

**RANCANG BANGUN APLIKASI E-LOGBOOK MPPD BERBASIS
MOBILE (STUDI KASUS: FAKULTAS KEDOKTERAN UNIVERSITAS
MUSLIM INDONESIA)**



**MUH.IKHSAN
H071191049**



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

**RANCANG BANGUN APLIKASI E-LOGBOOK MPPD BERBASIS
MOBILE (STUDI KASUS: FAKULTAS KEDOKTERAN UNIVERSITAS
MUSLIM INDONESIA)**

**MUH. IKHSAN
H071191049**



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

**RANCANG BANGUN APLIKASI E-LOGBOOK MPPD BERBASIS
MOBILE (STUDI KASUS: FAKULTAS KEDOKTERAN UNIVERSITAS
MUSLIM INDONESIA)**

**MUH. IKHSAN
H071191049**

Skripsi

Diajukan sebagai salah satu syarat untuk mencapai gelar sarjana

Program Studi Sistem Informasi

pada

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

SKRIPSI
RANCANG BANGUN APLIKASI E-LOGBOOK MPPD BERBASIS MOBILE
(STUDI KASUS: FAKULTAS KEDOKTERAN UNIVERSITAS MUSLIM
INDONESIA)

MUH. IKHSAN
H071191049

Skripsi,

Telah dipertahankan di hadapan Panitia Ujian Sarjana Sistem Informasi pada
21 Juni 2022 dan dinyatakan telah memenuhi syarat kelulusan

pada

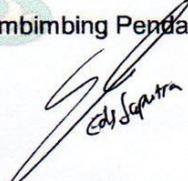
Program Studi Sistem Informasi
Departemen Matematika
Fakultas Matematika dan Ilmu Pengetahuan Alam
Universitas Hasanuddin
Makassar

Mengesahkan:

Pembimbing Tugas Akhir,

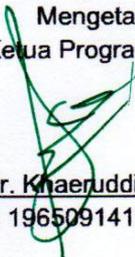
Pembimbing Pendamping,


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


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



Mengetahui:
Ketua Program Studi,


Dr. Khaeruddin, M.Sc.
NIP. 196509141991031003

PERNYATAAN KEASLIAN SKRIPSI DAN PELIMPAHAN HAK CIPTA

Dengan ini saya menyatakan bahwa, skripsi berjudul “Rancang Bangun Aplikasi E-Logbook MPPD Berbasis Mobile (Studi Kasus: Fakultas Kedokteran Universitas Muslim Indonesia)” adalah benar karya saya dengan arahan dari pembimbing (Edy Saputra Rusdi, S.Si, M.Si. sebagai Pembimbing Utama dan Muhammad Sadno, S.Si., M.Si. sebagai Pembimbing Pendamping). 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, 20 Juni 2024



Muh.Ikhsan
H071191049

Ucapan Terima Kasih

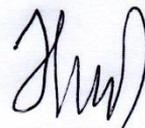
Puji Syukur penulis panjatkan ke hadirat Tuhan yang Maha Esa atas segala limpahan Rahmat dan karunia-Nya, yang telah memberikan kesempatan dan kelancaran bagi penulis dalam penyelesaian tugas akhir yang berjudul "Rancang Bangun Aplikasi E-Logbook MPPD Berbasis Mobile (Studi Kasus: Fakultas Kedokteran Universitas Muslim Indonesia)" ini. Dengan berbagai rintangan yang dihadapi saat menyelesaikan tugas ini, tidak lupa untuk penulis mengucapkan terima kasih atas kontribusi dan bantuannya kepada:

1. Rektor Universitas Hasanuddin, **Bapak Prof. Dr. Ir. Jamaluddin Jompa, M.Sc.** beserta jajarannya.
2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam, **Dr. Eng. Amiruddin** beserta jajarannya.
3. Ketua Departemen Matematika Fakultas Matematika Dan Ilmu Pengetahuan Alam Universitas Hasanuddin, **Bapak Prof. Dr. Nurdin, S.Si., M.Si.**, atas seluruh ilmu dan saran-saran yang telah diberikan.
4. Ketua Program Studi Sistem Informasi, Bapak **Dr. Khaeruddin, M.Sc.** atas seluruh ilmu dan saran-saran yang telah diberikan.
5. Ketua Program Studi Sistem Informasi periode sebelumnya, Bapak **Dr. Hendra, S.Si., M.Kom.** yang telah senantiasa membantu dan memberikan arahan selama masa studi penulis.
6. Pembimbing Utama penulis sekaligus Penasehat Akademik Penulis, Bapak **Edy Saputra Rusdi, S.Si, M.Si** yang telah senantiasa membantu, membimbing, dan memberikan arahan selama masa studi penulis khususnya dalam masa penyusunan skripsi.
7. Pembimbing Pertama penulis **Bapak Muhammad Sadno, S.Si., M.Si.** yang telah senantiasa membantu, membimbing, dan memberikan arahan selama masa studi penulis khususnya dalam masa penyusunan skripsi ini.
8. Kedua dosen penguji, Bapak **Dr. Muhammad Hasbi, M.Sc**, dan Ibu **Rozalina Amran, S.T., M.Eng**, yang telah memberikan kritik dan masukan yang bermanfaat dalam penelitian tugas akhir ini sehingga oleh karenanya skripsi ini dapat tersusun dengan baik.
9. Bapak/Ibu **Dosen Program Studi Sistem Informasi** beserta seluruh tenaga pendidik yang telah memberikan ilmu dan mendidik penulis selama masa perkuliahan. Serta kepada seluruh staf dan pegawai **Departemen Matematika** yang telah membantu penulis terutama dalam segala proses administrasi.
10. Kedua orang tua penulis, Bapak **Asiruddin** dan Ibu **Sutriani** yang tidak pernah lelah mendoakan, memotivasi, dan memberikan dukungan moral dan materi. Begitupula seluruh keluarga besar penulis.
11. Seluruh teman-teman program studi **Sistem Informasi Angkatan 2019** khususnya **Andi Ilhamsyah, Alip Setya, Bayu Ajid, Eurico Devon,**

- Muhamad Fajri, Fatwa Anugrah, Muhammad Takdim, Muammar Ahlan, Rafly Fatur, Rafly Masloman, Richard Enrico, Silverius Sony, Sultan, Taufiq Goe, Theodarryl, dan Yusuf Syam** yang senantiasa memberikan bantuan dan dukungan selama perkuliahan hingga selesainya penulisan skripsi penulis.
12. Teman-teman, serta bapak dan ibu posko **KKNT 110 Pengembangan UMKM Bantaeng Posko 6** yang selalu memberikan dukungan dan motivasi kepada penulis dalam menyelesaikan skripsi penulis.
 13. Teman-teman Kartjis.Id, **Reski Anugrah, Nizar Sabri** yang telah memberikan keleluasaan dan dukungan bagi penulis dalam mengerjakan skripsi sambil membangun usaha bersama.

Penulis menyadari bahwa skripsi ini masih jauh dari sempurna dikarenakan terbatasnya pengalaman dan pengetahuan yang dimiliki 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, 20 Juni 2024



Muh. Ikhsan

ABSTRAK

MUH. IKHSAN. **Rancang Bangun Aplikasi E-Logbook MPPD Berbasis Mobile (Studi Kasus: Fakultas Kedokteran Universitas Muslim Indonesia)** (dibimbing oleh Edy Saputra Rusdi, S.Si, M.Si. dan Muhammad Sadno, S.Si., M.Si.).

Fakultas Kedokteran Universitas Muslim Indonesia (FK UMI) masih menggunakan *logbook* konvensional berbasis kertas untuk mencatat pengalaman klinis Mahasiswa Program Profesi Dokter (MPPD). Hal ini menimbulkan berbagai kendala dalam aksesibilitas, pembaruan, dan distribusi data. Penelitian ini bertujuan merancang dan membangun aplikasi E-Logbook MPPD berbasis *mobile* menggunakan metode *waterfal*. Adapun dalam pengimplementasiannya, penulis menggunakan *framework* Flutter, sehingga aplikasi dapat digunakan di Android dan iOS. Pengujian sistem dengan menggunakan *Black Box Testing* menunjukkan bahwa aplikasi telah memenuhi seluruh kebutuhan fungsional. Selain itu mengetahui kepuasan pengalaman pengguna dilakukan pengujian *User Acceptance Test*, pada user mahasiswa diperoleh skor sebesar 91.93%, pengujian terhadap user supervisor diperoleh skor sebesar 89.04%, pengujian terhadap *user* tim CEU diperoleh skor sebesar 90%, pengujian terhadap *user* kepala bagian 94.64%, dan pengujian terhadap *user* tim kordik 88.64%.

Kata Kunci: *logbook*, MPPD, aplikasi *mobile*, Flutter, *black box testing*, *User Acceptance Test*, *Waterfall*, *pendidikan kedokteran*

ABSTRACT

MUH. IKHSAN. **Design and Development of a Mobile-Based E-Logbook Application for Professional Medical Students (Case Study: Faculty of Medicine, Universitas Muslim Indonesia)** (supervised by Edy Saputra Rusdi, S.Si, M.Si. and Muhammad Sadno, S.Si., M.Si.).

The Faculty of Medicine at Universitas Muslim Indonesia (FK UMI) still uses conventional paper-based logbooks to record the clinical experiences of professional medical students (MPPD). This presents various challenges in terms of accessibility, updates, and data distribution. This study aims to design and develop a mobile-based E-Logbook MPPD application using the waterfall method. For its implementation, the author used the Flutter framework, making the application usable on both Android and iOS platforms. System testing using Black Box Testing showed that the application meets all functional requirements. Additionally, user satisfaction was assessed through a User Acceptance Test, yielding scores of 91.93% from students, 89.04% from supervisors, 90% from the CEU team, 94.64% from department heads, and 88.64% from the coordination team.

Keywords: logbooks, professional medical students, mobile-based application, Flutter, black box testing, User Acceptance Test, Waterfall, medical education.

DAFTAR ISI

	Halaman
HALAMAN JUDUL	ii
PERNYATAAN PENGAJUAN.....	ii
HALAMAN PENGESAHAN.....	iii
PERNYATAAN KEASLIAN SKRIPSI	iv
UCAPAN TERIMA KASIH.....	ii
ABSTRAK	vii
ABSTRACT	viii
DAFTAR ISI	ix
DAFTAR TABEL	xi
DAFTAR GAMBAR	xiii
DAFTAR LAMPIRAN	xvii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah	2
1.4 Tujuan Penelitian	3
1.5 Teori.....	3
1.6 Penelitian Terkait	12
BAB II METODE PENELITIAN.....	16
2.1 Waktu dan Lokasi Penelitian.....	16
2.3 Metode Pengembangan Sistem.....	17
2.4 Instrumen Penelitian	19
2.5 Perancangan Sistem.....	19
2.6 Perancangan <i>User Interface</i>	22
BAB III HASIL DAN PEMBAHASAN	33
3.1 Implementasi Sistem.....	33
3.2 Struktur Database	33
3.3 Implementasi <i>Activity Diagram</i>	39
3.4 Implementasi <i>User Interface</i>	70
3.5 Pengujian Sistem	96

BAB IV KESIMPULAN DAN SARAN	113
4.1 Kesimpulan	113
4.2 Saran	113
DAFTAR PUSTAKA	114
LAMPIRAN	117

DAFTAR TABEL

Nomor Urut	Halaman
1. Daftar Bagian dan Lama Stase di FK UMI	4
2. Daftar dokumen atau data pada <i>logbook</i> FK-UMI	5
3. Kelebihan <i>Black-Box Testing</i>	10
4. Penelitian Terkait.....	14
5. <i>Timeline</i> Penelitian	16
6. Daftar Tabel	35
7. Tabel <i>User</i>	36
8. Tabel <i>Students</i>	36
9. Tabel <i>Daily Activity</i>	37
10. Tabel <i>Activities</i>	38
11. Tabel <i>Clinical Records</i>	38
12. Tabel <i>Weekly Assessments</i>	38
13. <i>Interval</i> dan Predikat untuk Nilai Mahasiswa	62
14. Halaman <i>Login</i>	96
15. Halaman Registrasi Akun <i>Student</i>	97
16. Halaman <i>Forgot Password</i>	97
17. Halaman <i>Input Password</i> Baru.....	97
18. Halaman Verifikasi OTP	97
19. Halaman <i>Notification</i>	98
20. Halaman <i>History</i>	98
21. Halaman <i>Home</i> (Menu Utama) <i>Students</i>	98
22. Halaman Pilih Departemen Aktif	98
23. Halaman Daftar Pengajuan <i>Small Group Learning</i> (SGL).....	99
24. Halaman Tambah Pengajuan SGL Baru	99
25. Halaman <i>Daily Activity</i> Perhari	100
26. Halaman <i>Update Daily Activity</i>	100
27. Halaman Daftar Pengajuan <i>Clinical Record</i>	100
28. Halaman Detail Pengajuan <i>Clinical Record</i>	101
29. Halaman Tambah <i>Clinical Record</i>	101
30. Halaman Tambah Case Mini-CEX	101

