

**PENGEMBANGAN SISTEM PENGELOLAAN KEGIATAN  
MAGANG MAHASISWA POLMAN NEGERI BABEL  
BERBASIS *ANDROID* DENGAN METODE *PROTOTYPE***

**PROYEK AKHIR**

Laporan akhir ini dibuat dan diajukan untuk memenuhi salah satu syarat kelulusan Sarjana Terapan/Diploma IV Politeknik Manufaktur Negeri Bangka Belitung



Disusun oleh:

Lita Rahmania

NIM: 1062216

**POLITEKNIK MANUFAKTUR NEGERI  
BANGKA BELITUNG  
TAHUN 2025**

**LEMBAR PENGESAHAN**

**PENGEMBANGAN SISTEM PENGELOLAAN KEGIATAN MAGANG  
MAHASISWA POLMAN NEGERI BABEL BERBASIS *ANDROID*  
DENGAN METODE *PROTOTYPE***

Oleh:

Lita Rahmania / 1062216

Laporan ini telah disetujui dan disahkan sebagai salah satu syarat kelulusan  
Program Sarjana Terapan/Diploma IV Politeknik Manufaktur Negeri Bangka  
Belitung

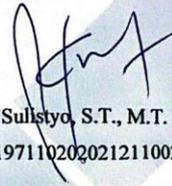
Menyetujui,

Pembimbing 1



Riki Afriansyah, S.T., M.T.  
NIP. 199004042019031013

Pembimbing 2



Eko Sulistyono, S.T., M.T.  
NIP. 197110202021211002

Penguji 1



Yang Agita Rindri, M. Eng  
NIP. 198609282022032003

Penguji 2



Zanu Saputra, M. Tr. T  
NIP. 1983110320140441001

## PERNYATAAN BUKAN PLAGIAT

Yang bertanda tangan di bawah ini:

Nama Mahasiswa : Lita Rahmania NIM : 1062216

Dengan Judul :Pengembangan Sistem Pengelolaan Kegiatan Magang Mahasiswa Polman Negeri Babel Berbasis *Android* Dengan Metode *Prototype*

Menyatakan bahwa laporan akhir ini adalah hasil kerja saya sendiri dan bukan merupakan plagiat. Pernyataan ini kami buat dengan sebenarnya dan bila ternyata dikemudian hari ternyata melanggar pernyataan ini, saya bersedia menerima sanksi yang berlaku.

Sungailiat, 03 Juli 2025

Lita Rahmania

## ABSTRAK

*Program Praktik Kerja Lapangan (PKL) atau magang di Politeknik Manufaktur Negeri Bangka Belitung dilaksanakan pada semester 6 untuk program Diploma III dan semester 8 untuk program Diploma IV. Namun, pelaksanaan dan pengelolaan magang masih dilakukan secara manual, yang sering menimbulkan berbagai kendala dalam hal administrasi, pemantauan, dan dokumentasi. Kondisi ini menyulitkan mahasiswa, dosen wali, mitra industri, serta komisi magang dalam berkoordinasi secara efisien. Penelitian ini bertujuan untuk mengembangkan sistem pengelolaan kegiatan magang berbasis aplikasi Android dengan pendekatan metode prototype. Sistem ini dirancang untuk memfasilitasi berbagai kebutuhan, seperti pencatatan aktivitas magang melalui logbook digital, penjadwalan sidang magang, serta unggah dan unduh dokumen secara langsung melalui aplikasi. Dengan desain antarmuka yang user-friendly dan akses yang fleksibel, aplikasi ini diharapkan mampu meningkatkan efisiensi dan efektivitas pengelolaan magang. Hasil pengembangan menunjukkan bahwa sistem yang dibangun berhasil memenuhi kebutuhan pengguna dalam mengelola aktivitas magang. Pengujian sistem menggunakan metode Black Box dan User Acceptance Testing (UAT) menunjukkan bahwa fitur-fitur utama berfungsi dengan baik, serta mendapat tanggapan positif dari pengguna. Aplikasi ini tidak hanya meningkatkan transparansi dan akuntabilitas, tetapi juga mengurangi risiko kehilangan data dan mempermudah proses kolaborasi antar pihak. Adapun hasil dari pengujian menyatakan sistem diterima dan berhasil membuktikan bahwa responden cenderung puas dan setuju bahwa sistem yang dikembangkan bermanfaat dan mudah digunakan dengan mendapatkan 85,4% pengujian sistem ini memberikan kontribusi positif dalam mendukung pelaksanaan magang yang lebih terstruktur dan profesional di Politeknik Manufaktur Negeri Bangka Belitung.*

*Kata kunci: Magang, Sistem Pengelolaan, Android, Prototype, Monitoring, Pembaruan.*

## ABSTRAC

*The Field Work Practice (PKL) or internship program at the Bangka Belitung State Manufacturing Polytechnic is carried out in semester 6 for Diploma III programs and semester 8 for Diploma IV programs. However, the implementation and management of internships are still carried out manually, which often causes various obstacles in terms of administration, monitoring, and documentation. This condition makes it difficult for students, guardian lecturers, industry partners, and internship commissions to coordinate efficiently. This research aims to develop an Android application-based internship management system with a prototype method approach. This system is designed to facilitate various needs, such as recording internship activities through digital logbooks, scheduling internship sessions, and uploading and downloading documents directly through the application. With a user-friendly interface design and flexible access, this application is expected to improve the efficiency and effectiveness of internship management. The development results show that the system built successfully meets the needs of users in managing internship activities. System testing using Black Box and User Acceptance Testing (UAT) methods showed that the main features functioned well, and received positive feedback from users. This application not only increases transparency and accountability, but also reduces the risk of data loss and simplifies the collaboration process between parties. The results of the test stated that the system was accepted and succeeded in proving that respondents tended to be satisfied and agreed that the developed system was useful and easy to use by getting 85.4% of testing. This system makes a positive contribution in supporting the implementation of a more structured and professional internship at the Bangka Belitung State Manufacturing Polytechnic.*

*Keywords: Internship, Management System, Android, Prototype, Monitoring, Update*

## KATA PENGANTAR

Segala puji dan syukur penulis panjatkan kepada Allah SWT atas segala berkat, rahmat, taufik serta hidayah-Nya yang tiada terkira, sehingga penulis dapat menyelesaikan laporan tugas akhir ini dengan judul “**Pengembangan Sistem Pengelolaan Kegiatan Magang Mahasiswa Polman Negeri Babel Berbasis Android Dengan Metode Prototype**”. Penyusunan tugas akhir ini merupakan salah satu prasyarat yang harus dipenuhi atau diselesaikan sebagai syarat untuk menyelesaikan pendidikan Diploma IV di Politeknik Manufaktur Negeri Bangka Belitung.

Ucapan terima kasih penulis kepada semua pihak yang telah membantu dalam menyelesaikan laporan tugas akhir ini sehingga dapat diselesaikan. Ucapan terima kasih penulis sampaikan kepada :

1. Ucapan syukur kepada Allah SWT.
2. I Made Andik Setiawan, M.Eng, Ph.D. selaku Direktur Politeknik Manufaktur Negeri Bangka Belitung.
3. Riki Afriansyah M.T selaku Dosen Pembimbing Utama Proyek Akhir.
4. Eko Sulisty, M.T selaku Dosen Pembimbing Kedua Proyek Akhir.
5. Yang Agita Rindri, S.Kom., M.Eng selaku Kepala jurusan informatika dan bisnis Politeknik Manufaktur Negeri Bangka Belitung.
6. Sidhiq Andriyanto, S.T., M.Kom selaku Kepala Prodi Diploma IV Teknologi Rekayasa Perangkat Lunak Politeknik Manufaktur Negeri Bangka Belitung.
7. Ucapan terima kasih yang tak terhingga penulis sampaikan kepada Ayah, Ibu, dan seluruh keluarga tercinta atas segala doa, kasih

sayang, dukungan moril maupun materil yang tiada henti, yang menjadi sumber kekuatan dan semangat dalam menyelesaikan Tugas Akhir ini.

8. Bapak dan Ibu seluruh Dosen Prodi Diploma IV Teknologi Rekayasa Perangkat Lunak, yang telah memberikan ilmu pengetahuan, mendidik, dan membimbing penulis selama perkuliahan.
9. Kepada mahasiswa 3 TRPL A yang telah membantu memberikan masukan dan bantuan dalam menyelesaikan Laporan Tugas Akhir ini.

Tentu penulis menyadari masih banyak kekurangan dalam Laporan Akhir ini, oleh karena itu kritik dan saran sangat diharapkan guna memperbaiki laporan ini sehingga menjadi lebih baik. Akhir kata penulis mengucapkan terima kasih, semoga laporan tugas akhir ini bermanfaat.

Sungailiat, 03 Juli 2025

Penulis

## DAFTAR ISI

LEMBAR PENGESAHAN .....	<b>Error! Bookmark not defined.</b>
PERNYATAAN BUKAN PLAGIAT .....	ii
ABSTRAK .....	iv
ABSTRAC .....	v
KATA PENGANTAR .....	vi
DAFTAR ISI.....	viii
DAFTAR TABEL.....	xi
DAFTAR GAMBAR .....	xii
DAFTAR LAMPIRAN.....	xiii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang Masalah.....	1
1.2 Rumusan Masalah .....	3
1.3 Batasan Masalah.....	3
1.4 Tujuan Proyek Akhir .....	4
BAB II DASAR TEORI .....	5
2.1 Tinjauan Pustaka .....	5
2.2 Sistem .....	21
2.3 Pengelolaan .....	22
2.4 Magang.....	22
2.5 <i>Android</i> .....	23
2.6 Metode Prototype .....	23
2.7 Teori Perancangan Sistem.....	24
2.7.1 <i>Unified Modeling Language (UML)</i> .....	24
2.7.2 <i>Use Case Diagram</i> .....	25
2.7.3 <i>Activity Diagram</i> .....	26
2.8 Metode Pengujian.....	27
2.8.1 ( <i>User Acceptance Testing</i> ) UAT .....	28
2.8.2 <i>Black Box Testing</i> .....	29

BAB III METODE PELAKSANAAN .....	30
3.1 Identifikasi Masalah .....	31
3.2 Studi Literatur.....	31
3.3 Pengumpulan Dan Analisis Kebutuhan Sistem.....	31
3.4 Perancangan Sistem.....	33
3.5 Pengembangan Sistem Menggunakan Metode <i>Prototype</i> .....	33
3.6 Pengujian, Pengumpulan dan Analisa data .....	36
3.6.1 Metode Pengujian .....	36
3.6.2 UAT dan Blackbox Testing .....	36
3.7 Metode Pengumpulan Dan Analisis Data.....	37
BAB IV PEMBAHASAN.....	38
4.1 Hasil Identifikasi Kebutuhan Sistem.....	38
4.1.2 Wawancara.....	38
4.1.3 Observasi .....	39
4.1.4 Studi Dokumen .....	39
4.2 Hasil Pembuatan <i>Prototype</i> .....	39
4.2.1 <i>Use Case Diagram</i> .....	42
4.2.2 <i>Activity Diagram</i> .....	44
4.2.3 <i>Entity Relationship Diagram</i> .....	47
4.2.4 Rancangan Antar Muka .....	48
4.3 Hasil Evaluasi <i>Prototype</i> .....	55
4.4 Hasil Pengkodean <i>Prototype</i> .....	56
4.4.1 Tampilan software .....	57
4.5 Hasil Evaluasi Sistem.....	71
4.6 Hasil Menguji Sistem .....	71
4.6.1 <i>Blackbox Testing</i> .....	71
4.6.2 <i>User Acceptance Testing (UAT)</i> .....	73
4.7 Penggunaan Sistem.....	76
BAB V KESIMPULAN DAN SARAN.....	77
5.1 Kesimpulan.....	77
5.2 Saran .....	77

DAFTAR PUSTAKA .....	78
DAFTAR RIWAYAT HIDUP.....	81
LAMPIRAN.....	82



## DAFTAR TABEL

Tabel 2. 1 Penelitian Sebelumnya.....	5
Tabel 2. 2 Simbol <i>Use Case</i> .....	26
Tabel 2. 3 Simbol <i>Activity Diagram</i> .....	27
Tabel 4. 1 <i>Blackbox Testing</i> Simagang.....	72
Tabel 4. 2 UAT Simagang .....	73
Tabel 4. 3 Tabel Perhitungan .....	75



## DAFTAR GAMBAR

Gambar 3. 1 Metode Pelaksanaan .....	30
Gambar 4. 1 Metode <i>Prototyping</i> .....	40
Gambar 4. 2 <i>Use Case Diagram</i> .....	43
Gambar 4. 3 <i>Activity Logbook</i> dan absen.....	45
Gambar 4. 4 <i>Activity Diagram</i> Sidang .....	46
Gambar 4. 5 <i>Entity Relationship Diagram</i> .....	48
Gambar 4. 6 Rancang Antar Muka <i>splash</i> .....	49
Gambar 4. 7 Rancang Antar Muka <i>login</i> .....	50
Gambar 4. 8 Rancang Antar Muka <i>Dashboard</i> Mahasiswa .....	51
Gambar 4. 9 <i>Dashboard</i> Penguji.....	52
Gambar 4. 10 <i>Dashboard</i> Mitra .....	53
Gambar 4. 11 <i>Dashboard</i> Dosen Wali .....	54
Gambar 4. 12 <i>Dashboard</i> Admin.....	55
Gambar 4. 13 Tampilan <i>Software (Home)</i> .....	<b>Error! Bookmark not defined.</b>
Gambar 4. 14 Tampilan <i>Software (Login)</i> .....	<b>Error! Bookmark not defined.</b>
Gambar 4. 15 Tampilan <i>Software (Dashboard</i> Dosen Wali)	<b>Error! Bookmark not defined.</b>
Gambar 4. 16 Tampilan <i>Software (Dashboard</i> Admin)	<b>Error! Bookmark not defined.</b>
Gambar 4. 17 Tampilan <i>Software (Dashboard</i> Mitra)	<b>Error! Bookmark not defined.</b>
Gambar 4. 18 Tampilan <i>Software (Dashboard</i> Mahasiswa)	<b>Error! Bookmark not defined.</b>
Gambar 4. 19 Tampilan <i>Software (Dashboard</i> Mahasiswa)	<b>Error! Bookmark not defined.</b>

## DAFTAR LAMPIRAN

Lampiran 1 Daftar Riwayat Hidup Penulis.....	81
Lampiran 2 Sample Jawaban Kusioner.....	82
Lampiran 3 Uji <i>blackbox</i> .....	84



# BAB I PENDAHULUAN

## 1.1 Latar Belakang Masalah

Magang merupakan salah satu program penting yang diselenggarakan oleh institusi pendidikan untuk meningkatkan kompetensi, pengalaman kerja, serta keterampilan soft skill mahasiswa sebelum mereka terjun ke dunia industri secara langsung. Program ini menjadi bagian dari kurikulum akademik yang bertujuan untuk mengintegrasikan teori yang diperoleh di bangku perkuliahan dengan praktik di lapangan. Mahasiswa yang menempuh pendidikan di Politeknik Manufaktur Negeri Bangka Belitung juga diwajibkan untuk mengikuti kegiatan magang, yang dilaksanakan pada semester enam untuk program Diploma III dan semester delapan untuk program Diploma IV. Dalam proses pelaksanaan magang, mahasiswa harus melalui beberapa tahapan administrasi yang melibatkan pihak Komisi Magang, mulai dari pengajuan tempat magang hingga tahap akhir berupa sidang evaluasi. Saat ini, seluruh proses tersebut masih dilakukan secara konvensional. Mahasiswa diwajibkan untuk datang langsung ke Komisi Magang guna menyampaikan informasi terkait tempat magang yang diinginkan, serta menyerahkan dokumen-dokumen pendukung [2]. Sistem konvensional ini tidak hanya memakan waktu dan tenaga, tetapi juga memiliki berbagai kelemahan seperti keterbatasan akses informasi, ketidakteraturan data, serta risiko kehilangan dokumen penting.

