

**RANCANG BANGUN APLIKASI PENGELOLAAN PRAKTIKUM DAN  
ASISTENSI BERBASIS MOBILE  
(STUDI KASUS: LABORATORIUM RPL UNIVERSITAS HASANUDDIN)**

**MUHAMMAD FAJRI RASID  
H071191051**



**PROGRAM STUDI SISTEM INFORMASI  
DEPARTEMEN MATEMATIKA  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS HASANUDDIN  
MAKASSAR  
2024**

**RANCANG BANGUN APLIKASI PENGELOLAAN PRAKTIKUM DAN  
ASISTENSI BERBASIS MOBILE  
(STUDI KASUS: LABORATORIUM RPL UNIVERSITAS HASANUDDIN)**

**MUHAMMAD FAJRI RASID  
H071191051**

Skripsi

Sebagai salah satu syarat untuk memperoleh gelar sarjana  
Program Studi Sistem Informasi

Pada

**PROGRAM STUDI SISTEM INFORMASI  
DEPARTEMEN MATEMATIKA  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS HASANUDDIN  
MAKASSAR  
2024**

## SKRIPSI

**RANCANG BANGUN APLIKASI PENGELOLAAN PRAKTIKUM DAN  
ASISTENSI BERBASIS MOBILE  
(STUDI KASUS: LABORATORIUM RPL UNIVERSITAS HASANUDDIN)**

**MUHAMMAD FAJRI RASID**  
**H071191051**

**Skripsi,**

Telah dipertahankan di depan Panitia Ujian Sarjana Sistem Informasi  
Pada 29 Oktober 2024  
dan dinyatakan telah memenuhi syarat kelulusan  
pada

**Program Studi Sistem Informasi  
Departemen Matematika  
Fakultas Matematika dan Ilmu Pengetahuan Alam  
Universitas Hasanuddin  
Makassar**



Mengesahkan:  
Pembimbing Tugas Akhir,

A handwritten signature in black ink, appearing to be 'Edy Saputra Rusdi'.

**Edy Saputra Rusdi, S.Si., M.Si.**  
NIP. 199104102020053001

Mengetahui:  
Ketua Program Studi,

A handwritten signature in black ink, appearing to be 'Prof. Drs. Jeffry Kusuma'.

**Prof. Drs. Jeffry Kusuma, Ph. D.**  
NIP. 196411121987031002

## PERNYATAAN KEASLIAN SKRIPSI DAN PELIMPAHAN HAK CIPTA

Dengan ini saya menyatakan bahwa, skripsi yang berjudul "Rancang Bangun Aplikasi Pengelolaan Praktikum dan Asistensi Berbasis Mobile (Studi Kasus: Laboratorium RPL Universitas Hasanuddin)" adalah benar karya saya dengan arahan dari dosen pembimbing (Edy Saputra Rusdi, S.Si., M.Si.). Karya ilmiah ini belum diajukan dan tidak sedang diajukan dalam bentuk apa pun kepada perguruan tinggi mana pun. Sumber informasi yang berasal atau dikutip dari karya yang diterbitkan maupun tidak diterbitkan dari penulis lain telah disebutkan dalam teks dan dicantumkan dalam Daftar Pustaka skripsi ini. Apabila di kemudian hari terbukti atau dapat dibuktikan bahwa sebagian atau keseluruhan skripsi ini adalah karya orang lain, maka saya bersedia menerima sanksi atas perbuatan tersebut berdasarkan aturan yang berlaku.

Dengan ini saya melimpahkan hak cipta (hak ekonomis) dari karya tulis saya berupa skripsi ini kepada Universitas Hasanuddin.

Makassar, 29 Oktober 2024

  
FCF20ALX164109313 ammad Fajri Rasid  
H071191051

## UCAPAN TERIMA KASIH

Puji syukur penulis panjatkan ke hadirat Tuhan yang Maha Esa atas segala limpahan rahmat dan karunia-Nya, yang telah memberikan kesempatan dan kelancaran bagi penulis dalam menyelesaikan tugas akhir yang berjudul “Rancang Bangun Aplikasi Pengelolaan Praktikum dan Asistensi Berbasis Mobile (Studi Kasus: Laboratorium RPL Universitas Hasanuddin)”. Dengan berbagai macam tantangan yang dihadapi ketika menyelesaikan tugas akhir ini, penulis juga ingin mengucapkan terima kasih atas kontribusi dan bantuan yang telah diberikan oleh:

1. Kedua orang tua penulis, Bapak **Abdul Rasid** dan Ibu **Hamsiah** serta kakak penulis, **Andi Kiki Rezki** dan **Andi Nurlinda Sari** yang tidak pernah lelah mendoakan, memotivasi, dan memberikan dukungan moral maupun materi.
2. Ketua Program Studi Sistem Informasi, Bapak **Prof. Drs. Jeffry Kusuma, Ph. D.** atas seluruh ilmu dan saran yang telah diberikan.
3. Pembimbing Tugas Akhir sekaligus Penasehat Akademik penulis, Bapak **Edy Saputra Rusdi, S.Si., M.Si.** yang senantiasa telah membantu membimbing dan memberikan arahan selama masa studi penulis, khususnya dalam masa penyusunan skripsi.
4. Kedua dosen penguji, Bapak **Dr. Hendra, S.Si., M.Kom.** dan Bapak **Jeriko Gormantara, S.Si., M.Si.** yang telah memberikan kritik serta saran yang bermanfaat, sehingga skripsi ini dapat tersusun dengan baik.
5. Para **Dosen Program Studi Sistem Informasi** beserta seluruh tenaga pendidik yang telah memberikan ilmu dan mendidik penulis selama masa perkuliahan, serta seluruh staf dan pegawai **Departemen Matematika** yang telah membantu penulis dalam berbagai hal terkait proses administrasi.
6. Seluruh teman-teman **Mahasiswa Program Studi Sistem Informasi Angkatan 2019** yang telah menemani penulis selama masa perkuliahan.
7. Segenap asisten dan praktikan dari **Laboratorium RPL Program Studi Sistem Informasi** karena telah membantu penulis dalam menguji sistem yang dikembangkan pada penelitian ini.
8. **Muhammad Alif Setya Prakasa S.Kom.**, selaku *software developer* yang membantu penulis dalam mengembangkan *back-end* dari sistem yang dibuat pada penelitian ini.
9. Terkasih **Wd. Ananda Lesmono**, sebagai seseorang yang selalu menemani, membantu, dan memberikan segala dukungannya kepada penulis dalam mengerjakan skripsi ini hingga selesai.
10. Sahabat-sahabat Fokus Hidup alias Mount Hua Sect, **Andi Ilhamsyah, Alif Setya, Bayu Ajid, Eurico Devon, Fatwa Anugrah, Muhammad Ikhsan, Muhammad Sulthan, Muhammad Takdim, Muammar Ahlan, Rafly Fatur, Rafly Masloman, Reski Anugrah, Richard Enrico, Silverius Sony, Taufiq Goe, Theodarryl, dan Yusuf Syam** yang senantiasa menemani serta memberikan dukungan selama perkuliahan hingga selesainya penulisan skripsi penulis.

Penulis menyadari bahwa skripsi ini masih jauh dari sempurna dikarenakan terbatasnya pengalaman dan pengetahuan yang dimiliki oleh penulis. Oleh karena itu, penulis mengharapkan segala bentuk saran serta masukan bahkan kritik yang membangun dari berbagai pihak. Semoga tulisan ini memberikan manfaat kepada semua pihak yang membutuhkan dan terutama untuk penulis.

Makassar, 29 Oktober 2024



*Muhammad Fajri Rasid*

## ABSTRAK

MUHAMMAD FAJRI RASID. **Rancang Bangun Aplikasi Pengelolaan Praktikum dan Asistensi Berbasis Mobile Studi Kasus: Laboratorium RPL Universitas Hasanuddin** (dibimbing oleh Edy Saputra Rusdi, S. Si., M. Si.).

Pengelolaan praktikum dan asistensi di Laboratorium Rekayasa Perangkat Lunak (RPL) Universitas Hasanuddin masih dilakukan secara manual, sehingga menimbulkan berbagai masalah seperti kesulitan mengakses materi, pencatatan absensi yang lama, dan kesalahan penginputan nilai. Penelitian ini bertujuan untuk merancang dan membangun aplikasi *mobile* guna mengatasi permasalahan tersebut, serta menguji dan mengevaluasi aplikasi yang telah dibangun. Penelitian ini menggunakan metode pengembangan sistem *waterfall*. Aplikasi dibangun dengan menggunakan *framework* Flutter, serta REST API untuk pengintegrasian *database*. Aplikasi ini menyediakan fitur untuk tiga jenis pengguna, yaitu admin, asisten, dan praktikan, yang meliputi pengelolaan data praktikum, absensi dengan *QR code*, *leaderboard*, ekspor data ke Excel, serta kartu kontrol asistensi digital. Pengujian fungsionalitas dilakukan dengan metode *black-box testing*, menunjukkan bahwa seluruh fitur telah berjalan dengan baik sesuai dengan masukan yang diberikan. *User Acceptance Testing* (UAT) menghasilkan skor 87,2% dari praktikan, 84,6% dari asisten, dan 87,2% dari admin, menunjukkan bahwa aplikasi telah memenuhi kebutuhan pengguna dan berhasil meningkatkan efisiensi pengelolaan praktikum dan asistensi di Laboratorium RPL Universitas Hasanuddin.

**Kata kunci:** rancang bangun, pengelolaan praktikum, aplikasi *mobile*

## ABSTRACT

MUHAMMAD FAJRI RASID. ***Design of Mobile-Based Practicum and Assistance Management Application Case Study: Hasanuddin University RPL Laboratory*** (supervised by Edy Saputra Rusdi, S. Si., M. Si.).

*The management of practicum and assistance in the Software Engineering Laboratory (RPL) of Hasanuddin University is still done manually, causing various problems such as difficulty accessing the material, long attendance recording, and score input errors. This research aims to design and build mobile applications to overcome these problems, as well as test and evaluate the applications that have been built. This research uses the waterfall system development method. The application was built using the Flutter framework, as well as the REST API for database integration. This application provides features for three types of users, namely admin, assistant, and practitioner, which include practicum data management, attendance with QR code, leaderboard, data export to Excel, and digital assistance control card. Functionality testing was done using the black-box testing method, showing that all features have run well according to the input. User Acceptance Testing (UAT) resulted in a score of 87.2% from practitioners, 84.6% from assistants, and 87.2% from admins, indicating that the application has met user needs and succeeded in increasing the efficiency of practicum management and assistance at the Hasanuddin University RPL Laboratory.*

**Keywords:** *design and build, practicum management, mobile application*

## DAFTAR ISI

<b>HALAMAN JUDUL .....</b>	<b>i</b>
<b>PERNYATAAN PENGAJUAN .....</b>	<b>ii</b>
<b>HALAMAN PENGESAHAN.....</b>	<b>iii</b>
<b>PERNYATAAN KEASLIAN SKRIPSI .....</b>	<b>iv</b>
<b>UCAPAN TERIMA KASIH .....</b>	<b>v</b>
<b>ABSTRAK.....</b>	<b>vii</b>
<b>ABSTRACT .....</b>	<b>viii</b>
<b>DAFTAR ISI .....</b>	<b>ix</b>
<b>DAFTAR TABEL .....</b>	<b>xi</b>
<b>DAFTAR GAMBAR .....</b>	<b>xiii</b>
<b>BAB I PENDAHULUAN.....</b>	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	3
1.3 Batasan Masalah.....	3
1.4 Tujuan Penelitian.....	3
1.5 Landasan Teori .....	4
1.5.1 Kerangka Penelitian Sistem Informasi.....	4
1.5.2 Laboratorium RPL Universitas Hasanuddin.....	4
1.5.3 Aplikasi <i>Mobile</i> .....	6
1.5.4 <i>Framework</i> Flutter.....	7
1.5.5 <i>Aplication Programming Interface</i> (API) .....	8
1.5.6 <i>QR Code</i> .....	9
1.5.7 <i>Black-Box Testing</i> .....	10
1.5.8 <i>User Acceptance Testing</i> (UAT).....	10
1.6 Penelitian Terkait .....	11
<b>BAB II METODE PENELITIAN.....</b>	<b>13</b>
2.1 Waktu dan Lokasi Penelitian .....	13
2.2 Instrumen Penelitian.....	14
2.2.1 Perangkat Keras .....	14
2.2.2 Perangkat Lunak.....	14
2.3 <i>Design Science Research</i> .....	14
2.4 Tahapan Penelitian .....	17
2.5 Metode Pengumpulan Data.....	18
2.5.1 Studi Literatur.....	18
2.5.2 Observasi.....	18
2.6 Metode Pengembangan Sistem.....	18
2.6.1 Analisis Kebutuhan ( <i>Requirements</i> ).....	19
2.6.2 Desain ( <i>Design</i> ) .....	19
2.6.3 Implementasi ( <i>Implementation</i> ) .....	19
2.6.4 Pengujian ( <i>Verification</i> ).....	19
2.6.5 Pemeliharaan ( <i>Maintenance</i> ).....	20
2.7 Perancangan Sistem .....	20

2.8 Perancangan <i>User Interface</i> .....	22
2.8.1 Halaman <i>Onboarding</i> dan <i>Dialog Login</i> .....	22
2.8.2 Halaman Menu Utama .....	23
2.8.3 Halaman Admin.....	24
2.8.4 Halaman Asisten .....	33
2.8.5 Halaman Praktikan.....	40
2.8.6 Halaman Asisten dan Praktikan .....	44
<b>BAB III HASIL DAN PEMBAHASAN .....</b>	<b>48</b>
3.1 Implementasi Sistem .....	48
3.2 Struktur <i>Database</i> .....	48
3.2.1 Skema <i>Database</i> .....	48
3.2.2 Struktur Tabel.....	49
3.3 Implementasi <i>Activity Diagram</i> .....	54
3.3.1 <i>Activity Diagram Login</i> .....	54
3.3.2 <i>Activity Diagram Admin</i> .....	55
3.3.3 <i>Activity Diagram Asisten</i> .....	74
3.3.4 <i>Activity Diagram Asisten dan Praktikan</i> .....	87
3.4 Implementasi <i>User Interface</i> .....	93
3.4.1 Halaman <i>Onboarding</i> dan <i>Dialog Login</i> .....	93
3.4.2 Halaman Menu Utama .....	94
3.4.3 Halaman Admin.....	95
3.4.4 Halaman Asisten .....	103
3.4.5 Halaman Praktikan.....	111
3.4.6 Halaman Asisten dan Praktikan .....	115
3.5 Pengujian Sistem .....	118
3.5.1 Pengujian <i>Black Box</i> .....	118
3.5.2 Pengujian UAT .....	129
<b>BAB IV KESIMPULAN DAN SARAN .....</b>	<b>134</b>
4.1 Kesimpulan .....	134
4.2 Saran .....	134
<b>DAFTAR PUSTAKA.....</b>	<b>135</b>

## DAFTAR TABEL

Nomor Urut	Halaman
Tabel 1. Keunggulan <i>Framework</i> Flutter .....	7
Tabel 2. Kelebihan <i>Black-Box Testing</i> .....	10
Tabel 3. Jadwal Pengembangan Sistem.....	13
Tabel 4. Jadwal Penulisan Tugas Akhir.....	13
Tabel 5. Instrumen Perangkat Keras Penelitian.....	14
Tabel 6. Instrumen Perangkat Lunak Penelitian .....	14
Tabel 7. Elemen-elemen <i>Design Science</i> .....	15
Tabel 8. Tabel <i>User</i> .....	49
Tabel 9. Tabel <i>Profile</i> .....	50
Tabel 10. Tabel <i>Practicum</i> .....	50
Tabel 11. Tabel <i>Classroom</i> .....	51
Tabel 12. Tabel <i>Group</i> .....	51
Tabel 13. Tabel <i>Meeting</i> .....	52
Tabel 14. Tabel <i>Attendance</i> .....	52
Tabel 15. Tabel <i>Score</i> .....	53
Tabel 16. Tabel <i>Assistance</i> .....	53
Tabel 17. Tabel <i>Control Card</i> .....	53
Tabel 18. Tabel <i>Exam Score</i> .....	54
Tabel 19. Pengujian Fitur <i>Login</i> .....	119
Tabel 20. Pengujian Fitur Tambah Pengguna Melalui Formulir .....	119
Tabel 21. Pengujian Fitur Tambah Pengguna Melalui <i>File Excel</i> .....	119
Tabel 22. Pengujian Fitur Lihat Detail Pengguna.....	120
Tabel 23. Pengujian Fitur Edit Pengguna.....	120
Tabel 24. Pengujian Fitur Hapus Pengguna .....	120
Tabel 25. Pengujian Fitur Reset <i>Password</i> Pengguna .....	120
Tabel 26. Pengujian Fitur Tambah Praktikum .....	120
Tabel 27. Pengujian Fitur Lihat Detail Praktikum .....	121
Tabel 28. Pengujian Fitur Edit Praktikum .....	121
Tabel 29. Pengujian Fitur Hapus Praktikum.....	121
Tabel 30. Pengujian Fitur Lihat Detail Kelas .....	121
Tabel 31. Pengujian Fitur Tambah Peserta Kelas .....	122
Tabel 32. Pengujian Fitur Hapus Peserta Kelas .....	122
Tabel 33. Pengujian Fitur Tambah Pertemuan.....	122
Tabel 34. Pengujian Fitur Lihat Detail Pertemuan .....	122
Tabel 35. Pengujian Fitur Edit Pertemuan .....	123
Tabel 36. Pengujian Fitur Hapus Pertemuan .....	123
Tabel 37. Pengujian Fitur Akses Kehadiran Pertemuan .....	123
Tabel 38. Pengujian Fitur Ekspor Kehadiran Pertemuan.....	123
Tabel 39. Pengujian Fitur Akses Rekap Nilai .....	123
Tabel 40. Pengujian Fitur Ekspor Rekap Nilai .....	124
Tabel 41. Pengujian Fitur Tambah Grup Asistensi .....	124