31. Halaman Daftar Pengajuan SGL.....	102
32. Halaman Detail SGL yang Diajukan.....	102
33. Halaman Daftar Pengajuan <i>Daily Activity</i>	102
34. Halaman Detail Pengajuan <i>Daily Activity</i>	103
35. Halaman Daftar Pengajuan <i>Clinical Record</i>	103
36. Halaman Detail Pengajuan <i>Daily Activity</i>	103
37. Halaman Penilaian Mini-CEX.....	103
38. Halaman Penilaian <i>Final Score</i>	104
39. Halaman Detail Pengajuan SGL.....	104
40. Halaman Daftar Pengajuan <i>Check-in</i>	105
41. Halaman Daftar Pengajuan <i>Check-out</i>	105
42. Halaman Daftar Nilai Mingguan.....	105
43. Halaman Daftar <i>Daily Activity</i> Mahasiswa.....	106
44. Pilihan Jawaban UAT.....	106
45. Tingkat Keberhasilan UAT.....	107
46. Hasil Pengujian UAT <i>User MPPD</i>	107
47. Hasil Pengujian UAT <i>User Supervisor</i>	108
48. Hasil Pengujian UAT <i>User Tim CEU</i>	110
49. Hasil Pengujian UAT <i>User Kepala Bagian</i>	110
50. Hasil Pengujian UAT <i>User Tim Kordik</i>	111

DAFTAR GAMBAR

Nomor Urut	Halaman
1. Contoh Logbook FK UMI.....	4
2. Komponen Sistem LBS	9
3. Alur Penelitian	17
4. Metode <i>Waterfall</i>	17
5. <i>Use Case Diagram</i> Role Mahasiswa.....	20
6. <i>Use Case Diagram</i> Role Supervisor.....	21
7. <i>Use Case Diagram</i> Role Tim Kordik.....	22
8. Desain <i>Wireframe</i> Halaman <i>Login dan Register</i>	23
9. Desain <i>Wireframe</i> Halaman Menu Utama	24
10. Desain <i>Wireframe</i> Halaman Menu Daftar <i>Task</i> atau <i>Submission</i>	25
11. Desain <i>Wireframe</i> Halaman <i>Form Submission</i> MPPD	26
12. Desain <i>Wireframe</i> Halaman Detail <i>Submission</i>	27
13. Desain <i>Wireframe</i> Halaman Detail SGL	28
14. Desain <i>Wireframe</i> Halaman Detail <i>Daily Activity</i>	28
15. Desain <i>Wireframe</i> Halaman Pilih Lokasi	29
16. Desain <i>Wireframe</i> Halaman <i>Assessment</i>	30
17. Desain <i>Wireframe</i> Halaman Verifikasi <i>Check-In</i> dan <i>Check-Out</i>	31
18. Desain <i>Wireframe</i> Halaman Riwayat.....	32
19. <i>Entity Relation Diagram</i> Database E-Logbook	34
20. <i>Activity Diagram</i> <i>Login</i>	39
21. <i>Activity Diagram</i> <i>Logout</i>	40
22. <i>Activity Diagram</i> <i>Register</i> Akun Mahasiswa	41
23. <i>Activity Diagram</i> <i>Forgot Password</i>	42
24. <i>Activity Diagram</i> <i>Kelola Profil</i>	43
25. <i>Activity Diagram</i> <i>Pilih Departemen Aktif</i>	44
26. <i>Activity Diagram</i> <i>Mengajukan Check-In</i> Departemen	44
27. <i>Activity Diagram</i> <i>Mengajukan Check-Out</i> Departemen	45
28. <i>Activity Diagram</i> <i>Menampilkan Daftar Pengajuan SGL</i>	45

29. <i>Activity Diagram</i> Menambahkan SGL.....	46
30. <i>Activity Diagram</i> Menyunting SGL.....	47
31. <i>Activity Diagram</i> Menghapus SGL.....	48
32. <i>Activity Diagram</i> Menambahkan Topik ke SGL Tertentu.....	48
33. <i>Activity Diagram</i> Menampilkan <i>Daily Activity</i> Perminggu.....	49
34. <i>Activity Diagram</i> Memperbaharui Status <i>Daily Activity</i>	50
35. <i>Activity Diagram</i> Menampilkan Detail <i>Clinical Record</i>	51
36. <i>Activity Diagram</i> Menambah <i>Clinical Record</i>	51
37. <i>Activity Diagram</i> Memperbaharui <i>Clinical Record</i>	52
38. <i>Activity Diagram</i> Menghapus <i>Clinical Record</i>	53
39. <i>Activity Diagram</i> Menambah Kasus.....	54
40. <i>Activity Diagram</i> Menampilkan Daftar Kasus	55
41. <i>Activity Diagram</i> Menghapus Kasus.....	55
42. <i>Activity Diagram</i> Membuka Referensi	56
43. <i>Activity Diagram</i> Menambahkan Data Mini-CEX.....	57
44. <i>Activity Diagram</i> Mengecek Data Nilai	58
45. <i>Activity Diagram</i> Menampilkan Daftar Pengajuan SGL	58
46. <i>Activity Diagram</i> Verifikasi Topik SGL.....	59
47. <i>Activity Diagram</i> Verifikasi <i>Daily Activity</i>	60
48. <i>Activity Diagram</i> Verifikasi <i>Clinical Record</i>	60
49. <i>Activity Diagram</i> Verifikasi Pengajuan Kasus.....	61
50. <i>Activity Diagram</i> Melakukan Penilaian Mini-CEX	62
51. <i>Activity Diagram</i> Melakukan Penilaian <i>Personal Behavior</i>	63
52. <i>Activity Diagram</i> Menampilkan Detail Mahasiswa Bimbingan	64
53. <i>Activity Diagram</i> Verifikasi <i>Check-In</i> atau <i>Check-Out</i>	65
54. <i>Activity Diagram</i> Verifikasi SGL atau CST.....	66
55. <i>Activity Diagram</i> Melakukan Penilaian <i>Final Score</i>	67
56. <i>Activity Diagram</i> Melakukan Penilaian Nilai Mingguan.....	68
57. <i>Activity Diagram</i> Melihat Daftar Minggu <i>Daily Activity</i> Mahasiswa	69
58. <i>Activity Diagram</i> Mengecek <i>History</i>	69
59. <i>Activity Diagram</i> Mengecek Notifikasi	70

60.	Halaman <i>Login</i> dan <i>Register</i>	71
61.	Halaman <i>Reset Password</i>	71
62.	Halaman Notifikasi	72
63.	Halaman <i>Activity History</i>	73
64.	Halaman Menu Utama <i>Student</i>	74
65.	Halaman Departemen Aktif	74
66.	Halaman Menu Utama SGL dan CST	75
67.	Halaman Pengajuan SGL dan CST.....	75
68.	Halaman Tambah SGL (Kiri) dan Tambah Topik SGL (Kanan)	76
69.	Halaman Daftar <i>Clinical Record</i>	77
70.	Halaman Tambah <i>Clinical Record</i>	78
71.	Halaman Detail <i>Clinical Record</i>	78
72.	Halaman Daftar <i>Scientific Session</i> (Kiri) dan Halaman Tambah <i>Scientific Session</i> (Kanan).....	79
73.	Halaman Utama <i>Daily Activity</i> Mahasiswa	80
74.	Halaman Daftar <i>Daily Activity</i>	80
75.	Halaman Tambah atau Sunting <i>Daily Activity</i>	81
76.	Halaman Pilih Lokasi <i>Daily Activity</i>	81
77.	Halaman Daftar Kasus dan <i>Pop Up</i> Tambah Kasus	82
78.	Halaman Menu <i>Assessments</i>	83
79.	Halaman Nilai <i>Weekly Assessments</i>	83
80.	Halaman Nilai <i>Personal Behavior</i>	84
81.	Halaman Nilai <i>Mini-CEX</i> (kiri) dan Halaman Nilai <i>Scientific Assignment</i> (kanan)	84
82.	Halaman Nilai Akhir Mahasiswa	85
83.	Halaman Sunting Data Diri Mahasiswa	85
84.	Halaman Utama Supervisor	86
85.	Halaman Daftar Pengajuan SGL dan CST	87
86.	Halaman Daftar SGL yang Diajukan Mahasiswa	87
87.	Halaman Daftar Pengajuan <i>Daily Activity</i> (Kiri) dan Halaman Detail Pengajuan <i>Daily Activity</i> (Kanan)	88
88.	Halaman Detail Pengajuan <i>Clinical Record</i>	89

89. Halaman Daftar Pengajuan <i>Case</i> (Kiri) dan Halaman Daftar <i>Case</i> yang Diajukan (Kanan).....	89
90. Halaman Daftar Pengajuan <i>Problem Consultation</i> (Kiri) dan Halaman Daftar <i>Problem Consultation</i> yang Diajukan (Kanan).....	90
91. Halaman Penilaian <i>Scientific Assignment</i>	91
92. Halaman Penilaian <i>Personal Behavior</i>	91
93. Halaman Menu Mahasiswa Bimbingan	92
94. Halaman Verifikasi <i>Check-In</i>	92
95. Halaman Penilaian Final Score	93
96. Tampilan <i>Card</i> Verifikasi <i>SGL</i>	94
97. Halaman Utama Tim Kordik	94
98. Halaman Kelola Minggu <i>Daily Activity</i> Mahasiswa	95
99. Halaman Penilaian <i>Weekly Assessments</i>	95

DAFTAR LAMPIRAN

Nomor Urut	Halaman
1. <i>Entity Relational Diagram Database</i> E-Logbook	117
2. Struktur Tabel <i>Database</i> E-Logbook	117
3. Daftar <i>Activity Diagram</i> E-Logbook	117
4. Daftar <i>User Interface</i> E-Logbook.....	118
5. Daftar Hasil <i>Black-Box Testing</i> E-Logbook	118
6. Hasil UAT <i>User</i> MPPD	118
7. Hasil UAT <i>User</i> Supervisor	119
8. Hasil UAT <i>User</i> Tim CEU	119
9. Hasil UAT <i>User</i> Kepala Bagian	119
10. Hasil UAT <i>User</i> Tim Kordik	120

BAB I PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi informasi telah membawa perubahan mendasar dalam berbagai aspek kehidupan manusia. Salah satu dampak signifikan dari perkembangan ini adalah transformasi digital. Hampir semua sektor dari kegiatan yang dilakukan oleh manusia dialihkan ke sistem digital termasuk sektor pendidikan. Dengan adanya transformasi digital segala proses yang dilakukan secara manual dan tradisional berevolusi menjadi sistem digital yang lebih efisien. Di Indonesia sendiri digitalisasi dalam aspek pendidikan telah diatur dalam Undang-undang (UU) Nomor 20 Tahun 2003 tentang Sistem Pendidikan Nasional.

Dalam konteks pendidikan kedokteran, *Logbook* atau biasa disebut rekam medis merupakan salah satu pelayanan penunjang medis di rumah sakit yang menjadi dasar penilaian mutu pelayanan medik (Nuraini, 2018). *Logbook* dirancang agar menjadi sebuah dokumen penting yang berisi catatan pencapaian kompetensi Mahasiswa Program Profesi Dokter (MPPD) selama melaksanakan praktik klinis. *Logbook* dapat mendorong agar MPPD tertib dan disiplin mencatat informasi-informasi yang berkaitan dengan peningkatan kompetensi klinis yang dimiliki (Yahya, 2019). *Logbook* juga berfungsi sebagai alat evaluasi yang membantu fakultas kedokteran memastikan bahwa mahasiswa mencapai kompetensi yang diperlukan dalam praktek medis.

Fakultas Kedokteran Universitas Muslim Indonesia (FK UMI) merupakan salah satu lembaga yang menerapkan penggunaan *logbook* dalam pengelolaan data kinerja dan penilaian MPPD. Salah satu misi yang menjadi fokus FK UMI adalah menyelenggarakan tata kelola dan sistem manajemen mutu berstandar nasional dan internasional dengan berbasis teknologi informasi (Fakultas Kedokteran Universitas Muslim Indonesia, 2022). Namun, hingga saat ini, FK UMI masih menggunakan *logbook* konvensional berbasis kertas untuk mencatat pengalaman klinis mahasiswa. Penggunaan *logbook* konvensional ini membawa beberapa kendala dalam pelaksanaannya, termasuk masalah efisiensi serta manajemen data yang kurang praktis. *Logbook* berbasis kertas sering kali menghadirkan kesulitan dalam hal aksesibilitas, pembaruan, dan distribusi data yang efektif. Untuk mengatasi kendala dalam penggunaan *logbook* konvensional solusi yang sangat relevan dan modern adalah mengadopsi *logbook* digital. *Logbook* digital dapat memberikan manfaat signifikan mulai dari pencatatan yang lebih mudah, dokumentasi lebih rapi dan mudah diakses, distribusi data lebih mudah dan lainnya.