Berdasarkan observasi, penulis mengidentifikasi sejumlah permasalahan yang kerap muncul selama proses pengelolaan magang. Di antaranya adalah kesulitan mahasiswa dalam mendapatkan informasi terkini mengenai prosedur pengajuan magang, proses verifikasi yang berjalan lambat, serta risiko kehilangan dokumen fisik. Selain itu, pihak akademik juga menyatakan bahwa hingga saat ini belum tersedia sistem informasi berbasis *website* maupun *aplikasi* yang dapat digunakan untuk mendukung administrasi magang secara *digital*. Situasi ini menyebabkan seluruh proses masih dijalankan secara manual, yang berdampak pada rendahnya

efisiensi serta akurasi dalam pengelolaan data magang mahasiswa. Masalah serupa juga ditemukan dalam beberapa penelitian terdahulu. Ketika sistem pengelolaan kegiatan magang belum berbasis digital, sering terjadi kesulitan dalam mencari informasi, kehilangan data, dan terjadinya duplikasi dokumen [3].

Studi lain menunjukkan bahwa perancangan sistem informasi pemantauan magang dapat membantu mengatasi tantangan-tantangan tersebut dengan mengurangi aktivitas administratif konvensional, serta menjadikan informasi lebih terpusat dan mudah diakses [6]. Menyikapi situasi tersebut, diperlukan suatu solusi yang berbentuk aplikasi untuk mengelola kegiatan magang secara *digital*, efisien, dan terintegrasi. Pemilihan aplikasi berbasis Android dianggap tepat karena dapat menyediakan akses yang mudah bagi pengguna dari berbagai tempat. Diharapkan aplikasi ini dapat membantu mahasiswa dalam mendapatkan informasi magang dengan cepat, mengupload dokumen persyaratan, dan mencatat aktivitas harian lewat jurnal.

Selain itu, *aplikasi* ini juga mempermudah Komisi Magang dan dosen pembimbing dalam melakukan pengawasan, validasi data, serta penjadwalan sidang magang. Fitur-fitur inti yang dirancang mencakup jurnal harian mahasiswa, pengingat jadwal, dan kemampuan untuk mengupload serta mendownload dokumen langsung melalui *aplikasi*. Dalam penelitian sebelumnya, sistem pengelolaan magang umumnya masih dikembangkan dalam bentuk website dan belum tersedia dalam versi aplikasi *android*, padahal *aplikasi mobile* menawarkan fleksibilitas yang lebih tinggi bagi mahasiswa dan pihak terkait [4]. Maka dari itu, pada penelitian ini dikembangkan sistem berbasis *Android* yang dilengkapi dengan fitur-fitur tambahan yang mendukung kebutuhan pengguna secara menyeluruh. Pengembangan sistem ini menggunakan metode *prototype*, yaitu pendekatan dalam pengembangan perangkat lunak yang menekankan pada pembuatan model awal dari sistem untuk mendapatkan umpan balik dari pengguna secara cepat.

Metode ini memungkinkan pengembangan sistem dilakukan secara iteratif dan adaptif terhadap kebutuhan pengguna [5]. *Prototype* bukanlah sistem akhir, melainkan model yang terus diuji, dievaluasi, dan disesuaikan hingga mencapai hasil yang diinginkan [11]. Dengan metode ini, sistem yang dirancang diharapkan

mampu memenuhi seluruh kebutuhan proses magang secara *digital*, mempermudah kolaborasi antara mahasiswa, dosen wali, dan Komisi Magang, serta meningkatkan efektivitas dan efisiensi dalam pengelolaan kegiatan magang di lingkungan Politeknik Manufaktur Negeri Bangka Belitung.

## 1.2 Rumusan Masalah

Berdasarkan uraian pada latar belakang masalah, dirumuskan pokok permasalahan yaitu:

1. Bagaimana merancang dan mengembangkan sistem informasi berbasis aplikasi *android* yang dapat mendukung proses administrasi dan pemantauan kegiatan magang mahasiswa di Politeknik Manufaktur Negeri Bangka Belitung?
2. Bagaimana sistem ini dapat mempermudah mahasiswa dalam mengakses informasi magang, mengunggah dokumen persyaratan, serta mencatat aktivitas magang secara *digital* melalui jurnal harian?
3. Bagaimana penerapan metode *prototype* dapat digunakan dalam proses pengembangan sistem untuk menghasilkan *aplikasi* yang sesuai dengan kebutuhan pengguna?

## 1.3 Batasan Masalah

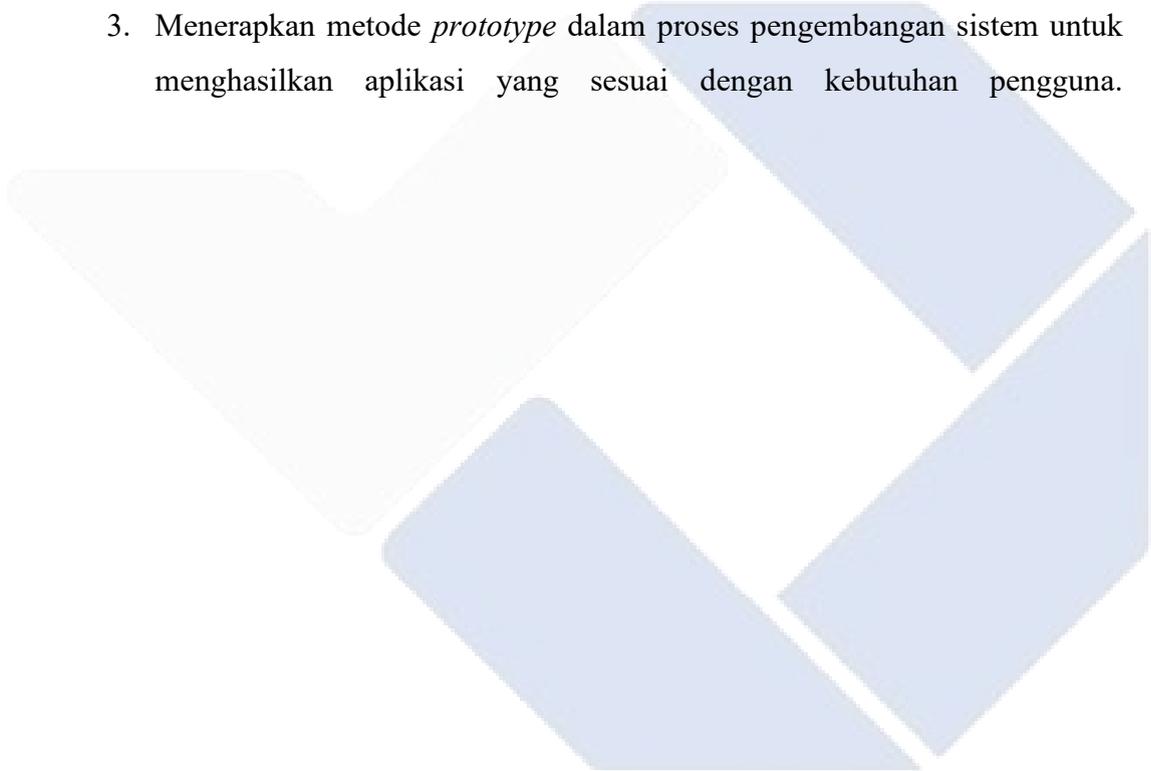
Untuk memastikan penelitian ini lebih terarah dan sesuai dengan ruang lingkup yang dapat dikaji secara mendalam, penelitian ini menetapkan beberapa batasan masalah:

1. Fitur utama yang dikembangkan dalam aplikasi ini mencakup jurnal harian mahasiswa, fasilitas untuk mengunggah dan mengunduh dokumen.
2. Pengembangan sistem dilakukan dalam bentuk aplikasi berbasis *Android* dengan menerapkan metode *prototype*, guna memastikan proses perancangan dapat disesuaikan secara iteratif berdasarkan kebutuhan dan umpan balik dari pengguna.

## 1.4 Tujuan Proyek Akhir

Adapun tujuan proyek akhir ini sebagai berikut:

1. Mengembangkan dan mengimplementasikan sistem informasi berbasis *Android* untuk mendukung proses administrasi dan pemantauan kegiatan magang mahasiswa di Politeknik Manufaktur Negeri Bangka Belitung.
2. Menerapkan teknologi *mobile* untuk mempermudah mahasiswa dalam mengakses informasi magang, mengunggah dokumen persyaratan, serta mencatat aktivitas magang secara digital melalui jurnal harian.
3. Menerapkan metode *prototype* dalam proses pengembangan sistem untuk menghasilkan aplikasi yang sesuai dengan kebutuhan pengguna.



## BAB II DASAR TEORI

### 2.1 Tinjauan Pustaka

Tinjauan pustaka ini difokuskan pada berbagai penelitian terdahulu yang memiliki keterkaitan erat dengan topik yang diangkat dalam penelitian ini. Bagian ini menyajikan ringkasan dari sejumlah hasil studi sebelumnya yang relevan, yang digunakan sebagai dasar dan acuan dalam mengembangkan penelitian lebih lanjut:

Tabel 2. 1 Penelitian Sebelumnya

No	Judul	Hasil
1.	Aplikasi pengelolaan magang berbasis web menggunakan metode prototyping pada pt. Glory industrial semarang	Verdinata & Evanata (2023) membahas tentang pengembangan sistem informasi berbasis <i>web</i> untuk membantu proses pengelolaan mahasiswa magang di PT. Glory Industrial Semarang yang selama ini masih dilakukan secara manual dan kurang efisien. Penelitian ini menghasilkan sebuah aplikasi dengan fitur masuk, <i>dashboard</i> , menambah berkas, manajemen nilai, dan pencetakan laporan, yang dikembangkan menggunakan <i>Framework CodeIgniter</i> dan metode pengembangan <i>prototyping</i> guna menyesuaikan kebutuhan pengguna secara bertahap. Output dari aplikasi ini dapat memudahkan perusahaan dalam mencatat, memilih, serta mengawasi

		<p>mahasiswa magang secara terorganisir dan efektif. Keunggulan jurnal ini terletak pada topik yang relevan, pendekatan prototyping yang adaptif, pemanfaatan teknologi terkini, serta desain sistem yang mencakup berbagai proses krusial dalam pengelolaan magang. Akan tetapi, jurnal ini juga memiliki kelemahan, seperti susunan yang kurang rapi, penjelasan teknis yang tidak mendalam, penyajian gambar yang kurang teratur, dan ketiadaan penilaian atau tanggapan dari pengguna akhir terhadap sistem yang dibuat.</p>
2.	<p>Sistem pengelolaan kegiatan magang mahasiswa polman negeri babel</p>	<p>Umayu Alakbar Yusa (2022) membahas tentang pelaksanaan kegiatan magang mahasiswa, yaitu sistem yang masih bersifat konvensional sehingga menimbulkan berbagai hambatan seperti hilangnya data, penggandaan dokumen, dan kesulitan dalam memperoleh informasi mengenai proses magang. Melalui penerapan metode <i>prototyping</i>, penulis menciptakan dan mengembangkan Sistem Manajemen Kegiatan Magang Mahasiswa berbasis <i>web</i> yang mencakup berbagai fitur penting seperti permohonan magang, pengunggahan dokumen, pemantauan, pelaporan, evaluasi, dan pengumuman jadwal sidang. Sistem ini telah diuji</p>

	<p>dengan metode <i>blackbox</i> dan beta testing, dan hasilnya menunjukkan bahwa semua fungsi sistem beroperasi 100% sesuai harapan, serta memperoleh tingkat kepuasan pengguna sebesar 88%, yang menandakan bahwa sistem sangat cocok untuk digunakan. Keunggulan jurnal ini terletak pada ruang lingkup fitur sistem yang sangat komprehensif dan sesuai dengan kebutuhan akademik, pengaturan data yang sistematis, serta penilaian sistem yang dilaksanakan secara menyeluruh dan didukung dengan data kuantitatif yang tepat. Jurnal ini melibatkan pemahaman yang mendalam mengenai proses bisnis magang yang rumit dan berhasil menyatukannya dalam satu sistem informasi yang terintegrasi. Akan tetapi, kekurangannya terletak pada penggunaan bahasa yang sangat deskriptif dan panjang di beberapa bagian, yang mungkin membingungkan pembaca awam dan membuat aspek teknis yang krusial menjadi kurang terlihat. Di samping itu, walaupun sistem telah diuji, jurnal tidak memberikan penjelasan rinci mengenai skenario pengguna atau tantangan nyata saat menerapkan sistem di lingkungan kampus, yang seharusnya dapat memperkuat analisis efektivitas sistem.</p>
--	--

3.	Desain dan Analisa Sistem Magang di Prodi Teknik Informasi Universitas Widya Kartika Menggunakan UML	<p>Trisno &amp; Hari (2021) membahas tentang perancangan sistem magang di lingkungan Prodi Teknik Informatika Universitas Widya Kartika dengan pendekatan <i>Unified Modeling Language</i> (UML) sebagai alat bantu utama dalam proses analisis dan desain sistem. UML diterapkan untuk menyusun rancangan sistem informasi magang dengan membuat class diagram, sequence diagram, dan activity diagram. Studi ini bertujuan untuk merancang struktur sistem magang secara komprehensif, meliputi entitas mahasiswa, dosen pembimbing, perusahaan mitra, serta dokumen penilaian, yang dibuat secara modular dan selaras dengan prinsip OOP (<i>Object Oriented Programming</i>). Outputnya adalah model sistem magang yang siap dijalankan dalam bentuk <i>aplikasi digital</i> dengan struktur basis data dan proses yang telah dirancang secara terintegrasi. Kelebihan jurnal ini terletak pada dokumentasi desain sistem yang sangat detail dan menyeluruh, sehingga sangat mendukung proses pengembangan sistem ke tahap implementasi teknis, serta penggunaan UML yang sesuai untuk menyusun kompleksitas proses magang menjadi lebih teratur. Di</p>

