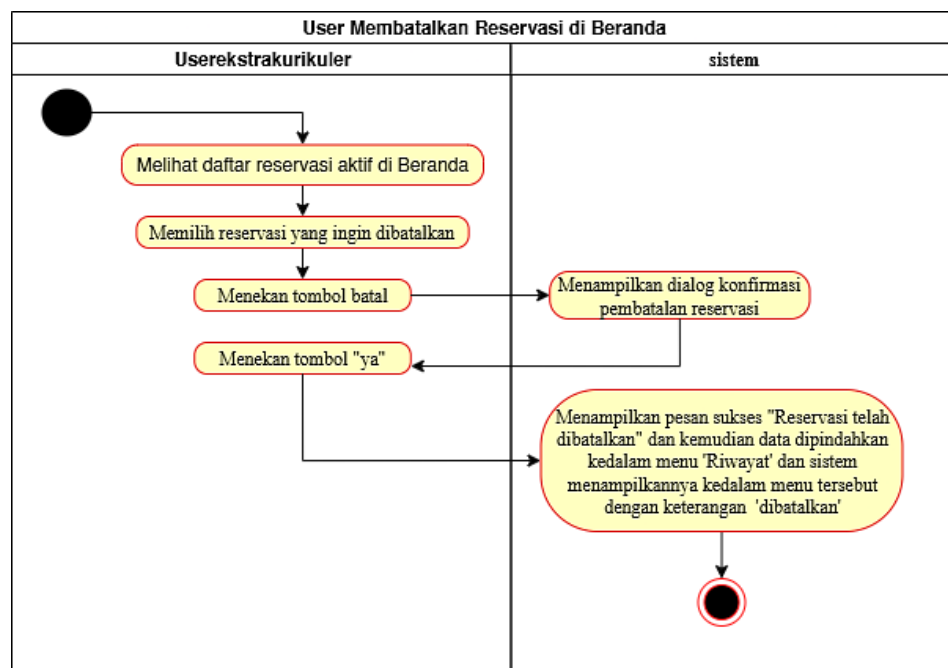
Gambar 3.11 mengilustrasikan alur kerja saat *User* Ekstrakurikuler mengakses fitur riwayat reservasi. Fitur ini dirancang untuk memberikan transparansi penuh dan berfungsi sebagai arsip digital pribadi bagi setiap pengguna, memungkinkan mereka untuk melacak status dan detail dari setiap pengajuan reservasi yang pernah mereka buat, baik yang berhasil maupun yang gagal.

Prosesnya dimulai ketika *User* memilih untuk masuk ke menu 'Riwayat'. Sistem akan merespons dengan menampilkan daftar komprehensif dari semua transaksi reservasi milik pengguna tersebut, tanpa memandang statusnya—baik yang telah 'Disetujui', 'Ditolak', 'Dibatalkan', maupun yang sudah 'Selesai'. Tampilan awal yang lengkap ini memastikan tidak ada informasi yang terlewat. Dari sini, seperti yang ditunjukkan oleh *decision node* pertama, pengguna dapat langsung meninjau seluruh riwayatnya atau memilih untuk melakukan pencarian yang lebih spesifik menggunakan fitur filter. Jika pengguna memilih untuk memfilter data, sistem menyediakan beberapa opsi

untuk mempersempit pencarian, misalnya berdasarkan periode waktu seperti bulan tertentu.

Alur diagram menunjukkan bagaimana sistem akan memproses kriteria yang dipilih dan kemudian menyajikan kembali daftar riwayat yang sudah disaring sesuai permintaan. Kemampuan untuk memfilter ini memberikan kemudahan bagi pengguna, terutama saat mereka perlu mencari detail reservasi spesifik dari masa lalu atau sekadar ingin merekapitulasi kegiatan mereka dalam rentang waktu tertentu, yang meningkatkan kontrol pengguna atas data historis mereka.

- *Activity diagram userekstrakurikuler* membatalkan reservasi di beranda

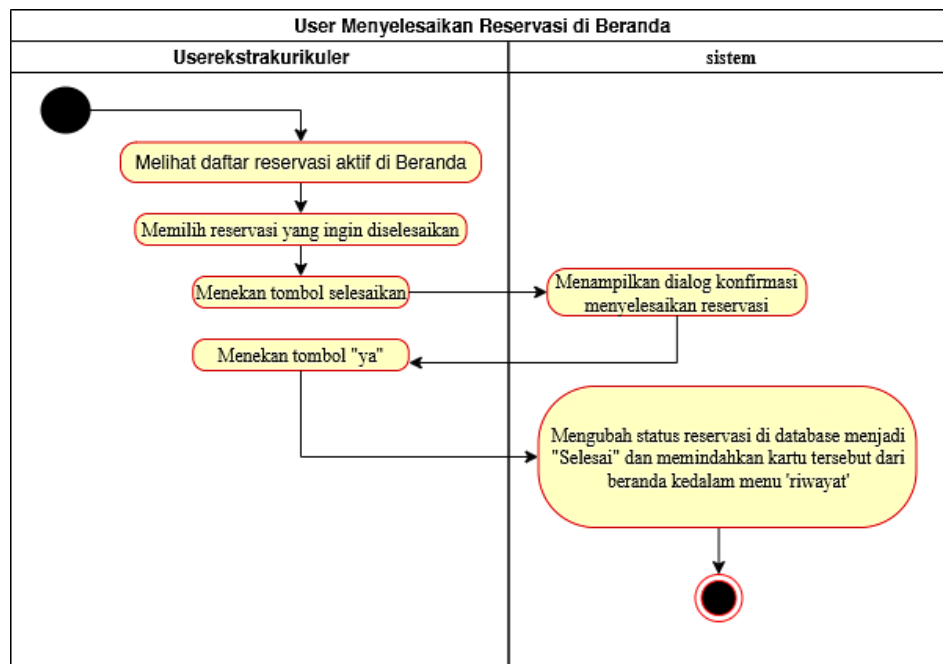


Gambar 3.12 *Activity diagram userekstrakurikuler* membatalkan reservasi di beranda.

Gambar 3.12 memodelkan alur kerja spesifik saat *User* Ekstrakurikuler melakukan pembatalan reservasi. Fungsionalitas ini memberikan fleksibilitas kepada pengguna jika terjadi perubahan rencana sebelum pengajuan mereka divalidasi oleh Supervisor. Proses ini terjadi di halaman Beranda, yang berfungsi sebagai dasbor untuk reservasi-reservasi yang masih aktif. Alur dimulai ketika *User* melihat daftar reservasi aktifnya di Beranda dan memilih

salah satu reservasi yang ingin dibatalkan. Setelah menekan tombol "Batal" yang tersedia pada kartu reservasi tersebut, Sistem akan merespons dengan menampilkan sebuah dialog konfirmasi. Langkah ini berfungsi sebagai lapisan pengaman untuk memastikan pengguna tidak salah menekan tombol atau membatalkan reservasi secara tidak sengaja. Setelah *User* memberikan konfirmasi "ya", Sistem akan mengeksekusi proses pembatalan, yang mencakup perubahan status reservasi menjadi "Dibatalkan" di *database* dan menampilkan pesan sukses. Sebagai langkah akhir, kartu reservasi tersebut akan dipindahkan dari Beranda ke menu Riwayat, sehingga Beranda hanya menampilkan reservasi yang masih relevan.