Beberapa penelitian terdahulu seperti yang dilakukan (Mathias, 2023) dan (Andry & Wijaya, 2020) telah membahas mengenai perancangan atau pembangunan *logbook* digital, namun umumnya masih terbatas pada fitur dasar, yaitu mahasiswa melakukan pengajuan verifikasi untuk kegiatan yang dilakukan, kemudian supervisor menyetujui pengajuan dan memberikan penilaian. Sehingga masih terdapat beberapa aspek yang dapat dikembangkan dalam pengadaan *logbook* digital di

masa depan. Terlebih lagi belum ada aturan umum untuk *logbook* yang digunakan di setiap universitas, sehingga terdapat serangkaian alur dan aturan tersendiri untuk *logbook* di Universitas Muslim Indonesia.

Untuk mengatasi permasalahan yang telah disebutkan, penulis menemukan solusi untuk membangun aplikasi *mobile* yang dirancang khusus untuk pengisian *logbook* di lingkup MPPD Universitas Muslim Indonesia. Aplikasi ini bertujuan untuk meningkatkan efisiensi dan akurasi dalam mencatat pengalaman klinis mahasiswa di FK UMI. Aplikasi berbasis *mobile* dipilih karena lebih mudah diakses dan digunakan oleh pengguna, serta dapat diakses kapan saja dan di mana saja. Dengan adanya aplikasi ini, diharapkan mahasiswa dan supervisor akan lebih mudah dan cepat dalam mengakses, mengisi, serta mengelola *logbook* mereka, sehingga memastikan pencatatan pengalaman klinis yang lebih akurat dan efisien. Penulis juga merancang beberapa fitur yang dapat menjadi pengembangan dari penelitian sebelumnya mulai dari absensi harian, riwayat kegiatan serta notifikasi untuk pengingat *task* yang perlu dilakukan. Hal ini merupakan langkah cepat menuju pendidikan kedokteran yang lebih modern, serta berlandaskan teknologi informasi di lingkungan FK UMI. Oleh karena itu, berdasarkan pertimbangan atas permasalahan yang terjadi, penulis melakukan perancangan dan pembangunan aplikasi *E-Logbook* Fakultas Kedokteran Universitas Muslim Indonesia untuk skripsi tugas akhir, dengan judul; "RANCANG BANGUN APLIKASI *E-LOGBOOK* MPPD BERBASIS *MOBILE* (STUDI KASUS: FAKULTAS KEDOKTERAN UNIVERSITAS MUSLIM INDONESIA)."

1.2 Rumusan Masalah

Dengan mengacu pada latar belakang yang telah disebutkan, maka rumusan masalah yang akan dibahas dalam penelitian ini sebagai berikut:

1. Bagaimana merancang dan membangun aplikasi *E-Logbook* MPPD Fakultas Kedokteran Universitas Muslim Indonesia?
2. Bagaimana pengujian aplikasi *E-Logbook* MPPD Fakultas Kedokteran Universitas Muslim Indonesia?
3. Bagaimana efektivitas aplikasi *E-Logbook* MPPD Fakultas Kedokteran Universitas Muslim Indonesia?

1.3 Batasan Masalah

Untuk mempersempit cakupan penelitian sesuai dengan maksud dan tujuan yang diinginkan, maka penelitian ini akan membatasi masalah sebagai berikut:

1. Aplikasi *E-Logbook* yang dikembangkan akan berbasis Android dan iOS serta dirancang menggunakan *framework* Flutter.
2. Sampel pengguna akan terbatas pada lingkungan Mahasiswa Program Profesi Dokter Fakultas Kedokteran Universitas Muslim Indonesia.

3. Peran (*role*) pengguna akan dibatasi pada MPPD atau mahasiswa, Supervisor selaku dosen pembimbing mahasiswa, Tim Koordinator Pendidikan (Kordik) selaku penanggung jawab pelaksanaan pendidikan di fakultas kedokteran, Tim *Clinical Education Unit* (CEU) selaku penanggung jawab evaluasi hasil belajar MPPD, dan Kepala Bagian selaku pimpinan bagian atau departemen tertentu di fakultas kedokteran.

1.4 Tujuan Penelitian

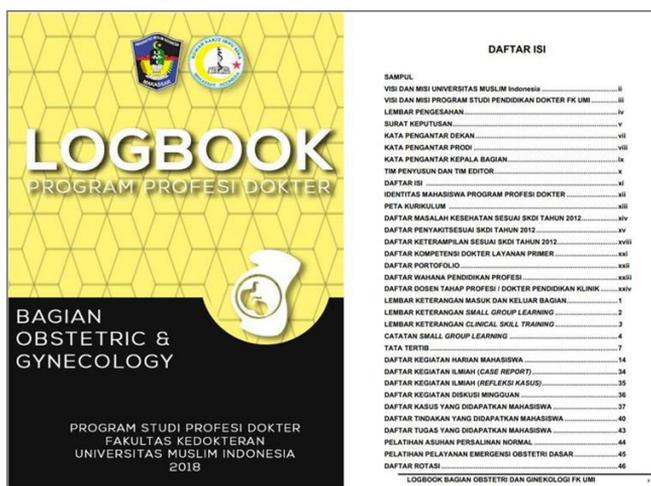
Berdasarkan latar belakang dan rumusan masalah yang telah disebutkan, penelitian ini bertujuan untuk mencapai hal-hal berikut:

1. Merancang dan membangun aplikasi *E-Logbook* yang sesuai dengan kebutuhan Fakultas Kedokteran Universitas Muslim Indonesia.
2. Melakukan pengujian aplikasi *E-Logbook* MPPD Fakultas Kedokteran Universitas Muslim Indonesia untuk mengevaluasi keandalan, dan kinerja aplikasi dalam mengelola data kegiatan mahasiswa secara akurat dan efisien.
3. Menilai efektivitas aplikasi *E-Logbook* MPPD Fakultas Kedokteran Universitas Muslim Indonesia dengan mengukur tingkat kepuasan pengguna, dan kemudahan akses informasi pada aplikasi.

1.5 Teori

1.5.1 *Logbook* Fakultas Kedokteran Universitas Muslim Indonesia

Dalam konteks pendidikan kedokteran, *logbook* merupakan alat yang sangat penting dalam pelatihan kedokteran, termasuk kegiatan MPPD. *Logbook* dapat membantu MPPD merekam dan merefleksikan aktivitas klinis mereka untuk digunakan sebagai bahan evaluasi dan pengembangan. Selain itu, *logbook* juga dapat digunakan sebagai bukti bahwa MPPD telah menyelesaikan kegiatan yang dipersyaratkan dalam program pelatihan. Oleh karena itu, penting bagi MPPD untuk mengisi *logbook* secara teratur dan akurat serta menjadikannya bagian integral dari pekerjaan klinis mereka (Mathias, 2023).



Gambar 1. Contoh Logbook FK UMI

Di Fakultas Kedokteran UMI, terdapat 15 bagian atau departemen berbeda yang perlu MPPD jalani tiap terjadi rotasi. Setiap bagian memiliki periode atau lama *stase* masing-masing. Berikut pada tabel 2.1 berisi daftar setiap bagian beserta lama *stase*.

Tabel 1. Daftar Bagian dan Lama *Stase* di FK UMI

No	Nama Departemen	Lama <i>Stase</i>
1	Anestesi	4 Minggu
2	Bedah	8 Minggu
3	Forensik	4 Minggu
4	Ilmu Kesehatan Masyarakat – Ilmu Kedokteran Komunitas (IKK-KK)	8 Minggu
5	Ilmu Kesehatan Anak	10 Minggu
6	Ilmu Penyakit Mata	4 Minggu
7	Interna	10 Minggu
8	Kardiologi	4 Minggu
9	Kesehatan Jiwa	4 Minggu
10	Kulit Kelamin	4 Minggu
11	Neurologi	4 Minggu
12	Obstetri dan Ginekologi	10 Minggu
13	Ortopedi	3 Minggu
14	Radiologi	3 Minggu

No	Nama Departemen	Lama Stase
15	Telinga Hidung Tenggorokan – Kepala Leher (THT-KL)	4 Minggu

Pada setiap departemen MPPD memiliki *logbook* tersendiri. Secara umum, setiap *logbook* tersebut mencakup data-data kegiatan mahasiswa selama menjalani studi di departemen tertentu. Dokumen atau data-data yang terdapat pada *logbook* ini yang sekaligus akan menjadi persyaratan untuk perpindahan departemen atau rotasi bagian. Berikut disajikan daftar data atau dokumen yang terdapat pada *logbook*.

Tabel 2. Daftar dokumen atau data pada *logbook* FK-UMI

No	Nama Dokumen	Deskripsi Dokumen	Syarat Perpindahan Stase atau Departemen
1	Lembar keterangan <i>check-in</i>	Pengajuan <i>check-in</i> dilakukan sebelum memulai kegiatan di suatu departemen. Lembar ini berisi data departemen tujuan, waktu pengajuan, dan diverifikasi oleh kepala bagian yang dituju.	Ya
2	Lembar keterangan <i>check-out</i>	Untuk berpindah dari departemen aktif MPPD membuat pengajuan <i>check-out</i> . Lembar ini juga diverifikasi oleh kepala bagian.	Ya
3	Daftar kegiatan harian	Selama stase berlangsung seluruh kegiatan yang dilakukan MPPD dicatat dan diverifikasi oleh <i>supervisor</i> terkait.	Ya
4	<i>Small group learning (SGL)</i>	SGL merupakan metode pembelajaran ketika MPPD bekerja dalam kelompok kecil dan menganalisis suatu masalah. Hasil dari kegiatan SGL dicatat dan diverifikasi oleh <i>supervisor</i> yang memandu diskusi.	Tidak
5	<i>Clinical skill training (CST)</i>	CST merupakan latihan keterampilan praktis yang dilakukan MPPD dengan melakukan suatu prosedur medis. Setiap CST dicatat dan diverifikasi oleh <i>supervisor</i> yang mengawasi praktek.	Tidak
6	Rekam medis	Rekam medis berisi catatan yang mendokumentasikan hasil pemeriksaan kesehatan yang dilakukan MPPD terhadap pasien tertentu.	Tidak
7	Daftar kasus	Daftar kasus yang ditemukan MPPD selama rotasi klinis di departemen	Ya

		tertentu. Setiap kasus memiliki tingkat kompetensi tertentu.	
8	Daftar tindakan	Daftar tindakan yang dilakukan atau dipraktikkan MPPD selama rotasi klinis di departemen tertentu. Setiap tindakan memiliki tingkat kompetensi tertentu/	Ya
9	Kegiatan ilmiah	Pada <i>logbook</i> terdapat beberapa kategori sesi ilmiah yang bisa diikuti MPPD misalnya refleksi kasus, <i>case report</i> , dan lainnya. Sesi ilmiah yang diikuti MPPD didokumentasikan dan diverifikasi oleh DPK	Ya
10	Nilai mingguan	Data penilaian MPPD untuk setiap minggu yang diberikan oleh tim kordinator pendidikan (kordik)	Ya
11	Nilai tugas ilmiah	MPPD mengajukan tugas ilmiah ketika mengikuti <i>case report</i> . Tugas ilmiah yang diajukan akan dinilai oleh DPK pembimbing	Tidak
12	Nilai evaluasi klinis (<i>mini-cex</i>)	Pada ujian (Mini-CEX) MPPD mengajukan sebuah kasus ujian dan dinilai oleh DPK penguji.	Tidak
13	Nilai <i>personal behavior</i>	Pada setiap departemen dilakukan penilaian terhadap <i>personal behavior</i> MPPD dengan beberapa indikator pencapaian.	Tidak
14	Nilai OSCE dan CBT	Pada lembar nilai akhir terdapat penilaian untuk ujian CBT (<i>Post test</i>) dan CBT yang telah diikuti MPPD. Verifikasi nilai akhir ini dilakukan oleh CEU	Tidak
15	Daftar konsultasi masalah	Terdapat lembar konsultasi untuk masalah-masalah yang dihadapi MPPD selama rotasi klinis di departemen tertentu. <i>Supervisor</i> atau DPK dapat memberikan solusi untuk masalah yang diajukan MPPD.	Tidak

Berdasarkan Tabel 2 dokumen atau berkas wajib yang menjadi syarat untuk perpindahan departemen terdiri dari lembar keterangan *check-in* dan *check-out*, lembar daftar kegiatan harian, nilai mingguan, daftar kasus, daftar tindakan, dan daftar sesi ilmiah yang semuanya telah terverifikasi.

1.5.2 Aplikasi *Mobile*

Aplikasi *mobile* adalah perangkat lunak yang dikembangkan khusus untuk *platform mobile* tertentu, seperti iOS, Android, *Windows Phone*, dan lainnya. Aplikasi ini dirancang untuk digunakan pada *smartphone*, *tablet*, *smartwatch*, dan perangkat *mobile* lainnya. Pengembangan aplikasi *mobile* melibatkan pemahaman tentang metode dan alat-alat yang digunakan untuk membuat aplikasi di berbagai *platform mobile* (Nafasov et al., 2022).

Salah satu keunggulan utama aplikasi *mobile* adalah ketersediaannya di perangkat seluler yang senantiasa berada pada genggaman pengguna. Fleksibilitas ini memungkinkan akses dan penggunaan aplikasi dengan mudah di berbagai situasi dan lokasi, menciptakan pengalaman pengguna yang lebih praktis dan efisien, serta dapat memenuhi kebutuhan sehari-hari dengan lebih baik.