Tabel 42. Pengujian Fitur Lihat Detail Grup Asistensi.....	124
Tabel 43. Pengujian Fitur Edit Grup Asistensi.....	124
Tabel 44. Pengujian Fitur Hapus Grup Asistensi .....	124
Tabel 45. Pengujian Fitur Akses Kartu Kontrol .....	125
Tabel 46. Pengujian Fitur Kelola Tata Tertib Lab.....	125
Tabel 47. Pengujian Fitur Kelola <i>File</i> Modul dan Soal Praktikum .....	125
Tabel 48. Pengujian Fitur Input Kehadiran Melalui <i>Scanner</i> .....	126
Tabel 49. Pengujian Fitur Input Kehadiran Secara Manual .....	126
Tabel 50. Pengujian Fitur Input Nilai Kuis, Respons, dan Ujian Lab .....	126
Tabel 51. Pengujian Fitur Atur Batas Waktu Asistensi .....	126
Tabel 52. Pengujian Fitur Input Nilai Tugas Praktikum .....	127
Tabel 53. Pengujian Fitur Input Kartu Kontrol Praktikan .....	127
Tabel 54. Pengujian Fitur Kelola Info Tambahan .....	127
Tabel 55. Pengujian Fitur Menu Utama.....	128
Tabel 56. Pengujian Fitur Lihat <i>File</i> Pendukung .....	128
Tabel 57. Pengujian Fitur <i>Update</i> Profil .....	128
Tabel 58. Pengujian Fitur Ubah <i>Password</i> .....	129
Tabel 59. Kategori Jawaban UAT .....	129
Tabel 60. Tingkat Keberhasilan UAT .....	129
Tabel 61. Hasil Pengujian UAT <i>User</i> Praktikan.....	130
Tabel 62. Hasil Pengujian UAT <i>User</i> Asisten .....	131
Tabel 63. Hasil Pengujian UAT <i>User</i> Admin.....	132

## DAFTAR GAMBAR

Nomor Urut	Halaman
Gambar 1. Kerangka Penelitian Sistem Informasi .....	4
Gambar 2. Interaksi antara <i>Client</i> dan <i>Server</i> dengan REST API .....	8
Gambar 3. Penerapan <i>QR Code</i> pada Aplikasi <i>Mobile</i> .....	10
Gambar 4. Penerapan <i>Design Science Research</i> .....	15
Gambar 5. Tahapan Penelitian.....	17
Gambar 6. Tahapan Metode <i>Waterfall</i> .....	19
Gambar 7. <i>Use Case Diagram</i> .....	20
Gambar 8. Desain <i>Wireframe</i> Halaman <i>Onboarding</i> dan <i>Dialog Login</i> .....	23
Gambar 9. Desain <i>Wireframe</i> Halaman Menu Utama .....	24
Gambar 10. Desain <i>Wireframe</i> Halaman Data Pengguna.....	25
Gambar 11. Desain <i>Wireframe</i> Halaman Data Praktikum .....	26
Gambar 12. Desain <i>Wireframe</i> Halaman Data Kelas .....	27
Gambar 13. Desain <i>Wireframe</i> Halaman Data Pertemuan .....	28
Gambar 14. Desain <i>Wireframe</i> Halaman Data Absensi .....	29
Gambar 15. Desain <i>Wireframe</i> Halaman Rekap Nilai .....	30
Gambar 16. Desain <i>Wireframe</i> Halaman Data Grup Asistensi.....	31
Gambar 17. Desain <i>Wireframe</i> Halaman Kartu Kontrol.....	32
Gambar 18. Desain <i>Wireframe</i> Halaman Tata Tertib Lab .....	33
Gambar 19. Desain <i>Wireframe</i> Halaman Pertemuan Asisten .....	34
Gambar 20. Desain <i>Wireframe</i> Halaman <i>Scanner</i> dan <i>Dialog Absensi</i> .....	35
Gambar 21. Desain <i>Wireframe</i> Halaman Penilaian .....	36
Gambar 22. Desain <i>Wireframe</i> Halaman Asistensi Asisten .....	37
Gambar 23. Desain <i>Wireframe</i> Halaman <i>Leaderboard</i> Asisten.....	38
Gambar 24. Desain <i>Wireframe</i> Halaman Edit Info Tambahan.....	39
Gambar 25. Desain <i>Wireframe</i> Halaman Data Diri.....	40
Gambar 26. Desain <i>Wireframe</i> Halaman Pertemuan Praktikan .....	41
Gambar 27. Desain <i>Wireframe</i> Halaman Asistensi Praktikan .....	42
Gambar 28. Desain <i>Wireframe</i> Halaman <i>Leaderboard</i> Praktikan .....	43
Gambar 29. Desain <i>Wireframe</i> Halaman Kartu Identitas .....	44
Gambar 30. Desain <i>Wireframe</i> Halaman Info Tambahan .....	45
Gambar 31. Desain <i>Wireframe</i> Halaman Peserta Kelas .....	46
Gambar 32. Desain <i>Wireframe</i> Halaman Edit Profil .....	47
Gambar 33. Skema <i>Database</i> .....	49
Gambar 34. <i>Activity Diagram Login</i> .....	55
Gambar 35. <i>Activity Diagram</i> Lihat Pengguna.....	55
Gambar 36. <i>Activity Diagram</i> Tambah Pengguna Melalui Formulir .....	56
Gambar 37. <i>Activity Diagram</i> Tambah Pengguna Melalui <i>File Excel</i> .....	57
Gambar 38. <i>Activity Diagram</i> Edit Pengguna .....	58
Gambar 39. <i>Activity Diagram</i> Hapus Pengguna .....	59
Gambar 40. <i>Activity Diagram</i> Reset <i>Password</i> Pengguna .....	59
Gambar 41. <i>Activity Diagram</i> Lihat Praktikum .....	60

Gambar 42. <i>Activity Diagram</i> Tambah Praktikum .....	61
Gambar 43. <i>Activity Diagram</i> Edit Praktikum.....	62
Gambar 44. <i>Activity Diagram</i> Hapus Praktikum .....	63
Gambar 45. <i>Activity Diagram</i> Lihat Kelas .....	63
Gambar 46. <i>Activity Diagram</i> Tambah Peserta Kelas.....	64
Gambar 47. <i>Activity Diagram</i> Hapus Peserta Kelas .....	65
Gambar 48. <i>Activity Diagram</i> Lihat Pertemuan .....	65
Gambar 49. <i>Activity Diagram</i> Tambah Pertemuan.....	66
Gambar 50. <i>Activity Diagram</i> Edit Pertemuan .....	67
Gambar 51. <i>Activity Diagram</i> Hapus Pertemuan .....	68
Gambar 52. <i>Activity Diagram</i> Lihat Daftar Kehadiran.....	68
Gambar 53. <i>Activity Diagram</i> Ekspor Data Kehadiran.....	69
Gambar 54. <i>Activity Diagram</i> Lihat Daftar Nilai .....	69
Gambar 55. <i>Activity Diagram</i> Ekspor Data Nilai .....	70
Gambar 56. <i>Activity Diagram</i> Lihat Grup Asistensi .....	70
Gambar 57. <i>Activity Diagram</i> Tambah Grup Asistensi.....	71
Gambar 58. <i>Activity Diagram</i> Edit Grup Asistensi .....	72
Gambar 59. <i>Activity Diagram</i> Hapus Grup Asistensi .....	73
Gambar 60. <i>Activity Diagram</i> Akses Kartu Kontrol .....	73
Gambar 61. <i>Activity Diagram</i> Kelola Tata Tertib Lab .....	74
Gambar 62. <i>Activity Diagram Upload</i> Modul Pertemuan .....	75
Gambar 63. <i>Activity Diagram</i> Hapus Modul Pertemuan .....	76
Gambar 64. <i>Activity Diagram</i> Input Nilai Respons.....	77
Gambar 65. <i>Activity Diagram</i> Input Nilai Kuis .....	78
Gambar 66. <i>Activity Diagram</i> Input Kehadiran Secara Manual .....	79
Gambar 67. <i>Activity Diagram</i> Input Kehadiran Melalui <i>Scanner</i> .....	79
Gambar 68. <i>Activity Diagram Upload</i> Soal Praktikum .....	80
Gambar 69. <i>Activity Diagram</i> Hapus Soal Praktikum .....	81
Gambar 70. <i>Activity Diagram</i> Atur Batas Waktu Asistensi .....	82
Gambar 71. <i>Activity Diagram</i> Input Nilai Tugas Praktikum .....	83
Gambar 72. <i>Activity Diagram</i> Input Kartu Kontrol Praktikan.....	84
Gambar 73. <i>Activity Diagram Update</i> Tautan Kuis.....	85
Gambar 74. <i>Activity Diagram Update</i> Tautan Kuesioner .....	85
Gambar 75. <i>Activity Diagram Update</i> Info Ujian Lab .....	86
Gambar 76. <i>Activity Diagram</i> Input Nilai Ujian Lab.....	87
Gambar 77. <i>Activity Diagram</i> Lihat Profil .....	88
Gambar 78. <i>Activity Diagram</i> Edit Profil.....	88
Gambar 79. <i>Activity Diagram Update Password</i> .....	89
Gambar 80. <i>Activity Diagram</i> Pilih Kelas Mata Kuliah .....	90
Gambar 81. <i>Activity Diagram</i> Lihat Pertemuan Kelas .....	90
Gambar 82. <i>Activity Diagram</i> Lihat Kartu Kontrol .....	91
Gambar 83. <i>Activity Diagram</i> Lihat Rekap Nilai.....	91
Gambar 84. <i>Activity Diagram</i> Buka Tautan Kuis .....	92
Gambar 85. <i>Activity Diagram</i> Buka Tautan Kuesioner.....	92

Gambar 86. <i>Activity Diagram</i> Lihat Info Ujian Lab .....	93
Gambar 87. <i>Activity Diagram</i> Lihat Daftar Peserta Kelas .....	93
Gambar 88. Halaman <i>Onboarding</i> dan <i>Dialog Login</i> .....	94
Gambar 89. Halaman Menu Utama .....	95
Gambar 90. Halaman Data Pengguna .....	96
Gambar 91. Halaman Data Praktikum .....	97
Gambar 92. Halaman Data Kelas .....	97
Gambar 93. Halaman Data Pertemuan .....	98
Gambar 94. Halaman Data Absensi .....	99
Gambar 95. Halaman Rekap Nilai .....	100
Gambar 96. Halaman Data Grup Asistensi .....	101
Gambar 97. Halaman Kartu Kontrol .....	102
Gambar 98. Halaman Tata Tertib Lab .....	103
Gambar 99. Halaman Pertemuan Asisten .....	104
Gambar 100. Halaman <i>Scanner</i> dan <i>Dialog Absensi</i> .....	105
Gambar 101. Halaman Penilaian .....	106
Gambar 102. Halaman Asistensi Asisten .....	107
Gambar 103. Halaman Penilaian Tugas Praktikum .....	108
Gambar 104. Halaman <i>Leaderboard</i> Asisten .....	109
Gambar 105. Halaman Edit Info Tambahan .....	110
Gambar 106. Halaman Data Diri .....	111
Gambar 107. Halaman Pertemuan Praktikan .....	112
Gambar 108. Halaman Asistensi Praktikan .....	113
Gambar 109. Halaman <i>Leaderboard</i> Praktikan .....	114
Gambar 110. Halaman Kartu Identitas .....	115
Gambar 111. Halaman Info Tambahan .....	116
Gambar 112. Halaman Peserta Kelas .....	117
Gambar 113. Halaman Edit Profil .....	118

## BAB I

### PENDAHULUAN

#### 1.1 Latar Belakang

Pendidikan tinggi memiliki peran yang sangat penting dalam membentuk pengetahuan mahasiswa untuk menghadapi tantangan di dunia nyata. Selain mengajarkan konsep dan teori, keterampilan mahasiswa juga perlu dilatih agar mereka mampu menerapkan pengetahuan yang telah dipelajari. Salah satu bentuk penerapan konsep teoritis di lingkungan pendidikan tinggi, khususnya pada universitas, yaitu adanya mata kuliah yang disajikan dalam bentuk praktikum. Praktikum adalah bagian dari perkuliahan yang merupakan kegiatan terstruktur dan terjadwal dengan memberikan kesempatan kepada mahasiswa untuk mendapatkan pengalaman yang nyata agar mampu menguasai keterampilan yang terkait dengan suatu pengetahuan (Akib & Maulana, 2021). Pada umumnya, salah satu syarat yang harus dipenuhi oleh mahasiswa agar dapat meluluskan mata kuliah praktikum adalah dengan melakukan kegiatan asistensi (Athirah, 2017).

Laboratorium Rekayasa Perangkat Lunak (RPL) merupakan salah satu laboratorium yang berada di Universitas Hasanuddin. Laboratorium ini berada pada Program Studi Sistem Informasi, Departemen Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam. Berbagai kegiatan telah dilaksanakan di Laboratorium RPL, seperti perkuliahan, penelitian, praktikum, dan asistensi. Sebagian dari kegiatan tersebut telah menerapkan sistem yang terintegrasi dengan sistem informasi dalam mencatat dan mendokumentasikan setiap kegiatannya. Contohnya, pada kegiatan perkuliahan terdapat Sistem Kelola Pembelajaran (Sikola) dan pada kegiatan penelitian terdapat Sistem Informasi Tugas Akhir (SITA). Namun, untuk kegiatan praktikum dan asistensi masih dikelola secara manual dengan menggunakan komputer tanpa berbasis sistem informasi.

Pengelolaan praktikum dan asistensi di Laboratorium RPL masih dilakukan secara manual tanpa adanya sistem terintegrasi. Beberapa aktivitas praktikum, seperti penyaluran informasi, pendataan peserta, penjadwalan pertemuan, absensi kelas, penginputan nilai, dan penginputan rekapitulasi nilai praktikum masih dikelola menggunakan Excel atau *spreadsheet*. Hal tersebut dapat menimbulkan sejumlah masalah, baik kepada asisten laboratorium maupun peserta praktikum. Masalah yang timbul diantaranya adalah sulitnya mengakses materi, tugas, maupun sumber daya pendukung lainnya dikarenakan setiap dokumen tersebut hanya dapat diakses secara terpisah dan tidak terpusat pada satu tempat. Selain itu, proses pencatatan absensi praktikan seringkali membutuhkan waktu yang cukup lama karena asisten laboratorium harus memanggil nama praktikan satu per satu. Proses penginputan nilai praktikum dan kartu kontrol asistensi yang masih dilakukan menggunakan *software spreadsheet*, terkadang juga dapat menyulitkan asisten laboratorium sehingga kerap menimbulkan kesalahan, kebingungan dalam perhitungan, dan penundaan dalam pemberian umpan balik kepada praktikan.

Oleh karena itu, solusi yang tepat untuk mengatasi pengelolaan praktikum dan asistensi yang masih dilakukan secara manual adalah dengan mengembangkan suatu sistem yang mampu menyimpan, memproses, dan mendistribusikan informasi secara cepat dan akurat (Susanti & Arifin, 2012). Dalam konteks ini, penggunaan aplikasi berbasis *mobile* dapat memberikan solusi yang relevan terhadap permasalahan tersebut. Pengembangan aplikasi *mobile* dalam pendidikan tinggi sangat penting untuk meningkatkan efektivitas proses administrasi dan efisiensi manajemen kampus (Asrori et al., 2020). Selain itu, di era digital ini, penggunaan aplikasi *mobile* menjadi semakin penting karena banyak orang yang menggunakan *smartphone* untuk menjalankan tugas-tugas kompleks dengan mudah serta dapat mengakses informasi kapan saja dan di mana saja.

Beberapa penelitian terdahulu telah mengembangkan sistem dengan memanfaatkan berbagai teknologi dalam mengelola proses terkait praktikum ataupun asistensi. Seperti penelitian yang berjudul “Rancang Bangun Sistem Informasi Laboratorium (SILAB) Berbasis Web di Teknik Informatika UNSOED” yang dilakukan pada tahun 2013, Afuan dan Permadi berhasil mengembangkan sistem informasi laboratorium berbasis web untuk memudahkan mahasiswa Teknik Informatika UNSOED dalam mengelola kegiatan praktikum. Adapun penelitian yang berjudul “Rancang Bangun Sistem Penjadwalan dan Monitoring Pelaksanaan Praktikum dan Asistensi Laboratorium Jurusan Sistem Informasi dan Teknik Informatika UIN Alauddin Makassar” yang dilakukan oleh Athirah pada tahun 2017, dihasilkan sebuah aplikasi berbasis *mobile* yang dapat digunakan oleh mahasiswa untuk memperoleh informasi mengenai jadwal praktikum dan asistensi. Selanjutnya, penelitian yang dilakukan oleh Putri dan Supriyono pada tahun 2019 dengan judul “Rancang Bangun Sistem Presensi Berbasis QR Code Menggunakan *Framework* Codeigniter (Studi Kasus Kehadiran Asisten Praktikum)” dimana sistem yang dibuat dapat membantu proses absensi asisten praktikum di Program Studi Informatika FKI UMS menjadi lebih efektif dan efisien. Kemudian penelitian yang berjudul “Implementasi *Model View Controller* (MVC) dalam Aplikasi Manajemen Penilaian Asisten Laboratorium Berbasis *Mobile*” yang dilakukan pada tahun 2021, Akib dan Maulana berhasil membangun sistem berbasis *mobile* yang dapat mempermudah asisten laboratorium dalam mengelola nilai praktikum serta menyalurkannya kepada mahasiswa. Hingga penelitian yang dilakukan oleh Seanne, Putra, Wicaksono, dan Farida pada tahun 2022 yang berjudul “Perancangan Aplikasi Pengelolaan Tugas Praktikum Laboratorium Rekayasa Perangkat Lunak Berbasis Web di Jurusan Teknik Informatika Institut Teknologi Adhi Tama Surabaya”, diperoleh hasil berupa aplikasi berbasis web yang dapat memudahkan dalam melakukan kegiatan praktikum bagi setiap penggunanya, seperti admin mengelola sistem, mahasiswa melakukan praktikum secara *online*, dan asisten laboratorium melakukan penilaian secara *online*.

Berangkat dari permasalahan yang telah diuraikan, serta dengan mempertimbangkan beberapa penelitian terdahulu, maka penulis ingin mengembangkan sebuah aplikasi yang dapat membantu mengelola kegiatan praktikum dan asistensi melalui skripsi yang berjudul “Rancang Bangun Aplikasi

Pengelolaan Praktikum dan Asistensi Berbasis Mobile (Studi Kasus: Laboratorium RPL Universitas Hasanuddin)". Aplikasi ini bertujuan untuk meningkatkan efisiensi dan kemudahan dalam mengelola data terkait praktikum, seperti data peserta, kelas mata kuliah, jadwal praktikum, asistensi, absensi, penilaian, dan data-data pendukung lainnya. Aplikasi yang dikembangkan juga menawarkan berbagai fitur yang mampu memberikan kemudahan dalam melaksanakan aktivitas praktikum. Beberapa diantaranya termasuk fitur absensi yang memanfaatkan teknologi *QR code*, fitur *leaderboard* yang menampilkan daftar peringkat peserta praktikum, fitur ekspor data kehadiran dan rekapitulasi nilai ke dalam *file* Excel, serta fitur kartu kontrol yang dapat mengefisienkan kegiatan asistensi.