- *Activity diagram userekstrakurikuler menyelesaikan reservasi di beranda*

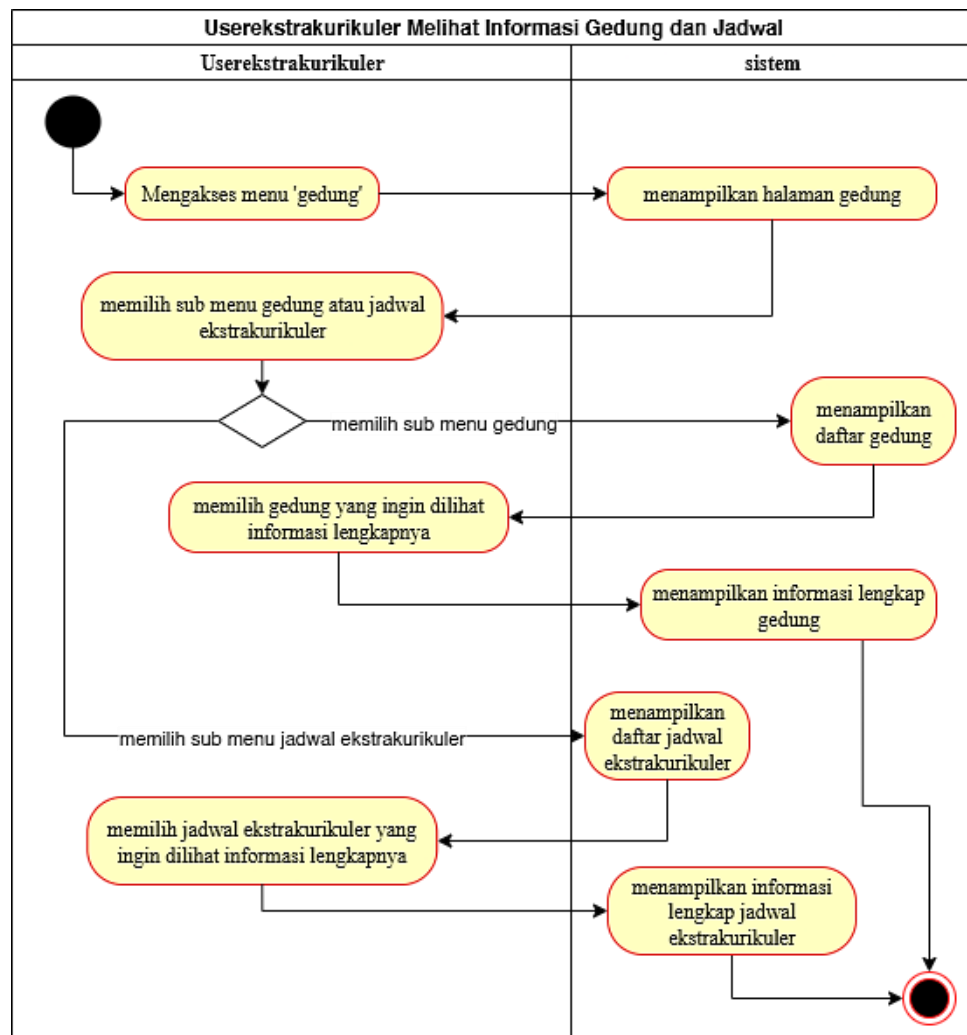


Gambar 3.13 *Activity diagram userekstrakurikuler menyelesaikan reservasi di beranda.*

Gambar 3.13 menggambarkan alur kerja untuk tahap akhir dari siklus sebuah reservasi yang berhasil, yaitu proses penyelesaian. Fungsionalitas ini memungkinkan pengguna untuk secara manual menandai bahwa sebuah kegiatan yang telah disetujui telah selesai dilaksanakan, yang membantu menjaga kebersihan dan relevansi data pada halaman Beranda. Proses ini, sama seperti pembatalan, terjadi di halaman Beranda. Alur dimulai ketika

User melihat daftar reservasi aktifnya dan memilih salah satu reservasi yang telah selesai digunakan (yang berstatus "Disetujui"). Setelah menekan tombol "Selesaikan", Sistem akan menampilkan dialog konfirmasi untuk memastikan tindakan tersebut. Setelah User memberikan konfirmasi "ya", Sistem akan melakukan dua aksi penting: pertama, mengubah status reservasi di *database* menjadi "Selesai", dan kedua, memindahkan kartu reservasi tersebut dari Beranda ke dalam menu Riwayat. Proses ini memastikan bahwa Beranda hanya menampilkan reservasi yang akan datang atau sedang menunggu konfirmasi sementara Riwayat berfungsi sebagai arsip lengkap.