Melihat ke arah sistem operasi *mobile*, Android adalah sistem operasi berbasis Linux yang digunakan sebagai pengelola sumber daya perangkat keras, baik untuk ponsel, *smartphone*, maupun tablet. Sebagai *platform* terbuka (*Open Source*), Android memberikan keleluasaan kepada para pengembang untuk menciptakan aplikasi mereka sendiri yang dapat diaplikasikan pada berbagai piranti bergerak (Lestari, 2017). Sistem operasi ini, yang awalnya dikembangkan oleh Android, Inc., diakuisisi oleh Google pada tahun 2005.

Di sisi lain, aplikasi iOS dirancang khusus untuk perangkat berbasis sistem operasi iOS, yang dikembangkan oleh Apple. Digunakan pada perangkat Apple seperti iPhone, iPad, dan iPod Touch, iOS memiliki ciri khasnya sendiri. Berbeda dengan Android yang bersifat *open source*, Apple Inc. tidak melisensikan sistem operasi iOS untuk digunakan pada perangkat keras lain, menciptakan eksklusivitas yang melekat pada produk Apple (Tangka et al., 2023).

Android dan iOS merupakan dua *platform mobile* paling populer di Indonesia. Berdasarkan data yang dilansir dari StatCounter Global Stats dari Januari 2023 sampai Oktober 2023 Android memegang pangsa pasar Indonesia sebesar 88,3% dan iOS sebesar 11,57% (Statcounter, 2022). Popularitas Android di Indonesia dapat dikaitkan dengan keterjangkauan harga perangkat Android, sementara popularitas iOS dikaitkan dengan citra merek Apple yang eksklusif dan premium.

1.5.3 *Framework Flutter*

Flutter adalah SDK atau *framework open source* yang dikembangkan oleh Google untuk membuat aplikasi yang dapat berjalan di sistem operasi Android dan iOS. Flutter menggunakan bahasa pemrograman Dart dalam pengkodean. Salah satu perbedaan utama Flutter dengan *framework* lainnya adalah bahwa semua kode Flutter dikompilasi menjadi kode *native* (Android NDK, LLVM, AOT compiled) tanpa interpreter dalam prosesnya, sehingga proses kompilasi menjadi lebih cepat (Krisnada & Tanone, 2020).

Flutter pertama kali diperkenalkan ke publik pada tahun 2017 dalam acara Google I/O. Versi pertama Flutter dikenal sebagai "Sky" dan hanya berjalan pada

sistem operasi Android. Flutter terus mengalami perkembangan dan pembaharuan, pada agustus 2023 Flutter merilis versi 3.0.2 hadir dengan berbagai peningkatan performa. Flutter dapat digunakan untuk mengembangkan aplikasi yang kompatibel dengan berbagai *platform* (*cross-platform*), seperti Android, iOS, web (dalam versi beta), macOS, Linux, Windows, dan perangkat *embedded*, semuanya dengan menggunakan satu kode sumber yang sama.

1.5.4 REST API

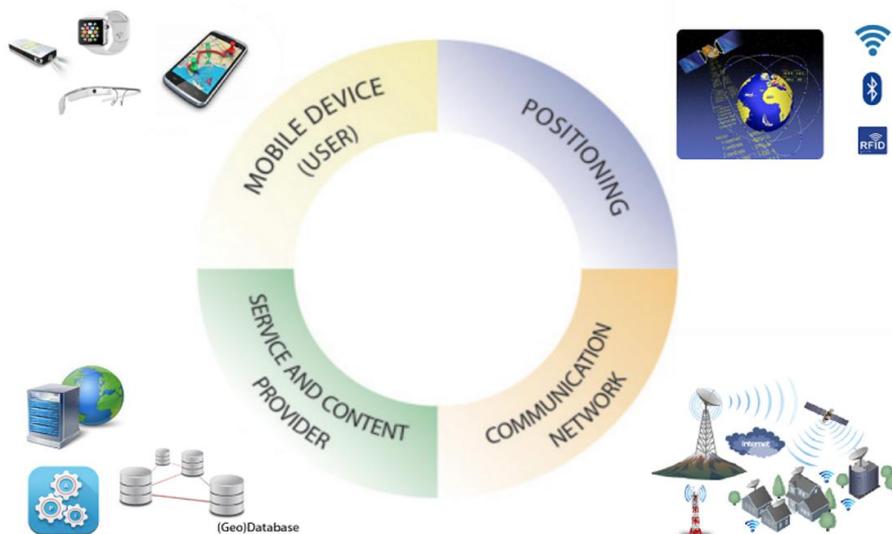
REST API (*Representational State Transfer Application Programming Interface* merupakan) merupakan sebuah layanan *web service* yang menawarkan arsitektur komunikasi untuk melakukan pertukaran data antara *client* dan *server* melalui protokol HTTP. Dalam proses ini, data yang disampaikan dan diterima akan diubah menjadi format JavaScript *Object Notation* (JSON), memungkinkan untuk komunikasi yang terstruktur dan kompatibel. Melalui *Uniform Request Identifier* (URI), *client* dapat mengirimkan permintaan kepada *server* yang akan memberikan *response* sesuai dengan permintaan yang diajukan. Beberapa metode HTTP yang umum digunakan dalam interaksi ini antara lain adalah *POST*, *PUT*, *GET* dan *DELETE* (Amin, 2018).

Implementasi REST API di dalam pengembangan aplikasi Flutter melibatkan penggunaan pustaka-pustaka seperti *dio* dan *http*. Pustaka-pustaka ini memfasilitasi pembuatan permintaan ke *endpoint* tertentu di *server* serta pengelolaan *response* yang diterima dari *server*, memungkinkan pengembang untuk berinteraksi dengan sumber daya yang ada di *server* secara efisien dan terstruktur. Dengan memanfaatkan pustaka tersebut, pengembang dapat melakukan serangkaian tindakan, mulai dari pengambilan data, pengiriman data, hingga manipulasi data dengan mudah dalam lingkungan Flutter.

1.5.5 Location-Based Service (LBS)

Location Based Service (LBS) atau layanan berbasis lokasi adalah layanan informasi yang dapat diakses dengan perangkat bergerak melalui jaringan dan mampu menampilkan posisi letak secara geografis keberadaan dari perangkat bergerak tersebut (Hasian et al., 2023).

Menurut (Huang & Gao, 2018) *Location-Based Services* (LBS) adalah aplikasi seluler yang menyediakan informasi tergantung pada lokasi pengguna. LBS terdiri dari empat komponen utama yang dapat dilihat pada Gambar 2.



Gambar 2. Komponen Sistem LBS

Perangkat seluler (*Mobile Device*). Perangkat seluler adalah alat portabel seperti *smartphone* dan *tablet* yang digunakan untuk mengakses informasi dan berbagai layanan. Pengguna dapat melakukan panggilan, mengirim pesan, serta menggunakan aplikasi berbasis internet.

Komponen pemosisian (*Positioning*). Komponen pemosisian, seperti GPS, menentukan lokasi pengguna saat ini menggunakan teknologi satelit dan jaringan. Informasi lokasi ini penting untuk navigasi dan layanan berbasis lokasi lainnya.

Jaringan komunikasi (*Communication Network*). Jaringan komunikasi memungkinkan perangkat seluler mengirim dan menerima data melalui koneksi seperti 4G, 5G, dan Wi-Fi. Jaringan ini menghubungkan pengguna dengan internet dan layanan digital lainnya.

Penyedia layanan dan konten (*Service and Content Provider*). Penyedia layanan dan konten memproses permintaan pengguna dan mengembalikan informasi atau layanan yang diperlukan. Mereka mencakup berbagai *platform* seperti media sosial, layanan *streaming*, dan toko aplikasi.

1.5.6 *Firestore Cloud Messaging (FCM)*

Firestore Cloud Messaging (FCM) merupakan salah satu layanan Google untuk pengiriman pesan atau pemberitahuan pesan kepada pengguna sistem melalui *web browser* atau *smartphone*. FCM menyediakan koneksi antar *server* ataupun antar *device*, sehingga pesan notifikasi dapat berjalan lancar di Android, iOS, dan *website* tanpa perlu biaya mahal. FCM mengharuskan pengguna dalam kondisi jaringan internet supaya pesan notifikasi aktif berjalan (Putro et al., 2022).

Implementasi FCM di Flutter adalah proses untuk mengirim dan menerima pesan *push* dari *server* FCM ke aplikasi Flutter. Proses ini dilakukan dengan

menggunakan pustaka *firebase messaging*. Pustaka ini menyediakan *tools* yang memungkinkan aplikasi Flutter untuk menginisialisasi *plugin* FCM, mendaftarkan *listener* untuk notifikasi, membuat notifikasi dan menampilkan notifikasi.

1.5.7 Black-Box Testing

Black-Box Testing merupakan pengujian yang berfokus pada spesifikasi fungsional dari perangkat lunak, *tester* dapat mendefinisikan kumpulan kondisi *input* dan melakukan pengujian pada spesifikasi fungsional program (Mustaqbal et al., 2015). Beberapa kelebihan dari *Black-Box Testing* dapat dilihat pada Tabel 3 berikut.

Tabel 3. Kelebihan *Black-Box Testing*

No	Kelebihan
1	Pengujian tidak harus memiliki pengetahuan tentang suatu bahasa pemrograman.
2	Pengujian dilakukan berdasarkan sudut pandang pengguna. Hal tersebut dilakukan agar dapat menemukan inkonsistensi dalam perangkat lunak.
3	Efisien untuk sistem yang besar karena pengujian tidak perlu memahami detail implementasi sistem untuk dapat mengujinya.
4	Pengujian tidak perlu memeriksa kode.
5	Memungkinkan pengujian dan pengembang bekerja secara independen tanpa mengganggu proses kerja satu sama lain.

1.5.8 User Acceptance Test (UAT)

User acceptance test (UAT) merupakan suatu proses pengujian oleh pengguna yang dimaksudkan untuk menghasilkan dokumen yang dijadikan bukti bahwa perangkat lunak yang dikembangkan telah dapat diterima oleh pengguna. Tingkat penerimaan sebuah sistem oleh pengguna dapat menjadi suatu tolak ukur untuk menilai penerimaan sebuah teknologi informasi oleh pengguna (Supriatna, 2019).

Skala Likert adalah alat ukur yang umum digunakan dalam UAT untuk mengukur pendapat dan persepsi pengguna terhadap suatu sistem. Skala ini terdiri dari serangkaian pernyataan yang terkait dengan aspek-aspek tertentu dari sistem, seperti fungsionalitas, kegunaan, dan kepuasan pengguna. Skala dalam penelitian ini menggunakan skala Likert dengan interval empat (*afour-point likert scale*) berupa skala numerik berkisar dari 1 hingga 4 pilihan penilaian yaitu "Sangat Setuju (SS)," "Setuju (S)," "Kurang Setuju (KS)," dan "Tidak Setuju (TS)" (Martillano et al., 2019).

Modifikasi terhadap skala Likert dimaksudkan untuk menghilangkan kelemahan yang terkandung oleh skala lima tingkat. Peniadaan kategori jawaban yang ditengah dilakukan untuk menghindari kategori *undecided* atau bisa diartikan belum memutuskan jawaban. Tersedianya jawaban pertengahan juga dapat

menimbulkan *centraltendency effect* atau kecenderungan memilih pilihan netral terutama bagi responden yang ragu-ragu dalam memilih jawaban. Tentu saja pilihan seperti ini tidak diharapkan dalam suatu instrument karena dapat menghilangkan banyak data penelitian dan mengurangi banyaknya informasi yang dapat dijangkau dari responden (Sari Tanjung et al., 2022).

1.5.9 Figma

Figma merupakan *platform* berbasis web yang berguna dalam membuat desain antarmuka pengguna (UI) dan pengalaman pengguna (UX). Dengan Figma, pengguna dapat membuat desain untuk aplikasi, situs web, dan berbagai komponen antarmuka pengguna yang dapat digunakan dalam proyek lainnya. Figma populer di kalangan desainer karena fleksibilitasnya, aksesibilitas *online*, dan fitur kolaboratifnya (Purbo et al., 2023).

1.5.10 Visual Studio Code

Visual Studio Code adalah sebuah *code editor* yang dibuat oleh Microsoft untuk berbagai sistem operasi. Visual Studio Code atau disingkat dengan VS Code merupakan perangkat lunak *open source* dibawah lisensi MIT. Diluncurkan sebagai pada 29 April 2015, Visual Studio Code telah berkembang menjadi salah satu proyek Microsoft dengan jumlah kontributor tertinggi di *github* (Setiawan, 2022).

Selain dukungan *debugging*, penyorotan *sintaks*, dan *git* terintegrasi, Visual Studio Code juga memiliki beragam fitur seperti bantuan penulisan kode, potongan kode yang dapat digunakan kembali, perbaikan kode, serta banyak ekstensi untuk berbagai bahasa pemrograman, *runtime*, lingkungan pengembangan, dan *cloud*. Pengguna dapat menyesuaikan tampilan, pintasan *keyboard*, preferensi, dan menambahkan ekstensi sesuai keinginan. Karena kecepatan, dukungan untuk banyak bahasa pemrograman dan kerangka kerja, serta fleksibilitasnya, banyak pengembang yang menggunakan Visual Studio Code.