		<p>samping itu, jurnal ini menyajikan pemahaman yang jelas mengenai hubungan antara entitas serta tata cara administratif yang diterapkan di lembaga. Namun, kelemahan yang cukup mencolok adalah kurangnya hasil implementasi atau pengujian sistem secara langsung, sehingga efektivitas desain hanya dianggap dari rancangan di atas kertas. Artikel ini juga memiliki pendekatan yang lebih teoretis dan tidak menyediakan data empiris atau tanggapan dari pengguna akhir, sehingga memerlukan tindak lanjut berupa pengembangan praktis agar manfaatnya dapat diuji secara nyata di lapangan.</p>
4.	<p>Pembangunan Aplikasi E-Magang Perguruan Tinggi dengan Memanfaatkan <i>SDLC SCRUM</i> Pada <i>Agile Project Management</i></p>	<p>Kustanto et al. (2021) membahas tentang pengembangan aplikasi e-Magang berbasis web melalui metode SDLC Scrum yang digabungkan dengan pendekatan manajemen proyek Agile untuk mengatasi kompleksitas administrasi magang di lingkungan universitas dengan jumlah mahasiswa yang banyak. Hasil pengembangan menunjukkan sistem dapat mengelola seluruh proses magang, mulai dari pengajuan proposal, penjadwalan pembimbing, pencetakan surat pengantar, sampai laporan magang dalam format digital dan terintegrasi, dengan memanfaatkan backlog dan sprint secara efisien dalam Scrum. Keunggulan jurnal ini</p>

		<p>terletak pada pendekatan terstruktur dengan metode pengembangan mutakhir yang sesuai untuk proyek skala besar, serta penggunaan antarmuka yang informatif bagi seluruh pemangku kepentingan. Akan tetapi, kelemahan terdapat pada kurangnya evaluasi terhadap penerapan sistem secara real di lapangan, serta tidak adanya ukuran kuantitatif untuk menguji performa yang menilai efektivitas sistem secara komprehensif.</p>
5.	<p>Pengembangan website absensi mahasiswa magang menggunakan metode prototype pada pt. Pertamina (persero) refinery unit iii plaju</p>	<p>Candra, Jemakmum, Wijaya, Udariansyah, &amp; Wydyanto, (2024) membahas tentang pengembangan sistem absensi <i>digital</i> bagi mahasiswa magang di PT Pertamina RU III Plaju sebagai jawaban atas masalah absensi manual yang tidak efektif dan rentan manipulasi. Dengan memanfaatkan pendekatan prototyping dan teknologi PHP-Laravel, sistem ini dibuat untuk menyimpan data kehadiran, profil mahasiswa, dan pembimbing secara digital serta memungkinkan proses pemantauan yang lebih tepat. Sebagai hasilnya, platform absensi ini terbukti dapat meningkatkan efisiensi pencatatan, keakuratan data kehadiran, dan pelaporan secara langsung. Keunggulan jurnal ini terletak pada keberhasilan penerapan sistem yang berlandaskan kebutuhan aktual di dunia</p>

		<p>industri serta keluwesan metode <i>prototyping</i> yang memungkinkan adanya iterasi dan umpan balik dari pengguna. Akan tetapi, jurnal ini memiliki kelemahan pada terbatasnya analisis keamanan sistem dan tidak dilengkapi dengan data kuantitatif hasil pengujian pengguna yang lebih mendalam sebagai pendukung efektivitas sistem.</p>
6.	Perancangan sistem informasi monitoring magang	<p>Yulianto &amp; Firdaus (2021) membahas tentang pengembangan sistem informasi pemantauan magang yang terdiri dari platform <i>website</i> dan aplikasi <i>mobile</i> untuk menggantikan cara manual dalam melaporkan kehadiran dan aktivitas harian mahasiswa, serta dilengkapi dengan integrasi teknologi GPS untuk memastikan validitas lokasi kehadiran. Sistem ini dirancang untuk mengkonsolidasikan informasi magang sehingga dapat diakses secara langsung oleh pengawas, serta mendukung pembuatan laporan magang berbasis aktivitas yang terekam secara digital. Temuan penelitian menunjukkan bahwa desain sistem dapat mendukung proses magang dengan lebih teratur dan efisien, sekaligus mengurangi peluang untuk manipulasi data. Kelebihan dari jurnal ini adalah inovasi penggunaan GPS untuk</p>

		<p>otentikasi kehadiran dan pendekatan ganda (<i>web</i> dan <i>mobile</i>) yang meningkatkan aksesibilitas bagi pengguna. Di sisi lain, kelemahannya terletak pada fase implementasi yang belum sepenuhnya dilaksanakan dan belum ada validasi melalui pengujian fungsional yang melibatkan pengguna akhir untuk menilai efektivitas sistem secara nyata.</p>
7.	<p>Penerapan Metode agile pada development aplikasi pengelola data magang berbasis web menggunakan framework Laravel.</p>	<p>Ramadhan et al. (2023) membahas tentang pendekatan Agile Scrum untuk mengembangkan <i>aplikasi web</i> pengelolaan data magang yang berbasis Laravel di Fakultas Ilmu Komputer Universitas Singaperbangsa Karawang untuk memenuhi kebutuhan pengelolaan data magang yang belum terautomasi. Aplikasi ini dibuat dalam waktu 21 hari melalui lima tahap sprint, yang menghasilkan fitur utama seperti pengajuan rekomendasi, manajemen logbook, penjadwalan pembimbing, serta pelaporan dan penilaian untuk magang. Keunggulan jurnal ini terletak pada penerapan metode Scrum yang mendalam dan terorganisir, disertai dengan dokumentasi visual dari setiap tahap sprint, serta suksesnya penggabungan berbagai fitur magang ke dalam satu sistem terpadu. Namun,</p>

		kelemahannya terletak pada kurangnya pengujian sistem secara langsung oleh pengguna <i>end-user testing</i> , serta tidak adanya analisis mengenai dampak penerapan sistem terhadap efisiensi kerja dari pihak-pihak yang menggunakan sistem tersebut.
8.	Pengembangan aplikasi magang mahasiswa untuk membantu proses administrasi	Sopiah & Al-Maudina (2022) membahas pembuatan aplikasi sistem informasi untuk magang mahasiswa di Universitas Bina Darma, yang bertujuan mengatasi masalah administratif dalam pendaftaran dan pencatatan Program Magang Mahasiswa Bersertifikat (PMMB) yang sebelumnya dilakukan secara manual dan tradisional. Penelitian ini mengadopsi pendekatan Web Engineering, melalui tahap komunikasi dengan pelanggan, perencanaan, pemodelan, konstruksi, hingga penerapan, yang menghasilkan sistem berbasis web yang dapat diakses oleh berbagai kalangan, termasuk Kepala Program Studi, dosen, mahasiswa, serta Kepala Unit Alumni dan Tracer Study. Dengan demikian, aplikasi ini memfasilitasi pengelolaan data magang secara teratur dan efektif, mempercepat pendaftaran di <i>website</i> PMMB FHCI sekaligus mengurangi kemungkinan kesalahan dalam memasukkan data mahasiswa. Keunggulan jurnal ini

		<p>terletak pada partisipasi berbagai pihak dalam proses pengembangan, penggunaan metodologi yang tepat, dan kapasitas sistem untuk meningkatkan transparansi serta kemudahan akses informasi tentang magang. Di samping itu, aplikasi ini dapat meningkatkan kesempatan mahasiswa untuk berpartisipasi dalam program PMMB dengan cara yang lebih adil dan terperinci. Walaupun demikian, kekurangan dari penelitian ini adalah terbatasnya pengujian sistem dalam konteks nyata, serta belum dilakukannya evaluasi lengkap mengenai pengalaman pengguna dan peningkatan kinerja secara kuantitatif setelah sistem diterapkan, yang bisa jadi penting untuk menyempurnakan solusi berbasis sistem informasi ke depannya.</p>
9.	<p>Rancang Bangun aplikasi pendaftaran peserta magang berbasis web studi kasus CV sinar pluit</p>	<p>Asri dan Christiawan (2023) membahas tentang pengembangan aplikasi registrasi magang berbasis web untuk mengatasi masalah proses seleksi manual di CV Sinar Pluit, yang selama ini bergantung pada pertemuan langsung dan pengumpulan dokumen fisik. Output dari riset ini adalah lahirnya sistem informasi berbasis web dengan penggunaan PHP dan framework CodeIgniter yang dapat mengelola pendaftaran, catatan harian,</p>

		<p>kuisisioner, serta status peserta magang secara efektif. Keuntungan dari sistem ini adalah potensinya dalam mempermudah proses bisnis magang, menurunkan risiko kehilangan data, dan mendukung otomatisasi tugas administratif, terutama dalam situasi yang tidak memungkinkan untuk bertemu secara langsung. Namun, kelemahannya terletak pada belum adanya penerapan fitur pelaporan evaluatif yang dinamis dan masih terbatasnya integrasi dengan sistem pendukung lain, seperti penjadwalan pembimbing atau notifikasi secara real-time.</p>
10.	<p>Rancang bangun sistem informasi monitoring kemajuan naskah dan Lokasi magang mahasiswa program studi sastra inggris uniryu menggunakan teknologi location based service</p>	<p>Ordiyasa et al. (2021) membahas tentang pengembangan sistem pemantauan kemajuan laporan serta lokasi magang mahasiswa Prodi Sastra Inggris UNRIYO dengan memanfaatkan teknologi Location Based Service (LBS) berbasis Android. Sistem ini dirancang untuk mengatasi masalah keterlambatan dalam penyelesaian laporan magang serta penyalahgunaan lokasi magang oleh mahasiswa. Pengembangan menunjukkan bahwa aplikasi ini mampu melacak posisi nyata mahasiswa saat menjalani kegiatan magang serta mengawasi kemajuan penulisan laporan</p>

		<p>secara digital, didukung oleh verifikasi dari dosen pembimbing dan supervisor lapangan. Keunggulannya adalah penggunaan teknologi GPS untuk akurasi pelacakan lokasi yang meningkatkan transparansi, serta proses verifikasi digital yang terstruktur antara mahasiswa, pembimbing, dan prodi. Kekurangan aplikasi adalah belum terintegrasinya dalam satu platform menyeluruh antara mahasiswa dan dosen, serta belum terdapat analisis performa sistem dari segi pengalaman pengguna maupun beban sistem pada penggunaan berskala besar.</p>
11.	Rancang bangun aplikasi monitoring kegiatan menggunakan prototype	<p>Aditya et al (2021) membahas tentang merancang dan mengembangkan aplikasi pemantauan aktivitas berbasis Android dengan pendekatan metode prototyping, yang mencakup tahap komunikasi, desain cepat, pembangunan prototipe, dan evaluasi serta perbaikan sistem. Tujuan utama aplikasi ini adalah untuk memantau kegiatan kelompok atau individu secara real-time menggunakan teknologi Global Positioning System (GPS) yang diterapkan dalam fitur pemantauan peta, serta pengelolaan absensi dan daftar kehadiran anggota. Temuan dari penelitian ini menunjukkan</p>

		<p>bahwa sistem dapat menggantikan cara monitoring manual yang memakan waktu dan tenaga banyak, serta meningkatkan ketepatan dalam pencatatan data aktivitas. Aplikasi ini memanfaatkan Firebase Realtime Database untuk penyimpanan data secara online, menggunakan JSON untuk parsing data dan Android Studio sebagai platform pengembangan. Kelebihan jurnal ini terletak pada penerapan teknologi Location-Based Services (LBS) yang sesuai dengan kebutuhan pemantauan terkini serta fleksibilitas sistem yang mendukung aktivitas berbasis lokasi. Di samping itu, desain struktur yang dibuat dengan UML menjelaskan alur interaksi antara pengguna dan sistem. Namun, kelemahan yang cukup mencolok dari penelitian ini adalah tidak adanya pengujian sistem oleh pengguna akhir dalam konteks nyata, sehingga belum bisa diyakini apakah fitur yang dibuat dapat memenuhi kebutuhan pengguna secara keseluruhan. Selain itu, jurnal tidak menyertakan metrik untuk mengevaluasi efektivitas atau tingkat kepuasan pengguna yang dapat mendukung argumen keberhasilan sistem.</p>
--	--	---

12.	<p>Pengembangan aplikasi pengelolaan data siswa berbasis android menggunakan metode prototype</p>	<p>Muhammad Alda (2023) Membahas tentang sebuah solusi digital melalui pengembangan aplikasi pengolahan data siswa berbasis Android di SMK Tritech Informatika, yang sebelumnya hanya mengandalkan aplikasi Microsoft Excel sebagai sarana pencatatan data siswa. Permasalahan utama yang diangkat adalah proses distribusi data yang lambat, keamanan data yang rendah, serta risiko kehilangan data akibat keterbatasan penyimpanan hanya pada satu perangkat. Melalui pendekatan metode prototyping, sistem dikembangkan menggunakan framework Kodular dan database Airtable, yang mampu memfasilitasi kebutuhan pengolahan data secara online dan real-time. Hasilnya, aplikasi ini dapat membantu staf tata usaha dalam menambah, menghapus, dan mengedit data siswa secara fleksibel, serta memungkinkan kepala sekolah atau pihak terkait untuk mengakses informasi tersebut secara cepat melalui smartphone Android. Keunggulan dari penelitian ini adalah pemanfaatan teknologi berbasis cloud dan low-code platform yang memudahkan proses pengembangan serta pemakaian aplikasi oleh pengguna non-teknis, ditambah kemampuan akses online yang meningkatkan efisiensi kerja.</p>
-----	---	---

		<p>Selain itu, pendekatan prototyping memungkinkan adanya iterasi pengembangan berdasarkan umpan balik awal dari pengguna. Namun, kelemahan dari jurnal ini adalah terbatasnya pengujian aplikasi dalam skala besar serta kurangnya penekanan terhadap aspek keamanan data yang sebenarnya menjadi perhatian utama dalam pengolahan data siswa, terutama bila melibatkan data sensitif. Evaluasi performa aplikasi dari sisi beban sistem, penggunaan memori, serta potensi peretasan juga belum dieksplorasi dalam jurnal ini</p>
13.	<p>Pengujian Aplikasi Absensi dan Kegiatan Magang Mahasiswa Menggunakan Metode BlackBox Testing</p>	<p>Apriansah et.al (2023) membahas tentang pengujian perangkat lunak pada aplikasi absensi dan kegiatan magang mahasiswa dengan menerapkan metode Black Box Testing, yang diarahkan pada teknik Equivalence Partitioning untuk mendeteksi kesalahan sistem berdasarkan input yang valid dan tidak valid. Studi dilaksanakan pada enam formulir utama dalam aplikasi, yaitu masuk, dasbor, data siswa, pengelolaan siswa, kehadiran, dan kegiatan magang, dengan pengujian untuk tiap-tiap alur kerja sistem. Hasil pengujian menunjukkan bahwa mayoritas fitur berfungsi dengan baik, meskipun ada beberapa kesalahan kecil</p>

	<p>yang tidak sesuai dengan desain awal. Kelebihan jurnal ini terletak pada akurasi proses pengujian fungsionalitas sistem tanpa harus memperhatikan struktur internal program, sehingga ideal untuk menilai kualitas aplikasi dari sudut pandang pengguna akhir. Di samping itu, jurnal ini secara jelas menyuguhkan skenario pengujian dan hasil yang diinginkan, yang sangat berguna sebagai bahan untuk evaluasi teknis. Akan tetapi, kelemahan jurnal ini terletak pada ruang lingkupnya yang hanya fokus pada pengujian tanpa menjelaskan proses pengembangan aplikasinya, serta tidak ada metrik performa atau statistik tingkat keberhasilan pengujian. Di samping itu, walaupun pengujian telah dilakukan pada beberapa bentuk, belum ada uji lanjut yang melibatkan pengguna sebenarnya atau skenario penggunaan dalam keadaan nyata, yang krusial untuk menilai kepuasan pengguna dan ketahanan aplikasi dalam jangka panjang.</p>
--	---

Berdasarkan hasil studi terhadap berbagai penelitian terdahulu, dapat disimpulkan bahwa pengembangan sistem informasi magang berbasis *digital*, baik melalui platform *website* maupun aplikasi *Android*, telah banyak dilakukan untuk menggantikan proses manual yang cenderung tidak efisien. Penelitian oleh Verdinata & Evanata (2023) dan Candra et al. (2024) menunjukkan bahwa sistem berbasis *web* dengan pendekatan *prototype* mampu memfasilitasi kebutuhan

pengelolaan magang, mulai dari pendaftaran, pengisian *logbook*, hingga pencetakan laporan. Sementara itu, pendekatan *Agile dan Scrum* seperti yang digunakan oleh Kustanto et al. (2021) serta Ramadhan et al. (2023) memberikan fleksibilitas dalam proses pengembangan, meskipun belum banyak yang membahas pengujian secara menyeluruh oleh pengguna akhir. Penelitian lain yang bersifat analitis, seperti yang dilakukan oleh Trisno & Hari (2021), lebih menitikberatkan pada desain sistem melalui UML tanpa disertai implementasi langsung. Beberapa penelitian juga telah menerapkan integrasi fitur *Location-Based Service (LBS)* dan absensi *digital* berbasis *mobile* seperti dalam penelitian Ordiyasa et al. (2021) dan Yulianto & Firdaus (2021), namun implementasinya masih terbatas pada pelacakan lokasi tanpa pelaporan lanjutan atau pemrosesan data lebih dalam.