- *Activity diagram Userekstrakurikuler* Melihat Informasi Gedung dan Jadwal

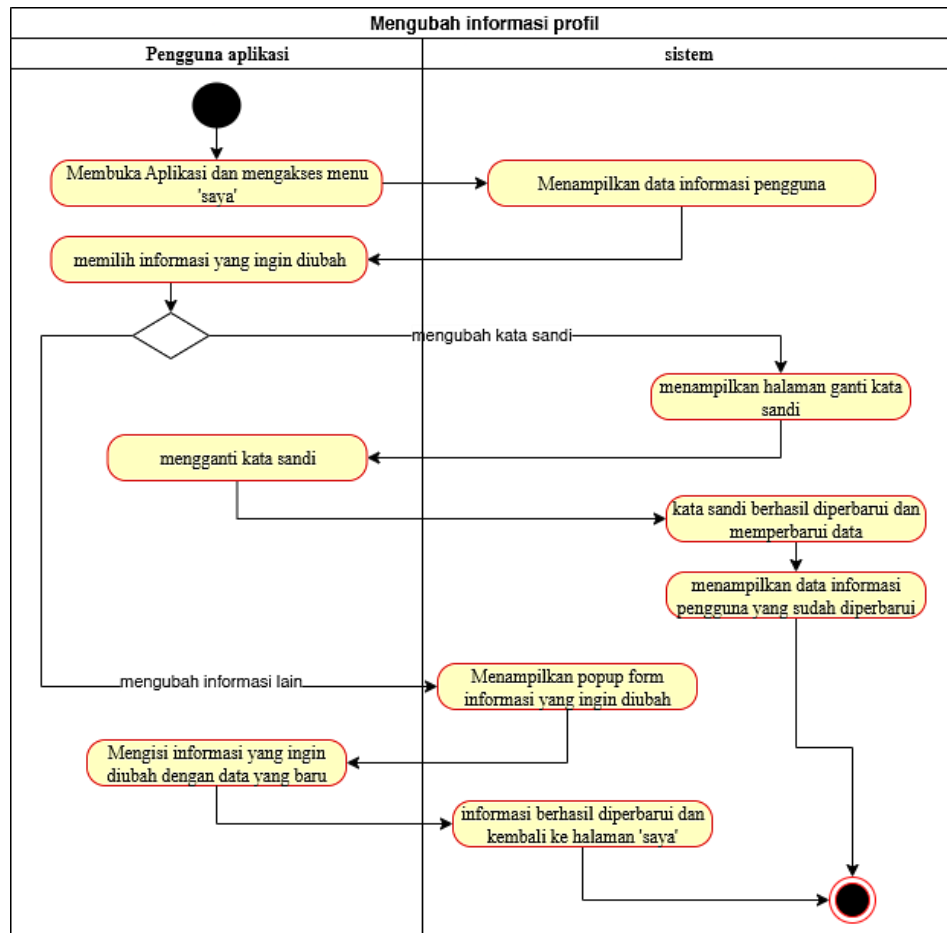


Gambar 3.14 *Activity diagram Userekstrakurikuler* Melihat Informasi Gedung dan Jadwal.

Gambar 3.14 Di atas menjelaskan alur kerja ketika pengguna ekstrakurikuler mengakses fitur informasi yang hanya bisa dilihat, tidak bisa diubah. Fungsi ini dibuat agar pengguna bisa mendapatkan informasi terlebih dahulu sebelum melakukan pendaftaran. Dengan demikian, pengguna bisa melihat detail tentang fasilitas dan jadwal kegiatan yang ada. Proses dimulai saat pengguna masuk ke menu utama informasi, yaitu menu 'Gedung'. Sistem akan merespons dengan menampilkan halaman yang menyediakan pilihan untuk memperoleh informasi lebih lanjut, yaitu sub-menu 'Gedung' atau 'Jadwal Ekstrakurikuler'. Titik ini disebut *decision node*, di mana alur akan terbagi berdasarkan pilihan yang dilakukan pengguna. Jika pengguna memilih menu 'Gedung', sistem akan menampilkan daftar semua gedung yang tersedia.

Pengguna kemudian bisa memilih gedung tertentu untuk melihat informasi lebih jelas, seperti kapasitas dan fasilitas yang ada. Alur kerja yang sama juga berlaku jika pengguna memilih menu 'Jadwal Ekstrakurikuler'; sistem akan menampilkan daftar jadwal, dan pengguna bisa memilih jadwal tertentu untuk melihat detailnya. Alur kerja yang sederhana dan terpisah ini membantu pengguna mencari informasi yang dibutuhkan secara mudah dan cepat.

- *Activity diagram* pengguna mengubah informasi profil



Gambar 3.15 *Activity diagram* pengguna mengubah informasi profil.

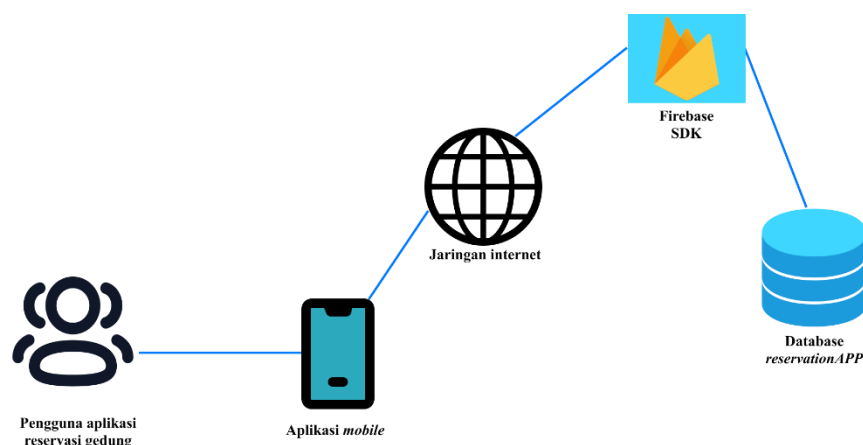
Gambar 3.15 Di atas menjelaskan alur kerja umum yang berlaku untuk semua pengguna aplikasi (*Superadmin*, *Supervisor*, dan *Userekstrakurikuler*) ketika mereka mengelola informasi akun pribadi. Fitur ini memberikan kesempatan bagi pengguna untuk memperbarui data mereka sendiri. Diagram ini menunjukkan bagaimana pengguna aplikasi berinteraksi dengan sistem dalam dua situasi utama: mengganti kata sandi dan mengedit informasi profil lainnya. Proses dimulai ketika pengguna mengakses menu pribadinya, yaitu menu "Saya". Sistem akan merespons dengan menampilkan data informasi pengguna yang sedang aktif. Dari halaman tersebut, pengguna memiliki beberapa pilihan untuk mengedit data yang berbeda, yang ditunjukkan oleh *node* keputusan dalam diagram. Pilihan ini memisahkan alur untuk data yang

bersifat sensitif (kata sandi) dengan data yang bersifat umum. Jika pengguna memilih untuk mengedit informasi profil umum, seperti nama atau nomor telepon, sistem akan menampilkan *pop-up* atau *form* khusus untuk memasukkan data baru. Setelah data diperbarui, sistem akan memberi pesan konfirmasi lalu kembali ke halaman profil utama. Jika pengguna memilih untuk mengubah kata sandi, sistem akan mengarahkan pengguna ke halaman khusus yang dirancang untuk mengganti kata sandi demi menjaga keamanan. Setelah proses selesai, sistem akan memberi konfirmasi dan menampilkan kembali informasi pengguna yang sudah diperbarui.

2. *Arsitektur sistem*

Arsitektur sistem mengacu pada struktur keseluruhan dari suatu sistem informasi, termasuk komponen-komponen utama, interaksi antara komponen-komponen tersebut, serta aturan dan prosedur yang mengatur operasi sistem. Dalam penelitian ini, arsitektur sistem mencakup beberapa aspek penting, yaitu pengguna dalam hal ini pengurus yang berkaitan dengan kegiatan ekstrakurikuler, aplikasi android, tampilan front end, dan backend server.

Arsitektur sistem yang digunakan dalam aplikasi reservasi gedung dengan studi kasus di SMAN 1 Tanjung Bintang memiliki beberapa bagian diantaranya adalah pengguna aplikasi reservasi gedung, aplikasi mobile, jaringan internet, firebase SDK, dan *database* reservation APP.