1.5.11 Postman

Postman adalah *platform* pengembangan dan pengujian API yang menawarkan berbagai fitur. Postman memungkinkan para pengembang untuk dengan mudah membuat dan berbagi API *requests* dan *collections*, mengotomatisasi pengujian, *mock* API, dan memantau kinerja. Postman juga menyediakan alat kolaborasi dan dokumentasi, memungkinkan tim untuk bekerja sama dengan lebih efisien dan mengkomunikasikan perilaku API dengan efektif (Demir, 2023).

1.6 Penelitian Terkait

Penelitian terkait pertama mengacu pada studi berjudul "Pengembangan Antarmuka Sistem Informasi *Logbook Coass* Program Studi Pendidikan Dokter di Fakultas Kedokteran Universitas XYZ" yang dilaksanakan oleh (Mathias, 2023). Dalam penelitian ini, aplikasi *logbook* dibangun menggunakan *framework* Flutter dan tersedia di *platform* Android dan iOS. Fitur- fitur yang dimiliki terbatas pada kemampuan MPPD untuk mengajukan penilaian dan mencatat *logbook*, serta pemberian nilai dan persetujuan atas pencatatan *logbook* oleh dosen. Metode penelitian yang digunakan pada penelitian ini dimulai dengan analisis kebutuhan, dilanjutkan dengan perancangan aplikasi menggunakan *Use Case Specification*, implementasi, dan kemudian diakhiri pengujian fungsionalitas perangkat lunak.

Penelitian terkait berikutnya dalam pengembangan sistem informasi *logbook* digital adalah studi yang berjudul "Perancangan Aplikasi *E-Logbook*: Studi Kasus di Fakultas Kedokteran Universitas Indonesia" yang dilakukan oleh (Andry & Wijaya, 2020). Penelitian ini bertujuan untuk merancang sebuah sistem informasi *logbook* digital berbasis *website* dengan menggunakan metode SDLC *Waterfall*. Rancangan website yang akan dibangun akan menggunakan *framework* Laravel dan basis data MySQL. Dalam analisis desain perancangan website *E-Logbook* ini, terdapat tiga jenis pengguna, yaitu PPDS atau mahasiswa yang mengirim *logbook*, supervisor yang memeriksa *logbook* yang dikirim oleh mahasiswa, dan SPS yang memantau perkembangan kegiatan mahasiswa. Karena penelitian ini masih dalam tahap perancangan, pengujian terhadap sistem yang dibangun belum dapat dilakukan.

Selanjutnya penelitian yang berjudul "Perancangan Sistem Informasi Penilaian *Coass* Dokter pada RSUD Pasar Rebo" yang dilakukan oleh (Nuryanis et al., 2022). Penelitian ini bertujuan untuk membuat sistem informasi yang dapat memudahkan pihak RSUD dalam mengelola data penilaian *coass*, dan membuat data *coass* lebih mudah diakses oleh bakordik. Pembangunan sistem informasi yang dilakukan pada penelitian ini menggunakan pendekatan *System Development Life Cycle* (SDLC) dengan metode *waterfall*. Sistem informasi penilaian *coass* yang dibangun berbasis desktop dan menggunakan bahasa Java.

Selanjutnya penelitian yang berjudul "Sistem Informasi Akademik Berbasis *Mobile* Menggunakan Flutter" yang dilakukan oleh (Sumihar & Theopilus, 2021). Dalam penelitian ini, para peneliti mengembangkan sebuah sistem informasi akademik (SIA) untuk mengelola dan menyebarkan informasi yang terkait dengan akademik mahasiswa. Tujuan pembuatan sistem informasi ini adalah karena SIA yang telah ada berbasis Android, sedangkan banyak mahasiswa yang menggunakan iOS. Perancangan aplikasi dimulai dengan mengidentifikasi kebutuhan fungsional, perancangan antarmuka pengguna (*user interface*), dan perancangan layanan *web RESTful API*.

Penelitian yang berjudul "Pengembangan Sistem Informasi Akademik Berbasis Android Menggunakan Flutter di Politeknik" yang dilakukan oleh (Hakim et al., 2019). Dalam penelitian ini, para peneliti melakukan pengembangan aplikasi akademik berbasis Android yang dibangun menggunakan *framework* Flutter. Analisis

aplikasi dilakukan dengan menggunakan *use case*, *class diagram*, ERD, dan kemudian dilanjutkan dengan perancangan antarmuka. Pengujian dilakukan dengan metode *black box*, dan hasilnya menunjukkan bahwa fitur-fitur yang dirancang berjalan dengan baik.

Perbandingan antara penelitian terdahulu dan penelitian ini dapat ditemukan secara rinci pada Tabel 2.1, yang menampilkan perbedaan mulai dari tujuan penelitian, hasil atau fitur aplikasi yang dibangun, metode penelitian yang diterapkan, hingga teknologi yang digunakan. Tujuan dari perbandingan ini adalah untuk menyoroti perbedaan serta perkembangan yang terjadi dari penelitian sebelumnya menuju penelitian yang sedang dilakukan.

Tabel 4. Penelitian Terkait

No	Nama Peneliti	Tahun	Judul	Hasil Penelitian	Perbandingan Penelitian
1	Mathias Ananda Arief Pradipto	2023	Pengembangan Antarmuka Sistem Informasi <i>Logbook Coass</i> Program Studi Pendidikan Dokter di Fakultas Kedokteran Universitas XYZ	Peneliti membangun sistem informasi <i>logbook coass</i> program studi pendidikan dokter fakultas kedokteran Universitas XYZ yang tersedia di <i>platform</i> Android dan iOS menggunakan <i>framework</i> Flutter.	Pada penelitian sebelumnya fitur terbatas penilaian dan pencatatan <i>logbook</i> . Sedangkan pada penelitian ini terdapat fitur tambahan mulai dari absensi atau pencatatan aktivitas harian yang disertai LBS, fitur <i>forgot password</i> , informasi statistik <i>logbook</i> mahasiswa serta fitur notifikasi untuk semua <i>role</i> .
2	Andry Johannes Fernandes dan Andy Wijaya	2020	Perancangan Aplikasi <i>E-Logbook</i> : Studi Kasus di Fakultas Kedokteran Universitas Indonesia	Peneliti berhasil membuat perancangan <i>logbook</i> digital berbasis <i>website</i> untuk studi kasus fakultas kedokteran Universitas Indonesia.	Pada penelitian sebelumnya <i>logbook</i> berbasis <i>website</i> dan dibangun dengan Laravel. Serta fitur terbatas pada pencatatan dan penilaian <i>logbook</i> . Sedangkan pada penelitian ini <i>logbook</i> berbasis <i>mobile</i> dan terdapat berbagai fitur tambahan seperti notifikasi, dan absensi.
3	Nuryanis Selly, Dewi Anjani, dan Rahmawati Yuliani	2022	Perancangan Sistem Informasi Penilaian <i>Coass</i> Dokter pada RSUD Pasar Rebo	Peneliti berhasil membuat perancangan sistem informasi penilaian <i>coass</i> dokter RSUD Pasar Rebo berbasis <i>desktop</i> menggunakan Java dan MySQL.	Pada penelitian sebelumnya sistem informasi yang dibangun berbasis <i>desktop</i> dan terbatas pada fitur penilaian <i>coass</i> oleh <i>administrator</i> . Sedangkan pada penelitian ini memiliki sistem dibangun berbasis <i>mobile</i> dan memiliki fitur yang lebih banyak mulai dari pencatatan, verifikasi, penilaian, absensi, dan lain-lain.

4	Yo'el Pieter Sumihar dan Arnan Abdiel Theopilus	2021	Sistem Informasi Akademik Berbasis <i>Mobile</i> Menggunakan Flutter	Peneliti berhasil membangun sistem informasi akademik berbasis <i>mobile</i> yang tersedia di <i>platform</i> iOS dan Android menggunakan <i>framework</i> Flutter. Fitur dari aplikasi yang dibangun mulai dari <i>e-learning</i> , presensi, <i>krs-online</i> , serta data akademik.	<p>Penelitian sebelumnya bertujuan untuk membangun sistem informasi akademik. Untuk sisi teknologi penelitian sebelumnya hanya mengimplementasikan <i>framework</i> Flutter dan REST API tanpa ada integrasi tambahan.</p> <p>Sedangkan penelitian ini bertujuan untuk membangun <i>logbook</i> digital dan memanfaatkan teknologi-teknologi lain selain Flutter mulai dari LBS yang diimplementasikan dalam presensi serta penggunaan FCM untuk memberikan notifikasi ke pengguna.</p>
5	Hakim Arip Rahmat, Kecitaan Harefa, dan Bambang Widodo	2019	Pengembangan Sistem Informasi Akademik Berbasis Android Menggunakan Flutter di Politeknik	Peneliti berhasil mengembangkan sistem informasi akademik berbasis Android menggunakan Flutter	<p>Penelitian sebelumnya bertujuan untuk membangun sistem informasi akademik dengan menggunakan <i>framework</i> Flutter dan REST API. Aplikasi yang dibuat hanya untuk <i>platform</i> Android, dan walaupun terdapat menu jadwal mata kuliah, namun tidak ada penerapan fitur notifikasi.</p> <p>Sementara itu, dalam penelitian ini, aplikasi dikembangkan berbasis <i>mobile</i> yang tersedia di dua <i>platform</i>, Android dan iOS. Terdapat fitur notifikasi yang berfungsi sebagai pengingat bagi pengguna.</p>

BAB II METODE PENELITIAN

2.1 Waktu dan Lokasi Penelitian

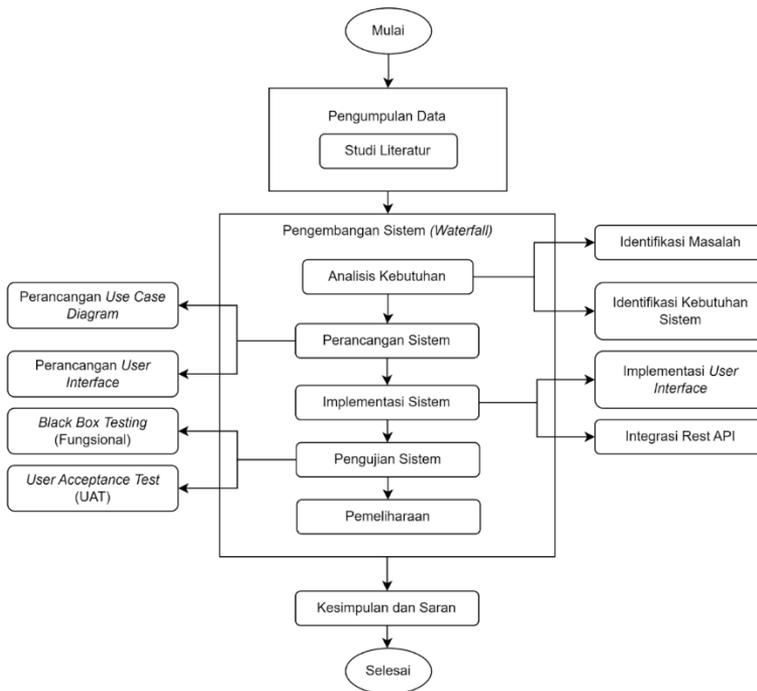
Penelitian ini dilakukan dari bulan Agustus 2023 sampai November 2023. Lokasi penelitian ini dilakukan di Fakultas kedokteran Universitas Muslim Indonesia dan Rumah Sakit Ibnu Sina.

Tabel 5. *Timeline* Penelitian

NO	Tahapan Penelitian	Sep				Okt				Nov				Des			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	Analisis Kebutuhan																
2	Desain Sistem																
3	Implementasi Sistem																
4	Pengujian Sistem																
5	Hasil dan Pemeliharaan																

2.2 Alur Penelitian

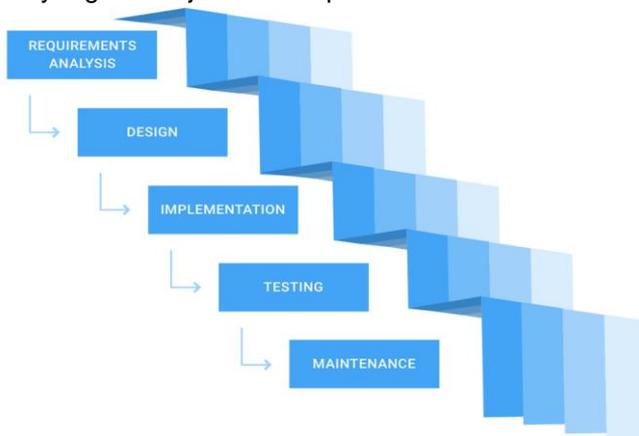
Alur penelitian adalah urutan langkah-langkah atau tahapan yang harus diikuti dalam menjalankan sebuah penelitian. Alur penelitian ini mencakup beberapa tahapan mulai dari proses pengumpulan data kemudian pengembangan sistem akan dilakukan dengan menggunakan metode *waterfall*. Setelah seluruh proses pengembangan sistem selesai penelitian diakhiri dengan membuat kesimpulan dan saran. Untuk lebih rincinya alur penelitian yang dilakukan seperti pada Gambar 3.