Secara umum, fitur yang paling sering dikembangkan dalam penelitian-penelitian sebelumnya adalah pendaftaran peserta magang, pengelolaan data mahasiswa, *logbook*, absensi, serta pengunggahan dokumen penunjang. Akan tetapi, sebagian besar masih memiliki keterbatasan dalam hal keamanan data, fleksibilitas peran pengguna (admin, dosen, mitra, mahasiswa), dan integrasi fungsionalitas dalam satu *platform* yang komprehensif. Beberapa jurnal seperti milik Apriansah et al. (2023) bahkan hanya menitikberatkan pada pengujian sistem tanpa menyentuh aspek pengembangan, sehingga belum memberikan gambaran utuh mengenai pemanfaatan sistem dalam konteks operasional sebenarnya.

Penelitian ini menghadirkan kontribusi kebaruan dengan mengembangkan sistem informasi pengelolaan magang berbasis *android* yang terintegrasi dan *responsif* terhadap kebutuhan semua peran pengguna. Fitur unggulan yang ditawarkan mencakup pemetaan lokasi otomatis saat absensi, pengisian *logbook* secara fleksibel melalui *android*.

## 2.2 Sistem

Sistem dapat dipahami sebagai kumpulan komponen yang saling terhubung dan bekerja sama untuk mencapai suatu tujuan. Di dalamnya terdapat elemen-elemen penting seperti masukan, proses, dan keluaran. Secara umum, sistem merupakan suatu kesatuan yang terdiri dari elemen atau variabel yang tersusun secara terorganisir, saling berinteraksi, dan saling bergantung satu sama lain. Dengan kata

lain, sistem adalah struktur yang terkoordinasi dan berfungsi melalui alur input, proses, hingga menghasilkan output untuk mencapai tujuan tertentu[2].

Sistem informasi magang ini dirancang sebagai satu kesatuan terintegrasi yang menghubungkan mahasiswa, dosen wali, mitra, admin, dan penguji. Alur input dimulai dari pengisian *logbook*, pengunggahan dokumen, hingga pencatatan absensi oleh mahasiswa. Proses berlangsung melalui validasi oleh mitra dan monitoring oleh dosen wali, kemudian menghasilkan output berupa status verifikasi, jadwal sidang, dan nilai magang. Semua elemen ini membentuk satu sistem kerja yang saling terkait dan mendukung pengelolaan magang secara *digital*.

### **2.3 Pengelolaan**

Pengelolaan berasal dari kata “*management*” yang artinya mengatur, jadi manajemen adalah proses pencapaian dengan tujuan yang diinginkan melalui berbagai aspek antara lain *planing*, *organizing*, dan *controlling*[2]. Dapat disimpulkan bahwa manajemen adalah proses pencapaian dengan tujuan yang diinginkan melalui berbagai aspek yaitu *planning*, *organizing* dan *controlling*.

Aplikasi ini mengelola seluruh aktivitas magang mulai dari pengajuan hingga pengawasan. Admin dan dosen wali berperan dalam memverifikasi dokumen, serta mengontrol pelaksanaan magang melalui fitur-fitur seperti dashboard monitoring, *logbook* validasi, dan jadwal sidang. Dengan begitu, seluruh proses manajemen magang terorganisir melalui sistem *Android* yang dirancang dalam *aplikasi* ini.

### **2.4 Magang**

Magang adalah program yang diselenggarakan oleh institusi pendidikan untuk meningkatkan kompetensi, pengalaman, dan *soft skill* siswa atau mahasiswa. Selain itu, program ini juga memberikan wawasan tentang dunia kerja sebelum mereka memasuki dunia profesional. Setiap perusahaan biasanya memiliki program magang dengan kriteria yang telah ditetapkan sesuai dengan kebijakan masing-masing [1].

Aplikasi ini memberikan fasilitas kepada mahasiswa untuk menjalankan kegiatan magang secara digital, termasuk mengajukan tempat magang, mengisi jurnal/*logbook*, melakukan absensi, dan berkomunikasi dengan mitra. Dengan

pemanfaatan fitur-fitur tersebut, mahasiswa mendapatkan pengalaman praktik lapangan yang tersistem, sementara mitra dan dosen wali dapat memantau dan memberikan penilaian secara langsung melalui aplikasi.

## **2.5 Android**

*Android* merupakan sistem operasi yang dikembangkan oleh Google dan dirancang khusus untuk perangkat mobile seperti smartphone dan tablet. Platform ini dilengkapi dengan *Google Play Store*, yang memiliki lebih dari satu miliar pengguna aktif di seluruh dunia. Dalam proses pengembangan aplikasi *Android*, para pengembang umumnya menggunakan *Integrated Development Environment* (IDE) sebagai alat bantu utama untuk menulis, menguji, dan menjalankan kode program secara efisien.

Sistem dikembangkan menggunakan *Android Studio* agar dapat digunakan di perangkat *mobile* seperti *smartphone*. Aplikasi ini mempermudah mahasiswa dan dosen dalam mengakses informasi magang di mana saja dan kapan saja, seperti mengisi logbook, melihat jadwal, serta memverifikasi aktivitas. Kemudahan akses ini menjadikan aplikasi lebih fleksibel dan efisien dibandingkan metode manual.

## **2.6 Metode Prototype**

*Prototyping* adalah pendekatan pengembangan sistem yang digunakan dalam penelitian ini. Metode *Prototyping* adalah metode pengembangan sistem dimana hasil Analisa sistem langsung diterapkan kedalam sebuah model tanpa menunggu seluruh sistem selesai, menghasilkan *prototype* dari perangkat lunak yang digunakan sebagai perantara pengembang dengan pengguna untuk berinteraksi [12].

Pengembangan aplikasi ini dilakukan melalui lima tahapan *prototyping*: komunikasi awal dengan pengguna, desain cepat, pembuatan *prototype*, evaluasi pengguna, dan perbaikan. Mahasiswa, mitra, dan dosen wali terlibat langsung dalam proses evaluasi sehingga pengembang bisa menyesuaikan fitur aplikasi berdasarkan masukan nyata pengguna, seperti penambahan validasi *logbook* oleh mitra dan pembatasan akses *logbook* hanya setelah absensi.

Pendekatan ini terdiri dari beberapa tahapan utama:

1. **Kebutuhan Awal:** Pengembang mengidentifikasi kebutuhan dasar dari sistem melalui komunikasi awal dengan pengguna.
2. **Desain Cepat:** Dibuatlah rancangan antarmuka awal sebagai visualisasi sistem (*mock-up*).
3. **Pembuatan Prototype:** Prototype dibangun berdasarkan desain cepat untuk menampilkan fungsi utama sistem secara sederhana.
4. **Evaluasi oleh Pengguna:** Pengguna mencoba prototype dan memberikan masukan.
5. **Perbaikan dan Iterasi:** *Prototype* disempurnakan berdasarkan *feedback* hingga sistem final terbentuk.

Dalam penelitian ini, metode *Prototype* dipilih karena sistem informasi magang membutuhkan masukan langsung dari berbagai peran (mahasiswa, dosen wali, mitra, admin dan penguji). Masukan tersebut sangat penting untuk memastikan sistem memenuhi kebutuhan pengguna secara akurat.

## **2.7 Teori Perancangan Sistem**

Sistem dirancang menggunakan pendekatan *Object Oriented Design* dengan bantuan diagram UML. Diagram tersebut digunakan untuk menggambarkan struktur sistem, proses bisnis, dan interaksi antar pengguna sistem, seperti mahasiswa, dosen wali, admin, dan mitra. Perancangan ini menjadi landasan pengembangan *aplikasi* agar sesuai dengan kebutuhan dan alur kerja magang yang sebenarnya.

### **2.7.1 Unified Modeling Language (UML)**

*Unified Modeling Language* (UML) adalah sebuah metode perancangan sistem yang menggunakan pendekatan analisa berbasis objek (*Object Oriented Design*) [3]. Tujuannya adalah untuk membantu dalam perancangan, pendokumentasian, serta pemahaman terhadap sistem perangkat lunak. UML memiliki beberapa diagram yang dapat digunakan selama proses perancangan sistem.

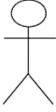
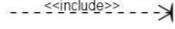
Dalam aplikasi ini, UML digunakan untuk mendeskripsikan interaksi antara pengguna dengan sistem. Diagram yang dibuat meliputi *Use Case Diagram* dan *Activity Diagram*, yang menggambarkan peran aktor (mahasiswa, admin, mitra, dosen wali, dan penguji), serta alur kegiatan seperti apply mitra, validasi logbook, dan penjadwalan sidang magang. Diagram ini membantu tim pengembang memahami dan merancang sistem sesuai kebutuhan pengguna. Beberapa diagram yang terdapat pada UML adalah sebagai berikut :

### **2.7.1.1 Use Case Diagram**

*Use Case Diagram* digunakan sebagai alat untuk memodelkan bagaimana sistem bekerja dalam proses pengembangannya. Diagram ini menggambarkan hubungan atau interaksi antara aktor pengguna atau entitas lain dengan sistem yang sedang dirancang, sehingga membantu memahami fungsi-fungsi utama yang harus dijalankan oleh sistem.

*Use Case Diagram* menggambarkan aktivitas yang dapat dilakukan oleh setiap aktor dalam sistem, seperti mahasiswa yang dapat mengisi *logbook* dan *apply* mitra, dosen wali yang dapat melihat dan menilai, serta mitra yang memverifikasi dan memberi nilai. Diagram ini digunakan untuk memastikan setiap fitur dalam aplikasi sesuai dengan kebutuhan masing-masing pengguna. Berikut ini adalah Tabel 2.2 yang menunjukkan simbol-simbol pada *use case* diagram.

Tabel 2. 2 Simbol *Use Case*

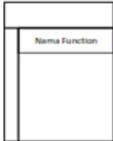
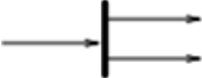
Simbol	Nama	Uraian
	<i>Actor</i>	Simbol yang mewakili pengguna dalam sistem
	<i>Use Case</i>	Menggambarkan adanya interaksi antara sistem dan <i>actor</i>
	<i>Include</i>	Menampilkan sebuah skenario <i>use case</i> sebagai komponen dari fungsi <i>use case</i> lainnya.
	<i>Ekstensi</i>	Menyoroti skenario <i>use case</i> yang berperan sebagai tambahan fungsi dari <i>use case</i> lainnya.
	Asosiasi	Koneksi antara <i>actor</i> dengan <i>use case</i> .
	Generalisasi	Menyajikan <i>actor</i> yang memiliki kekhususan untuk dalam suatu <i>use case</i> .

### 2.7.1.2 Activity Diagram

*Activity Diagram* adalah salah satu diagram yang juga terdapat pada UML. *Activity Diagram* berguna untuk menunjukkan aliran dari suatu aktivitas ke aktivitas lainnya. *Activity Diagram* juga dapat membantu orang-orang yang terlibat dalam pembuatan sistem, baik dari sisi pengembang maupun dari sisi pengguna *end user* supaya mampu memiliki pemahaman yang sama [3]. *Activity Diagram* menggambarkan alur proses dari fitur-fitur utama aplikasi, seperti pengajuan mitra, validasi logbook, serta pelaksanaan sidang. Diagram ini menjelaskan langkah-langkah dan aliran proses antar aktor secara jelas, dan menjadi panduan dalam menyusun alur interaksi di dalam *aplikasi*.

Berikut ini adalah Tabel 2.3 yang menunjukkan simbol-simbol pada *Activity Diagram*.

Tabel 2. 3 Simbol *Activity Diagram*

Simbol	Nama	Uraian
	<i>Swimlane</i>	Pengelompokkan aktivitas masing – masing <i>actor</i> dalam sistem
	<i>Start Point</i>	Simbol mengawali aktivitas sistem yang terjadi
	<i>Activity</i>	Menggambarkan suatu proses dalam sistem
	<i>Penggabungan</i>	Menggambarkan beberapa aktivitas yang digabung menjadi satu
	<i>State Transition</i>	Menggambarkan antara aktivitas
	<i>Fork</i>	Menggambarkan sebuah aktivitas dimulai dan disertai dua atau lebih aktivitas lainnya.
	<i>End Point</i>	Simbol aktivitas sistem telah berakhir.

## 2.8 Metode Pengujian

Pada tahap akhir pengembangan sistem, dilakukan proses pengujian untuk memastikan bahwa seluruh fitur dan fungsi dari sistem telah berjalan sesuai dengan kebutuhan yang telah ditentukan sejak awal. Fase pengujian ini menjadi langkah

krusial dalam siklus pengembangan perangkat lunak karena bertujuan mengevaluasi kesiapan sistem sebelum diterapkan dalam lingkungan nyata. Pengujian dilakukan untuk memverifikasi apakah sistem telah bekerja dengan benar, stabil, serta sesuai dengan harapan pengguna akhir. Dalam penelitian ini, proses pengujian dilakukan dengan menerapkan dua pendekatan, yaitu *User Acceptance Testing (UAT)* sebagai metode pengujian utama, dan *Black Box Testing* sebagai teknik pengujiannya.

Sistem diuji untuk memastikan bahwa fitur seperti *logbook*, *apply* mitra, validasi, dan upload dokumen berfungsi dengan benar. Pengujian dilakukan untuk menghindari bug, memastikan input diterima atau ditolak sesuai logika sistem, dan untuk mengetahui apakah sistem memenuhi harapan pengguna sebelum diimplementasikan secara penuh di kampus.