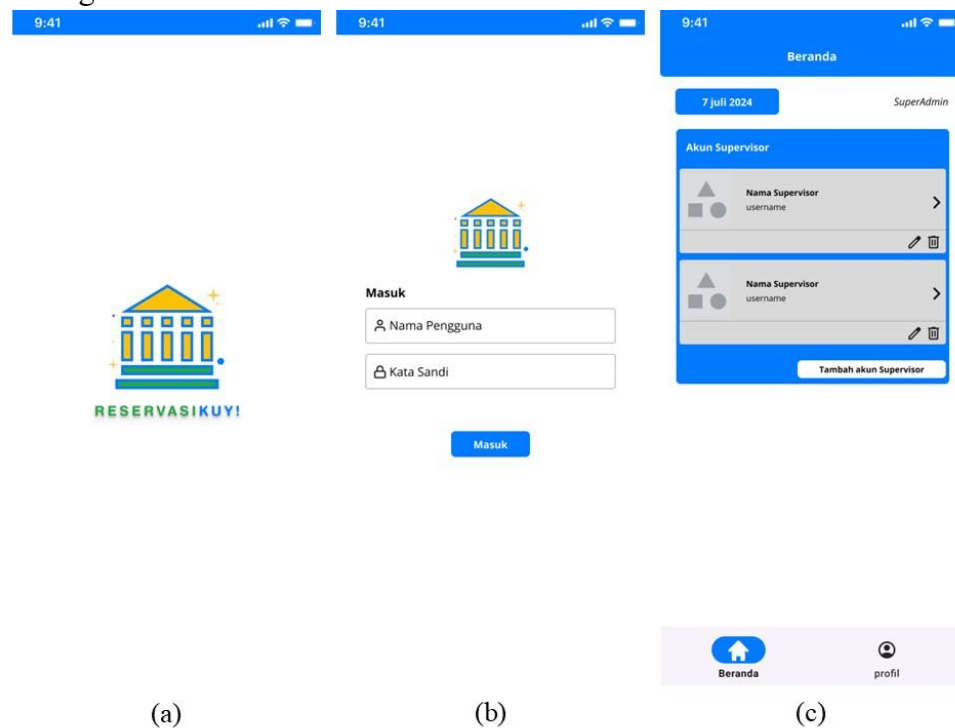


Gambar 3.16 Arsitektur sistem reservasi gedung studi kasus SMAN 1 Tanjung Bintang.

Dalam gambar 4.16 terdapat pengguna aplikasi reservasi gedung yang dapat mengakses aplikasi mobile yang kemudian aplikasi reservasi gedung membutuhkan jaringan internet untuk yang akan menghubungkan aplikasi dengan firebase SDK dan kemudian dapat mengakses *database* aplikasi reservasi gedung studi kasus SMAN 1 Tanjung Bintang dan menampilkan data yang tersedia.

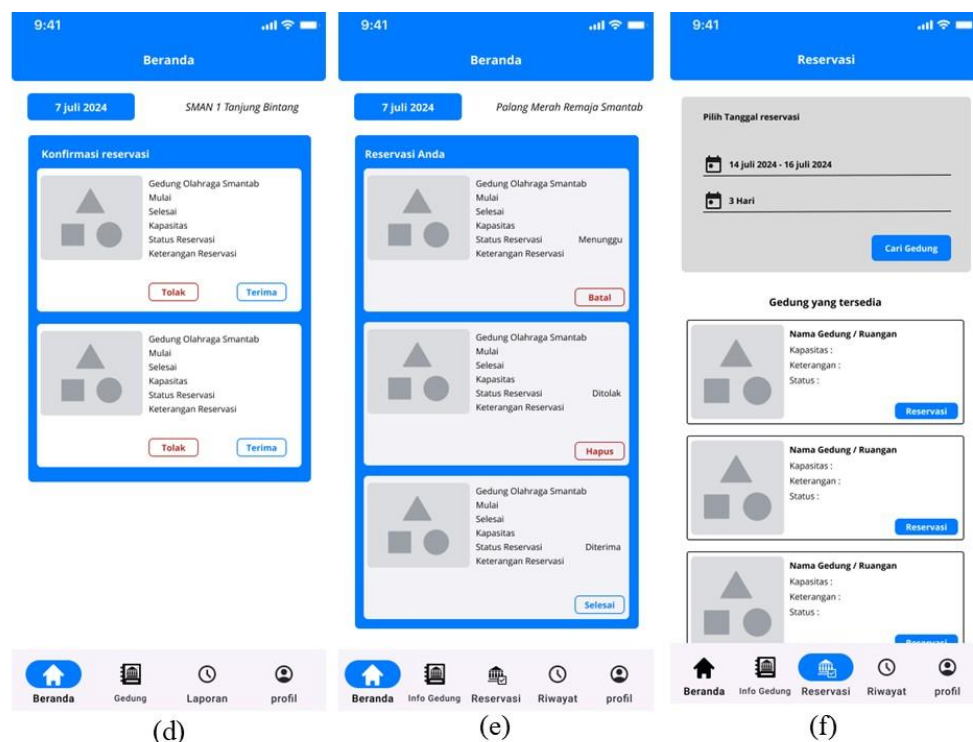
3. Wireframe

Tahap perancangan antarmuka dengan membuat *wireframe design* Sebelum memasuki tahap implementasi visual (*user interface*), diawali dengan membuat serangkaian *wireframe*. *Wireframe* berperan sebagai kerangka dasar (*blueprint*) tata letak halaman yang menekankan fungsionalitas, alur pengguna, serta penempatan elemen-elemen utama tanpa terpengaruh oleh detail desain visual seperti warna atau jenis huruf. Berikut merupakan beberapa detail penting dari *wireframe* yang sudah dirancang untuk aplikasi reservasi gedung untuk kegiatan ekstrakurikuler berbasis android studi kasus SMAN 1 Tanjung Bintang.



Gambar 3.17 *wireframe design* aplikasi ‘RESERVASIKUY’. Keterangan (a) *splash screen* (b) halaman masuk aplikasi (c) halaman beranda *superadmin*.

Gambar 3. 17 di atas menunjukkan desain *wireframe* untuk alur awal aplikasi 'RESERVASIKUY', yang terdiri dari tiga tahapan utama: layar pembuka, proses *login*, dan halaman utama untuk peran *Superadmin*. Alur dimulai dengan layar *splash screen* (a), yang bertugas sebagai panduan visual pertama dan menampilkan logo atau identitas merek aplikasi saat aplikasi pertama kali dijalankan. Selanjutnya, pengguna akan dialihkan ke halaman *login* (b), sebuah antarmuka yang dirancang standar untuk proses verifikasi identitas, di mana pengguna diminta memasukkan nama pengguna dan kata sandi. Setelah berhasil *login* sebagai *Superadmin*, pengguna akan masuk ke halaman beranda (c). Halaman ini dirancang sebagai *dashboard* utama *Superadmin* yang langsung menampilkan daftar seluruh akun Supervisor. Desain *wireframe* ini menunjukkan bahwa setiap item dalam daftar dilengkapi dengan ikon aksi untuk melihat detail, mengedit, dan menghapus, serta terdapat tombol utama di bagian bawah untuk menambah akun Supervisor baru, sehingga mencakup semua fungsionalitas CRUD yang dibutuhkan.