Gambar 3. Alur Penelitian

2.3 Metode Pengembangan Sistem

Metode pengembangan sistem adalah serangkaian langkah-langkah yang sistematis dan terstruktur untuk mengembangkan sistem informasi. Dalam penelitian ini metode pengembangan sistem yang akan digunakan adalah metode *waterfall*. Berikut Gambar 4 yang menunjukkan tahapan dari metode *waterfall*.



Gambar 4. Metode *Waterfall*

Sumber : <https://marketsplash.com/software-development-life-cycle/>

2.3.1 Analisis Kebutuhan (*Requirements Analysis*)

Tahap pertama dalam metode *waterfall* adalah mengidentifikasi masalah yang ingin diselesaikan dan mengidentifikasi kebutuhan sistem secara menyeluruh. Peneliti bekerja sama dengan *stakeholder* untuk memperoleh informasi yang diperlukan mengenai tantangan utama yang ingin diatasi oleh sistem yang akan dikembangkan.

2.3.2 Perancangan (*Design*)

Pada penelitian ini proses perancangan mencakup pembuatan *use case diagram* dan antarmuka pengguna (*user interface*). *Use case diagram* memiliki peranan penting dalam menggambarkan interaksi antara pengguna dan sistem secara visual. Sementara perancangan antarmuka pengguna atau *user interface* (UI) bertujuan memberikan gambaran yang jelas mengenai fitur-fitur yang akan tersedia dalam sistem, memastikan pengguna dapat dengan mudah memahami dan menggunakan sistem secara efektif.

2.3.3 Implementasi (*Implementation*)

Setelah selesai tahap perancangan, langkah implementasi dimulai. Di fase ini, peneliti menulis kode perangkat lunak berdasarkan spesifikasi dan desain yang telah disusun sebelumnya. Tahap implementasi ini melibatkan penggunaan Flutter untuk mengembangkan aplikasi *mobile*, fokusnya adalah pada implementasi antarmuka pengguna yang telah direncanakan sebelumnya serta integrasi dengan layanan yang disediakan oleh REST API. Proses ini memungkinkan aplikasi untuk berinteraksi dengan layanan-layanan yang tersedia di REST API untuk memperoleh atau mengirim data yang diperlukan.

2.3.4 Pengujian (*Testing*)

Tahap pengujian adalah tahap ketika sistem yang telah dikembangkan diuji secara menyeluruh. Peneliti akan melakukan pengujian menggunakan metode *Black Box Testing* dan *User Acceptance Testing* (UAT). *Black box testing* bertujuan untuk menguji dan memverifikasi bahwa sistem berfungsi sesuai dengan spesifikasi dan kebutuhan, serta mengidentifikasi kesalahan yang mungkin terjadi. Sedangkan UAT adalah pengujian yang melibatkan pengguna akhir atau *stakeholder* untuk menguji perangkat lunak sebelum dirilis ke lingkungan produksi.

2.3.5 Pemeliharaan (*Maintenance*)

Setelah aplikasi selesai dirancang dan diuji, tahap pemeliharaan dimulai. Pada tahap ini, fokus utamanya adalah mengidentifikasi dan memperbaiki bug atau kesalahan

yang mungkin muncul selama pengujian aplikasi, serta melakukan pemeliharaan rutin untuk memastikan kinerja aplikasi tetap optimal.

2.4 Instrumen Penelitian

2.4.1 Kebutuhan Perangkat Keras

Perangkat keras yang akan digunakan adalah laptop HP Pavilion Gaming 15 dengan spesifikasi sebagai berikut:

No	Nama Perangkat	Spesifikasi
1	<i>Processor</i>	Intel Core i7 Gen 9
2	RAM	16 GB
3	SSD	256 GB
4	HDD	1TB

3.4.2 Kebutuhan Perangkat Lunak

Perangkat lunak yang akan digunakan akan adalah sebagai berikut:

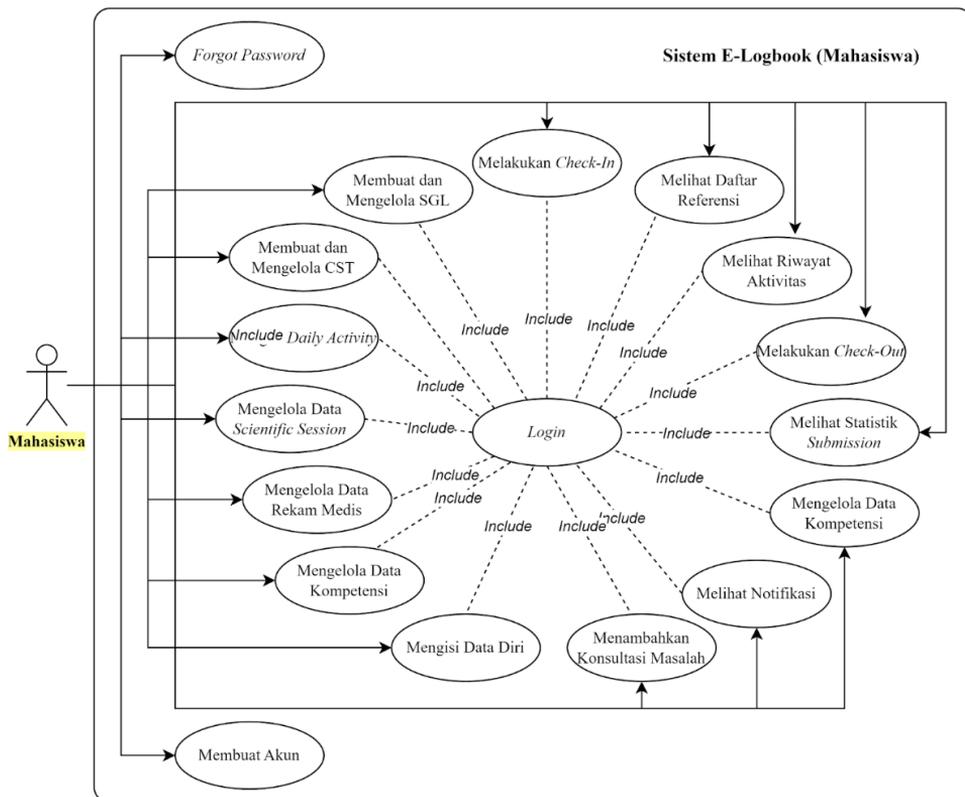
No	Nama Perangkat Lunak	Detail
1	Windows 11 Home 64 Bit	Sistem operasi
2	Visual Studio Code	Kode <i>editor</i>
3	Postman	<i>Tools</i> untuk <i>testing</i> API
4	Draw.io	<i>Tools</i> untuk membuat <i>diagram</i>
5	Brave	<i>Web Browser</i>
6	Figma	<i>Tools</i> untuk membuat perancangan <i>wireframe</i> dan <i>user interface</i>

2.5 Perancangan Sistem

Perancangan sistem penelitian ini menggunakan *use case diagram*. *Use case diagram* digunakan untuk menggambarkan interaksi antara sistem atau perangkat lunak dengan aktor-aktor eksternal atau pengguna.

2.5.1 *Use case diagram* role mahasiswa

Seperti pada Gambar 5 terdapat aktor mahasiswa. Aktor mahasiswa dapat melakukan registrasi akun, *forgot password*, dan *login*. Setelah *login* barulah mahasiswa bisa menggunakan fitur-fitur utama dari sistem. Beberapa aksi yang dapat dilakukan oleh aktor mahasiswa setelah *login* ialah mengisi data diri, melakukan *check in* dan *check out*, mengisi atau menambahkan *submission* (laporan *task* yang pernah dilakukan) ke dalam *E-Logbook*. Serta melihat referensi, riwayat, notifikasi, dan statistik diri.



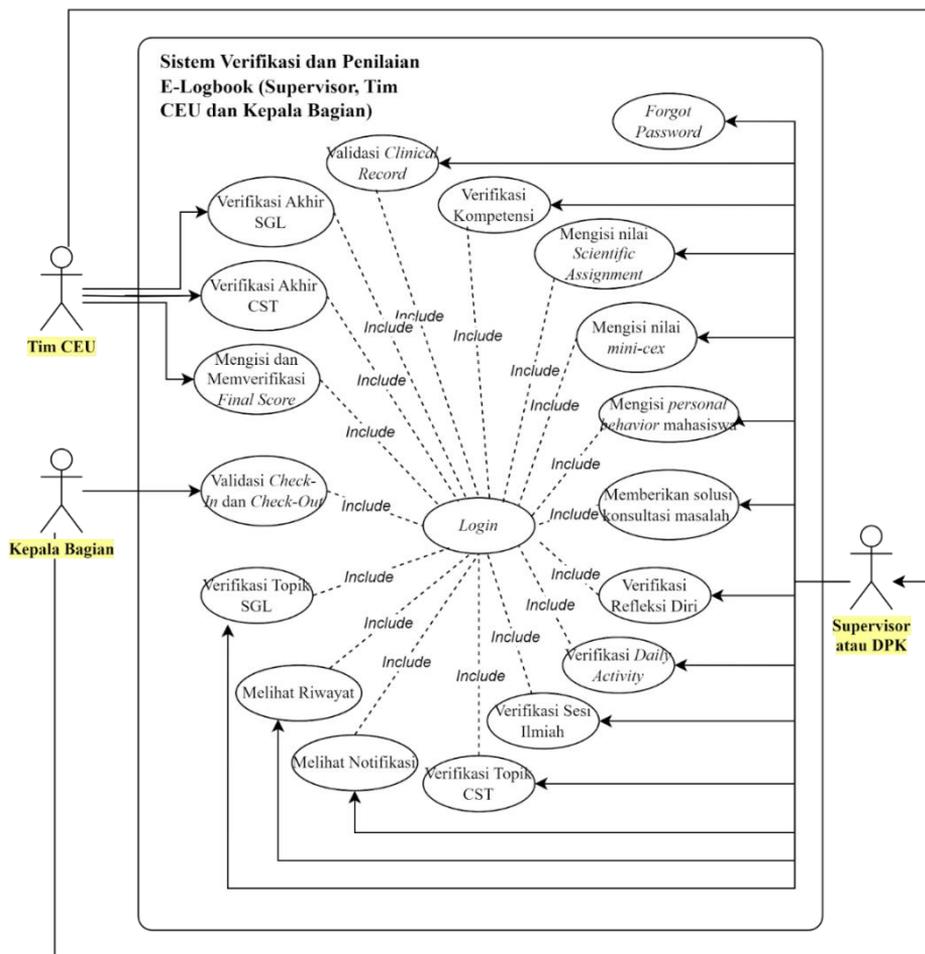
Gambar 5. Use Case Diagram Role Mahasiswa

2.5.2 Use case diagram role supervisor

Dalam *role* supervisor terdapat 3 aktor berbeda, hal ini dikarenakan pada *role* supervisor terdapat 2 *badge* tambahan yang dapat dimiliki yaitu tim CEU dan kepala bagian. Jadi, terdapat 4 kondisi untuk *role* supervisor yaitu supervisor, supervisor sekaligus CEU, supervisor sekaligus kepala bagian, dan supervisor, sekaligus CEU dan kepala bagian.

Supervisor perlu melakukan *login* terlebih dahulu sebelum melakukan aksi aksi lainnya. Terdapat beberapa aksi yang dapat dilakukan supervisor tanpa *badge*, aksi ini secara umum terdiri dari validasi task atau data yang ditambahkan mahasiswa dan melakukan penilaian kepada mahasiswa. aksi validasi yang dapat dilakukan diantaranya validasi topik SGL, validasi topik CST, validasi *daily activity*, validasi *clinical record*, validasi *scientific session*, validasi *self reflection*, dan validasi *competency*. Sedangkan penilaian yang dapat dilakukan supervisor tanpa *badge* yaitu menilai *scientific assignment*, menilai *mini-cex*, dan menilai *personal behavior*. Terakhir tentu saja aktor ini dapat mengakses riwayat kegiatan serta notifikasi jika terdapat *task* baru yang perlu dilakukan.

Untuk supervisor dengan *badge* terdapat aksi khusus yang dapat dilakukan yaitu menilai *final score*, validasi akhir SGL dan validasi akhir CST. Sedangkan supervisor dengan *badge* kepala bagian memiliki aksi tambahan selain aksi yang dapat dilakukan oleh supervisor yaitu melakukan validasi *check-in* dan validasi *check-out*. *Use case diagram* dari role supervisor dapat dilihat pada Gambar 6.