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

Pengujian dalam penelitian ini menerapkan metode *UAT (User Acceptance Testing)*. *User Acceptance testing* adalah tahapan untuk memastikan bahwa Solusi yang dikembangkan dalam sistem telah sesuai untuk pengguna. Terdapat berbagai macam metode UAT seperti pengujian alfa, pengujian beta, pengujian penerimaan regulasi, dan pengujian kotak hitam [2]. Penelitian ini menerapkan pendekatan *black box testing* dalam pelaksanaan UAT, yaitu dengan menguji sistem dari aspek antarmuka dan fungsi, tanpa harus melihat atau memahami struktur internal kode program. Dengan metode ini, pengguna mengevaluasi sistem berdasarkan situasi-situasi nyata dan memberikan tanggapan apakah sistem telah memenuhi kebutuhan atau masih perlu ditingkatkan.

UAT dilakukan oleh mahasiswa, dosen wali, dan mitra dengan cara mencoba langsung aplikasi. Pengguna memberikan penilaian terhadap kegunaan fitur-fitur seperti pengisian *logbook*, upload dokumen, dan validasi absensi. Hasilnya, sistem dinilai layak dan siap digunakan, dengan tingkat kepuasan pengguna mencapai 85,4%.

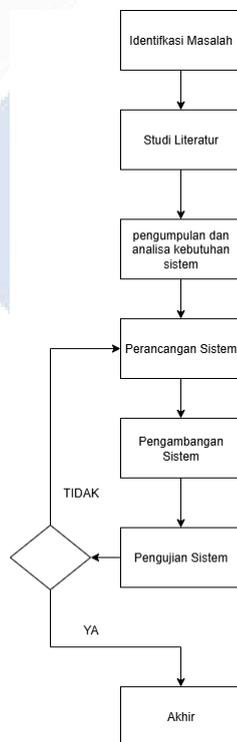
### **2.8.2 Black Box Testing**

Metode Pengujian *Black Box* adalah teknik yang digunakan untuk mendeteksi kesalahan dalam aplikasi sistem, termasuk kesalahan pada fungsi dan menu yang tidak tersedia. *Black Box testing* berfokus pada pengujian fungsionalitas sistem aplikasi dengan menggunakan data acak sebagai masukan untuk mendapatkan hasil yang jelas. Jika data yang dimasukkan salah, sistem informasi akan menolak dan tidak menyimpannya dalam database, sebaliknya, jika data tersebut benar, maka akan diterima dan disimpan dalam *database*. [13] Dalam penelitian ini, *Black Box Testing* diterapkan sebagai metode pengujian dalam pelaksanaan *User Acceptance Testing (UAT)*. Dengan pendekatan ini, pengujian dilakukan dengan memberikan berbagai jenis data ke sistem untuk memastikan respons yang diharapkan. Hasil dari pengujian ini menjadi pedoman penting dalam menilai kestabilan dan keandalan sistem sebelum digunakan secara resmi.

### BAB III

#### METODE PELAKSANAAN

Metode pelaksanaan adalah tahapan terencana yang bertujuan untuk menjelaskan secara rinci tahap-tahap pelaksanaan proyek akhir, dari identifikasi masalah hingga penulisan laporan akhir. Metode ini bertujuan untuk menyediakan kerangka kerja yang jelas dan terarah dalam mengatasi masalah yang telah ditetapkan di awal penelitian. Setiap fase dilaksanakan secara bertahap dan saling terhubung untuk memastikan bahwa seluruh proses pengembangan sistem berlangsung sesuai dengan sasaran yang telah ditentukan. Agar memperjelas jalannya proses secara *visual*, berikut adalah *flowchart* pelaksanaan proyek akhir yang menggambarkan langkah-langkah dari awal hingga akhir secara sistematis dan teratur. Berikut ini adalah Gambar 3.1 yang menunjukkan *Flowchart* metode pelaksanaan.



Gambar 3. 1 Metode Pelaksanaan

### 3.1 Identifikasi Masalah

Masalah yang dihadapi dalam pengelolaan kegiatan magang di Politeknik Manufaktur Negeri Bangka Belitung adalah belum tersedianya sistem informasi berbasis aplikasi yang dapat mengelola proses administrasi dan monitoring secara *digital*. Proses yang masih dilakukan secara manual menimbulkan beberapa kendala seperti keterlambatan informasi, sulitnya koordinasi antara mahasiswa, dosen wali, dan pihak mitra, serta risiko kehilangan dokumen. Mahasiswa mengalami kesulitan dalam mengakses informasi magang secara *real-time*, mencatat kegiatan harian, serta mengunggah dokumen. Di sisi lain, komisi magang dan dosen wali juga kesulitan dalam melakukan validasi dan pemantauan kegiatan magang. Oleh karena itu, diperlukan suatu sistem berbasis *android* yang mampu mengatasi permasalahan tersebut secara efisien dan terintegrasi.

### 3.2 Studi Literatur

Studi literatur dilakukan untuk memperoleh landasan teori dan referensi yang relevan dalam pengembangan sistem informasi magang menggunakan metode *prototype*. Beberapa penelitian sebelumnya menjadi acuan penting dalam studi ini. Verdinata & Evanata (2023) mengembangkan aplikasi pengelolaan magang berbasis web menggunakan metode *prototyping*, yang terbukti membantu proses administrasi magang secara *digital* meskipun belum diuji secara langsung oleh pengguna akhir. Umayu Alakbar Yusa (2022) juga menerapkan metode *prototyping* dalam membangun sistem manajemen kegiatan magang dengan tingkat kepuasan pengguna sebesar 88%, menunjukkan efektivitas pendekatan ini dalam merancang fitur yang sesuai kebutuhan. Trisno & Hari (2021) menekankan pentingnya pemodelan sistem menggunakan UML untuk menggambarkan struktur dan interaksi antar entitas dalam sistem magang, meskipun penelitian mereka bersifat teoretis tanpa implementasi nyata.

Selain itu, Aditya et al. (2021) menunjukkan bahwa metode *prototype* mampu menghasilkan sistem monitoring kegiatan berbasis *Android* yang fleksibel terhadap masukan pengguna. Di sisi lain, Apriansah et al. (2023) menyoroti pentingnya pengujian sistem menggunakan *blackbox* testing guna memastikan fungsi-fungsi dalam aplikasi berjalan sesuai harapan. Berdasarkan studi-studi tersebut, penelitian

ini merancang sistem informasi magang berbasis *Android* dengan pendekatan *prototype*, pengujian blackbox, dan perancangan sistem berbasis UML, sehingga diharapkan mampu menjawab kebutuhan pengguna secara praktis dan menyeluruh.

### 3.3 Pengumpulan Dan Analisis Kebutuhan Sistem

Pada tahap ini dilakukan pengumpulan data kebutuhan sistem sebagai dasar dalam proses pengembangan sistem informasi magang berbasis *Android*. Metode yang digunakan dalam pengumpulan data adalah:

#### a) Wawancara

Wawancara merupakan salah satu metode yang digunakan untuk mengumpulkan informasi secara langsung melalui sesi tanya jawab antara peneliti dan responden. Teknik ini memungkinkan peneliti mendapatkan data yang lebih mendalam dan relevan sesuai kebutuhan[2]. Dalam penelitian ini, proses pengumpulan data dilakukan dengan mewawancarai pihak Komisi Magang Polman Negeri Babel. Hasil dari wawancara tersebut mengungkapkan bahwa belum terdapat sistem yang mendukung proses pemantauan magang secara terstruktur. Komisi magang menyampaikan bahwa pemantauan kegiatan mahasiswa selama magang masih dilakukan secara manual dan tidak terpusat, sehingga menyulitkan dalam memantau perkembangan setiap mahasiswa secara *real time*. Selain itu, pengarsipan dokumen magang seperti *logbook*, surat rekomendasi, dan laporan kegiatan juga dirasakan cukup merepotkan karena harus dilakukan secara manual dan berpotensi menimbulkan risiko kehilangan atau kerusakan data.

#### b) Observasi

Observasi merupakan metode pengumpulan data yang dilakukan melalui pengamatan langsung terhadap aktivitas atau proses yang berlangsung dalam suatu organisasi, dengan mencatat berbagai informasi penting yang mendukung jalannya penelitian. Hasil observasi menunjukkan bahwa proses pencatatan aktivitas *logbook* masih dilakukan secara konvensional menggunakan dokumen, yang menyulitkan mitra dan dosen wali dalam melakukan validasi harian. Selain itu, tidak adanya

integrasi data antar pihak menyebabkan komunikasi dan pelaporan kegiatan menjadi terhambat.

### **3.4 Perancangan Sistem**

Pada tahap perancangan sistem, dilakukan pembuatan desain awal sistem dalam bentuk *Use Case Diagram* dan *Activity Diagram*, sebagai representasi *visual* dari keseluruhan proses dan struktur sistem yang akan dikembangkan. Desain ini disusun untuk menggambarkan secara sistematis bagaimana sistem akan beroperasi, siapa saja aktor yang terlibat, serta bagaimana interaksi antar aktor dengan fitur-fitur dalam sistem berlangsung.

*Use Case Diagram* digunakan untuk memetakan hubungan antara pengguna sistem, seperti mahasiswa, dosen wali, admin, mitra, dan penguji sidang, dengan fungsi-fungsi utama yang tersedia dalam aplikasi. *Activity Diagram* menggambarkan alur aktivitas secara terperinci mulai dari pengajuan magang, pengisian *logbook*, proses *verifikasi*, hingga pelaksanaan sidang.

Perancangan diagram ini tidak hanya berfungsi sebagai dokumentasi teknis, tetapi juga menjadi pedoman utama bagi pengembang dalam membangun sistem agar tetap selaras dengan kebutuhan pengguna. Dengan adanya *blueprint visual* yang jelas, proses pengembangan menjadi lebih terarah, minim revisi, serta memungkinkan komunikasi yang lebih efektif antara tim pengembang dan pihak pemangku kepentingan.

### **3.5 Pengembangan Sistem Menggunakan Metode *Prototype***

Metode penelitian yang diterapkan dalam pengembangan sistem Pengelolaan Kegiatan Magang Mahasiswa di Politeknik Manufaktur Negeri Bangka Belitung adalah Metode *Prototyping*. Metode *Prototyping* dipilih karena bersifat iteratif dan adaptif terhadap perubahan kebutuhan pengguna, sebagaimana dijelaskan oleh Pressman (2014), bahwa metode ini efektif dalam lingkungan yang kompleks dan dinamis. Dalam konteks sistem magang yang melibatkan berbagai aktor dengan kebutuhan berbeda, metode ini memungkinkan komunikasi berkelanjutan dan pengembangan yang fleksibel.

Pemilihan metode ini didasarkan pada karakteristiknya yang iteratif dan fleksibel, memungkinkan pengembangan sistem secara bertahap dengan masukan langsung dari pengguna. Pendekatan ini sangat sesuai untuk proyek sistem informasi yang melibatkan berbagai peran pengguna, termasuk mahasiswa, dosen wali, admin, dan mitra .

Dengan menggunakan metode *prototype*, pengembangan sistem dimulai dengan merancang sebuah model awal yang berfungsi sebagai representasi sederhana dari sistem yang akan dibangun. Model ini berisi antarmuka dasar dan alur kerja utama yang ditujukan untuk memberi gambaran nyata kepada pengguna terhadap sistem yang direncanakan. Setelah itu, *prototype* diuji secara langsung oleh pengguna untuk memperoleh *feedback*, baik terkait fitur, tampilan, hingga kemudahan penggunaan. *Feedback* tersebut menjadi dasar untuk melakukan perbaikan dan penyempurnaan sistem secara bertahap hingga mencapai hasil akhir yang benar-benar sesuai dengan kebutuhan pengguna.

Kelebihan metode ini terletak pada kemampuannya untuk menyesuaikan sistem dengan cepat terhadap perubahan atau masukan selama proses pengembangan. Pengembang tidak perlu menunggu hingga sistem selesai sepenuhnya untuk diuji oleh pengguna, melainkan dapat melibatkan pengguna sejak awal, sehingga risiko kesalahan desain atau kesenjangan kebutuhan dapat diminimalisir. Dalam konteks sistem informasi magang Polman Negeri Babel, metode *prototyping* memungkinkan pengembangan aplikasi *android* yang responsif terhadap kebutuhan nyata, seperti pengajuan tempat magang, pengisian *logbook* harian, validasi absensi, pengunggahan laporan, hingga penjadwalan dan penilaian sidang magang.

Adapun tahapan pelaksanaan metode *Prototype* dalam proyek ini dilakukan melalui lima langkah utama berikut:

#### 1. Komunikasi Awal dengan Pengguna

Tahap ini dimulai dengan menggali kebutuhan sistem dari para pengguna melalui wawancara, observasi, dan studi dokumen magang. Pihak yang dilibatkan dalam komunikasi awal meliputi mahasiswa, dosen wali, admin, dan mitra. Dari tahapan ini diperoleh informasi terkait hambatan pada sistem

manual, seperti keterlambatan proses verifikasi, kesulitan mengakses informasi, dan kerumitan dalam pelaporan aktivitas magang.

## 2. Desain Cepat (*Quick Design*)

Setelah kebutuhan dikumpulkan, dibuatlah desain awal berupa *mock-up* antarmuka sistem menggunakan *tools* seperti Figma. Desain ini mencakup halaman *login*, dashboard peran pengguna (mahasiswa, mitra, dosen wali, admin dan penguji), menu pengisian *logbook*, pengunggahan dokumen, dan validasi *logbook*. Desain awal ini difokuskan untuk memberikan gambaran navigasi dan alur sistem secara sederhana, sehingga mudah dipahami oleh pengguna.

## 3. Pembuatan Prototype

*Prototype* awal dikembangkan dengan mengimplementasikan antarmuka dan beberapa fungsi utama ke dalam aplikasi berbasis *Android* menggunakan *Android Studio*. Fitur utama yang dibangun pada tahap ini meliputi: input *logbook*, pengunggahan dokumen, jadwal sidang, dan validasi *logbook*. *Prototype* ini masih bersifat simulasi dan digunakan untuk uji coba awal.

## 4. Evaluasi dan Umpan Balik

Setelah *prototype* dapat digunakan, dilakukan evaluasi langsung oleh pengguna akhir melalui simulasi penggunaan. Mahasiswa diminta mencoba fitur *logbook* dan unggah dokumen, sedangkan dosen wali dan mitra mencoba validasi *logbook* serta melihat status magang. Berdasarkan evaluasi, beberapa masukan muncul, seperti kebutuhan fitur seperti melihat penilaian mahasiswa, *acc logbook* dan absensi di mitra, menu *logbook* hanya bisa digunakan jika mahasiswa masuk dan tambah mahasiswa belum ada. Masukan ini dicatat dan menjadi dasar untuk revisi selanjutnya.

## 5. Revisi dan Finalisasi

Berdasarkan hasil evaluasi pengguna terhadap *prototype* awal, dilakukan serangkaian perbaikan untuk menyempurnakan sistem sesuai kebutuhan pengguna. Beberapa masukan penting yang diterima antara lain perlunya penambahan fitur untuk melihat penilaian mahasiswa, pengajuan *logbook* dan absensi yang dapat disetujui *acc* oleh mitra, serta pembatasan akses menu

*logbook* agar hanya dapat digunakan oleh mahasiswa yang sudah melakukan absensi. Selain itu, fitur penambahan data mahasiswa oleh admin juga belum tersedia dalam versi awal.