## 1.2 Rumusan Masalah

Dengan mengacu pada latar belakang yang telah diuraikan, maka rumusan masalah yang dapat diambil adalah sebagai berikut:

1. Bagaimana cara merancang dan membangun aplikasi pengelolaan praktikum dan asistensi berbasis *mobile* guna mengatasi permasalahan yang ada di Laboratorium RPL Universitas Hasanuddin?
2. Bagaimana cara menguji dan mengevaluasi aplikasi pengelolaan praktikum dan asistensi berbasis *mobile* yang telah dibangun?

## 1.3 Batasan Masalah

Agar penelitian yang dilakukan lebih fokus dan terarah sehingga memperoleh hasil sesuai yang diharapkan, maka perlu dilakukan pembatasan masalah sebagai berikut:

1. Lokasi penelitian berfokus pada Laboratorium RPL yang berada di Universitas Hasanuddin.
2. Aplikasi yang dibuat tidak terintegrasi dengan sistem informasi akademik universitas maupun program studi yang bersangkutan.
3. Aplikasi dibangun menggunakan *framework* Flutter dan dirancang khusus untuk pengguna perangkat *mobile* seperti *smartphone* dan tablet yang menggunakan sistem operasi Android maupun iOS.
4. Aplikasi pengelolaan praktikum dan asistensi berbasis *mobile* ini hanya berfokus pada pengembangan di sisi *client* atau *front-end*, sehingga implementasi mengenai *database* tidak dibahas secara mendalam.

## 1.4 Tujuan Penelitian

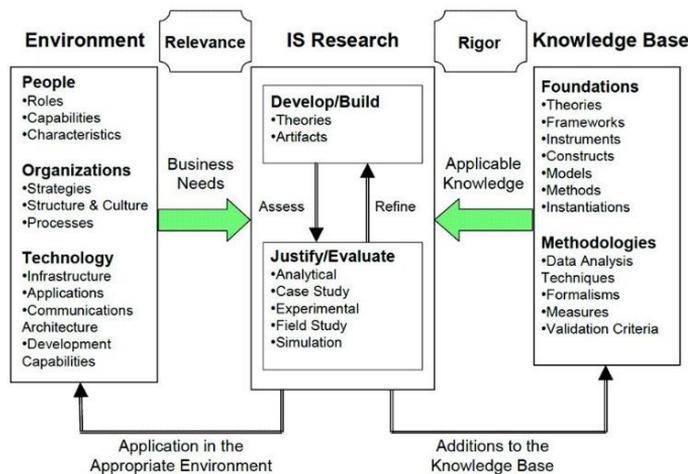
Berdasarkan masalah yang telah dirumuskan, maka tujuan dari penelitian ini adalah sebagai berikut:

1. Untuk merancang dan membangun aplikasi pengelolaan praktikum dan asistensi berbasis *mobile* yang dapat mengatasi permasalahan yang ada di Laboratorium RPL Universitas Hasanuddin.
2. Untuk menguji dan mengevaluasi aplikasi pengelolaan praktikum dan asistensi berbasis *mobile* yang telah dibangun.

## 1.5 Landasan Teori

### 1.5.1 Kerangka Penelitian Sistem Informasi

Sistem Informasi adalah kumpulan elemen atau subsistem yang saling berkaitan dan berhubungan untuk mengelola data sehingga data tersebut menjadi berarti bagi penerima dan bermanfaat untuk pengambilan keputusan di masa sekarang atau di masa mendatang (Nugroho, 2015). Kerangka Penelitian Sistem Informasi merupakan panduan untuk memahami, melaksanakan, dan mengevaluasi penelitian desain *science* dalam bidang sistem informasi (Hevner et al., 2004). Berdasarkan desain *science* penelitian sistem informasi yang dikemukakan oleh Hevner, kerangka penelitian sistem informasi terdiri dari *environment*, *IS research*, dan *knowledge base*.



**Gambar 1.** Kerangka Penelitian Sistem Informasi

Sumber: (Hevner et al., 2004)

*Environment* mendefinisikan ruang lingkup masalah dimana fenomena yang menarik terjadi, yang mencakup *people*, *organizations*, dan *technology*. *IS research* merupakan upaya penelitian yang dilakukan dengan menerapkan *behavioral science* melalui penggunaan berbagai teori yang menjelaskan atau membenarkan masalah bisnis, serta *design science* untuk membangun dan mengevaluasi artefak (perangkat lunak, perangkat keras, atau proses bisnis) yang dirancang guna memenuhi kebutuhan bisnis yang telah diidentifikasi. *Knowledge base* meliputi semua landasan teoritis, termasuk metodologi penelitian maupun teori inti.

### 1.5.2 Laboratorium RPL Universitas Hasanuddin

Laboratorium Rekayasa Perangkat Lunak (RPL) merupakan salah satu laboratorium yang berada di Universitas Hasanuddin. Laboratorium ini berada di bawah naungan Program Studi Sistem Informasi, Departemen Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam. Laboratorium RPL berfokus pada keahlian melakukan pengembangan dan kemampuan mengelola proyek perangkat lunak. Berdasarkan

data yang diperoleh dari halaman web Direktori Laboratorium UNHAS, sebanyak 80% fungsi Laboratorium RPL digunakan untuk kepentingan pengajaran dan 20% sisanya digunakan untuk kepentingan riset.

Praktikum termasuk dalam kegiatan pendukung pembelajaran dan merupakan kegiatan rutin yang wajib diikuti oleh mahasiswa (Gunawan et al., 2023). Pada Program Studi Sistem Informasi Universitas Hasanuddin, terdapat beberapa mata kuliah wajib yang mengharuskan mahasiswanya untuk mengikuti kegiatan praktikum dari mata kuliah tersebut. Beberapa mata kuliah praktikum yang pengelolaannya dilakukan di Laboratorium RPL, antara lain Algoritma dan Pemrograman, Pemrograman Berorientasi Objek, Sistem Basis Data, Pemrograman Web, dan Pemrograman Mobile (Sadno, 2023).

Pelaksanaan praktikum dipimpin oleh koordinator laboratorium, yaitu seorang mahasiswa yang diberi tanggung jawab oleh kepala laboratorium untuk mengatur kegiatan praktikum agar berjalan dengan baik dan lancar. Koordinator laboratorium dibantu oleh para asisten laboratorium, yaitu mahasiswa-mahasiswa yang terpilih melalui proses pendaftaran dan seleksi untuk setiap mata kuliah praktikum. Adapun peserta praktikum atau biasa disebut sebagai praktikan adalah mahasiswa aktif yang mengambil mata kuliah praktikum yang ada pada semester yang sedang berjalan (Tone, 2017).

Proses pelaksanaan praktikum di Laboratorium RPL Universitas Hasanuddin dapat diuraikan menjadi tiga kegiatan yang berbeda, yaitu sebagai berikut.

**Pertemuan kelas.** Pertemuan kelas merupakan kegiatan rutin yang dijadwalkan sekali dalam seminggu untuk memberikan pengajaran dan pemahaman kepada praktikan terhadap materi praktikum. Umumnya, bahan ajar yang diberikan kepada praktikan berupa modul dan *slide* presentasi yang dibuat oleh asisten laboratorium. Pertemuan kelas mencakup aktivitas-aktivitas, seperti absensi peserta, *pre-test*, penjelasan materi, *live coding*, latihan soal, *post-test*, dan pemberian tugas praktikum. Pertemuan kelas biasanya dilaksanakan kurang lebih sebanyak 10 kali atau disesuaikan dengan pembagian materi mingguan yang telah ditetapkan sebelumnya.

**Asistensi tugas praktikum.** Dalam konteks praktikum, asistensi mengacu pada peran mahasiswa yang membantu dalam proses praktikum, baik dengan memberikan bimbingan kepada peserta praktikum maupun dengan membantu dosen dalam proses berjalannya praktikum (Taruna & Susatyo, 2018). Pada Laboratorium RPL, praktikan diharuskan untuk melakukan kegiatan asistensi setiap minggu kepada asisten laboratorium mereka sebagai persyaratan kelulusan mata kuliah praktikum. Fokus utama dari kegiatan asistensi ini adalah membahas mengenai tugas praktikum yang telah diberikan pada saat pertemuan kelas sebelumnya. Dalam konteks ini, asisten laboratorium berperan dalam membimbing dan mengajarkan praktikan mereka, serta menilai tugas praktikum yang telah mereka kerjakan. Untuk memastikan bahwa kegiatan asistensi selalu terpantau oleh asisten laboratorium, maka kehadiran asistensi dicatat dalam kartu kontrol asistensi yang dimiliki oleh setiap praktikan.

**Final praktikum.** Final praktikum adalah tahap akhir yang dilakukan praktikan setelah menyelesaikan seluruh rangkaian pertemuan dan asistensi. Kegiatan ini bertujuan untuk menguji dan menilai pemahaman praktikan terhadap keseluruhan materi yang telah diajarkan. Bentuk pelaksanaan final praktikum dapat bervariasi tergantung pada mata kuliah praktikum ataupun kebijakan yang ditetapkan oleh asisten laboratorium. Penerapan final praktikum biasanya berbentuk ujian lab berbasis *live coding*, proyek individu, ataupun proyek kelompok. Selesaiannya final praktikum menandai berakhirnya seluruh rangkaian kegiatan praktikum yang telah berlangsung selama satu semester.

### 1.5.3 Aplikasi *Mobile*

Aplikasi *mobile* adalah sebuah perangkat lunak atau *software* yang dirancang khusus untuk dijalankan pada perangkat *mobile* seperti *smartphone* dan tablet. Keunggulan utama aplikasi *mobile* terletak pada kemampuannya yang dapat memberikan akses informasi dan layanan secara cepat dan mudah. Dalam pengembangannya, aplikasi *mobile* telah memanfaatkan berbagai fitur yang disediakan pada perangkat *mobile*, seperti kamera, GPS, dan konektivitas jaringan. Dengan mengintegrasikan aplikasi ke fitur-fitur tersebut, maka kita mampu menyelesaikan tugas-tugas kompleks dengan mudah serta dapat mengakses informasi kapan saja dan di mana saja (Lee et al., 2004).

Agar dapat beroperasi, aplikasi *mobile* perlu dijalankan pada perangkat *mobile* yang memiliki sistem operasi yang kompatibel dengan aplikasi tersebut. Saat ini, sistem operasi yang paling populer untuk perangkat *mobile* adalah Android dan iOS. Berikut ini adalah uraian dari kedua sistem operasi tersebut.

**Android.** Sistem operasi Android adalah platform perangkat lunak yang dikembangkan oleh Google. Sistem operasi ini berdasarkan pada kernel Linux yang dirancang untuk perangkat *mobile* seperti *smartphone* dan tablet (Simanjuntak, 2018). Android bersifat *open source*, yang berarti kode sumbernya dapat dimodifikasi dan didistribusikan secara bebas oleh siapa saja. Keunggulan sistem operasi Android terletak pada fleksibilitasnya, yaitu dapat disesuaikan untuk berbagai perangkat dan ukuran layar. Hal tersebut menjadikan Android banyak digunakan oleh berbagai produsen perangkat *mobile* di dunia. Android juga menyediakan fitur seperti antarmuka pengguna yang intuitif serta memiliki platform distribusi aplikasi resmi yang terkenal, yaitu Google Play Store. Berbagai versi Android terus berkembang, mulai dari Android 1.0 hingga Android 14 dengan terus menawarkan berbagai pembaruan dan peningkatan fitur.

**iOS.** Sistem operasi iOS adalah platform perangkat lunak yang dikembangkan oleh Apple Inc. khusus untuk perangkat *mobile* mereka, seperti iPhone, iPad, dan iPod Touch. Berbeda dengan Android, iOS tidak bersifat *open source*, yang berarti kode sumbernya tidak tersedia untuk umum dan hanya Apple yang memiliki kendali penuh terhadap pengembangan, peningkatan, dan distribusi dari sistem operasi ini. Hal tersebut membuat iOS bersifat tertutup dan eksklusif untuk produk-produk Apple saja. Kelebihan utama dari iOS antara lain menawarkan antarmuka pengguna yang

sederhana dan elegan serta memiliki fasilitas kamera yang berkualitas tinggi dengan hasil gambar yang lebih realistis (Setyaningsih, 2023). Selain itu, iOS juga dikenal memiliki tingkat keamanan yang tinggi dan kontrol yang ketat terhadap kualitas aplikasi yang masuk ke App Store (Widia, 2023). Sistem Operasi iOS telah mengalami sejumlah perkembangan, mulai dari iPhone OS 1 hingga iOS 17 dengan fitur dan fungsionalitas yang terus diperbarui.

#### 1.5.4 Framework Flutter

Flutter adalah sebuah kerangka kerja (*framework*) sumber terbuka yang dikembangkan oleh Google untuk membangun aplikasi *multi-platform* dari satu basis kode. Flutter dapat digunakan untuk mengembangkan aplikasi di berbagai platform, termasuk Android, iOS, Windows, Mac, Linux, dan Web. Flutter ditulis dalam bahasa Dart, yaitu bahasa pemrograman yang juga dikembangkan oleh Google yang dioptimalkan untuk aplikasi klien pada platform apa pun.

Agar aplikasi Flutter dapat berjalan pada berbagai platform, kode Dart yang ditulis akan dikompilasi menjadi kode *native*. Pada Android, proses kompilasi dilakukan menggunakan Android NDK (*Native Development Kit*). Sedangkan pada iOS, kode dikompilasi dengan bantuan LLVM (*Low-Level Virtual Machine*). Setelah kode disesuaikan dengan setiap perangkat, barulah aplikasi dapat dijalankan (Uma, 2022).

Flutter sangat populer dan banyak diminati oleh para developer aplikasi. Hal tersebut tentunya tidak lepas dari kelebihan yang terdapat pada *framework* ini. Dalam *website* resminya, disebutkan bahwa terdapat tiga keunggulan utama yang dimiliki oleh Flutter, yaitu sebagai berikut.

**Tabel 1.** Keunggulan *Framework* Flutter

Sumber: (Flutter, 2024)

No	Keunggulan	Deskripsi
1	Cepat	Kode Flutter yang dibuat nantinya dikompilasi menjadi kode mesin ARM atau Intel serta JavaScript, menjadikan kinerja Flutter sangat cepat pada perangkat apa pun.
2	Produktif	Flutter menyediakan fitur <i>hot reload</i> yang memungkinkan kita untuk melihat perubahan tampilan <i>User Interface</i> (UI) dari aplikasi yang dibuat secara <i>real-time</i> tanpa perlu melakukan <i>rebuild</i> .
3	Fleksibel	Flutter memungkinkan kita untuk mengontrol setiap piksel dari tampilan UI yang dibuat agar responsif pada berbagai ukuran layar dan adaptif pada berbagai jenis perangkat.

Flutter dirilis oleh Google pada tahun 2015. Versi pertama Flutter dikenal dengan nama *Sky Engine* yang awalnya hanya diperuntukkan untuk pengembangan aplikasi pada platform Android. Flutter baru tersedia secara publik pada tahun 2017 ketika versi *alpha* (0.0.6) dirilis. Kemudian pada bulan Desember 2018, barulah

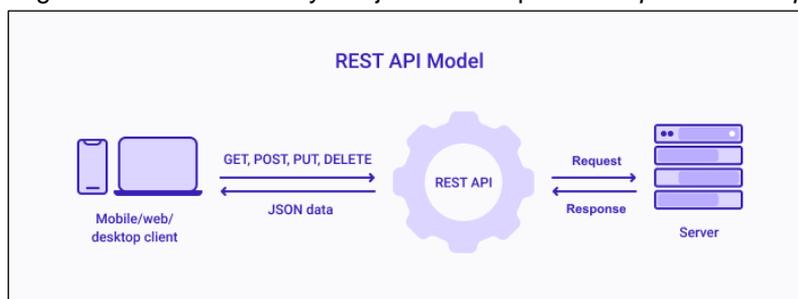
Flutter versi *stable* (1.0) dirilis dan telah mendukung pengembangan aplikasi *multi-platform*. Flutter terus mengalami perkembangan hingga pada bulan Mei 2024, Flutter merilis versi *stable* (3.19.5) dengan menawarkan berbagai pembaruan fitur serta peningkatan performa.

### 1.5.5 *Application Programming Interface* (API)

*Application Programming Interface* (API) adalah suatu cara dimana dua atau lebih program komputer saling berinteraksi satu sama lain. API merupakan antarmuka perangkat lunak yang menawarkan layanan ke perangkat lunak lainnya (Reddy, 2011). API memungkinkan aplikasi yang dibuat dapat mengakses fitur atau layanan dari aplikasi lain dengan memanfaatkan kumpulan perintah, fungsi, maupun data yang tersedia pada aplikasi tersebut. Salah satu bentuk penerapan API adalah *web service* atau biasa juga disebut web API.

*Web service* merupakan API yang berisi sekumpulan basis data (*database*) dan perangkat lunak (*software*) yang dapat diakses melalui jaringan internet. *Web service* memungkinkan aplikasi untuk berkomunikasi dan bertukar data melalui web tanpa memandang sumber *database*, bahasa yang digunakan, dan pada platform apa data tersebut dikonsumsi (Kusuma, 2021). *Web service* biasanya disajikan dalam bentuk REST API, yaitu jenis API yang mengikuti prinsip desain atau gaya arsitektur *Representational State Transfer* (REST). Dalam dunia pemrograman, istilah API dan REST API seringkali disamakan meskipun keduanya memiliki cakupan dan implementasi yang berbeda.

Dalam membuat suatu aplikasi (*web*, *mobile*, maupun *desktop*), terkadang kita perlu membutuhkan data yang berasal dari internet yang sifatnya dinamis. Maka dari itu, API berperan sebagai penghubung antara aplikasi (*client*) dan server (*database*) dalam melakukan proses pertukaran data melalui internet. Interaksi antara klien dan server dengan REST API umumnya terjadi melalui proses *request* dan *response*.