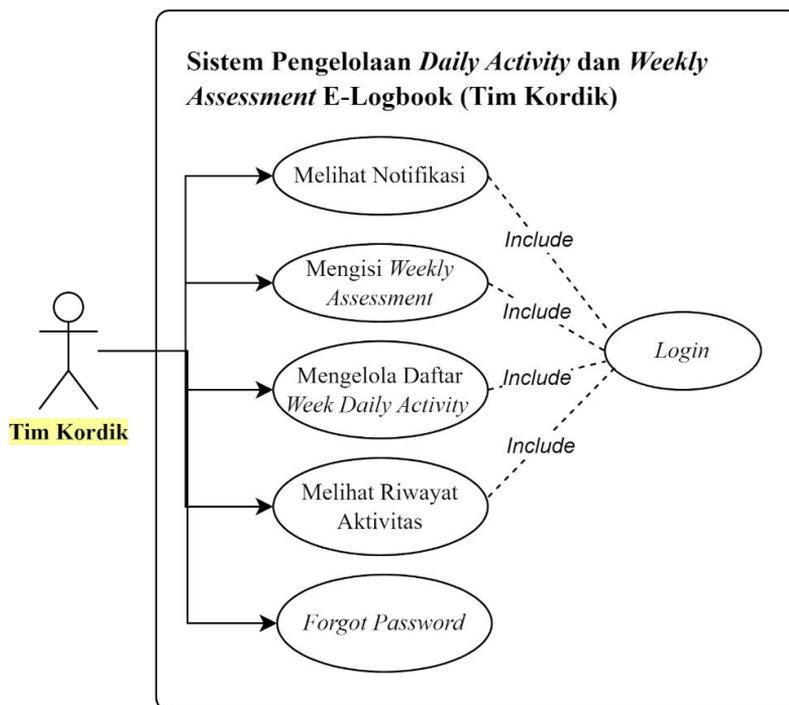


Gambar 6. Use Case Diagram Role Supervisor

2.5.3 Use case diagram role tim Koordinator Pendidikan (Kordik)

Seperti pada Gambar 7 terdapat aktor tim kordik. Tugas utama tim kordik adalah menambahkan minggu ke setiap departemen untuk keperluan *daily activity*, dan menilai *weekly assessment* (nilai mingguan). Tim kordik perlu *login* terlebih dahulu sebelum dapat mengakses menu utama *role* tim kordik pada aplikasi *E-Logbook*. Setelah *login* tim kordik dapat mengakses halaman untuk mengisi *weekly*

assessment, menambah minggu untuk tiap departemen, dan melihat *history* kegiatan yang pernah dilakukan. Selain itu tanpa tim kordik dapat mengakses fitur *forgot password* tanpa *login*.



Gambar 7. Use Case Diagram Role Tim Kordik

2.6 Perancangan User Interface

Perancangan *User Interface* (UI) merupakan bentuk tampilan visual yang berhubungan langsung dengan para pengguna agar mereka dapat dengan jelas memahami dan merasakan bagaimana fitur-fitur yang akan dihadirkan pada sistem akan bekerja. Dalam konteks penelitian ini, rancangan UI untuk *E-Logbook* MPPD berbasis *mobile* mencakup pengembangan *wireframe*. *Wireframe* ini berfungsi sebagai kerangka dasar untuk mengilustrasikan komponen-komponen utama, tata letak, dan navigasi yang akan ada dalam aplikasi *mobile* tersebut. Adapun rancangan UI berupa *wireframe* pada penelitian mengenai *E-Logbook* MPPD berbasis *mobile* adalah sebagai berikut.

2.6.1 Halaman *Login* dan *Register*

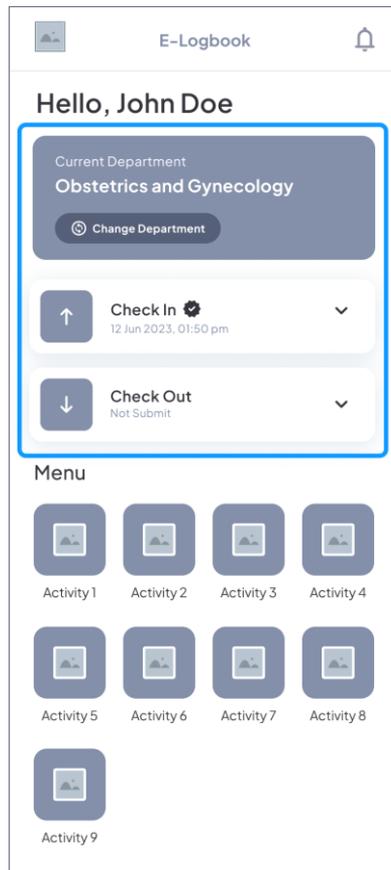
Halaman *login* merupakan antarmuka dari aplikasi yang dirancang khusus untuk memungkinkan pengguna untuk memasukkan informasi identifikasi pribadi mereka, seperti nama pengguna dan kata sandi, guna mengakses sistem atau layanan yang terproteksi. Komponen halaman *login* umumnya terdiri dari *form* untuk memasukkan nama pengguna atau id dan kata sandi, tombol masuk. Halaman *register* sendiri merupakan antarmuka yang digunakan untuk membuat akun baru. Komponen halaman *register* terdiri dari beberapa *form* untuk memasukkan data diri pengguna. Desain antarmuka halaman *login* dan *register* pada aplikasi *E-Logbook* dapat dilihat pada Gambar 8 berikut.

The image shows two wireframe designs side-by-side. The left design is for a 'Login' page, featuring a 'username' input field, a 'password' input field with a visibility toggle, a 'Forgot your password?' link, a 'Log In' button, and a 'Dont have an account? Register here' link. The right design is for a 'Sign Up' page, featuring a 'Sign Up' title, 'Username', 'Student ID', 'Fullname', 'Email', 'Password', and 'Confirm Password' input fields, a 'Sign Up' button, and an 'Already have an account? Login here' link. Blue arrows connect the 'Register here' link on the login page to the 'Sign Up' button on the register page, and the 'Login here' link on the register page to the 'Log In' button on the login page.

Gambar 8. Desain *Wireframe* Halaman *Login* dan *Register*

2.6.2 Halaman Menu Utama

Menu utama merupakan halaman yang terdiri dari daftar menu yang dapat diakses di aplikasi seperti menu *daily activity*, menu *competency* dan lainnya. Secara umum menu yang muncul di mahasiswa dan supervisor akan sama namun Gambar 9. Menu mahasiswa akan diarahkan ke pembuatan atau pengajuan *submission*. Sedangkan supervisor, kepala CEU, tim kordik, dan kepala bagian akan diarahkan ke halaman untuk verifikasi atau pemberian nilai. Desain *Wireframe* halaman menu utama dapat dilihat pada



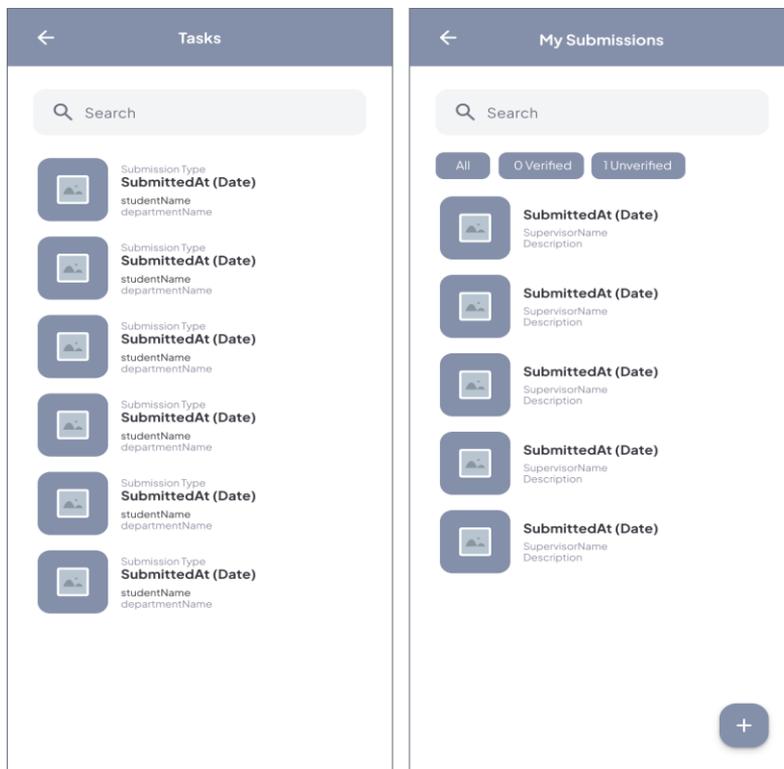
Gambar 9. Desain *Wireframe* Halaman Menu Utama

Komponen lain yang hanya ada di menu mahasiswa dapat dilihat pada bagian yang ditandai kotak biru pada Gambar 3.6. Komponen tersebut adalah bagian untuk melihat departemen yang sedang dijalani (aktif) dan dua *expansion tile* untuk melakukan permintaan *check in* dan *check out*.

2.6.3 Halaman Daftar *Task* atau *Submission*

Halaman daftar *task* dan *submission* merupakan dua halaman berbeda halaman daftar *task* merupakan halaman pada akun supervisor, dan tim CEU yang menampilkan daftar pengajuan mahasiswa yang perlu diverifikasi atau dinilai. Sedangkan halaman daftar *submission* merupakan halaman yang menampilkan *submission* yang pernah dikirim atau diajukan mahasiswa. Namun desain antarmuka halaman daftar *task* dan *submission* secara umum sama, komponennya terdiri dari bagian pencarian, dan daftar *card* (*task* atau *submission*). Perbedaannya hanya pada informasi yang diberikan di *card* serta terdapat komponen tambahan pada *submission* yaitu tombol untuk membuat *submission* baru dan 3 *chip* yang dapat

digunakan untuk melakukan *filter* dan memberikan informasi terkait jumlah *submission* yang telah diverifikasi dan belum.



Gambar 10. Desain *Wireframe* Halaman Menu Daftar *Task* atau *Submission*

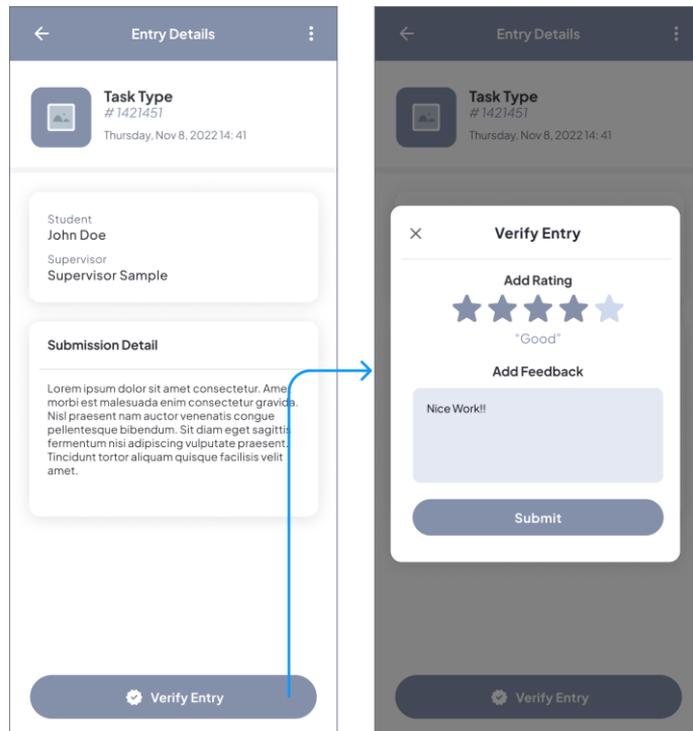
2.6.4 Halaman Tambah atau Sunting *Submission*

Desain halaman tambah atau sunting *submission* secara umum terdiri dari beberapa formulir untuk membuat atau mengubah *submission*. Terdapat beberapa tipe komponen formulir yang dapat digunakan seperti formulir *input* teks, *dropdown*, tombol *upload file*, dan lainnya. Perbedaan antara halaman-halaman *submission* hanyalah judul di *AppBar* dan kombinasi komponen formulir yang digunakan. Desain umum *wireframe* halaman tambah atau sunting *submission* oleh MPPD dapat dilihat pada Gambar 11.