Gambar 3.17 *wireframe design* aplikasi 'RESERVASIKUY'. Keterangan (d) halaman beranda supervisor (e) halaman beranda *user* ekstrakurikuler (f) halaman reservasi pada *user* ekstrakurikuler.(lanjutan)

Gambar 3. 18 menunjukkan desain wireframe untuk antarmuka utama dari dua aktor, yaitu Supervisor dan *Userekstrakurikuler*, yang menekankan perbedaan dalam proses kerja dan fungsi mereka. Halaman beranda Supervisor (d) dirancang sebagai pusat pengendalian untuk memvalidasi ajuan reservasi. Di sini, Supervisor dapat melihat langsung daftar ajuan yang masuk dan memerlukan tindakan. Wireframe ini menampilkan tombol aksi "Tolak" dan "Terima" yang jelas, sehingga memudahkan Supervisor dalam membuat keputusan secara cepat dan efisien. Sementara itu, halaman beranda *Userekstrakurikuler* (e) berfungsi sebagai dasbor untuk memantau status reservasi yang sedang berlangsung. Halaman ini menampilkan kartu reservasi yang diajukan oleh pengguna, lengkap dengan status terkini seperti "Menunggu", "Ditolak", atau "Diterima". Selain itu, halaman ini juga menyediakan tombol aksi seperti "Batal" atau "Selesai". Wireframe halaman reservasi (f) menunjukkan proses awal yang dilalui *Userekstrakurikuler* untuk mengajukan pemesanan. Halaman ini dirancang secara terstruktur, mulai dari pemilihan tanggal yang kemudian akan menampilkan daftar gedung yang tersedia di periode tersebut. Dengan demikian, pengguna hanya dapat memilih jadwal yang valid.

Wireframe yang tampil di atas adalah gambaran dari alur kerja utama dalam aplikasi. Untuk gambaran yang lebih lengkap, rancangan wireframe untuk semua halaman aplikasi, termasuk halaman profil dan halaman pendukung lainnya, telah disertakan secara lengkap dalam Lampiran.

3.4.3. Perancangan papan *kanban*

Perancangan papan *kanban* dalam pembuatan aplikasi Reservasi Gedung untuk Kegiatan Ekstrakurikuler dengan studi kasus di SMAN 1 Tanjung Bintang bertujuan untuk menginisiasi dimulainya pengerjaan pembuatan aplikasi dengan menggunakan metode *Kanban*. Pada tahap ini hal pertama yang dilakukan adalah identifikasi alur kerja, menentukan alat dan platform *kanban* yang akan digunakan, kemudian mulai membuat kolom *kanban*, serta menentukan batasan dalam *Work in*

Progress (WIP) yang akan dilakukan. Struktur papan *Kanban* tersebut kemudian dilengkapi dengan menggunakan serangkaian kartu tugas (*task card*) yang merupakan hasil pemecahan dari *user story*, detailnya dapat dilihat pada tabel dibawah ini.

Tabel 3.6 daftar kartu tugas (*task card*).

ID	Fitur Utama	Deskripsi	Kategori
TC-01	Install Android Studio	Melakukan penginstalan IDE (<i>Integrated Development Environment</i>) yang akan digunakan.	Pengaturan Lingkungan Pengembangan (<i>Development Environment Setup</i>)
TC-02	Install Flutter SDK	Melakukan penginstalan <i>flutter</i> sesuai dengan <i>website</i> resminya.	Pengaturan Lingkungan Pengembangan (<i>Development Environment Setup</i>)
TC-03	Install Dart	Melakukan penginstalan <i>dart</i> untuk menunjang <i>flutter</i> .	Pengaturan Lingkungan Pengembangan (<i>Development Environment Setup</i>)
TC-04	Mengintegrasikan <i>flutter</i> , <i>dart</i> serta <i>plugin</i> yang diperlukan ke dalam android studio.	Menambahkan semua yang diperlukan dalam melakukan pengembangan aplikasi yang akan dilakukan.	Pengaturan Lingkungan Pengembangan (<i>Development Environment Setup</i>)
TC-05	Membuat <i>wireframe</i> .	Membuat <i>wireframe</i> sesuai dengan arahan.	Finalisasi <i>Wireframe (low fidelity)</i>
TC-06	Meninjau <i>wireframe</i> supervisor.	Meninjau dan kemudian menyesuaikan <i>design wireframe</i> sesuai dengan arahan perbaikan yang ada pada aktor supervisor.	Finalisasi <i>Wireframe (low fidelity)</i>
TC-07	Meninjau <i>wireframe</i> <i>userekstrakurikuler</i> .	Meninjau dan kemudian menyesuaikan <i>design wireframe</i> sesuai dengan arahan perbaikan yang ada pada aktor <i>userekstrakurikuler</i> .	Finalisasi <i>Wireframe (low fidelity)</i>
TC-08	Meninjau <i>wireframe</i> <i>superadmin</i> .	Meninjau dan kemudian menyesuaikan <i>design wireframe</i> sesuai dengan arahan perbaikan yang ada pada aktor <i>superadmin</i> .	Finalisasi <i>Wireframe (low fidelity)</i>

Tabel 3.6 daftar kartu tugas (*task card*). (lanjutan)

ID	Fitur Utama	Deskripsi	Kategori
TC-09	Buat proyek baru <i>Flutter</i> untuk <i>development</i> .	Membuat proyek baru <i>flutter</i> untuk mengembangkan aplikasi reservasi gedung untuk kegiatan ekstrakurikuler berbasis android.	Pengaturan Lingkungan Pengembangan (<i>Development Environtment Setup</i>)
TC-10	Integrasi <i>Firebase</i> .	Melakukan integrasi <i>firebase</i> dengan <i>flutter</i> dalam IDE android studio.	Pengaturan Lingkungan Pengembangan (<i>Development Environtment Setup</i>)
TC-11	<i>Splash screen</i>	Membuat <i>splash screen</i> untuk tampilan awal ketika membuka aplikasi.	<i>Developing</i>
TC-12	<i>slicing</i> halaman <i>login</i> untuk semua aktor.	Membuat halaman <i>login</i> dengan menerapkan <i>wireframe</i> .	<i>Developing</i>
TC-13	Membuat semua aktor dapat melakukan <i>login</i> menggunakan nama pengguna dan kata sandi.	Menghubungkan antarmuka <i>login</i> dengan layanan <i>Firebase</i> untuk memvalidasi kredensial (nama pengguna dan kata sandi) dari semua aktor dan dapat masuk sesuai dengan rolenya.	<i>Developing</i>
TC-14	membuat halaman beranda/ <i>home</i> (<i>slicing</i>).	Dalam halaman beranda/ <i>home</i> , akan terdapat beberapa tampilan beranda yang berbeda antara <i>user</i> , <i>superivisor</i> dan juga <i>superadmin</i> , sesuai dengan tugas dan kewenangan dari masing masing aktor.	<i>Developing</i>
TC-15	membuat halaman profil (<i>slicing</i>).	Menerapkan <i>wireframe</i> ke dalam <i>coding</i> UI dari tampilan profil dengan <i>superadmin</i> dan Supervisor memiliki halaman tambahan untuk CRUD akun supervisor oleh <i>superadmin</i> dan <i>userektrakurikuler</i> oleh supervisor.	<i>Developing</i>