**Gambar 2.** Interaksi antara *Client* dan *Server* dengan REST API

Sumber: (Voximplant, 2020)

**Permintaan (*Request*).** Klien mengirim permintaan (*request*) ke server melalui protokol HTTP. Aksi yang dilakukan klien terhadap sumber daya (*resource*) ditentukan oleh metode HTTP yang digunakan. Adapun metode HTTP yang sering dipakai, antara lain GET untuk mengambil data, POST untuk menambahkan data, PUT untuk memperbarui data, dan DELETE untuk menghapus data. Selain itu, URL

atau *endpoint* dari sumber daya yang diinginkan beserta *header* yang mungkin diperlukan, juga merupakan bagian yang dikirim saat melakukan permintaan ke server.

**Respons (*Response*).** Setelah menerima dan memproses permintaan dari klien, server akan memberikan respons (*response*) yang meliputi informasi status (kode status HTTP), data yang diminta (jika ada), dan *header* yang mungkin berisi informasi tambahan. Respons dari server kemudian dikirim kembali ke klien dalam format data yang umumnya berbentuk JSON. Data dengan format JSON ini perlu diserialisasi terlebih dahulu, kemudian diuraikan ke dalam bentuk model atau objek, sesuai dengan bahasa pemrograman yang digunakan dalam membuat aplikasi klien. Setelah itu, barulah data tersebut dapat digunakan sesuai kebutuhan.

### 1.5.6 QR Code

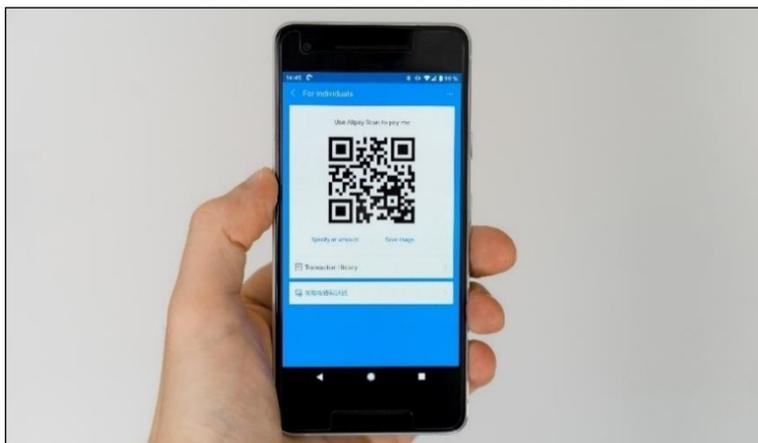
*QR Code (quick-response code)* adalah jenis *barcode* dua dimensi yang diciptakan oleh perusahaan Jepang, Denso Wave, pada tahun 1994 (Hung et al., 2019). *QR code* terdiri dari kotak-kotak hitam yang disusun dalam kisi-kisi persegi dengan latar belakang putih. *QR code* dapat dibaca oleh perangkat pemindaian seperti kamera *smartphone* atau alat pemindai *barcode*. *QR code* biasanya digunakan untuk menyimpan berbagai jenis informasi, seperti teks, URL, nomor telepon, email, lokasi, dan lain-lain.

*QR code* memiliki kemampuan untuk mengoreksi kesalahan, yang berarti informasi yang tersimpan pada *QR code* masih dapat dikembalikan bahkan jika kodenya kotor atau rusak (Irawan & Adriantantri, 2018). Tercatat bahwa *QR code* masih dapat dipindai bahkan jika terjadi kesalahan hingga 30% pada gambar. Karena adanya koreksi kesalahan ini, kita dapat membuat *QR code* yang artistik dengan menambahkan warna, logo, dan fitur lain ke dalam blok *QR code*. Hal ini bertujuan agar *QR code* lebih mudah dibaca dan menarik perhatian pengguna. Meskipun hiasan tersebut dianggap sebagai kesalahan, tetapi kode tetap dapat dipindai dengan benar (Chan, 2011).

*QR code* bekerja dengan cara menggunakan teknologi matriks dua dimensi. Informasi yang disimpan dalam *QR code* direpresentasikan dalam bentuk pola hitam dan putih. Pola tersebut kemudian dipindai oleh kamera *smartphone* atau perangkat lain yang memiliki fitur pemindai *QR code*. Perangkat pemindai kemudian menerjemahkan pola tersebut menjadi informasi yang disimpan dalam *QR code*, lalu menampilkannya pada layar perangkat.

Karena dinilai lebih cepat, praktis, efektif, dan efisien, *QR code* banyak diterapkan pada berbagai aspek kehidupan. Contohnya pada pembayaran elektronik, dimana pelanggan dapat memindai *QR code* pada tagihan atau layar penjual untuk melakukan pembayaran secara digital melalui *e-wallet* atau *mobile banking*. *QR code* juga seringkali ditempatkan pada materi pemasaran atau iklan untuk memberikan akses cepat ke situs web dari produk yang diiklankan. Selain itu, *QR code* dapat diterapkan dalam kegiatan absensi atau kehadiran, dimana

pengguna dapat memindai *QR code* pada platform yang disediakan untuk mengonfirmasi kehadirannya pada kegiatan yang sedang diikuti.



**Gambar 3.** Penerapan *QR Code* pada Aplikasi *Mobile*

### 1.5.7 *Black-Box Testing*

*Black-Box Testing* merupakan metode pengujian yang dilakukan untuk mengamati hasil *input* dan *output* dari suatu aplikasi atau perangkat lunak tanpa perlu mengetahui struktur kode yang ada di baliknya. Pengujian ini umumnya dilakukan pada tahap akhir pembuatan perangkat lunak untuk mengetahui apakah perangkat lunak dapat berfungsi dengan baik (Setiawan, 2021). *Black-box testing* dilakukan berdasarkan spesifikasi aplikasi, seperti tampilan, fungsionalitas, dan kesesuaian antara alur fungsi dengan proses bisnis.

Pada *black-box testing*, tidak diperlukan tim teknis atau sumber daya khusus. Hal terpenting adalah penguji bisa memahami perspektif pengguna. Beberapa kelebihan yang dapat diperoleh saat *black-box testing* diterapkan pada pengujian aplikasi adalah sebagai berikut.

**Tabel 2.** Kelebihan *Black-Box Testing*

Sumber: (Setiawan, 2021)

No	Kelebihan
1	Penguji tidak perlu pengetahuan teknis seputar teknologi atau bahasa pemrograman yang digunakan dalam pembuatan aplikasi.
2	Pengembang dan penguji mampu bekerja sama tanpa harus mengganggu aktivitas pekerjaan utama masing-masing.
3	Pengujian dilakukan sesuai perspektif pengguna dan hasilnya dapat digunakan untuk mengidentifikasi berbagai inkonsistensi pada aplikasi.

### 1.5.8 *User Acceptance Testing (UAT)*

*User Acceptance Testing (UAT)* merupakan proses evaluasi yang dilakukan oleh klien atau pengguna akhir untuk memastikan apakah solusi yang dibuat dalam sistem

telah memenuhi kebutuhan mereka (Suprpto, 2021). UAT tidak berfokus pada pengujian fungsi dan menu yang dibuat oleh *developer*, melainkan berfokus pada kebutuhan bisnis. Pengujian ini menggunakan skenario dan data yang mewakili penggunaan aktual setelah sistem dirilis.

Tujuan utama UAT adalah untuk memastikan bahwa aplikasi telah memenuhi persyaratan pengguna dan dapat digunakan secara efektif dan efisien. UAT juga dapat digunakan untuk mengidentifikasi masalah atau kekurangan dalam aplikasi yang dapat diperbaiki sebelum aplikasi tersebut dirilis ke publik. Evaluasi kegunaan, fungsionalitas, desain sistem, serta memverifikasi apakah aplikasi itu ramah pengguna dan dapat menangani tugas di kondisi yang nyata adalah beberapa fungsi penting dari dijalankannya UAT (Juliandiny, 2023).

Hasil UAT biasanya didokumentasikan dalam laporan penerimaan yang berisi informasi mengenai apakah aplikasi telah layak digunakan secara publik atau belum. Jika UAT berhasil, maka aplikasi dianggap siap untuk produksi dan dapat diluncurkan. Namun jika masih terdapat masalah, maka perlu dilakukan perbaikan sebelum melanjutkan ke tahap produksi.

## 1.6 Penelitian Terkait

Penelitian pertama berjudul “Rancang Bangun Sistem Informasi Laboratorium (SILAB) Berbasis Web di Teknik Informatika UNSOED” yang dilakukan oleh Afuan dan Permadi pada tahun 2013. Pada penelitian ini, dikembangkan sebuah sistem informasi laboratorium berbasis web yang diberi nama SILAB untuk memudahkan mahasiswa Teknik Informatika UNSOED dalam mengelola kegiatan praktikum. SILAB dibuat dengan bahasa pemrograman PHP, basis data MySQL, serta menggunakan metode *waterfall* dalam pengembangannya. SILAB memiliki tiga level pengguna, yaitu mahasiswa, dosen/asisten, dan admin. Fitur-fitur yang terdapat pada SILAB, antara lain pengelolaan data peserta praktikum, pengelolaan jadwal, pengelolaan dosen/asisten, pengelolaan presensi, pengelolaan nilai, serta pengelolaan rekap vakasi kegiatan praktikum.

Penelitian selanjutnya berjudul “Rancang Bangun Sistem Penjadwalan dan Monitoring Pelaksanaan Praktikum dan Asistensi Laboratorium Jurusan Sistem Informasi dan Teknik Informatika UIN Alauddin Makassar” yang dilakukan oleh Athirah pada tahun 2017. Sistem yang dikembangkan bertujuan untuk membantu proses penyampaian informasi jadwal praktikum dan asistensi agar lebih mudah dan efisien. Sistem ini diterapkan dalam sebuah aplikasi *mobile*, khusus untuk platform Android. Metode pengumpulan data yang digunakan adalah observasi dan dokumentasi. Kemudian dalam merancang dan mengembangkan aplikasi, penulis menggunakan metode *waterfall*. Pada tahap akhir pengembangan, aplikasi diuji menggunakan metode *black-box testing*.

Penelitian berikutnya dengan judul “Rancang Bangun Sistem Presensi Berbasis QR Code Menggunakan *Framework* Codeigniter (Studi Kasus Kehadiran Asisten Praktikum)” yang dilakukan oleh Putri dan Supriyono pada tahun 2019. Dalam riset tersebut, peneliti mengembangkan sistem presensi berbasis *QR code* yang bertujuan

untuk memudahkan proses presensi asisten praktikum serta rekapitulasi kehadiran oleh laboran di Program Studi Informatika FKI UMS. Metode yang digunakan oleh penulis dalam merancang sistem tersebut adalah *research and development*. Sistem yang dibangun berbasis website dengan menggunakan *framework* Codeigniter dan *database* MySQL. Sistem dibuat dengan memanfaatkan teknologi *QR code* agar asisten dapat melakukan presensi di sistem yang sama. Sistem yang dihasilkan memiliki fitur *QR code scanner* dari kartu tiap asisten pada *webcam* di *Personal Computer* (PC).

Penelitian berikutnya berjudul “Implementasi *Model View Controller* (MVC) Dalam Aplikasi Manajemen Penilaian Asisten Laboratorium Berbasis *Mobile*” yang dilakukan oleh Akib dan Maulana pada tahun 2021. Penelitian ini menghasilkan sebuah sistem berbasis *mobile* yang dapat mempermudah asisten laboratorium dan dosen dalam mengelola nilai praktikum, mendapatkan absen secara tepat waktu, serta dapat memberikan informasi kepada mahasiswa mengenai nilai yang mereka dapatkan. Tahapan penelitian ini, dimulai dari analisis sistem, kemudian dilanjutkan dengan membuat alur aplikasi berdasarkan konsep MVC, implementasi (pengkodean), dan diakhiri dengan pengujian aplikasi menggunakan metode *black-box* dan kuesioner. Berdasarkan hasil pengujian, aplikasi ini dinilai dapat membantu dari segi pengaksesan nilai praktikum, penyerahan laporan, serta proses pencatatan data kehadiran menjadi lebih cepat setelah diterapkan absensi *online*.

Terakhir, penelitian yang berjudul “Perancangan Aplikasi Pengelolaan Tugas Praktikum Laboratorium Rekayasa Perangkat Lunak Berbasis Web di Jurusan Teknik Informatika Institut Teknologi Adhi Tama Surabaya” yang dilakukan oleh Seanne, Putra, Wicaksono, dan Farida pada tahun 2022. Dalam penelitian ini, penulis mengembangkan sebuah sistem manajemen informasi yang dapat memudahkan praktikan ataupun dosen/asisten lab untuk melakukan pengelolaan data, pemantauan praktikan, dan penyampaian informasi yang cepat dan efisien. Aplikasi ini dibuat dengan bahasa pemrograman PHP melalui bantuan *framework* Laravel dan *database* MySQL. Metode yang digunakan dalam pengembangan aplikasi ini adalah metode *waterfall*. Berdasarkan hasil uji coba dan analisa yang dilakukan, diperoleh kesimpulan bahwa aplikasi yang dikembangkan dapat berjalan dengan baik. Aplikasi ini juga memudahkan dalam melakukan praktikum bagi seluruh pengguna, seperti admin dapat mengelola sistem, mahasiswa dapat melakukan praktikum secara *online*, dan asisten laboratorium dapat melakukan penilaian secara *online*.

## BAB II

### METODE PENELITIAN

#### 2.1 Waktu dan Lokasi Penelitian

Pengembangan sistem pada penelitian ini dilaksanakan mulai dari minggu pertama bulan Februari hingga minggu terakhir bulan Mei 2024. Lokasi penelitian dilakukan di Laboratorium Rekayasa Perangkat Lunak (RPL), Program Studi Sistem Informasi, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Hasanuddin.

**Tabel 3.** Jadwal Pengembangan Sistem

No	Tahapan Penelitian	2024															
		Feb				Mar				Apr				Mei			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	Pengumpulan Data																
2	Analisis Kebutuhan																
3	Desain Sistem																
4	Implementasi Sistem																
5	Pengujian Sistem																
6	Hasil dan Pemeliharaan																

Sementara itu, penulisan tugas akhir skripsi ini dilakukan mulai dari awal bulan Mei hingga akhir bulan Agustus 2024.

**Tabel 4.** Jadwal Penulisan Tugas Akhir

No	Bagian (Bab)	2024															
		Mei				Jun				Jul				Agu			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	Pendahuluan																
2	Metode Penelitian																
3	Hasil dan Pembahasan																
4	Kesimpulan dan Saran																

## 2.2 Instrumen Penelitian

### 2.2.1 Perangkat Keras

Perangkat keras (*hardware*) yang digunakan dalam menunjang penelitian ini adalah sebagai berikut.

**Tabel 5.** Instrumen Perangkat Keras Penelitian

No	Nama Perangkat	Spesifikasi
1	Laptop Asus Vivobook Ultra 15 OLED K513	Prosesor Intel Core i5-1135G7, kartu grafis Intel Iris Xe, RAM 8GB, dan penyimpanan SSD M.2 NVMe PCIe 512GB.
2	<i>Smartphone</i> Vivo Y15	Prosesor Octa-core 2.0GHz, Android versi 11, RAM 4GB, dan penyimpanan internal 64GB.

### 2.2.2 Perangkat Lunak

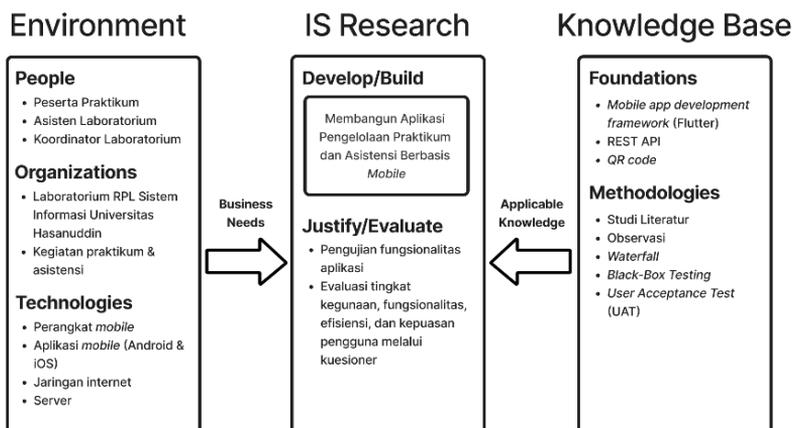
Adapun perangkat lunak (*software*) yang digunakan oleh peneliti adalah sebagai berikut.

**Tabel 6.** Instrumen Perangkat Lunak Penelitian

No	Nama Perangkat	Detail
1	Windows 11 Home (64-bit)	Sistem operasi yang berjalan pada perangkat keras (laptop)
2	Android Emulator (Pixel 6a)	<i>Smartphone</i> virtual untuk membantu proses <i>debugging</i> atau <i>testing</i> aplikasi
3	Figma	<i>Software</i> untuk merancang <i>user interface</i>
4	Visual Studio Code	Teks <i>editor</i> untuk melakukan pengkodean
5	Postman	<i>Software</i> untuk melakukan <i>testing</i> API
6	Google Chrome	<i>Web browser</i>

## 2.3 Design Science Research

*Design Science Research* adalah paradigma atau metodologi yang berfokus pada pengembangan dan evaluasi pengetahuan preskriptif untuk merancang sebuah solusi yang dapat meningkatkan kinerja fungsional (Hevner, 2007). Penerapan *design science* dalam penelitian ini dapat dilihat pada gambar berikut.



**Gambar 4.** Penerapan *Design Science Research*

Berikut adalah uraian dari masing-masing elemen *design science* yang telah diterapkan dalam penelitian ini.