Gambar 11. Desain *Wireframe* Halaman *Form Submission* MPPD

2.6.5 Halaman Detail *Submission* (Rekam Medis, Sesi Ilmiah, dan Refleksi Diri)

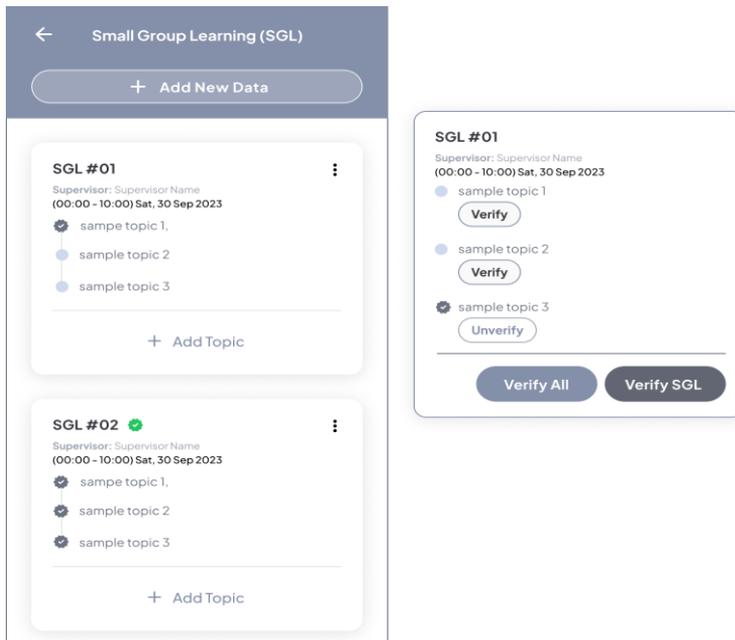
Halaman detail rekam medis, sesi ilmiah, dan refleksi diri memiliki desain antarmuka yang mirip. Komponen dari halaman tersebut terdiri dari bagian *header* yang memberikan informasi tentang jenis *task* atau *submission*, kemudian terdapat beberapa *section* yang memberikan detail informasi dari *submission*. Pada akun supervisor halaman ini memiliki komponen lain yaitu tombol verifikasi di pojok bawah yang akan membuka *pop up* untuk memverifikasi *submission* mahasiswa. Desain antarmuka halaman detail *submission* (rekam medis, sesi ilmiah, dan refleksi diri) dapat dilihat pada Gambar 11 berikut.



Gambar 12. Desain *Wireframe* Halaman Detail *Submission*

2.6.6 Halaman Detail SGL atau CST

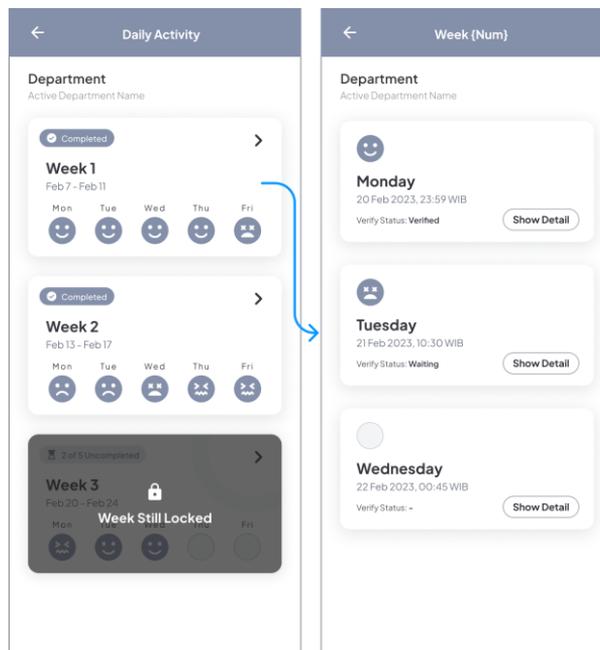
Halaman detail SGL dan CST memiliki desain antarmuka yang sama. Pada Gambar 12 menampilkan desain antarmuka halaman SGL. Komponen dari desain antarmuka halaman ini terdiri tombol tambah data dan daftar SGL atau CST yang pernah ditambahkan mahasiswa. Setiap *card* SGL atau CST memberikan informasi berupa status verifikasi, tombol untuk menghapus dan menyunting, jadwal SGL atau CST, daftar topik pada SQL atau CST tersebut, serta tombol untuk menambah topik. Untuk akun supervisor, dan tim CEU letak perbedaan antarmuka mulai dari tombol tambah SGL atau CST yang tidak ada dan tambahan komponen yang dapat dilihat pada gambar sebelah kanan yaitu terdapat tombol untuk memverifikasi topik dan memverifikasi SGL atau CST.



Gambar 13. Desain Wireframe Halaman Detail SGL

2.6.7 Halaman *Daily Activity*

Halaman *daily activity* atau kegiatan harian memiliki desain antarmuka seperti pada Gambar 14.

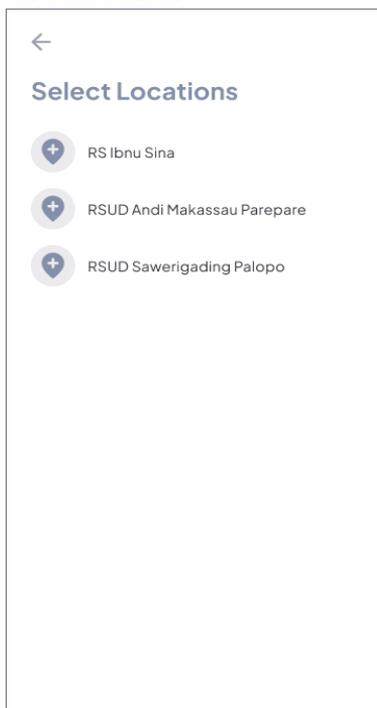


Gambar 14. Desain Wireframe Halaman Detail *Daily Activity*

Halaman *daily activity* memiliki terdiri dari 2 halaman berbeda. Halaman pertama merupakan halaman daftar minggu, yang komponennya terdiri dari nama departemen berjalan dan daftar *card* kegiatan per minggu. *Card* minggu ini memiliki informasi berupa status pengisian *daily activity* atau hari yang telah dilakukan presensi pada minggu tersebut. Jika salah satu *card* minggu ditekan halaman akan diarahkan ke detail kegiatan harian minggu tersebut. Halaman detail ini berisi daftar *card* yang memberikan informasi *daily activity* perharinya.

2.6.8 Halaman Pilih Lokasi

Halaman pilih lokasi merupakan halaman yang menampilkan daftar lokasi yang dapat dipilih untuk melakukan presensi pada *daily activity*. Halaman ini memiliki desain antarmuka seperti pada Gambar 15. Komponen dari halaman ini terdiri dari formulir pencarian dan daftar item lokasi.

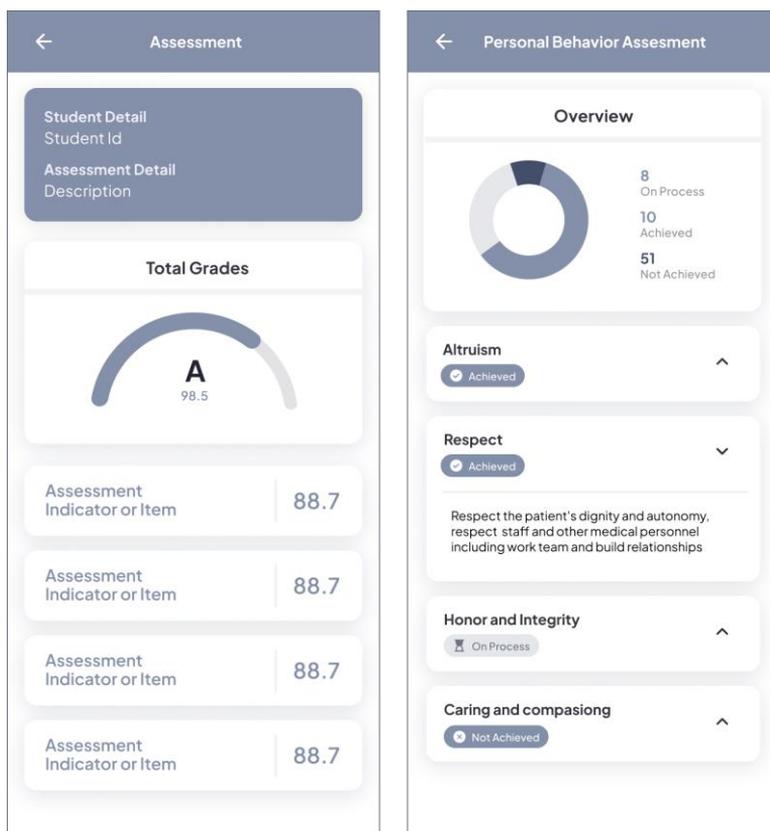


Gambar 15. Desain *Wireframe* Halaman Pilih Lokasi

2.6.9 Halaman *Assessment*

Halaman *assessment* merupakan halaman yang menampilkan informasi penilaian mahasiswa. Perbedaan halaman *assessment* mahasiswa dan supervisor hanya terletak pada akun supervisor yang memiliki akses untuk mengubah nilai mahasiswa. Pada Gambar 16 terlihat 2 jenis halaman *assessment* yang kiri merupakan desain antarmuka halaman *assessment* untuk penilaian sesi ilmiah, penilaian *mini-cex*,

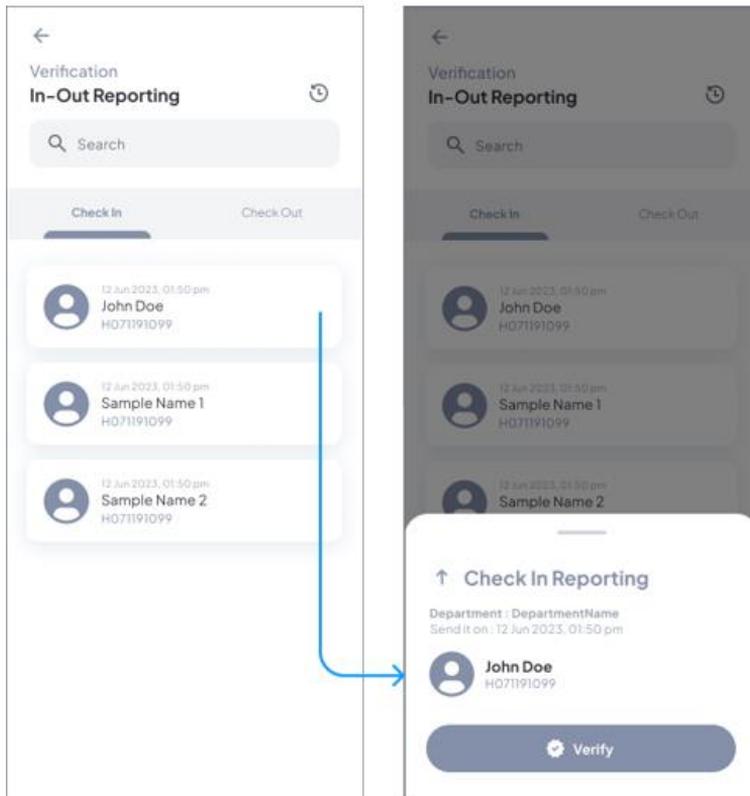
penilaian mingguan, penilaian *final score*. Sedangkan untuk gambar antarmuka yang kanan merupakan desain antarmuka halaman penilaian *personal behavior*. Secara umum komponen halaman penilaian terdiri dari informasi mahasiswa, statistik atau *overview* penilaian, dan daftar item penilaian.



Gambar 16. *Desain Wireframe Halaman Assessment*

2.6.10 Halaman Verifikasi *Check-In* dan *Check-Out*

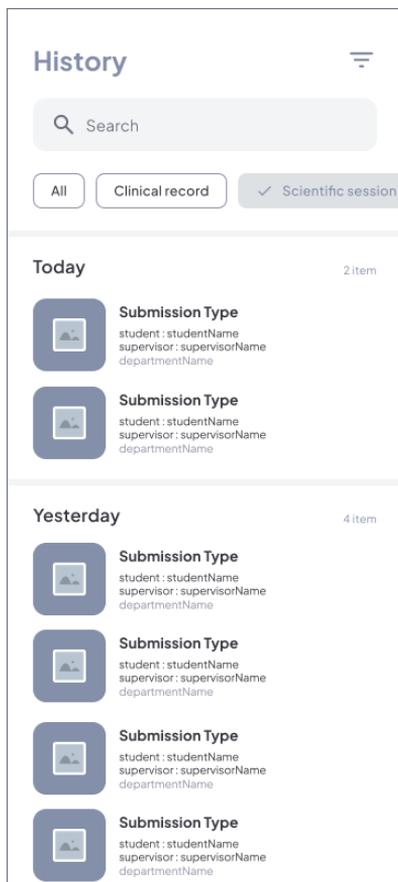
Halaman *check-in* dan *check-out* merupakan halaman yang terdapat pada akun tim kepala bagian untuk melihat daftar pengajuan *check-in* atau *check-out* dan melakukan verifikasi terhadap pengajuan tersebut. Desain antarmuka halaman ini dapat dilihat pada Gambar 17, halaman sebelah kiri merupakan *wireframe* yang menampilkan halaman daftar pengajuan *check-in* dan halaman sebelah kanan merupakan tampilan saat salah satu item pengajuan ditekan yang membuat *bottom sheet* untuk melakukan verifikasi *check-in* terbuka.



Gambar 17. Desain *Wireframe* Halaman Verifikasi *Check-In* dan *Check-Out*

2.6.11 Halaman Riwayat

Halaman riwayat merupakan halaman yang menampilkan daftar riwayat aktivitas yang dilakukan pengguna. Komponen antarmuka halaman ini terdiri dari formulir pencarian, daftar *chip* kategori aktivitas untuk melakukan *filter*, serta daftar *card* riwayat yang terpisah-pisah berdasarkan kategori waktu verifikasi atau pengajuan. Pada Gambar 18 menampilkan desain *wireframe* halaman riwayat aktivitas.



Gambar 18. Desain *Wireframe* Halaman Riwayat