Tabel 3.6 daftar kartu tugas (*task card*). (lanjutan)

ID	Fitur Utama	Deskripsi	Kategori
TC-16	<i>Slicing</i> halaman informasi gedung pada aktor <i>userekstrakurikuler</i> dan supervisor.	Membuat halaman informasi gedung dengan mengimplementasikan <i>wireframe</i> .	<i>Developing</i>
TC-17	<i>Slicing</i> halaman reservasi serta pemilihan tanggal reservasi untuk <i>userekstrakurikuler</i> .	Membuat halaman reservasi serta pemilihan tanggal reservasi dengan mengimplementasikan <i>wireframe</i> .	<i>Developing</i>
TC-18	<i>Slicing</i> halaman riwayat dan juga laporan dalam <i>userekstrakurikuler</i> dan juga supervisor.	Membuat halaman riwayat dan juga laporan dengan mengimplementasikan <i>wireframe</i> .	<i>Developing</i>
TC-19	CURD akun supervisor oleh <i>superadmin</i> dan CURD akun <i>userekstrakurikuler</i> oleh supervisor.	Fitur untuk <i>superadmin</i> agar bisa CRUD akun supervisor pada halaman berandanya kemudian supervisor dapat melakukan CRUD akun <i>userekstrakurikuler</i> dalam halaman profilnya.	<i>Developing</i>
TC-20	Aktor dapat mengambil informasi profil dari sistem dan sistem menampilkan ke dalam aplikasi.	Membuat fitur untuk aktor dapat mengambil informasi seputar informasi dari profil yang sudah ada di firestore <i>database</i> dan juga mengedit kemudian menampilkan kedalam aplikasi ReservasiKuy	<i>Developing</i>
TC-21	Aktor supervisor dapat CRUD data gedung dan jadwal ekstrakurikuler.	Membuat fitur agar aktor supervisor dapat CRUD data seputar gedung dan juga jadwal ekstrakurikuler kemudian dapat ditampilkan ke dalam akun <i>userekstrakurikuler</i> juga sesuai dengan instansinya.	<i>Developing</i>

Tabel 3.6 daftar kartu tugas (*task card*). (lanjutan)

ID	Fitur Utama	Deskripsi	Kategori
TC-22	<i>Userekstrakurikuler</i> melakukan reservasi.	Membuat fitur memilih jadwal reservasi gedung pada <i>userekstrakurikuler</i> dan menampilkan gedung yang tersedia dan sukses melakukan reservasi.	<i>Developing</i>
TC-23	Status reservasi dalam beranda <i>userekstrakurikuler</i> serta supervisor.	Menampilkan status reservasi pada beranda <i>userekstrakurikuler</i> dan juga dalam beranda supervisor sesuai dengan instansinya. Dalam beranda <i>userekstrakurikuler</i> terdapat fitur membatalkan reservasi yang sudah diajukan dan juga menyelesaikan reservasi.	<i>Developing</i>
TC-24	Supervisor Mengkonfirmasi reservasi.	Supervisor dapat menerima atau menolak reservasi yang diajukan oleh <i>userekstrakurikuler</i> dan juga informasi pengajuan reservasi dalam beranda <i>userekstrakurikuler</i> statusnya berubah.	<i>Developing</i>
TC-25	Riwayat <i>userekstrakurikuler</i> dan Laporan supervisor.	Menambahkan menu Riwayat Reservasi dalam <i>user</i> ekstrakurikuler dan juga Laporan dalam supervisor sesuai dengan agensinya (instansi) serta menampilkannya.	<i>Developing</i>
TC-26	Pengujian Aplikasi dengan <i>Blackbox testing Decision table</i> .	Melakukan <i>Blackbox testing Decision table</i> pada aplikasi <i>Reservasikuy</i> .	<i>Testing</i>
TC-27	Cek Aplikasi	cek kembali keseluruhan aplikasi	<i>Developing</i>
TC-28	Persiapan untuk UEQ	Menemui pihak dari SMAN 1 Tanjung bintang dan membicarakannya serta menyiapkan dokumen dari UEQ serta finalisasi aplikasi sebelum dilakukan UEQ.	<i>Testing</i>

Tabel 3.6 daftar kartu tugas (*task card*). (lanjutan)

ID	Fitur Utama	Deskripsi	Kategori
TC-29	Pelaksanaan UEQ.	Melakukan testing aplikasi dengan menggunakan UEQ pada aplikasi <i>Reservasikuy</i> .	<i>Testing</i>
TC-30	Hasil dari UEQ.	Mengolah data dari kuisisioner UEQ serta mendapatkan hasilnya.	<i>Testing</i>

3.4.4. Pengembangan dengan menggunakan *Kanban (Develop)*

Papan *Kanban* digunakan untuk mendukung pengelolaan tugas dalam pengembangan aplikasi reservasi gedung dengan studi kasus di SMAN 1 Tanjung Bintang secara mandiri. Metode ini dipilih karena memberikan visualisasi yang jelas terhadap alur kerja, membantu menentukan prioritas, serta mempermudah proses pemantauan dan evaluasi progres.

Pengembangan dengan menggunakan *kanban (develop)* dalam pembuatan aplikasi Reservasi Gedung untuk Kegiatan Ekstrakurikuler dengan studi kasus di SMAN 1 Tanjung Bintang bertujuan untuk menginisiasi dimulainya pengerjaan pembuatan aplikasi dengan menggunakan metode *Kanban*. Pada tahap ini hal pertama yang dilakukan adalah identifikasi alur kerja, menentukan alat dan platform *kanban* yang akan digunakan, kemudian mulai membuat kolom *kanban*, serta menentukan batasan dalam *Work in Progress (WIP)* yang akan dilakukan yang di mana dalam *develop* ini memiliki batasan (2) WIP.

Dalam pembuatan aplikasi reservasi gedung untuk kegiatan ekstrakurikuler studi kasus di SMAN 1 Tanjung Bintang terdapat beberapa kolom. Untuk menentukan *backlog* diperlukan *user story* yang menjadi kolom pertama yang kemudian menjadi pedoman dalam penentuan *backlog* yang digunakan dalam pengembangan aplikasi. Kemudian terdapat kolom *work on Wait (WOW)* atau *To do* yang berisi tugas-tugas yang perlu dikerjakan pada pengembangan aplikasi sistem informasi reservasi gedung untuk kegiatan ekstrakurikuler di SMAN 1 Tanjung Bintang. Selanjutnya kolom *Work in Progress* yang menampung tugas-tugas yang sedang dikerjakan. Untuk menjaga fokus dan efisiensi, hanya dua tugas

yang boleh berada di kolom ini pada satu waktu. Batasan ini membantu memastikan bahwa setiap tugas dapat diselesaikan dengan maksimal sebelum beralih ke pekerjaan lain. Membatasi, dan kemudian mengoptimalkan jumlah pekerjaan yang terdapat dalam *WIP* yang masuk sangat penting untuk kesuksesan dengan *Kanban*. Selanjutnya Kolom *Failed/Pending* yang digunakan untuk menyimpan tugas yang mengalami hambatan atau perlu peninjauan lebih lanjut. Kolom terakhir adalah *Done* yang menampung tugas-tugas yang telah selesai dikerjakan. Kolom ini menunjukkan hasil kerja yang telah dicapai dan membantu memantau progres secara keseluruhan.