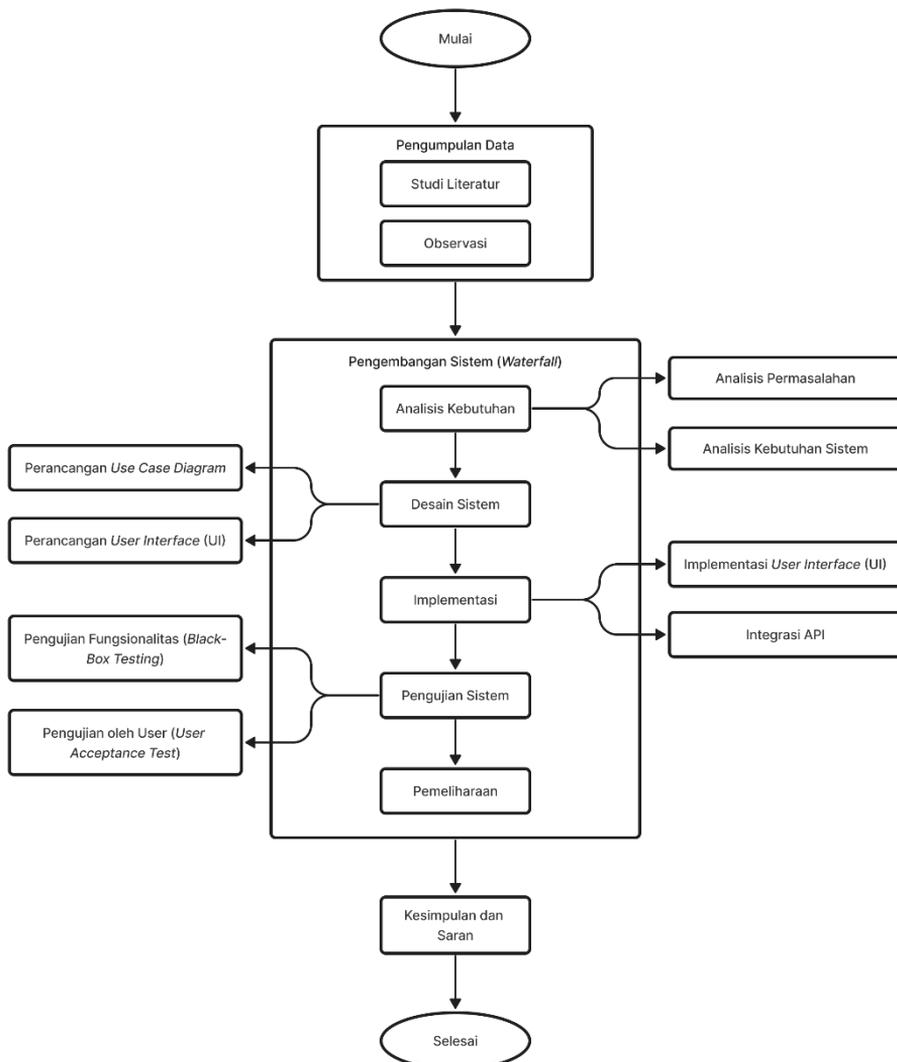
**Tabel 7.** Elemen-elemen *Design Science*

No	Elemen	Unsur	Deskripsi	Penerapan pada Penelitian
1	Lingkungan ( <i>Environment</i> )	Orang ( <i>People</i> )	Individu atau kelompok yang terlibat dalam penelitian.	Peserta praktikum, asisten laboratorium, dan koordinator laboratorium.
		Organisasi ( <i>Organizations</i> )	Konteks dimana penelitian dilakukan, termasuk struktur, budaya, proses, maupun sumber daya yang tersedia.	Laboratorium RPL Sistem Informasi Universitas Hasanuddin. Di Laboratorium RPL terjadi kegiatan praktikum dan asistensi yang rutin dilaksanakan setiap minggu.
		Teknologi ( <i>Technology</i> )	Infrastruktur, perangkat lunak, ataupun perangkat keras yang digunakan dalam penelitian.	Perangkat <i>mobile</i> berupa <i>smartphone</i> atau tablet yang terhubung dengan jaringan internet agar aplikasi dapat terhubung dengan server.

No	Elemen	Unsur	Deskripsi	Penerapan pada Penelitian
2	Penelitian Sistem Informasi ( <i>IS Research</i> )	Membangun ( <i>Build</i> )	Pembuatan artefak, yaitu solusi yang dirancang untuk mengatasi masalah yang telah diidentifikasi.	Aplikasi pengelolaan praktikum dan asistensi berbasis <i>mobile</i> .
		Mengevaluasi ( <i>Evaluate</i> )	Evaluasi artefak untuk mengukur efektivitas dan efisiensinya dalam mencapai tujuan penelitian.	Memberikan kuesioner kepada masing-masing peran pengguna untuk mengukur tingkat kegunaan, efektivitas, efisiensi, dan kepuasan.
3	Basis Pengetahuan ( <i>Knowledge Base</i> )	Fondasi ( <i>Foundations</i> )	Teori, konsep, maupun kerangka kerja yang mendasari penelitian.	<i>Framework</i> pengembangan aplikasi <i>mobile</i> Flutter, REST API, dan <i>QR code</i> .
		Metodologi ( <i>Methodologies</i> )	Metode-metode yang digunakan untuk melakukan penelitian.	Metode pengumpulan data dengan studi literatur dan observasi, metode pengembangan sistem dengan <i>waterfall</i> , serta pengujian atau evaluasi sistem dengan menggunakan metode <i>black-box testing</i> dan <i>User Acceptance Testing</i> (UAT).

## 2.4 Tahapan Penelitian

Agar penelitian dapat berjalan secara terstruktur dan terencana, maka perlu dibuat suatu tahapan atau alur yang harus diikuti sebagai panduan pelaksanaan penelitian. Penelitian ini sendiri memiliki beberapa tahapan yang dimulai dari proses pengumpulan data dengan metode studi literatur dan observasi, kemudian dilanjutkan dengan proses pengembangan sistem menggunakan metode *waterfall*. Berdasarkan metode *waterfall*, tahapan-tahapan yang harus dilakukan dalam mengembangkan sebuah sistem, yaitu dimulai dari analisis kebutuhan, desain sistem, implementasi sistem, pengujian sistem, dan diakhiri dengan pemeliharaan sistem. Untuk lebih jelasnya, tahapan dalam penelitian ini dapat dilihat pada gambar di bawah.



**Gambar 5.** Tahapan Penelitian

## **2.5 Metode Pengumpulan Data**

### **2.5.1 Studi Literatur**

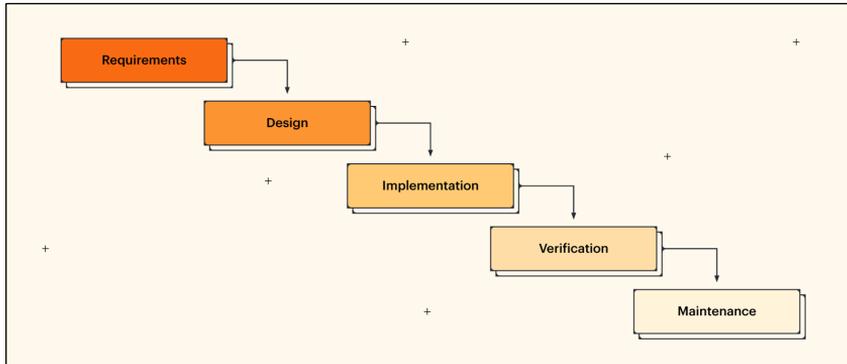
Studi literatur merupakan metode yang digunakan untuk mengumpulkan data dari berbagai sumber tertulis yang relevan dengan suatu topik penelitian (Habsy, 2017). Pada penelitian ini, penulis melakukan studi literatur dengan mengumpulkan berbagai data yang terkait dengan topik pengembangan aplikasi pengelolaan praktikum dan asistensi melalui jurnal dan artikel. Setelah data terkumpul, penulis menganalisis temuan dari literatur-literatur tersebut dengan memperhatikan beberapa aspek, seperti metode penelitian, fitur aplikasi, teknologi yang digunakan, dan hasil penelitian. Hasil analisis tersebut kemudian diintegrasikan ke dalam latar belakang penelitian.

### **2.5.2 Observasi**

Observasi adalah metode pengumpulan data melalui pengamatan langsung terhadap berita atau peristiwa yang ada di lapangan (Prayogo, 2021). Jenis observasi yang dilakukan dalam penelitian ini adalah observasi partisipatif, dimana peneliti terlibat secara aktif dalam situasi yang diamati atau yang digunakan sebagai sumber data penelitian. Berdasarkan studi kasus penelitian ini, penulis telah berperan aktif sebagai asisten laboratorium dalam kegiatan praktikum dan asistensi yang dilaksanakan di Laboratorium RPL Universitas Hasanuddin. Dengan terlibat secara langsung, penulis dapat memperoleh informasi terkait kegiatan praktikum dan asistensi yang dilaksanakan.

## **2.6 Metode Pengembangan Sistem**

Seperti yang telah diuraikan dalam tahapan penelitian, pengembangan sistem pada penelitian ini dilakukan dengan menggunakan metode *waterfall*. Metode *waterfall* adalah salah satu model dari siklus hidup pengembangan sistem yang sering digunakan dalam pengembangan sistem informasi atau perangkat lunak. Model ini menggambarkan pendekatan yang sistematis dan berurutan dimana pelaksanaannya dilakukan secara bertahap. Tahapan selanjutnya hanya dapat dimulai setelah tahapan sebelumnya diselesaikan (Wahid, 2020). Adapun tahapan-tahapan dalam metode *waterfall* dapat dilihat pada gambar berikut.



**Gambar 6.** Tahapan Metode *Waterfall*

Sumber: (Lucidchart, 2024)

### 2.6.1 Analisis Kebutuhan (*Requirements*)

Tahap ini merupakan awal dari pengembangan sistem yang bertujuan untuk memahami kebutuhan pengguna dan sistem yang akan dikembangkan. Pada tahap ini, peneliti mengidentifikasi permasalahan berdasarkan hasil observasi untuk memperoleh informasi terkait sistem yang sedang berjalan sehingga peneliti mampu memahami dan menentukan spesifikasi dari sistem yang ingin dikembangkan.

### 2.6.2 Desain (*Design*)

Setelah kebutuhan teridentifikasi, tahap selanjutnya adalah merancang arsitektur, komponen, dan juga antarmuka pengguna (*user interface*) dari sistem yang akan dibangun. Tahap ini bertujuan untuk memberikan panduan yang jelas mengenai cara mengkonstruksi setiap bagian dari sistem sesuai dengan kebutuhan dan spesifikasi yang telah ditentukan.

### 2.6.3 Implementasi (*Implementation*)

Tahap implementasi merupakan pelaksanaan dari tahap desain. Pada tahap ini, desain dari sistem yang telah dibuat diimplementasikan ke dalam kode program dengan menggunakan *framework* Flutter. Selain itu, proses integrasi data melalui API juga dilakukan pada tahap ini.

### 2.6.4 Pengujian (*Verification*)

Setelah sistem selesai diimplementasikan, tahap pengujian dilakukan untuk memastikan bahwa sistem yang dibangun telah memenuhi kebutuhan pengguna dan berfungsi dengan baik sesuai dengan spesifikasi yang telah ditetapkan sebelumnya. Pengujian sistem pada penelitian ini dilakukan dengan menggunakan metode *black-box testing* dan *User Acceptance Testing* (UAT). *Black-box testing* dilakukan untuk menguji fungsionalitas sistem, sedangkan UAT dilakukan untuk menguji solusi yang telah diterapkan dalam sistem. Pengujian sistem juga dilakukan untuk

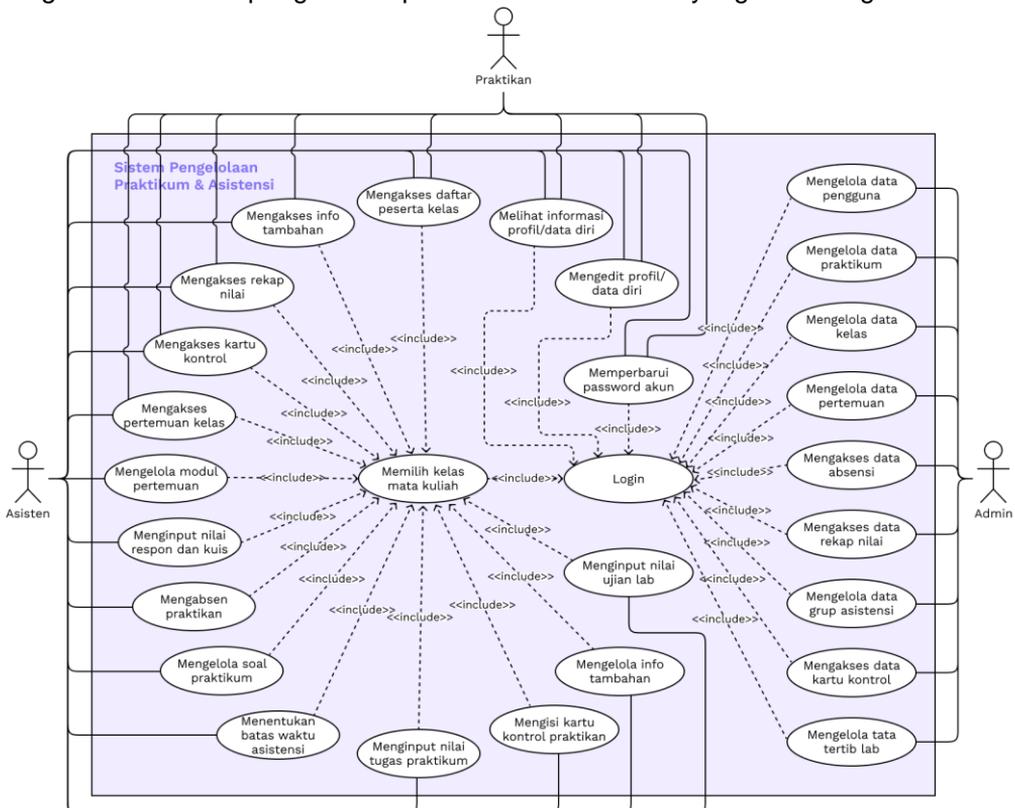
mengidentifikasi dan memperbaiki *bug* atau kesalahan yang mungkin muncul pada aplikasi sebelum dirilis kepada pengguna.

## 2.6.5 Pemeliharaan (*Maintenance*)

Tahap akhir dalam pengembangan sistem adalah pemeliharaan. Tahap ini bertujuan untuk memperbaiki kesalahan dan kekurangan yang ditemukan pada sistem atau aplikasi setelah digunakan oleh pengguna. Tahap pemeliharaan juga dapat mencakup penambahan fitur baru, peningkatan kinerja sistem, serta pembaruan sistem jika diperlukan.

## 2.7 Perancangan Sistem

Perancangan sistem pada penelitian ini divisualisasikan melalui *use case diagram*. *Use case diagram* memberikan gambaran umum mengenai fungsionalitas sistem dari perspektif pengguna (aktor) dan bagaimana aktor berinteraksi dengan *use case* dalam sistem tersebut. *Use case diagram* juga memungkinkan kita untuk melihat aktivitas atau tindakan apa saja yang dapat dilakukan oleh masing-masing peran pengguna pada aplikasi yang dibuat. Gambar di bawah menunjukkan *use case diagram* dari sistem pengelolaan praktikum dan asistensi yang dirancang.



Gambar 7. Use Case Diagram

Dapat dilihat pada gambar di atas, terdapat tiga pengguna atau aktor dengan peran yang berbeda, yaitu Admin, Asisten, dan Praktikan. Ketiga aktor tersebut harus melakukan *login* terlebih dahulu agar dapat mengakses fitur-fitur lain, sesuai dengan peran mereka masing-masing. Berikut ini adalah uraian dari ketiga aktor yang terdapat pada sistem.

**Praktikan.** Praktikan atau peserta praktikum adalah mahasiswa yang mengambil dan melaksanakan kegiatan mata kuliah praktikum. Setelah melakukan *login*, praktikan dapat melihat data diri, memperbarui data diri, dan mengubah *password* akun mereka. Untuk dapat melakukan aksi lain, praktikan harus memilih kelas mata kuliah terlebih dahulu, sesuai dengan pembagian kelas yang ditentukan oleh Admin. Setelah memilih kelas, barulah praktikan dapat melakukan beberapa tindakan lain, seperti

- a. Mengakses pertemuan kelas;
- b. mengakses kartu kontrol asistensi;
- c. mengakses rekap nilai praktikum;
- d. mengakses info tambahan (berupa informasi mengenai kuis, kuesioner, dan ujian lab); dan
- e. mengakses daftar peserta kelas.

**Asisten.** Asisten merupakan mahasiswa yang terpilih melalui proses pendaftaran dan seleksi yang ditugaskan untuk membantu pelaksanaan kegiatan praktikum dan asistensi. Asisten mampu melihat data diri, memperbarui data diri, dan mengubah *password* akun mereka setelah *login* ke dalam sistem. Asisten juga diharuskan untuk memilih kelas mata kuliah tertentu, sesuai dengan kelas yang diajarkan agar dapat melakukan tindakan lebih lanjut. Beberapa aksi yang dapat dilakukan oleh asisten setelah memilih kelas mata kuliah adalah

- a. mengakses pertemuan kelas;
- b. mengelola modul pertemuan;
- c. menginput nilai respons dan nilai kuis;
- d. mengabsen praktikan;
- e. menginput soal praktikum;
- f. menentukan batas waktu asistensi;
- g. menginput nilai tugas praktikum/asistensi;
- h. mengakses dan mengisi kartu kontrol asistensi praktikan;
- i. mengakses seluruh rekap nilai praktikan;
- j. mengakses dan mengelola info tambahan (berupa informasi mengenai kuis, kuesioner, dan ujian lab);
- k. menginput nilai ujian lab; dan
- l. mengakses daftar peserta kelas.

**Admin.** Admin atau administrator adalah pengguna yang memiliki hak untuk mengelola seluruh data terkait kegiatan praktikum dan asistensi. Dalam konteks penelitian ini, koordinator laboratorium adalah satu-satunya orang yang berperan sebagai admin. Berbeda dengan pengguna lain, admin tidak dapat melakukan aksi seperti memperbarui data diri dan mengubah *password* akun sendiri. Fokus utama seorang admin, yaitu

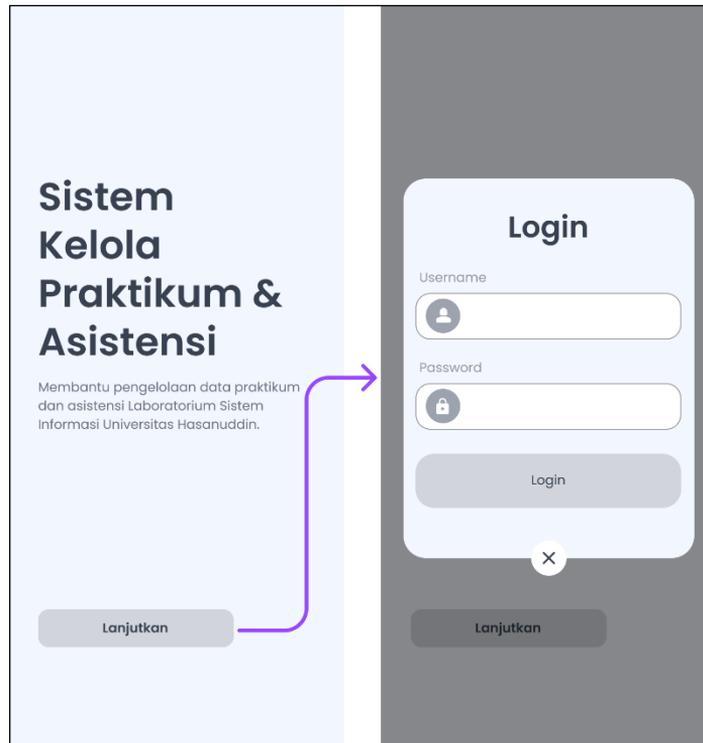
- a. mengelola data pengguna;
- b. mengelola data praktikum;
- c. mengelola data kelas;
- d. mengelola data pertemuan;
- e. mengakses data absensi;
- f. mengakses rekap nilai;
- g. mengelola data grup asistensi;
- h. mengakses kartu kontrol asistensi; dan
- i. mengelola tata tertib lab.