Menanggapi hal tersebut, tim pengembang melakukan penyesuaian pada struktur navigasi *aplikasi* dan menambahkan logika akses pada fitur *logbook* agar lebih selektif. Fitur baru seperti tampilan penilaian, proses validasi mitra, serta form tambah mahasiswa kemudian diimplementasikan dalam versi final *prototype*. Setelah proses revisi selesai, sistem kembali diuji dan dinyatakan telah sesuai dengan kebutuhan serta ekspektasi mayoritas pengguna.

### **3.6 Pengujian, Pengumpulan dan Analisa data**

Tahapan ini merupakan fase terakhir setelah sistem selesai dikembangkan. Pada periode ini, serangkaian pengujian dilakukan untuk memastikan bahwa semua fungsi dalam sistem beroperasi sesuai dengan kebutuhan dan harapan pengguna. Pengujian dilakukan dengan metode yang tepat, seperti uji kotak hitam, untuk mengevaluasi fungsionalitas sistem tanpa mempertimbangkan struktur *internal* kode. Selain itu, data juga dikumpulkan dari hasil pengujian melalui berbagai cara, termasuk pengamatan langsung, kuesioner, dan wawancara dengan pengguna sistem. Data yang diperoleh kemudian dianalisis secara menyeluruh untuk menilai performa sistem, mengidentifikasi kelemahan, serta memberikan dasar untuk perbaikan dan pengambilan keputusan selanjutnya.

#### **3.6.1 Metode Pengujian**

Penelitian ini melibatkan dua jenis pengujian, yaitu UAT dan *Black Box*, yang masing-masing bertujuan menilai efektivitas sistem dan menguji kebenaran hipotesis yang telah ditetapkan.

#### **3.6.2 UAT dan Blackbox Testing**

*User Acceptance Testing (UAT)* dilaksanakan untuk memastikan bahwa sistem telah dikembangkan sesuai dengan kebutuhan dan ekspektasi pengguna. Dalam pelaksanaannya, metode *Blackbox Testing* digunakan untuk mengevaluasi

fungsionalitas sistem berdasarkan respons terhadap input yang diberikan, tanpa memperhatikan struktur internal perangkat lunak.

### **3.7 Metode Pengumpulan Dan Analisis Data**

Penelitian yang dilakukan dalam proyek akhir ini mengadopsi pendekatan kuantitatif, dengan metode pengumpulan data yang mencakup wawancara, observasi, dan survei menggunakan kuesioner.

#### **a. Observasi dan Wawancara**

Observasi dilakukan secara langsung terhadap proses administrasi magang di lingkungan Politeknik Manufaktur Negeri Bangka Belitung, guna mengidentifikasi alur kerja yang masih bersifat manual, kendala yang sering muncul, serta potensi solusi berbasis aplikasi *Android*. Wawancara dilakukan terhadap pihak-pihak yang terlibat langsung dalam kegiatan magang, yaitu Komisi Magang, dosen wali, dan mahasiswa. Informasi dari hasil wawancara dijadikan dasar untuk merumuskan kebutuhan sistem.

#### **b. Survei Kuesioner**

Untuk mengukur kepuasan pengguna terhadap *aplikasi* yang telah dikembangkan, dilakukan survei menggunakan kuesioner tertutup yang disusun berdasarkan pernyataan positif terkait aspek kegunaan, kemudahan penggunaan, kehandalan, dan manfaat sistem.

## **BAB IV**

### **PEMBAHASAN**

#### **4.1 Hasil Identifikasi Kebutuhan Sistem**

Dalam proses pengembangan sistem informasi pengelolaan kegiatan magang, dilakukan pengumpulan data dengan beberapa metode, yaitu wawancara, observasi, serta studi dokumen pendukung seperti SOP dan panduan magang. Pengumpulan data ini bertujuan untuk menggali informasi yang akurat terkait kebutuhan sistem, permasalahan yang dihadapi dalam proses administrasi magang, serta alur kerja yang berlaku saat ini. Berikut penjelasan tiap metode:

##### **4.1.2 Wawancara**

Wawancara dilakukan secara langsung dengan pihak yang terkait dalam pengelolaan administrasi magang, dengan tujuan untuk mengidentifikasi berbagai permasalahan yang selama ini muncul dalam pelaksanaan kegiatan magang mahasiswa. Hasil wawancara mengungkapkan beberapa kendala utama yang cukup signifikan. Pertama, terdapat hambatan dalam memantau kegiatan magang mahasiswa secara langsung karena belum tersedia sistem yang mendukung pelaporan dan pemantauan secara *real-time*. Kedua, proses pengarsipan dokumen administrasi seperti surat pengantar, *logbook*, dan laporan akhir masih dilakukan secara manual dalam bentuk cetak, yang menyebabkan kesulitan dalam pencarian data dan menambah beban kerja administratif.

Sering terjadi kehilangan dokumen penting akibat tidak adanya sistem *digital* terintegrasi yang mampu menyimpan dan mengelola data secara aman dan terstruktur. Permasalahan-permasalahan ini mengindikasikan perlunya pengembangan sistem informasi berbasis aplikasi *android* yang mampu mengintegrasikan seluruh proses administrasi dan pemantauan magang secara efisien, terpusat, serta mudah diakses oleh seluruh pihak yang terlibat.

#### **4.1.3 Observasi**

Observasi dilakukan terkait proses pengelolaan magang. Tahapan yang diperhatikan mencakup pengajuan lokasi magang, pengumpulan berkas, pengisian logbook, serta pelaksanaan sidang magang. Terungkap bahwa mayoritas kegiatan dilakukan secara konvensional dengan menggunakan dokumen fisik, yang sangat rentan terhadap kesalahan, keterlambatan, dan kehilangan informasi. Hasil observasi ini mendukung hasil wawancara, sekaligus menguatkan urgensi adanya sistem yang mampu mengotomatisasi serta mengintegrasikan proses pengelolaan magang.

#### **4.1.4 Studi Dokumen**

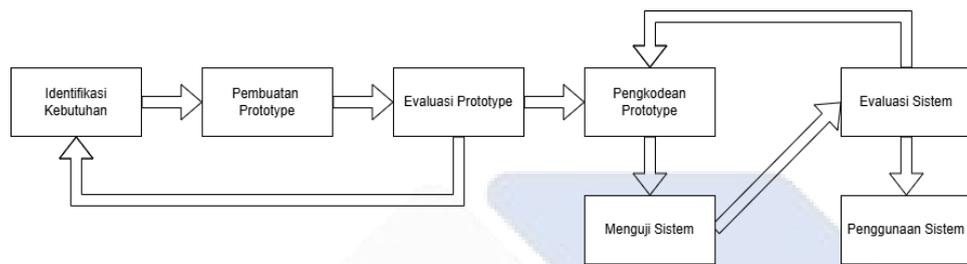
Untuk mendalami struktur dan aturan baku dalam pelaksanaan kegiatan magang, dilakukan juga studi terhadap dokumen-dokumen resmi seperti *Standar Operasional Prosedur (SOP)* dan *Panduan Magang Mahasiswa*. Studi dokumen ini bertujuan untuk memastikan bahwa sistem yang dikembangkan nantinya sejalan dengan kebijakan dan prosedur yang berlaku di institusi. Dari hasil studi dokumen, diperoleh gambaran alur resmi proses magang, termasuk persyaratan administrasi, tahapan yang harus dilalui mahasiswa, tugas masing-masing aktor (mahasiswa, dosen wali, mitra, admin dan penguji), serta format pelaporan dan penilaian. Informasi ini menjadi dasar dalam merancang *use case diagram*, *activity diagram*, dan fitur-fitur dalam *aplikasi* agar sesuai dengan ketentuan yang berlaku di Polman Negeri Babel.

#### **4.2 Hasil Pembuatan Prototype**

Pengembangan sistem informasi pengelolaan kegiatan magang mahasiswa Politeknik Manufaktur Negeri Bangka Belitung dilakukan dengan pendekatan Metode *Prototyping*. Metode ini digunakan untuk memungkinkan proses pengembangan sistem dilakukan secara bertahap, melalui serangkaian iterasi yang disesuaikan dengan masukan dari pengguna. Model ini sangat cocok digunakan untuk sistem yang kompleks dan berorientasi pada pengguna, karena setiap tahapan

dapat diuji secara langsung dan diperbaiki sebelum masuk ke tahap implementasi *final*.

Berikut ini adalah Gambar 4.1 yang menunjukkan Metode *Prototyping*



Gambar 4. 1 Metode *Prototyping*

Berikut penjelasan mendetail setiap tahapan:

1. Identifikasi kebutuhan

Tahap awal ini merupakan proses penting untuk memahami secara menyeluruh permasalahan yang dihadapi pengguna. Dalam konteks pengembangan sistem magang Polman Negeri Babel, proses identifikasi dilakukan dengan melakukan wawancara kepada pihak Komisi Magang, dosen wali, dan mahasiswa, serta observasi langsung terhadap proses administrasi magang yang berjalan saat ini. Dari hasil kegiatan ini, diperoleh gambaran mengenai kebutuhan sistem, seperti fitur pengajuan tempat magang secara *online*, pengisian jurnal dan *logbook* harian, validasi absensi, penilaian mitra, serta penjadwalan sidang magang. Informasi ini menjadi dasar untuk menentukan ruang lingkup dan tujuan pengembangan sistem.

2. Pembuatan *Prototype*

Setelah kebutuhan terkumpul dan dianalisis, pengembang mulai merancang *prototype* awal dari sistem. *Prototype* merupakan tampilan awal (*mock-up*) yang mencerminkan struktur menu, navigasi, dan alur interaksi antar pengguna dalam *aplikasi*, seperti mahasiswa yang ingin mengunggah dokumen atau mitra yang ingin memberikan penilaian. Pada tahap ini, desain belum bersifat *final* dan masih dalam bentuk simulasi sederhana menggunakan *tools* seperti Figma atau desain antarmuka

di *Visual Studio* . Tujuannya adalah memberikan gambaran nyata kepada pengguna tentang sistem yang akan dikembangkan.

### 3. Evaluasi Prototype

Pada tahap ini, *prototype* diuji langsung oleh pengguna untuk mengetahui apakah alur, fitur, dan tampilan sudah sesuai dengan ekspektasi mereka. Misalnya, mahasiswa mencoba menu *logbook* dan memberikan masukan jika tampilan terlalu rumit atau fitur tidak berfungsi sebagaimana mestinya. Begitu juga dosen wali dan mitra dapat memberikan pendapat tentang kemudahan akses data absensi atau penilaian. Setiap masukan dari pengguna akan dicatat oleh tim pengembang dan digunakan sebagai dasar untuk revisi *prototype* agar lebih sesuai dengan kebutuhan sebenarnya. Evaluasi sistem yang telah dikembangkan diuji lebih lanjut untuk memastikan bahwa semua fungsi berjalan sesuai.

### 4. Pengkodean *Prototype*

Jika pengguna sudah menyetujui alur dan tampilan *prototype*, maka pengembang mulai melakukan pengkodean . Ini merupakan proses mengimplementasikan sistem yang sesungguhnya dalam bentuk aplikasi *Android*. Pengembangan dilakukan dengan mengacu pada *prototype* yang telah dievaluasi. Fitur utama seperti login multi-akun (mahasiswa, mitra, dosen), pengisian *logbook* dan absensi, validasi *logbook*.

### 5. Evaluasi Sistem

Setelah sistem dikembangkan, tim melakukan uji fungsionalitas secara menyeluruh. Evaluasi ini dirancang untuk memastikan bahwa semua fitur beroperasi dengan lancar dan tidak ada fungsi yang tidak ada atau tidak berjalan sebagaimana mestinya. Pengujian bisa dilakukan oleh pengembang sendiri (*developer testing*) atau tim QA (*quality assurance*) untuk melihat apakah sistem memenuhi standar yang telah ditentukan, baik dari segi performa, keamanan, maupun kegunaan.

### 6. Menguji Sistem

Tahap ini merupakan pengujian sistem oleh pengguna akhir (*User Acceptance Testing / UAT*) yang melibatkan mahasiswa, dosen wali, dan mitra. Mereka akan menggunakan sistem dalam skenario nyata untuk menilai kemudahan

penggunaan, akurasi data, dan efektivitas alur kerja. Jika ditemukan kekurangan seperti bug, error, atau fitur yang tidak sesuai kebutuhan, maka pengembang akan kembali memperbaiki sistem. Metode yang digunakan termasuk *Blackbox Testing* untuk melihat output yang dihasilkan berdasarkan input pengguna.

## 7. Penggunaan Sistem

Tahap akhir adalah implementasi sistem secara penuh di lingkungan kampus, di mana semua pihak yang terlibat mulai menggunakan aplikasi dalam kegiatan magang secara nyata. Mahasiswa akan mengisi *logbook* harian, mitra memvalidasi kehadiran, dan dosen wali memantau serta menilai kemajuan mahasiswa melalui aplikasi. Penggunaan sistem ini diharapkan dapat menggantikan sistem manual yang selama ini digunakan, sehingga proses magang menjadi lebih efisien. Setelah kebutuhan sistem teridentifikasi, dilakukan penyusunan *prototype* awal untuk memvisualisasikan bagaimana sistem akan bekerja. *Prototype* dirancang menggunakan alat bantu desain seperti Figma, kemudian dijadikan acuan dalam pengembangan aplikasi.