Pada tahapan pengembangan dengan menggunakan *kanban (develop)* terdapat beberapa serangkaian aktivitas utama yang dilakukan dalam pengembangan aplikasi yaitu Pengaturan Lingkungan Pengembangan (*Development Environment Setup*), Finalisasi *wireframe (low fidelity)*, *Developing* yang terdiri dari *user interface* dan *coding*, selanjutnya adalah testing dengan *Blackbox testing Decision table* dan *User Experience Questionnaire (UEQ)* yang di mana jika sudah sesuai dengan testing maka lanjut ke tahap *deploy* jika belum maka akan dikembalikan ke dalam tahapan *developing*. Dalam tahapan-tahapan tersebut diwakili dengan *task card* yang nantinya akan mengisi pada bagian kolom-kolom *WOW (Work on Wait)*, *WIP (Work in Progress)* dengan maksimal 2 *task card* yang bisa dikerjakan dalam satu waktu), *Failed/Pending* dan *Done*.

3.4.5. Deploy

Tahap Penyebaran atau *Deploy* adalah puncak dari seluruh proses pengembangan teknis dalam studi ini. Fase ini dijalankan setelah aplikasi berhasil melewati semua uji coba fungsional (*Black box Testing*) dan memperoleh hasil yang baik dari Pengujian Pengalaman Pengguna (*User Experience Questionnaire*). Fokus utama dari tahap penyebaran dalam kajian ini bukanlah untuk peluncuran komersial, melainkan untuk menyempurnakan perangkat lunak menjadi artefak yang stabil, bisa didistribusikan, dan siap untuk ditunjukkan. Kegiatan utama dalam tahap ini dimulai dengan kompilasi (*build*) seluruh kode sumber (*source code*) dari

aplikasi. Memanfaatkan alat dari *framework Flutter*, proyek aplikasi yang sebelumnya berada dalam mode pengembangan disusun menjadi paket aplikasi Android (*file .apk*) yang mandiri. *File .apk* ini merupakan bentuk akhir dan nyata dari aplikasi "RESERVASIKUY" yang dapat *diinstal* pada semua perangkat Android yang memenuhi spesifikasi minimum tanpa perlu lingkungan pengembangan lagi.

3.4.6. Penulisan Laporan

Tahapan selanjutnya yaitu penulisan laporan. Tujuan dari penulisan laporan ini yaitu untuk mendokumentasikan penelitian yang telah dilakukan, agar bisa bermanfaat bagi pembaca.

V. KESIMPULAN & SARAN

5.1 Kesimpulan

Berdasarkan hasil implementasi, pengujian, dan pembahasan yang telah diuraikan pada bab-bab sebelumnya, maka dapat ditarik kesimpulan yang menjawab tujuan penelitian, maka dapat ditarik kesimpulan sebagai berikut:

1. Penelitian ini telah berhasil merancang dan membangun sebuah aplikasi sistem informasi reservasi gedung berbasis Android yang secara fungsional terbukti mampu menggantikan proses manual yang sebelumnya tidak efisien dan rentan terhadap kesalahan data di SMAN 1 Tanjung Bintang.
2. Sistem yang dibangun berhasil mengimplementasikan fitur penyajian informasi ketersediaan jadwal secara *real-time* dan transparan. Fungsionalitas ini, yang telah divalidasi melalui pengujian *black box*, secara efektif mampu mencegah terjadinya konflik jadwal antar pengguna.
3. Penelitian ini telah berhasil merancang sebuah sistem yang mampu menyediakan rekam jejak digital dari seluruh aktivitas reservasi. Fitur ini mempermudah proses pengawasan dan rekapitulasi data oleh pihak Supervisor sekolah, yang sebelumnya sulit dilakukan.
4. Berdasarkan hasil pengujian fungsionalitas dan pengalaman pengguna (UEQ), telah diukur bahwa aplikasi yang dibangun tidak hanya berjalan sesuai kebutuhan, tetapi juga memiliki tingkat penerimaan dan kepuasan yang sangat baik dari pengguna akhir.

5.2 Saran

Berdasarkan hasil penelitian dan potensi pengembangan di masa mendatang, penulis mengajukan beberapa saran yang dapat dijadikan acuan untuk iterasi pengembangan aplikasi selanjutnya sebagai berikut:

1. Pengembangan *Dashboard* Admin Berbasis Web: Guna mempermudah tugas manajemen yang dilakukan oleh Supervisor dan *Superadmin*, sangat disarankan untuk membangun sebuah antarmuka *dashboard* berbasis web. *Dashboard* ini akan menyediakan tampilan yang lebih luas dan perangkat kerja yang lebih optimal untuk pengelolaan data reservasi, pengguna, dan jadwal gedung melalui perangkat komputer atau laptop.
2. Implementasi fitur untuk menonaktifkan akun pengguna sebagai pengganti penghapusan permanen. Mekanisme *soft delete* ini penting untuk menjaga keutuhan data historis, seperti riwayat reservasi yang masih terkait dengan akun tersebut. Sebagai gantinya, data akun hanya diberi tanda "nonaktif" dan tidak dihapus sepenuhnya dari *database*. Hal ini memungkinkan pemulihan akun jika terjadi kesalahan, serta memastikan jejak audit data tetap lengkap. Penambahan fitur ini akan membuat sistem lebih aman, fleksibel, dan *handal* dalam mengelola data secara jangka panjang.

DAFTAR PUSTAKA

- [1] SIMON KEMP, “DIGITAL 2024: INDONESIA,” datareportal.com. Accessed: Jun. 23, 2024. [Online]. Available: <https://datareportal.com/reports/digital-2024-indonesia>
- [2] A. A. Prasetya, “Sistem Informasi Reservasi Gedung Serbaguna di Kota Palembang Berbasis Android,” *Jurnal Intelektualita: Keislaman, Sosial dan Sains*, vol. 7, no. 1, pp. 121–128, 2018.
- [3] S. I. M. Suryadharma, S. E. Triyani Budyastuti, and M. Ak, *Sistem Informasi Manajemen*. Uwais Inspirasi Indonesia, 2019.
- [4] N. Y. Arifin *et al.*, *Analisa Perancangan Sistem Informasi*. Cendikia Mulia Mandiri, 2022.
- [5] Irmayana, A. Salam, and Muammar, “Pengembangan Sistem Informasi Manajemen Pasien Berbasis Komputer untuk Meningkatkan Efisiensi di Rumah Sakit,” *Jurnal Manajemen Sistem Informasi (JMASIF)*, vol. 3, no. 1, pp. 33–42, Apr. 2024, doi: 10.59431/jmasif.v3i1.462.
- [6] S. F. Pane, I. H. Hardy, and E. C. Sujadi, *Pengembangan Smart Conveyor Pada Tracking Barang Berbasis IOT*, vol. 1. Kreatif, 2020.
- [7] I. B. Nuryaman, D. K. Mulyantoro, and P. P. Rumah, *Model Edukasi Radioterapi Berbasis Mobile Android pada Pasien yang Menjalani Terapi Radiasi Eksternal*. Penerbit Pustaka Rumah Cinta, 2022.
- [8] S. Herlinah and K. H. Musliadi, *Pemrograman Aplikasi Android dengan Android Studio, Photoshop, dan Audition*. Elex Media Komputindo, 2019.
- [9] J. Enterprise, *Pengantar Pemrograman Dart dan Flutter*. Elex Media Komputindo, 2021.
- [10] R. Syaputra and Y. P. W. Ganda, *Happy Flutter: Membuat Aplikasi Android dan iOS dengan Mudah menggunakan Flutter-UDACODING*. Udacoding, 2019.