## **2.8 Perancangan *User Interface***

Perancangan *user interface* adalah proses merancang antarmuka pengguna suatu sistem atau aplikasi. *User interface* yang dibuat pada penelitian ini berupa rancangan *wireframe*. *Wireframe* sendiri merupakan gambaran visual sederhana dari struktur dan tata letak halaman yang terdapat pada sebuah aplikasi. *Wireframe* digunakan untuk menggambarkan penempatan elemen seperti tombol, menu, gambar, teks, ataupun navigasi. Hasil dari perancangan *user interface* yang berupa *wireframe* pada penelitian ini adalah sebagai berikut.

### **2.8.1 Halaman *Onboarding* dan *Dialog Login***

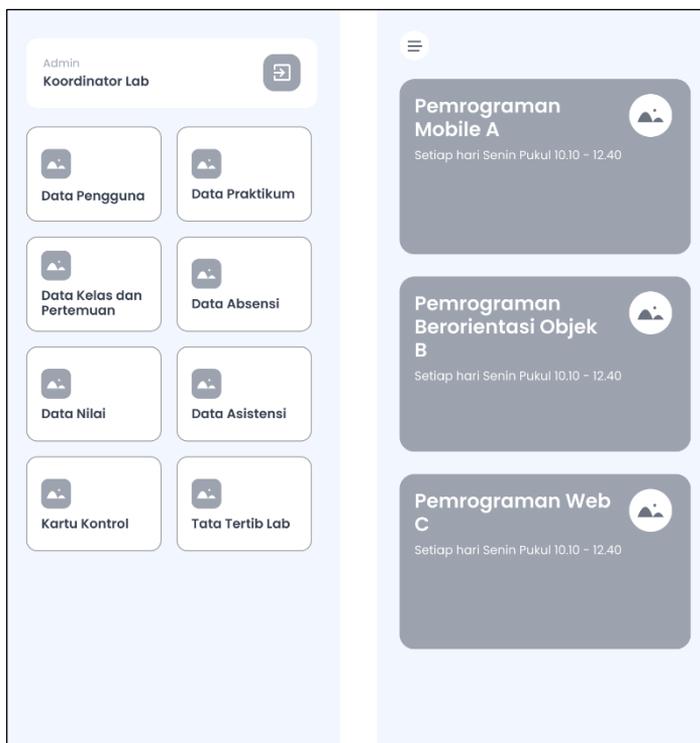
Halaman *onboarding* adalah halaman pengenalan yang ditampilkan saat pertama kali aplikasi dibuka. Pada halaman ini terdapat tombol “Lanjutkan” yang ketika ditekan akan menampilkan jendela atau *pop-up* di atas halaman tersebut. Dalam konteks aplikasi *mobile*, jendela yang muncul ini disebut *dialog*. Saat *dialog* untuk melakukan *login* muncul, pengguna perlu memasukkan *username* dan *password* akun mereka pada formulir untuk dapat masuk ke menu utama aplikasi.



**Gambar 8.** Desain Wireframe Halaman Onboarding dan Dialog Login

### 2.8.2 Halaman Menu Utama

Halaman menu utama merupakan halaman yang pertama kali muncul setelah pengguna berhasil melakukan *login*. Halaman ini memiliki tampilan yang berbeda antara admin dengan praktikan dan asisten. Halaman menu utama pada admin menampilkan beberapa menu untuk mengakses dan mengelola berbagai data, seperti data pengguna, data praktikum, data absensi, dan lain-lain. Sementara itu, halaman menu utama pada praktikan dan asisten tampilannya sama, yaitu menampilkan menu untuk memilih kelas praktikum. Pada gambar di bawah, halaman di sebelah kiri merupakan menu utama admin, sedangkan halaman di sebelah kanan merupakan menu utama praktikan dan asisten.

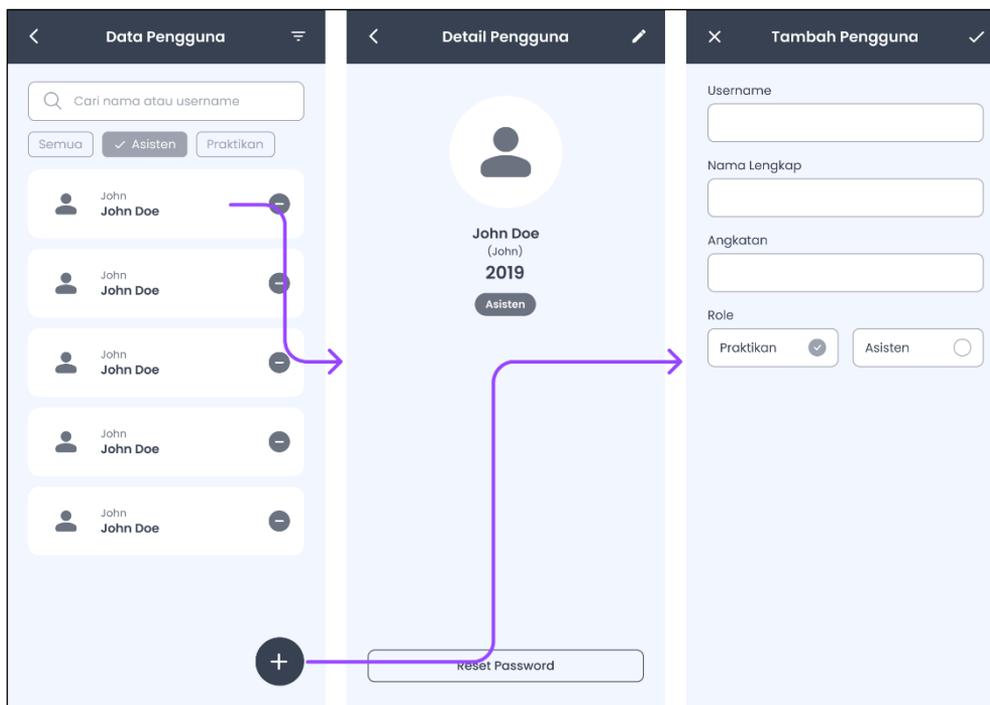


**Gambar 9.** Desain *Wireframe* Halaman Menu Utama

### 2.8.3 Halaman Admin

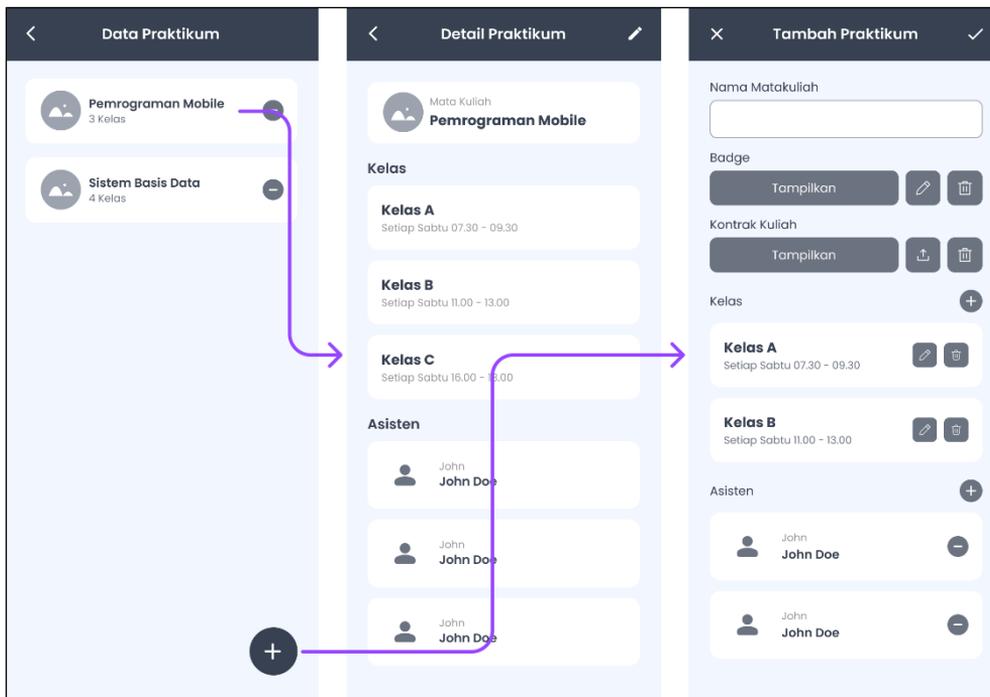
Berikut ini adalah rancangan *user interface* dari beberapa halaman yang hanya dapat diakses oleh peran admin.

**Halaman Data Pengguna.** Halaman data pengguna menampilkan daftar seluruh akun pengguna, baik praktikan maupun asisten. Pada halaman ini, admin mampu melihat, menambah, mengedit, dan menghapus akun pengguna. Untuk mempermudah pencarian data pengguna, terdapat fitur untuk melakukan *searching*, *filtering*, dan juga *sorting*. Selain itu, terdapat juga tombol untuk mereset *password* jika pengguna lupa dengan *password* akun mereka.



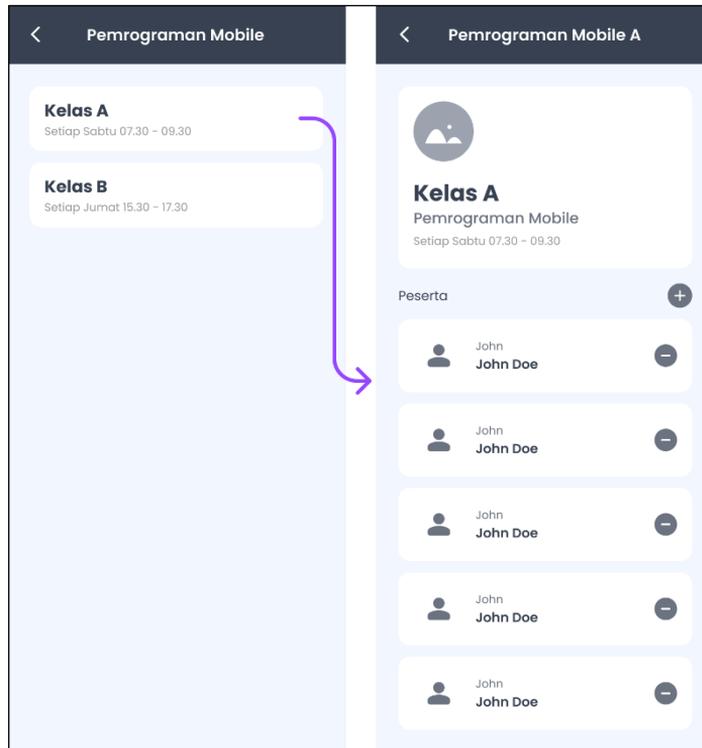
**Gambar 10.** Desain *Wireframe* Halaman Data Pengguna

**Halaman Data Praktikum.** Halaman data praktikum merupakan halaman dimana admin dapat melakukan pengelolaan terhadap data praktikum, diantaranya seperti melihat daftar dan detail praktikum, menambah, mengedit, serta menghapus praktikum. Data praktikum sendiri meliputi nama mata kuliah, *badge*, kontrak kuliah, daftar kelas, dan daftar asisten.



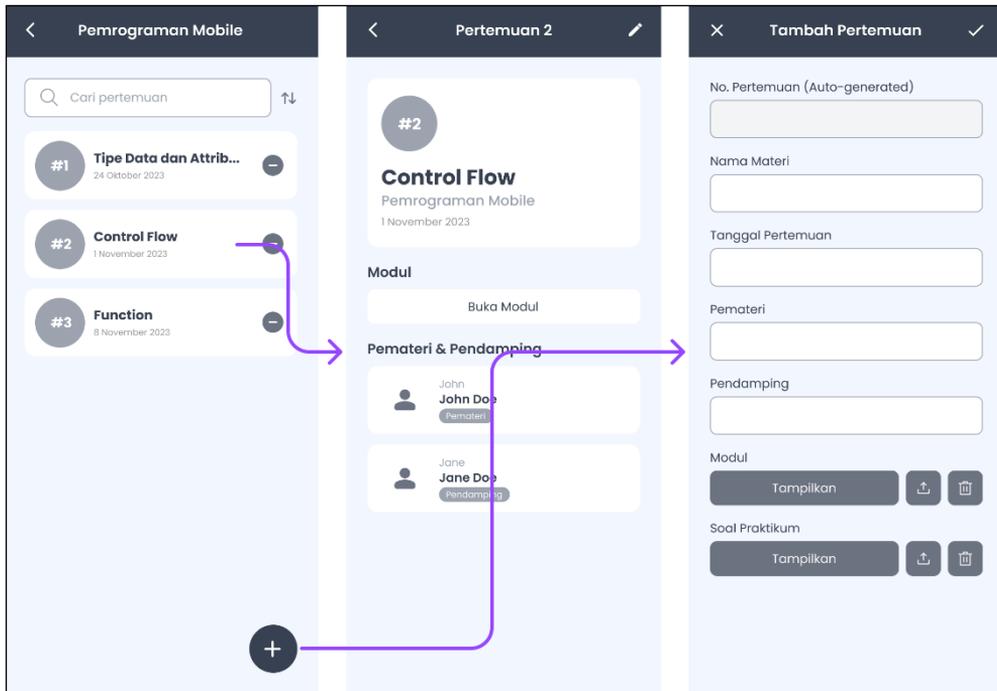
Gambar 11. Desain Wireframe Halaman Data Praktikum

**Halaman Data Kelas.** Di halaman data kelas, terdapat daftar kelas yang tersedia dalam suatu praktikum. Contohnya pada gambar di bawah, terdapat dua kelas yang tersedia pada praktikum pemrograman mobile, yaitu kelas A dan kelas B. Ketika salah satu *item* kelas ditekan, akan ditampilkan seluruh peserta praktikum yang terdaftar pada kelas tersebut. Admin mampu mengelola kelas praktikum dengan menambahkan atau mengeluarkan peserta.



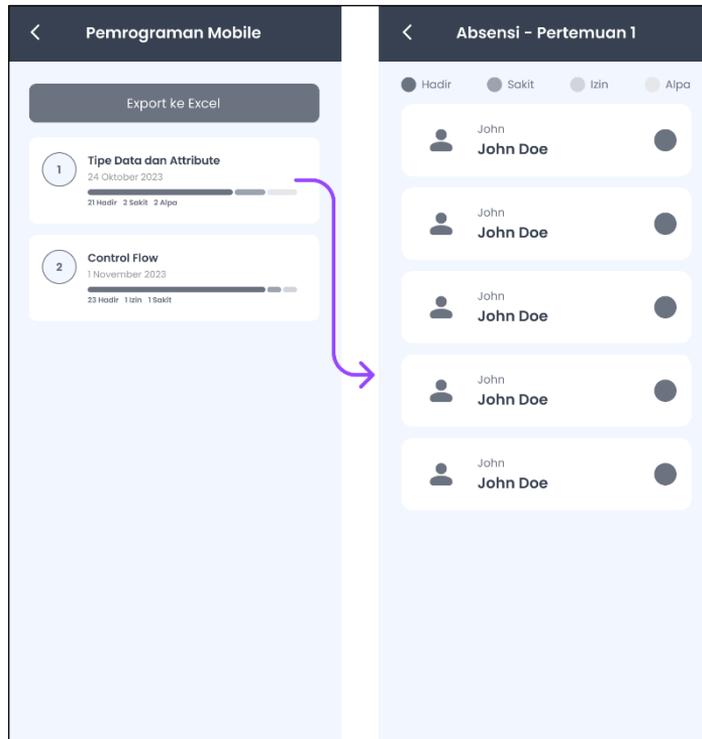
**Gambar 12.** Desain *Wireframe* Halaman Data Kelas

**Halaman Data Pertemuan.** Di halaman data pertemuan, terdapat daftar pertemuan yang tersedia dalam suatu praktikum. Daftar pertemuan ini dilengkapi dengan fitur pencarian dan pengurutan berdasarkan tanggal pertemuan. Admin juga dapat melihat detail, menambah, mengedit, dan menghapus pertemuan melalui halaman ini. Saat ingin menambah atau mengedit pertemuan, admin harus mengisi formulir yang mencakup nomor pertemuan, nama materi, tanggal pertemuan, pemateri, pendamping, modul, dan soal praktikum.



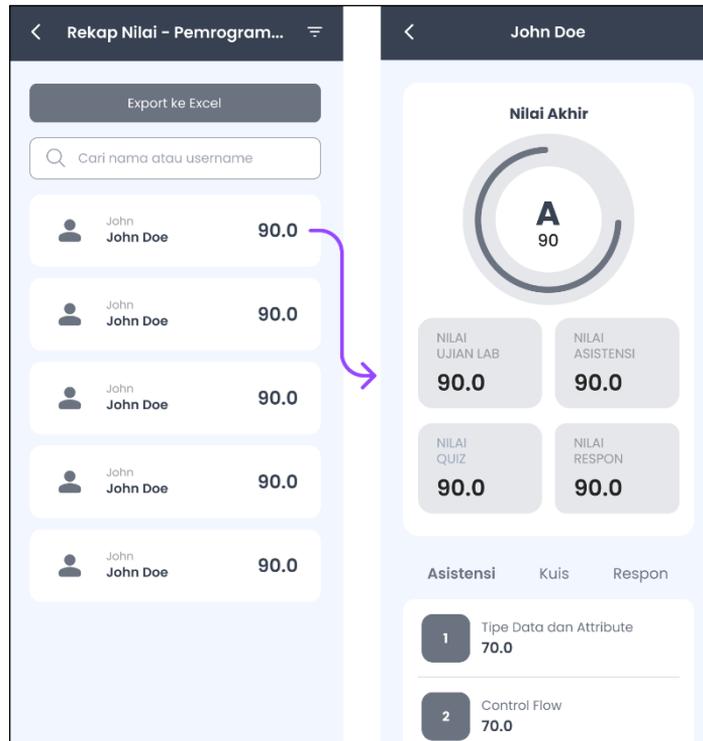
**Gambar 13.** Desain *Wireframe* Halaman Data Pertemuan

**Halaman Data Absensi.** Halaman data absensi menampilkan daftar absensi atau kehadiran peserta praktikum pada setiap pertemuan dalam suatu kelas. Halaman ini terdiri dari daftar pertemuan kelas yang merangkum jumlah praktikan yang hadir, alpa, sakit, dan izin. Ketika salah satu *item* pertemuan ditekan, akan ditampilkan daftar seluruh peserta kelas beserta keterangan absensi mereka pada pertemuan tersebut. Selain itu, terdapat juga tombol untuk mengekspor data absensi ke dalam *file* Excel.