Fitur-fitur utama yang ditampilkan dalam *prototype* meliputi:

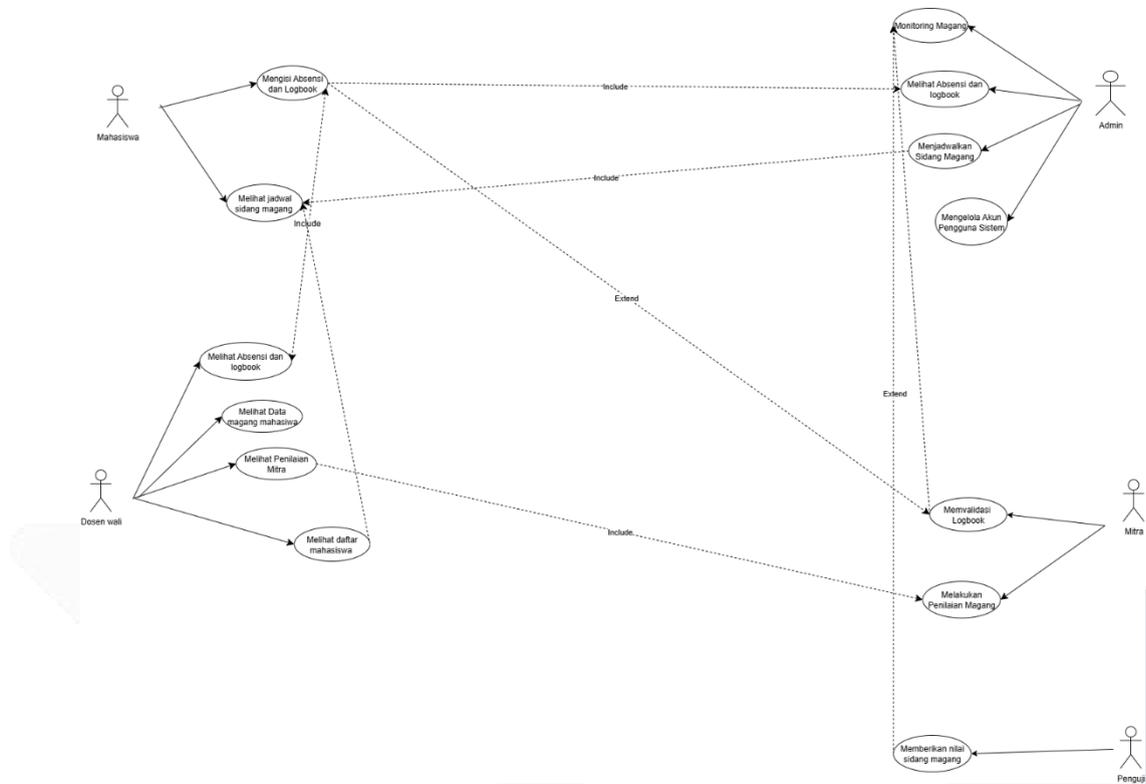
- a. Pengajuan tempat magang
- b. Pengisian *logbook* dan absensi harian
- c. Validasi *logbook* oleh mitra dan dosen wali
- d. input nilai dari penguji

Untuk memperkuat perancangan sistem, dilakukan pemodelan sistem menggunakan *Unified Modeling Language* (UML), yang terdiri dari:

### 4.2.1 Use Case Diagram

*Use Case Diagram* menggambarkan hubungan antara aktor (mahasiswa, dosen wali, admin, mitra dan penguji) dengan fitur utama sistem. Diagram ini menjadi dasar untuk memahami ruang lingkup interaksi pengguna dengan sistem, berikut menggambarkan alur *use case* dalam pengembangan sistem informasi magang berbasis Android yang digunakan dalam penelitian ini.

Berikut ini adalah Gambar 4.2 yang menunjukkan *Use Case Diagram*.



Gambar 4. 2 *Use Case Diagram*

Dari gambar diatas, Mahasiswa memiliki beberapa peran utama dalam sistem. Mahasiswa dapat mengisi absensi dan *logbook*, yang kemudian akan dimonitor oleh admin serta dilihat oleh dosen wali. Selain itu, mahasiswa dapat melihat jadwal sidang magang, yang informasinya dikelola oleh admin. Fitur-fitur ini membantu mahasiswa dalam menjalankan proses magang secara terstruktur, mulai dari dokumentasi kegiatan harian hingga mengikuti sidang akhir magang. Dosen wali memiliki akses terhadap berbagai informasi terkait mahasiswa bimbingannya. Mereka dapat melihat absensi dan *logbook*, melihat data magang mahasiswa, serta melihat penilaian dari mitra, yang mendukung proses monitoring dan pembimbingan. Selain itu, dosen wali juga dapat melihat daftar mahasiswa yang sedang melaksanakan magang, guna memudahkan pengawasan secara keseluruhan.

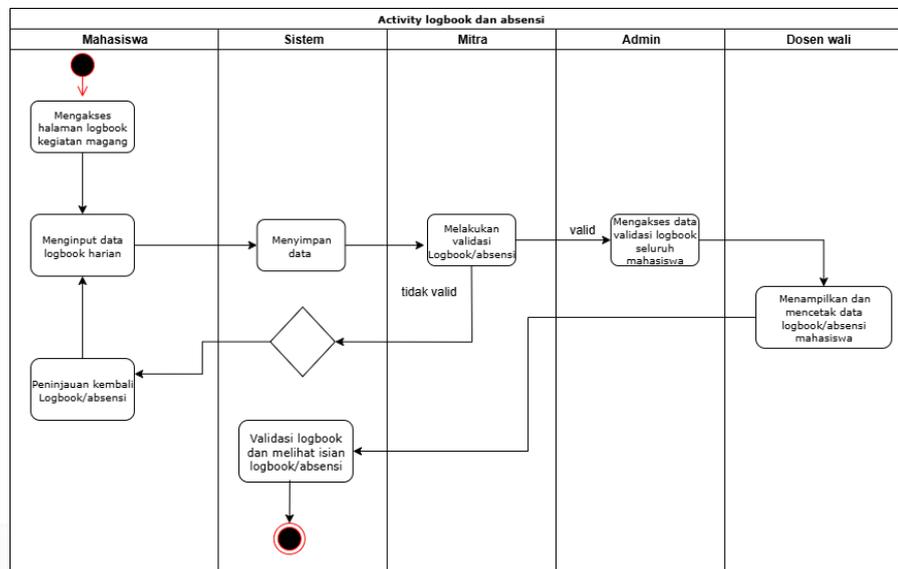
Admin berperan penting dalam pengelolaan sistem. Mereka dapat melakukan monitoring magang secara keseluruhan, termasuk melihat absensi dan *logbook* mahasiswa. Admin juga bertanggung jawab untuk melihat jadwal sidang magang, memastikan seluruh proses berjalan sesuai jadwal. Mitra sebagai pihak industri atau perusahaan tempat mahasiswa magang, memiliki tanggung jawab untuk memvalidasi *logbook* dan absensi mahasiswa. Setelah validasi, mitra juga dapat melakukan penilaian magang, yang akan menjadi salah satu indikator kinerja mahasiswa selama magang. Fitur ini sangat penting dalam proses evaluasi karena memberikan perspektif langsung dari dunia industri.

Penguji adalah aktor yang bertugas untuk memberikan nilai sidang magang kepada mahasiswa. Meskipun tidak terlibat langsung dalam proses harian, peran penguji sangat penting di tahap akhir magang karena menentukan hasil akhir dari pelaksanaan magang berdasarkan presentasi dan laporan mahasiswa.

#### **4.2.2 Activity Diagram**

Activity Diagram digunakan untuk menjelaskan alur proses bisnis utama dalam sistem, seperti proses *apply* magang, verifikasi *logbook*, serta pelaksanaan sidang magang. Diagram ini membantu pengembang dalam memahami skenario dan urutan aktivitas yang terjadi. *Activity diagram* menjelaskan alur proses dari aplikasi pengelolaan kegiatan magang mahasiswa. Terdapat tiga aktivitas dalam diagram ini yang mencakup peran mahasiswa, administrator, mitra, dosen wali, dan penguji.

Berikut ini Gambar 4.3 *Activity Logbook* dan absensi dalam pengembangan sistem informasi magang berbasis *Android*.



Gambar 4. 3 *Activity Logbook* dan absen

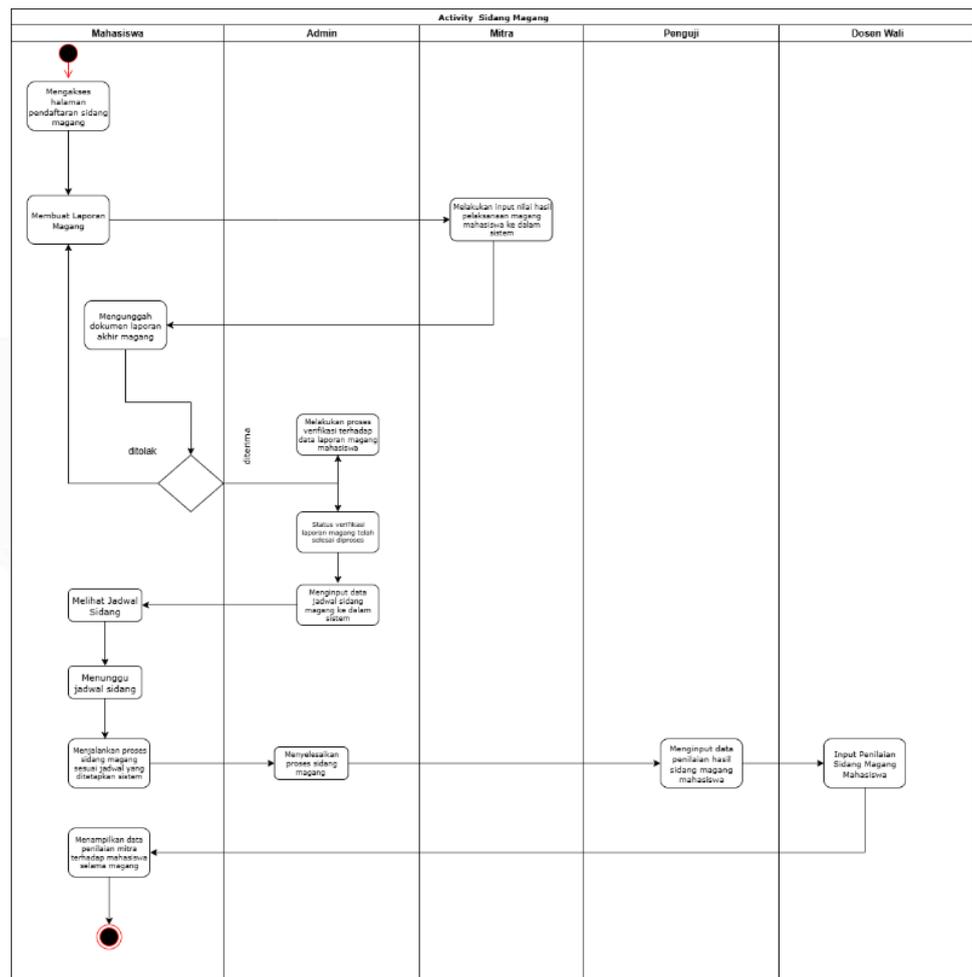
Dari gambar tersebut terlihat bahwa proses *logbook* dan absensi dilakukan sebagai bagian dari tahapan kegiatan magang.

a. *Aktivitas logbook* dan absensi

*Aktivitas logbook* dan absensi menjelaskan proses pencatatan kegiatan harian mahasiswa selama menjalani magang. Langkah-langkah dimulai ketika mahasiswa mengakses halaman sistem untuk mencatat aktivitas mereka setiap hari. Informasi yang dimasukkan oleh mahasiswa kemudian disimpan dalam sistem dan divalidasi secara otomatis. Apabila data *logbook* dan absensi tidak sesuai, sistem akan meminta mahasiswa untuk memperbaiki kesalahan dalam pengisian. Namun, jika data dianggap valid, mitra akan melakukan verifikasi terhadap isi *logbook* tersebut. Selanjutnya, admin bertugas mengarsipkan dan mengelola data *logbook* serta absensi yang telah diverifikasi. Dosen wali juga memiliki peran penting dalam meninjau serta mengevaluasi *logbook* dan absensi mahasiswa melalui tampilan antarmuka sistem. Diagram ini memperlihatkan keterlibatan lima aktor utama: mahasiswa, sistem, mitra, admin, dan dosen wali yang bekerja secara terintegrasi

untuk menjamin keakuratan, validitas, dan kelengkapan data kegiatan magang mahasiswa. Proses ini mendukung prinsip transparansi dan akuntabilitas dalam pelaporan kegiatan magang.

Berikut ini adalah Gambar 4.4 yang menunjukkan *Activity Diagram pada sidang magang*.



Gambar 4. 4 *Activity Diagram Sidang*

b. Aktivitas sidang magang

*Activity diagram* ini menggambarkan alur pelaksanaan sidang magang mahasiswa sebagai tahap akhir dari keseluruhan proses magang. Proses dimulai dari mahasiswa yang mengakses halaman pendaftaran sidang magang dan mengunggah dokumen laporan akhir magang ke dalam sistem. Admin akan melakukan verifikasi

terhadap dokumen yang diunggah. Jika dokumen ditolak, mahasiswa akan diberitahukan untuk melakukan perbaikan dan mengunggah ulang. Namun jika dokumen disetujui, maka admin akan menginput data jadwal sidang ke dalam sistem, yang kemudian dapat diakses oleh mahasiswa. Setelah menunggu jadwal, mahasiswa mengikuti sidang yang telah ditentukan. Pada tahap ini, mitra melakukan input nilai magang mahasiswa ke dalam sistem, sedangkan penguji dan dosen wali turut memberikan penilaian terhadap hasil sidang melalui sistem. Nilai tersebut akan dikompilasi sebagai hasil akhir penilaian magang mahasiswa. Diagram ini menampilkan keterlibatan semua pihak yang memiliki otoritas akademik dan administratif dalam proses penilaian akhir, serta menjamin objektivitas dan transparansi dalam evaluasi kinerja mahasiswa selama masa magang.

#### **4.2.3 Entity Relationship Diagram**

ERD ini menggambarkan sistem informasi manajemen magang mahasiswa berbasis *mobile* yang melibatkan berbagai peran, yaitu mahasiswa, mitra (tempat magang), admin, dan dosen wali. Entitas utama dalam sistem ini meliputi mahasiswa, mitra, lowongan, *apply\_mitra*, *logbook*, *jadwal\_sidang*, *penilaian\_magang*, serta beberapa entitas pendukung lainnya seperti *users*, *roles*, dan *dokumen\_magang*. Mahasiswa adalah pusat dari sistem ini, di mana mereka dapat melihat dan melamar lowongan magang dari mitra melalui entitas *apply\_mitra*. Setelah magang selesai, mitra memberikan penilaian melalui entitas *penilaian\_magang*, dan mahasiswa dapat mengikuti sidang magang yang dijadwalkan dalam entitas *jadwal\_sidang*. Dosen wali memiliki peran penting dalam melakukan monitoring terhadap mahasiswa melalui entitas *monitoring\_mahasiswa*. Sementara itu, admin berperan dalam pengelolaan data pengguna, pengaturan peran, dan pengelolaan referensi seperti jurusan, prodi, *tahun\_ajaran*, dan lainnya. Sistem ini juga mencatat histori pekerjaan, sesi login (*sessions*), migrasi database, dan validasi token *reset password*.



Berikut ini adalah Gambar 4.6 yang menunjukkan Rancang Antar Muka *splash*.



Gambar 4. 6 Rancang Antar Muka splash

Pada gambar diatas merupakan halaman awal dari *Aplikasi* magang yang ditampilkan saat pengguna pertama kali membuka aplikasi. Tampilan ini menyajikan sambutan sederhana berupa teks “Selamat Datang” dan sebuah tombol navigasi bertuliskan “Masuk ke akun anda” yang berfungsi untuk mengarahkan pengguna ke halaman login. Desainnya dibuat minimalis dengan latar belakang biru muda agar memberikan kesan ramah dan mudah dipahami oleh semua pengguna, baik mahasiswa, mitra, dosen wali, maupun penguji. Berikut ini adalah Gambar 4.7 yang menunjukkan Rancang Antar Muka *login*.



Gambar 4. 7 Rancang Antar Muka *login*

Setelah pengguna menekan tombol masuk, maka akan diarahkan ke halaman *login*. Di halaman ini, pengguna diminta untuk mengisi dua field utama, yaitu "*Username*" dan "*Password*", sebagai syarat untuk dapat mengakses sistem. Pesan "Selamat Datang" tetap ditampilkan di bagian atas untuk memberikan sambutan yang konsisten dan ramah kepada pengguna. Desain halaman ini dibuat minimalis dan mudah dipahami agar mempermudah proses autentikasi.