- [11] R. Syaputra and Y. P. Wiraganda, *Enjoy Flutter: Membuat Aplikasi Android dan iOS dengan Mudah Menggunakan Flutter Level Intermediate - UDACODING*. UDACODING, 2019. [Online]. Available: <https://books.google.co.id/books?id=R4qsDwAAQBAJ>
- [12] A. L. Ginting, M. E. SE, M. F. Hafa, and L. Ariyanti, *MEDIA DART GAME KEBERAGAMAN SUKU BANGSA BUDAYA INDONESIA*. Scopindo Media Pustaka, 2023.
- [13] D. J. Anderson and A. Carmichael, “Essential Kanban Condensed. Statewide Agricultural Land Use Baseline 2015(Vol. 1). Lean Kanban University Press,” 2016.
- [14] E. P. Yunastyo, A. B. Pranogyo, and D. T. Yulianto, “Kanban Mastery: Terobosan Revolusioner dalam Pengelolaan Persediaan untuk Meningkatkan Efisiensi Material Inventori,” 2024.
- [15] W. Zayat and O. Senvar, “Framework Study for Agile Software Development Via Scrum and Kanban,” *International Journal of Innovation and Technology Management*, vol. 17, no. 04, p. 2030002, May 2020, doi: 10.1142/S0219877020300025.
- [16] L. Magdalena *et al.*, *AGILE PRACTICE: PERAN TRELLO DALAM ADOPSI AGILE SCRUM PADA PENGEMBANGAN IDE BISNIS START UP*. CV. Green Publisher Indonesia, 2023.
- [17] E. Setyawati, ruby santamoko, agus handoko, P. Setiawan, H. Wijoyo, and A. Wibowo, *Manajemen Proyek Sistem Informasi*. 2021.
- [18] S. Rosalin, K. S. Rahayu, R. B. Utami, L. Edityastono, and R. Yuliawan, *Administrasi Perkantoran Berbasis Teknologi Informasi*. Universitas Brawijaya Press, 2022.
- [19] M. Muslihudin, *Analisis dan perancangan Sistem Informasi menggunakan model Terstruktur dan UML*. Penerbit Andi, 2016.
- [20] S. Maesaroh *et al.*, *Rekayasa Perangkat Lunak*. Sada Kurnia Pustaka, 2024. [Online]. Available: <https://books.google.co.id/books?id=Baj4EAAAQBAJ>
- [21] R. F. Purnomo, O. W. Purbo, and R. Z. A. Aziz, *Firestore: Membangun Aplikasi Berbasis Android*. Penerbit Andi, 2021.

- [22] D. B. P. Dessyanto BP, *Manajemen Jaringan Menggunakan Firebase Cloud Massaging Berbasis Android Teori dan Praktek*. LPPM UPN Veteran Yogyakarta, 2019.
- [23] M. Prabowo, *Metodologi pengembangan sistem informasi*. LP2M Press IAIN Salatiga, 2020.
- [24] S. Nidhra, “Black box and White Box Testing Techniques - A Literature Review,” *International Journal of Embedded Systems and Applications*, vol. 2, pp. 29–50, Jun. 2012, doi: 10.5121/ijesa.2012.2204.
- [25] S. Wicaksono, *Blackbox Testing Teori dan Studi Kasus*. 2022. doi: 10.5281/zenodo.7659674.
- [26] J. Joosten, A. E. Permanasari, and T. B. Adji, “The use of *decision table* for reducing complex rules in software testing,” *IOP Conf Ser Mater Sci Eng*, vol. 732, no. 1, p. 012086, Jan. 2020, doi: 10.1088/1757-899X/732/1/012086.
- [27] S. J. Putri, D. G. P. Putri, and W. H. Putra, “Analisis komparasi pada teknik *black box* testing (studi kasus: website lars),” *Journal of Internet and Software Engineering (JISE)*, vol. 5, no. 1, pp. 23–28, 2024.
- [28] M. Schrepp, “*User Experience Questionnaire* handbook,” *All you need to know to apply the UEQ successfully in your project*, pp. 50–52, 2015.
- [29] R. R. C. Putra and E. B. Perkasa, “Aplikasi Peminjaman Ruangan Rapat Kantor Gubernur Provinsi Kepulauan Bangka Belitung Berbasis Android,” *Jurnal Sisfokom (Sistem Informasi dan Komputer)*, vol. 8, no. 2, pp. 191–198, 2019.
- [30] K. D. Sukmana, K. Q. Fredlina, and P. T. H. Permana, “Model Aplikasi Peminjaman Ruangan Berbasis Web Pada Tingkat Fakultas di Perguruan Tinggi,” *Jutisi: Jurnal Ilmiah Teknik Informatika dan Sistem Informasi*, vol. 11, no. 2, pp. 393–402, 2022.
- [31] A. A. Prasetya, “Sistem Informasi Reservasi Gedung Serbaguna di Kota Palembang Berbasis Android,” *Jurnal Intelektualita: Keislaman, Sosial dan Sains*, vol. 7, no. 1, pp. 121–128, 2018.
- [32] Y. Novianto, N. Aulia, and D. Z. Abidin, “Perancangan Sistem Informasi Penyewaan Lapangan Badminton Pada Hall Gembira Kota Jambi Berbasis

- Web,” *Jurnal Manajemen Teknologi Dan Sistem Informasi (JMS)*, vol. 4, no. 2, pp. 803–812, 2024.
- [33] D. K. Hiuredhy and Y. R. Beeh, “Aplikasi Reservasi Ibadah Mawar Sharon Salatiga Menggunakan Flutter,” *JATISI (Jurnal Teknik Informatika dan Sistem Informasi)*, vol. 9, no. 3, pp. 1739–1751, 2022.
- [34] R. S. Pressman, *Software engineering: a practitioner’s approach*. Palgrave macmillan, 2005.
- [35] Rehkopf Max, “User stories with examples and a template,” atlassian.com. Accessed: Jul. 12, 2024. [Online]. Available: <https://www.atlassian.com/agile/project-management/user-stories>