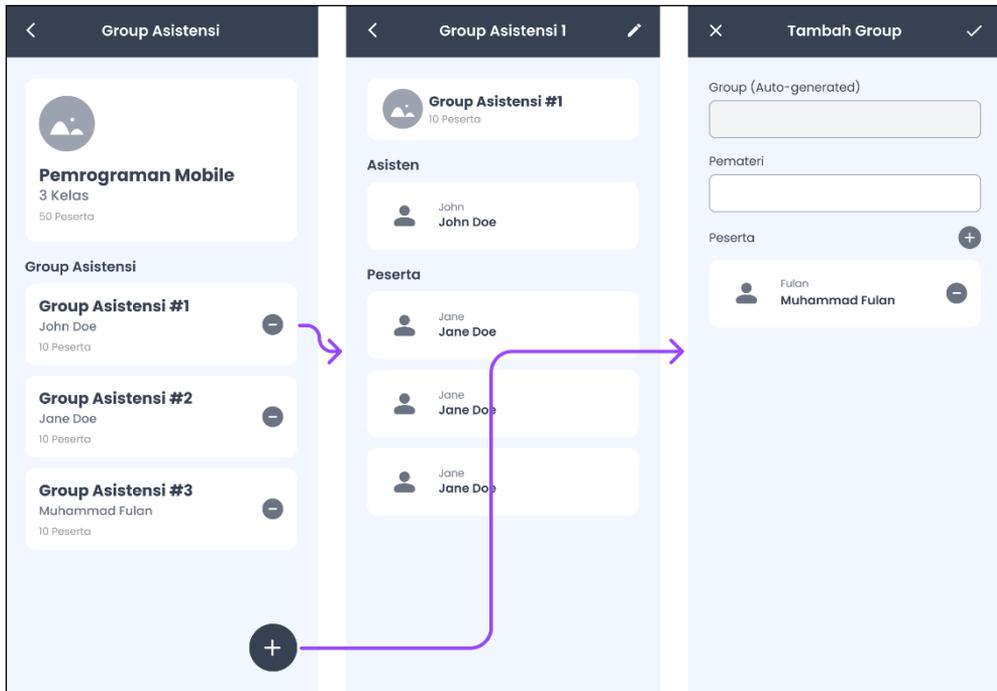
**Gambar 14.** Desain *Wireframe* Halaman Data Absensi

**Halaman Rekap Nilai.** Pada halaman rekap nilai, terdapat daftar rekapitulasi nilai praktikan dalam suatu praktikum. Nilai yang ditampilkan pada setiap *item* merupakan nilai akhir yang diperoleh praktikan. Nilai akhir ini dihitung dari rata-rata nilai ujian lab, nilai asistensi, nilai kuis, dan nilai respons. Tampilan daftar rekap nilai juga diperlengkap dengan adanya fitur pencarian, pengurutan, serta tombol untuk mengekspor rekap nilai praktikum ke dalam *file* Excel.



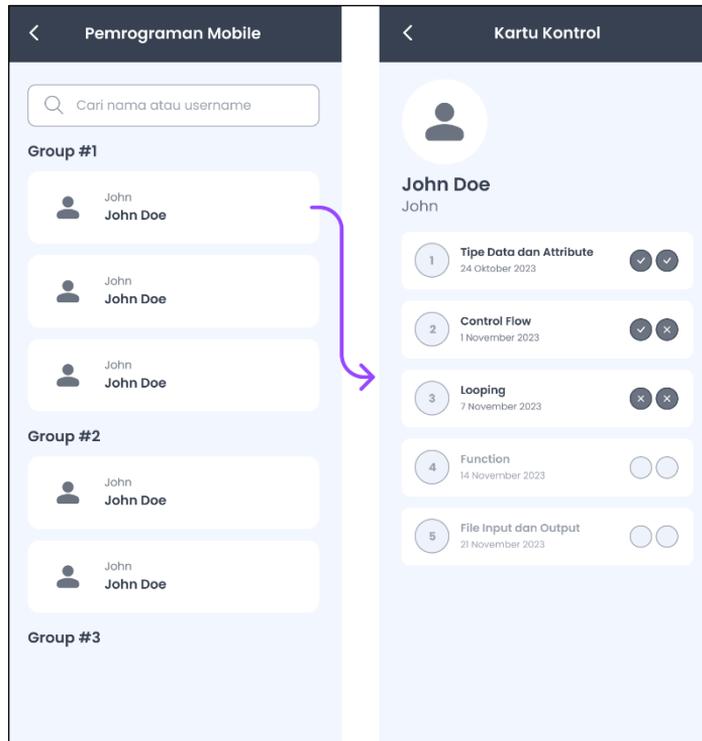
**Gambar 15.** Desain *Wireframe* Halaman Rekap Nilai

**Halaman Data Grup Asistensi.** Pada halaman ini, ditampilkan daftar grup asistensi pada suatu praktikum. Grup asistensi merupakan grup yang terdiri dari sejumlah praktikan dan dipimpin oleh seorang asisten praktikum untuk memberikan bimbingan, jawaban atas pertanyaan, serta menilai tugas praktikum dari para praktikannya. Di halaman ini, admin dapat mengelola data dengan menambah, mengedit, atau menghapus grup asistensi. Tidak hanya itu, admin juga dapat memilih asisten dan menentukan peserta praktikum yang akan dimasukkan pada grup asistensi.



**Gambar 16.** Desain *Wireframe* Halaman Data Grup Asistensi

**Halaman Kartu Kontrol.** Halaman kartu kontrol menampilkan daftar kartu kontrol asistensi yang dimiliki oleh seluruh praktikan dalam suatu praktikum. Kartu kontrol asistensi digunakan untuk mencatat kehadiran dan memberikan umpan balik kepada praktikan saat melakukan kegiatan asistensi. Tampilan daftar kartu kontrol asistensi dikelompokkan berdasarkan grup asistensi masing-masing praktikan. Pada saat salah satu *item* dari daftar tersebut ditekan, akan ditampilkan detail kartu kontrol asistensi praktikan di semua pertemuan.



**Gambar 17.** Desain *Wireframe* Halaman Kartu Kontrol

**Halaman Tata Tertib Lab.** Halaman tata tertib lab adalah halaman dimana admin mengelola aturan dan tata tertib laboratorium. Admin dapat mengunggah *file* tata tertib lab yang nantinya dapat dilihat oleh peserta praktikum. Selain itu, admin juga dapat menetapkan jumlah pengurangan nilai tugas praktikum ketika praktikan terlambat melakukan asistensi, serta menetapkan jumlah pengurangan nilai kuis ketika praktikan terlambat melakukan absensi.

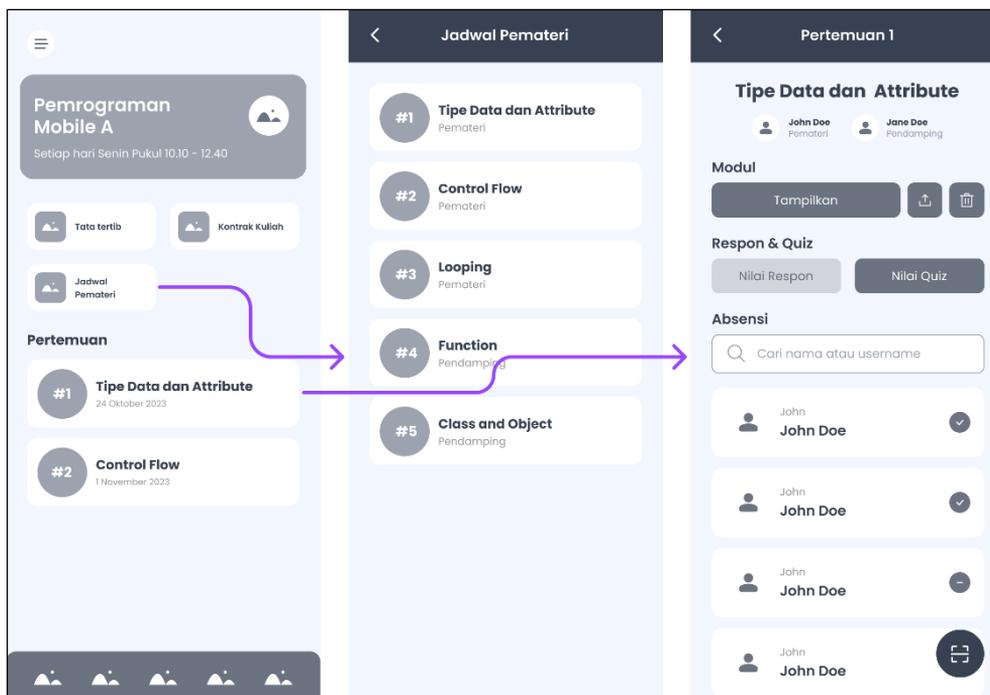


**Gambar 18.** Desain *Wireframe* Halaman Tata Tertib Lab

#### 2.8.4 Halaman Asisten

Rancangan *user interface* untuk halaman-halaman yang dimiliki oleh peran asisten adalah sebagai berikut.

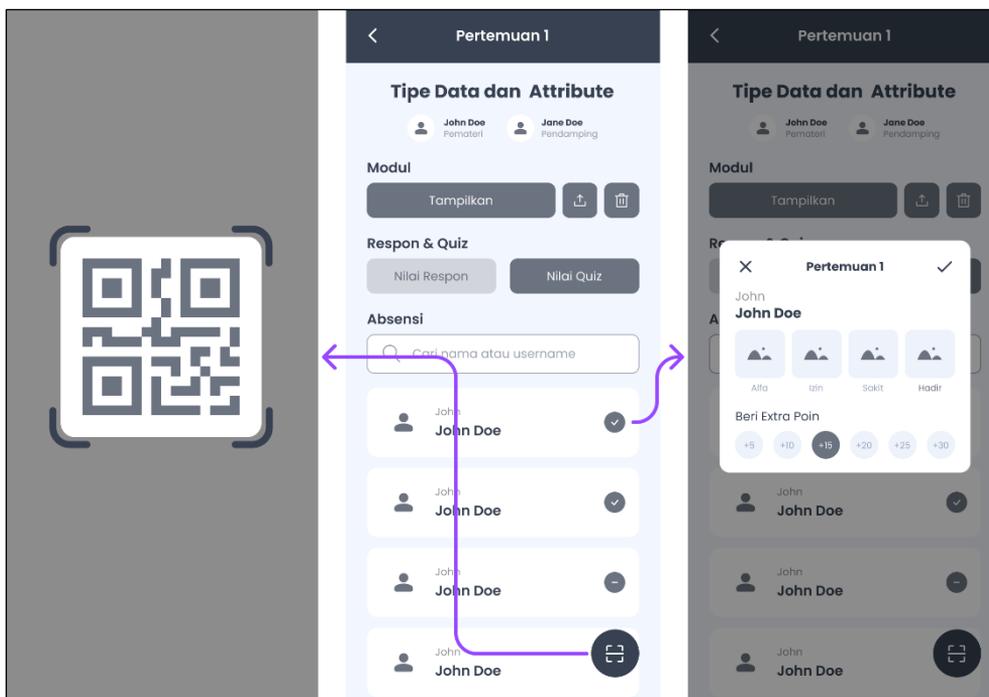
**Halaman Pertemuan.** Halaman pertemuan pada asisten menampilkan daftar pertemuan yang tersedia pada kelas yang dipilih melalui halaman menu utama. Pada halaman ini, asisten dapat melihat jadwal kelas, tata tertib lab, dan kontrak kuliah. Terdapat pula tombol “Jadwal Pemateri” yang berfungsi untuk melihat jadwal pertemuan dimana asisten tersebut bertindak sebagai pemateri atau pendamping. Ketika salah satu *item* pada daftar pertemuan ditekan, akan tampil halaman detail pertemuan yang memuat komponen-komponen, seperti tombol untuk mengunggah modul pertemuan, tombol untuk menuju ke halaman nilai kuis dan respons, serta daftar absensi peserta kelas.



**Gambar 19.** Desain *Wireframe* Halaman Pertemuan Asisten

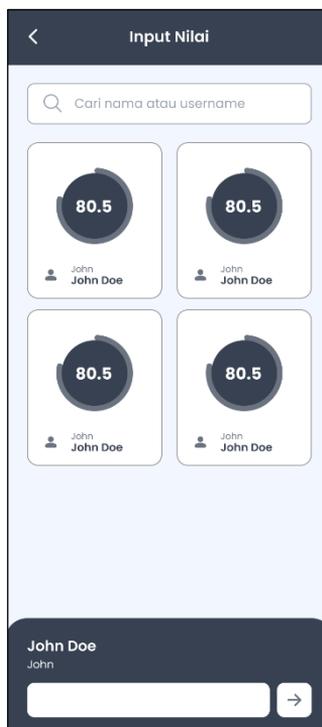
**Halaman *Scanner* dan *Dialog Absensi*.** Halaman *scanner* dan *dialog absensi* merupakan dua halaman berbeda yang digunakan untuk melakukan absensi pada peserta praktikum. Pada gambar di bawah, halaman *scanner* terletak di sebelah kiri, sedangkan *dialog absensi* terletak di sebelah kanan. Pada halaman *scanner*, terdapat fitur kamera yang mampu memindai *ID card* (*QR code*) yang dimiliki praktikan. Ketika *ID card* berhasil dipindai, maka praktikan tersebut dianggap hadir, dan data kehadirannya akan otomatis diperbarui.

Selain menggunakan *scanner*, asisten juga dapat menginput kehadiran praktikan secara manual melalui daftar absensi peserta. Ketika salah satu *item absensi* peserta ditekan, akan muncul *dialog absensi*. Melalui *dialog* tersebut, asisten mampu menentukan status kehadiran praktikan, seperti hadir, sakit, izin, atau alpa. Selain itu, asisten juga dapat memberikan nilai tambahan untuk kuis kepada praktikan dengan rentang nilai 5 hingga 30.



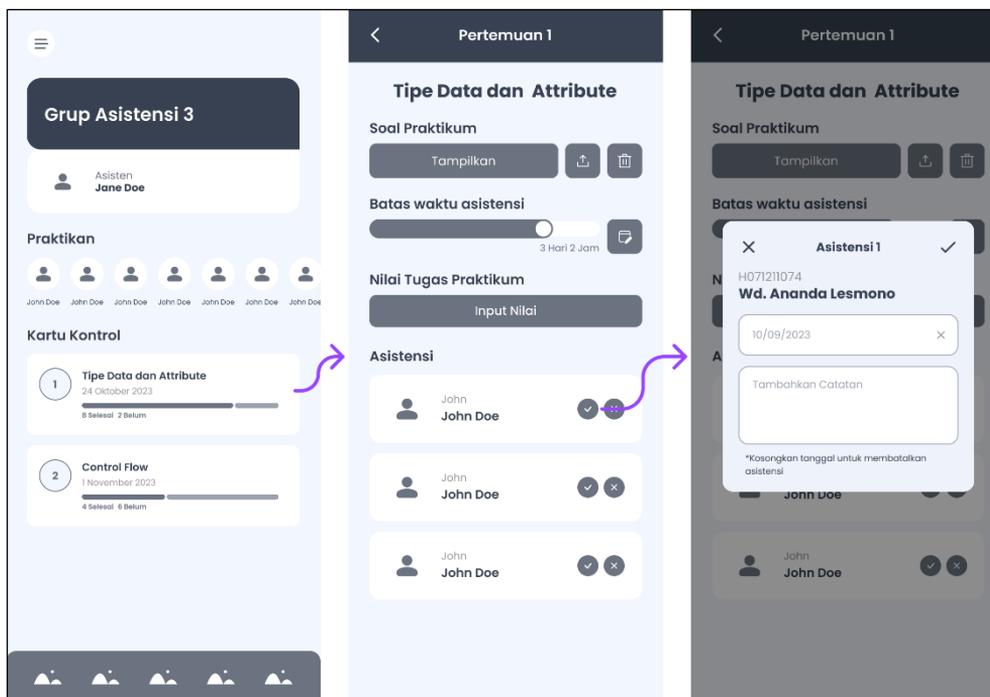
**Gambar 20.** Desain Wireframe Halaman Scanner dan Dialog Absensi

**Halaman Penilaian.** Halaman nilai kuis, respons, asistensi, dan ujian lab dikelompokkan menjadi halaman penilaian karena pada dasarnya keempat halaman tersebut memiliki tampilan yang sama. Seperti yang terlihat pada gambar di bawah, halaman ini menampilkan daftar nilai praktikan yang divisualisasikan dalam bentuk grafik lingkaran. Nilai kuis, respons, dan asistensi merupakan nilai praktikan yang terdapat dalam setiap pertemuan kelas. Sementara itu, nilai ujian lab merupakan nilai yang diperoleh praktikan dari hasil ujian lab pada suatu praktikum. Asisten dapat menginput nilai praktikan pada masing-masing halaman penilaiannya.



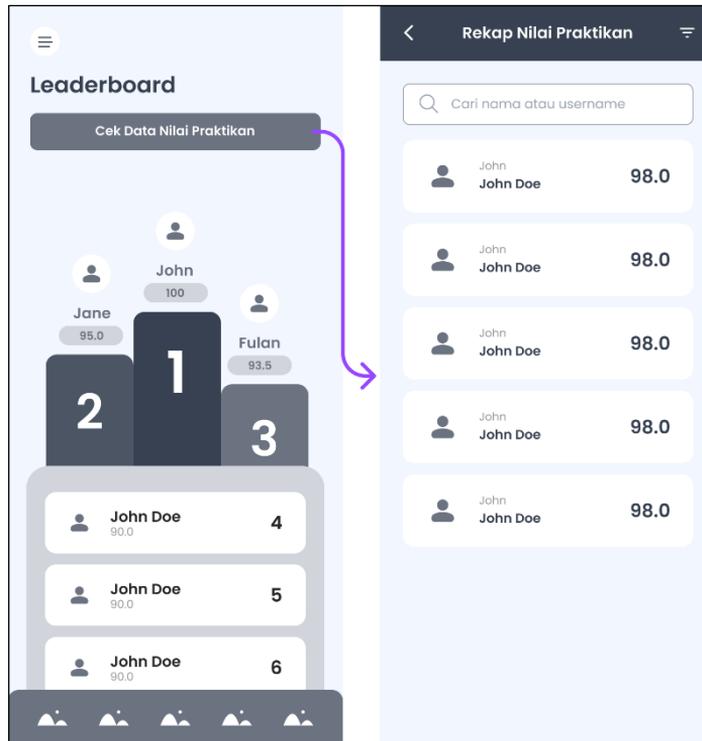
**Gambar 21.** Desain *Wireframe* Halaman Penilaian

**Halaman Asistensi.** Halaman asistensi adalah halaman yang digunakan untuk menunjang kegiatan asistensi antara praktikan dengan asisten. Bagian utama yang ditampilkan pada halaman ini adalah daftar kartu kontrol untuk setiap pertemuan kelas dalam sebuah grup asistensi. Asisten mampu melihat dan mengisi kartu kontrol asistensi untuk setiap pertemuan, serta memberikan umpan balik kepada praktiknya.



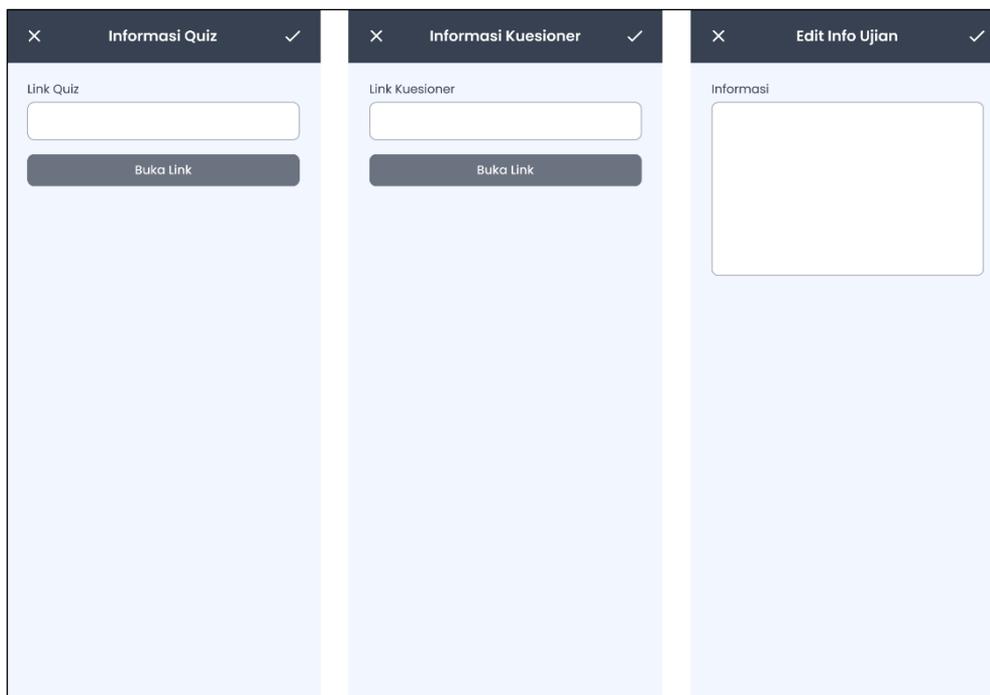
**Gambar 22.** Desain *Wireframe* Halaman Asistensi Asisten

**Halaman *Leaderboard*.** Halaman *leaderboard* adalah halaman yang menampilkan daftar peringkat peserta dalam suatu praktikum. Secara spesifik, halaman ini menampilkan 10 daftar peserta praktikum yang memiliki nilai tertinggi. Dari 10 daftar peserta tersebut, 3 peserta dengan nilai tertinggi akan diberi sorotan, kemudian diikuti oleh 7 peserta lainnya. Tujuan dibuatnya fitur *leaderboard* adalah untuk memberikan pengakuan bagi praktikan yang memiliki hasil kerja yang baik, serta memotivasi praktikan untuk terus berusaha agar memperoleh hasil yang lebih baik. Asisten dapat melihat *leaderboard* dan seluruh daftar rekap nilai praktikum.



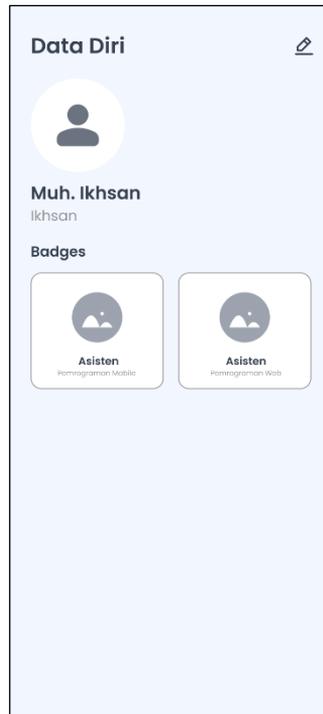
**Gambar 23.** Desain *Wireframe* Halaman *Leaderboard* Asisten

**Halaman Edit Info Tambahan.** Pada halaman edit info tambahan, asisten mampu mengelola beberapa informasi tambahan yang terkait dengan kegiatan praktikum. Aksi yang dapat dilakukan oleh asisten pada halaman ini, yaitu mengedit tautan kuis, tautan kuesioner, dan informasi terkait ujian lab.



**Gambar 24.** Desain *Wireframe* Halaman Edit Info Tambahan

**Halaman Data Diri.** Halaman data diri menampilkan informasi detail dari profil asisten. Data profil yang ditampilkan berupa nama lengkap, nama panggilan, foto profil, *link* profil Github, *link* profil Instagram, dan juga praktikum yang diampu. Di bagian *AppBar* terdapat tombol ikon untuk menuju ke halaman edit profil.

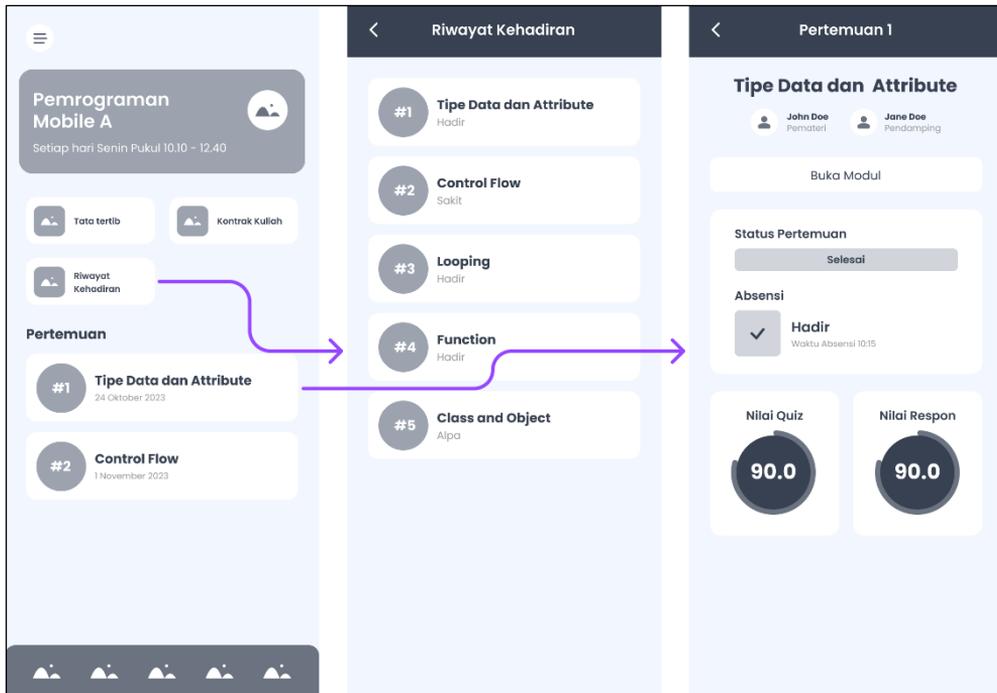


**Gambar 25.** Desain *Wireframe* Halaman Data Diri

### 2.8.5 Halaman Praktikan

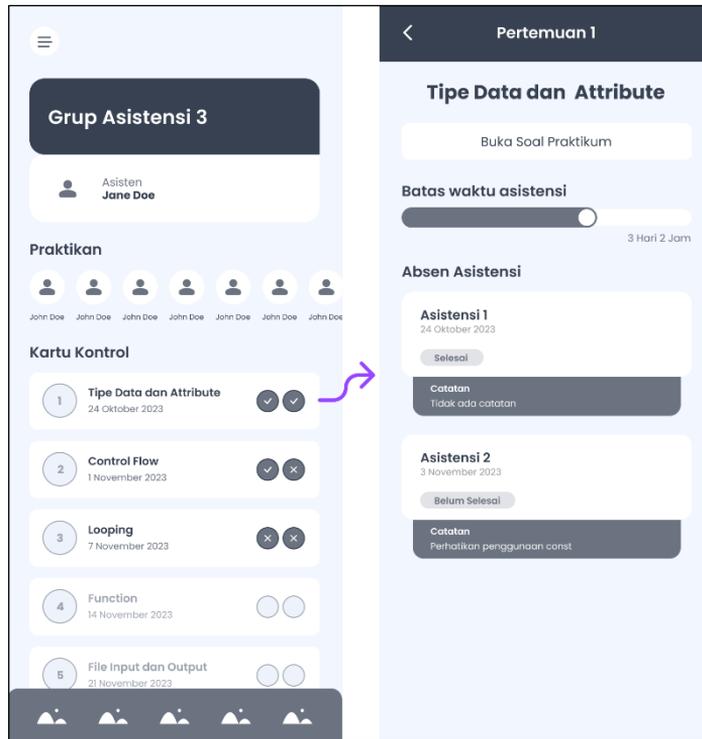
Rancangan *user interface* untuk halaman-halaman yang dimiliki oleh peran praktikan adalah sebagai berikut.

**Halaman Pertemuan.** Secara umum, halaman pertemuan pada praktikan memiliki tampilan yang mirip dengan halaman pertemuan pada asisten. Perbedaannya adalah, jika pada asisten terdapat tombol “Jadwal Pemateri”, maka pada praktikan tombol tersebut digantikan dengan tombol “Riwayat Kehadiran”. Tombol ini berfungsi untuk melihat riwayat kehadiran praktikan pada setiap pertemuan. Selain itu, halaman detail pertemuan pada praktikan juga berbeda dengan halaman detail pertemuan pada asisten. Halaman detail pertemuan pada praktikan memuat beberapa komponen, seperti status pertemuan dan absensi, grafik nilai kuis dan respons, serta tombol untuk melihat modul pertemuan.



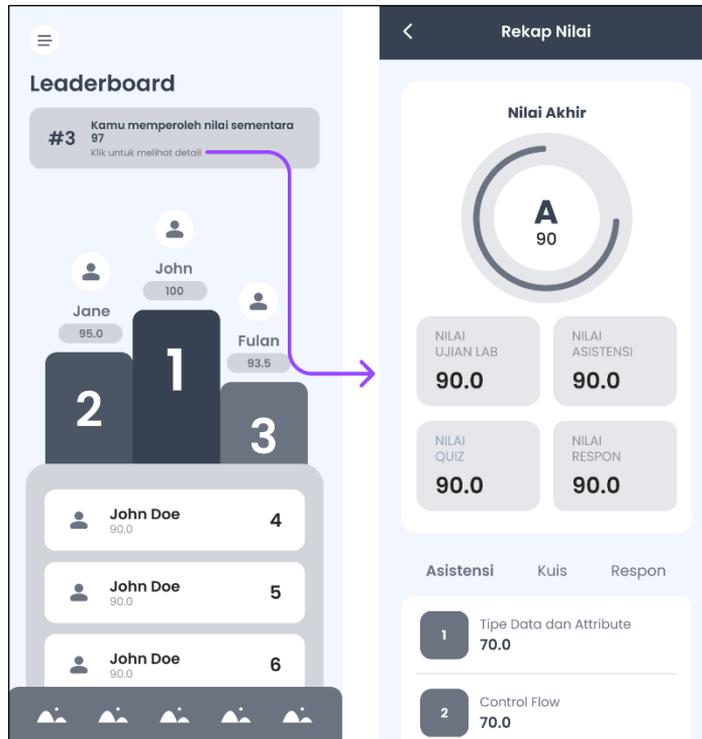
**Gambar 26.** Desain *Wireframe* Halaman Pertemuan Praktikan

**Halaman Asistensi.** Halaman asistensi pada praktikan menampilkan informasi mengenai grup asistensi dan kartu kontrol. Di halaman ini, praktikan dapat melihat asisten dan anggota praktikan lain sesama grup asistensinya. Praktikan juga dapat melihat kartu kontrol mereka beserta umpan balik yang diberikan oleh asisten setelah melakukan kegiatan asistensi. Selain itu, halaman ini memungkinkan praktikan untuk melihat soal praktikum dan batas waktu asistensi untuk setiap pertemuan kelas.



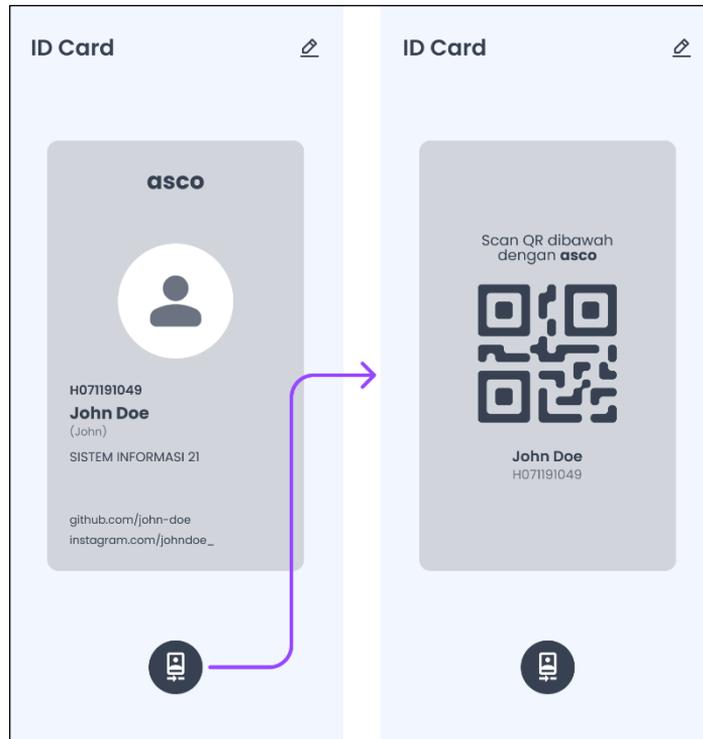
**Gambar 27.** Desain *Wireframe* Halaman Asistensi Praktikan

**Halaman *Leaderboard*.** Halaman *leaderboard* pada praktikan hampir sama dengan halaman *leaderboard* pada asisten. Bedanya, jika pada asisten seluruh daftar rekap nilai praktikan dapat dilihat, lain halnya dengan praktikan yang hanya dapat melihat rekap nilai yang dimiliki. Tidak hanya itu, praktikan juga mampu melihat peringkatnya sendiri pada *leaderboard*, baik jika ia masuk dalam 10 peserta dengan nilai tertinggi maupun tidak.



**Gambar 28.** Desain *Wireframe* Halaman *Leaderboard* Praktikan

**Halaman Kartu Identitas.** Halaman kartu identitas memiliki fungsi yang sama seperti halaman data diri pada asisten, yaitu menampilkan informasi detail dari profil pengguna. Perbedaannya hanya terletak dari segi desainnya saja. Sebagai tambahan, di halaman kartu identitas juga terdapat *QR code* peserta praktikum. *QR code* ini dapat dilihat dengan menekan tombol *flip* yang terletak di bagian bawah kartu. *QR code* inilah yang akan dipindai oleh asisten saat kegiatan absensi kelas berlangsung.

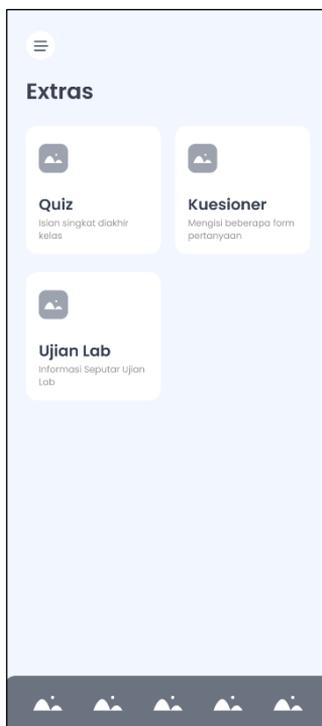


**Gambar 29.** Desain *Wireframe* Halaman Kartu Identitas

### 2.8.6 Halaman Asisten dan Praktikan

Halaman-halaman berikut adalah halaman umum yang dapat diakses oleh peran asisten maupun praktikan.

**Halaman Info Tambahan.** Halaman info tambahan menyediakan beberapa menu untuk mengakses informasi tertentu. Informasi yang tersedia di halaman ini meliputi tautan untuk mengerjakan kuis, tautan untuk mengisi kuesioner, dan informasi terkait pelaksanaan ujian lab. Untuk melihat informasi tersebut, pengguna perlu menekan menu yang relevan. Gambar di bawah menampilkan desain *wireframe* dari halaman info tambahan.



**Gambar 30.** Desain *Wireframe* Halaman Info Tambahan

**Halaman Peserta Kelas.** Halaman peserta kelas merupakan halaman yang menampilkan daftar seluruh asisten dan praktikan dalam suatu kelas. Halaman ini bertujuan agar praktikan dapat mengenal teman sesama kelasnya serta asisten-asisten yang terlibat dalam kelas tersebut. Desain *wireframe* halaman peserta kelas dapat dilihat pada gambar di bawah.