Berikut ini adalah Gambar 4.8 yang menunjukkan Rancang Antar muka *Dashboard* Mahasiswa.



Gambar 4. 8 Rancang Antar Muka *Dashboard* Mahasiswa

Pada gambar diatas merupakan halaman dashboard yang ditampilkan untuk pengguna dengan peran sebagai mahasiswa. Setelah berhasil *login*, mahasiswa diarahkan ke tampilan ini yang berisi beberapa menu utama seperti *History Apply*, *Logbook*, Absensi, Jadwal Sidang, dan Penilaian. Fitur *Logbook* dan Absensi memungkinkan mahasiswa mencatat aktivitas harian serta kehadiran selama magang, sedangkan fitur lainnya mendukung proses pelaporan dan evaluasi secara *digital*. Gambar di atas menunjukkan bahwa aktivitas *logbook* dan absensi termasuk dalam alur sistem.

Berikut ini adalah Gambar 4.9 yang menunjukkan Rancang Antar muka *Dashboard* Penguji.



Gambar 4. 9 *Dashboard* Penguji

Pada gambar diatas merupakan halaman *dashboard* untuk pengguna dengan peran sebagai penguji siding magang. Tampilan ini menyediakan tiga fitur utama, yaitu Jadwal Sidang, Penilaian Sidang, dan Laporan. Fitur ini membantu penguji dalam memantau jadwal mahasiswa yang diuji, memberikan nilai berdasarkan performa sidang, serta mengakses laporan akhir. Dengan desain yang sederhana namun fungsional, halaman ini mendukung kelancaran proses evaluasi akademik dalam program magang.

Berikut ini adalah Gambar 4.10 yang menunjukkan Rancang Antar *Dashboard* Mitra.



Gambar 4. 10 *Dashboard* Mitra

Pada gambar diatas merupakan halaman dashboard yang ditujukan untuk mitra atau perusahaan tempat mahasiswa melaksanakan magang. *Dashboard* ini menampilkan tiga fitur utama, yaitu *Data Logbook*, *Data Absensi*, dan *Penilaian*. Melalui halaman ini, mitra dapat memeriksa dan memverifikasi catatan kegiatan serta kehadiran mahasiswa selama magang berlangsung, sekaligus memberikan penilaian akhir. Gambar di atas menunjukkan bahwa aktivitas *logbook* dan absensi termasuk dalam alur sistem.

Berikut ini adalah Gambar 4.11 yang menunjukkan Rancang Antar *Dashboard* Dosen Wali.



Gambar 4. 11 *Dashboard* Dosen Wali

Pada gambar diatas merupakan halaman dashboard untuk pengguna dengan peran sebagai dosen wali. Dalam halaman ini tersedia tiga menu utama, yaitu Data Mahasiswa, Data Magang, dan Jadwal Sidang. Dosen wali dapat menggunakan halaman ini untuk memantau perkembangan mahasiswa bimbingan, meninjau data magang yang sedang berjalan, serta melihat jadwal sidang magang. Gambar di atas menunjukkan bahwa aktivitas *logbook* dan absensi termasuk dalam alur sistem, karena dosen wali turut terlibat dalam memvalidasi dan mengawasi aktivitas mahasiswa selama proses magang berlangsung.

Berikut ini adalah Gambar 4.12 yang menunjukkan Rancang Antar Muka *Dashboard Admin*.



Gambar 4. 12 *Dashboard Admin*

Pada gambar diatas merupakan halaman Dashboard Admin yang ditampilkan setelah pengguna dengan peran sebagai admin berhasil masuk ke dalam sistem. Pada halaman ini, admin diberikan akses penuh terhadap data pengguna dan pengelolaan sistem magang. Terdapat lima menu utama, yaitu Mahasiswa, Mitra, Komisi Magang, Admin, dan Penguji. Masing-masing menu memungkinkan admin untuk mengelola data dan akun pengguna sesuai dengan peran masing-masing, termasuk menambahkan, mengedit, atau menghapus data.

### 4.3 Hasil Evaluasi Prototype

Setelah *prototype* awal sistem berhasil disusun, dilakukan uji coba secara terbatas kepada perwakilan pengguna yang terdiri dari mahasiswa, dosen wali, dan admin. Pengujian ini dilakukan dengan cara mensimulasikan penggunaan antarmuka aplikasi dan mengamati bagaimana masing-masing peran menjalankan fungsinya di dalam sistem. Para pengguna diminta untuk mengeksplorasi *prototype* secara langsung serta memberikan umpan balik terkait beberapa aspek utama, seperti tampilan antarmuka *aplikasi*, navigasi menu, kelengkapan fitur yang

tersedia, dan tingkat kemudahan dalam penggunaan. Berdasarkan hasil evaluasi yang diperoleh, sebagian besar responden menyatakan bahwa keberadaan *prototype* ini sangat membantu mereka dalam memahami alur sistem yang akan dikembangkan. Namun demikian, pengguna juga memberikan sejumlah masukan yang konstruktif, di antaranya adalah perlunya penyederhanaan tampilan *logbook* agar lebih ringkas dan mudah diisi oleh mahasiswa, serta penambahan fitur *notifikasi* yang dapat memberikan informasi *real-time* mengenai status verifikasi dokumen atau *logbook* yang telah diajukan. Masukan tersebut menjadi dasar penting dalam proses iterasi desain berikutnya, agar sistem dapat lebih responsif terhadap kebutuhan pengguna dan mampu memberikan pengalaman yang optimal dalam proses pengelolaan magang.

#### 4.4 Hasil Pengkodean Prototype

Tahap ini merupakan fase implementasi sistem berdasarkan *prototype* yang sebelumnya telah divalidasi melalui evaluasi pengguna. Setelah mendapatkan umpan balik yang memadai, proses pengkodean dilaksanakan untuk mengubah desain statis menjadi aplikasi fungsional. Pengembangan sistem dilakukan dengan menggunakan *Android Studio* sebagai lingkungan pemrograman utama untuk sisi *frontend* atau antarmuka pengguna, sedangkan pada sisi *backend*, digunakan *framework Laravel* untuk membangun layanan *Application Programming Interface (API)* yang mendukung komunikasi data secara *real-time* antara aplikasi dan server.

Seluruh fitur yang telah dirancang dalam tahap *prototype* mulai diimplementasikan secara teknis, di antaranya fitur *login* dengan otentikasi *multi-level* yang membedakan hak akses antara mahasiswa, mitra, dosen wali, admin dan penguji. Sistem ini juga dilengkapi dengan mekanisme validasi berbasis peran (*role-based validation*), di mana setiap aktor dalam sistem memiliki hak dan tanggung jawab sesuai peran masing-masing. Dengan pendekatan ini, *aplikasi* yang dikembangkan tidak hanya memenuhi kebutuhan fungsional, tetapi juga menjaga keamanan, efisiensi, dan keteraturan dalam pengelolaan data magang.



Berikut ini adalah Gambar 4.14 yang menunjukkan Hasil Tampilan *Software Signup*



Gambar 4. 14 Tampilan *Software (Signup)*

Pada Gambar 4.14 menunjukkan halaman sambutan yang memperkenalkan aplikasi magang kepada pengguna. Melalui tampilan yang informatif, halaman ini mengajak pengguna untuk memulai perjalanan magang mereka dengan menonjolkan tiga keunggulan utama, yaitu kesempatan magang berkualitas, akses ke komunitas yang luas, serta peluang karir yang lebih cerah di masa depan. Pengguna diberikan opsi untuk melanjutkan ke proses *login* melalui tombol “Masuk ke Akun Anda,” yang menjadi pintu akses ke fitur-fitur *aplikasi*. Selain itu, halaman ini juga menyampaikan informasi bahwa dengan bergabung, pengguna secara otomatis menyetujui syarat, ketentuan, dan kebijakan *privasi aplikasi*, memastikan aspek legalitas penggunaan layanan. Dengan demikian, halaman ini tidak hanya menjadi pengantar *visual* tetapi juga memiliki fungsi penting dalam mengarahkan dan mendukung pengguna sebelum mereka masuk lebih jauh ke dalam *aplikasi*.

Berikut ini adalah Gambar 4.15 yang menunjukkan pengkodean untuk halaman *Login*.

```
Text(
  'Selamat Datang! 🌟',
  style: TextStyle(
    fontSize: 24,
    fontWeight: FontWeight.bold,
    color: Colors.blue[800],
  ), // TextStyle
), // Text
 SizedBox(height: 8),
Text(
  'Masuk untuk melanjutkan ke akun Anda',
  style: TextStyle(
    fontSize: 14,
    color: Colors.grey[600],
  ), // TextStyle
), // Text
SizedBox(height: 32),

// Username Field
_buildModernTextField(
  controller: _usernameController,
  label: 'NPW/NID/NIB/Email',
  icon: Icons.person_outline_rounded,
  isPassword: false,
),
SizedBox(height: 20),

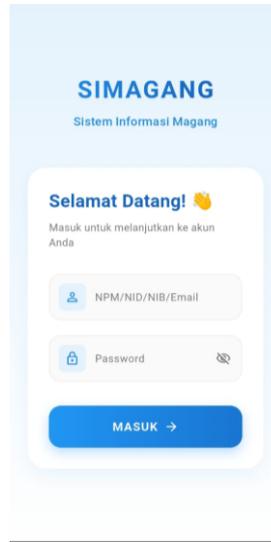
// Password Field
_buildModernTextField(
  controller: _passwordController,
  label: 'Password',
  icon: Icons.lock_outline_rounded,
  isPassword: true,
),
SizedBox(height: 32),

// Login Button
Container(
  width: double.infinity,
  height: 56,
```

Gambar 4. 15 Kode Tampilan Login

Pada Gambar 4.15 menunjukkan halaman bagian bawah tampilan halaman utama aplikasi yang menampilkan informasi persetujuan pengguna terhadap syarat dan kebijakan privasi. Komponen diawali dengan *widget SizedBox* untuk memberikan jarak *vertikal* dari elemen sebelumnya, lalu dilanjutkan dengan *Container* yang membungkus seluruh isi menggunakan padding dan dekorasi *visual*. Dekorasi tersebut menggunakan *BoxDecoration* dengan *LinearGradient* berwarna biru muda transparan, *borderRadius* untuk menciptakan sudut melengkung, serta border berwarna lembut untuk membentuk tampilan seperti kartu. Di dalamnya terdapat *Row* yang menyusun ikon dan teks secara *horizontal*. Ikon *info\_outline\_rounded* ditempatkan di dalam *Container* kecil berwarna biru muda transparan dengan sudut membulat, sehingga tampak seperti ikon informasi dalam latar berbentuk lingkaran. Di sebelahnya, digunakan *Expanded* agar teks dapat memanfaatkan ruang secara fleksibel.

Berikut ini adalah Gambar 4.16 yang menunjukkan Tampilan *Software login*



Gambar 4. 16 Tampilan *Software (Login)*

Pada Gambar 4.16 menunjukkan halaman *login* untuk sistem aplikasi magang yang bernama SIMAGANG (Sistem Informasi Magang). Pada halaman ini, pengguna diminta untuk memasukkan kredensial mereka berupa NPM, NID, NIB, atau email, serta *password*, agar dapat mengakses akun masing-masing. Fitur *input* dilengkapi dengan ikon yang memudahkan identifikasi kolom, serta tombol untuk menampilkan atau menyembunyikan *password* demi menjaga keamanan data pengguna. Tombol MASUK di bagian bawah menjadi aksi utama yang akan memproses data login dan mengarahkan pengguna ke halaman berikutnya bila autentikasi berhasil. Dengan demikian, halaman ini memiliki fungsi penting sebagai gerbang keamanan sistem, memastikan bahwa hanya pengguna terdaftar yang dapat mengakses informasi dan fitur yang tersedia di dalam *aplikasi*.

Berikut ini adalah Gambar 4.17 yang menunjukkan pengkodean untuk halaman *Dashboard* Dosen.

```
child: Row(
  mainAxisAlignment: MainAxisAlignment.spaceEvenly,
  children: [
    _buildNavItem(Icons.home_rounded, 'Beranda', 0, () {}),
    _buildNavItem(Icons.people_alt_rounded, 'Mahasiswa', 1, () {
      if (mounted) {
        Navigator.push(context, MaterialPageRoute(builder: (context) => MahasiswaBimbinganListPage()));
      }
    }),
    _buildNavItem(Icons.work_rounded, 'Magang', 2, () {
      if (mounted) {
        Navigator.push(context, MaterialPageRoute(builder: (context) => DataMagangListPage()));
      }
    }),
    _buildNavItem(Icons.book_rounded, 'Logbook', 3, () {
      if (mounted) {
        Navigator.push(context, MaterialPageRoute(builder: (context) => LogbookListPageDosen()));
      }
    }),
  ],
), // Row
; // Container

get _buildNavItem(IconData icon, String label, int index, VoidCallback onTap) {
  final isSelected = _currentIndex == index;
  return GestureDetector(
    onTap: () {
      setState(() {
        _currentIndex = index;
      });
      onTap();
    },
    child: Container(
      padding: EdgeInsets.symmetric(vertical: 8, horizontal: 10),
      decoration: BoxDecoration(
        color: isSelected ? Colors.blue[600] : Colors.transparent,
        borderRadius: BorderRadius.circular(16),
      ), // BoxDecoration
      child: Column(
        mainAxisAlignment: MainAxisAlignment.min,
        mainAxisAlignment: MainAxisAlignment.center,
        children: [
```

Gambar 4. 17 Kode Tampilan *Dashboard Dosen*

Pada Gambar 4.17 menunjukkan halaman Aplikasi Flutter berdasarkan nilai dari variabel route. Saat pengguna memilih salah satu menu seperti “logbook”, “absensi”, atau “mahasiswa”, aplikasi akan menentukan halaman tujuan yang sesuai menggunakan struktur switch-case. Setiap case memetakan nilai route ke halaman tertentu, misalnya 'logbook' akan diarahkan ke LogbookListPageDosen(). Jika halaman yang dituju tidak null dan widget masih aktif (mounted), maka proses navigasi dilakukan menggunakan Navigator.push() dengan MaterialPageRoute. Di dalamnya, digunakan efek transisi SlideTransition yang membuat halaman baru muncul dengan animasi geser dari kanan ke kiri. Hal ini menjelaskan mengapa saat pengguna mengetuk menu pada antarmuka, halaman berpindah dengan efek animasi yang halus dan menarik.

Berikut ini adalah Gambar 4.18 yang menunjukkan pengkodean untuk halaman Dashboard Dosen Wali .



Gambar 4. 18 Tampilan *Software* (Dashboard Dosen Wali)

Pada Gambar 4.18 menunjukkan Halaman *dashboard* untuk Dosen Wali dalam aplikasi SIMAGANG (Sistem Informasi Magang). Fungsinya adalah sebagai pusat informasi bagi dosen wali untuk memantau aktivitas mahasiswa bimbingan. Selain itu, terdapat menu utama yang memudahkan akses langsung ke fitur Data Mahasiswa dan Data Magang, serta navigasi di bagian bawah layar yang memungkinkan dosen wali berpindah cepat antar halaman seperti Beranda, Mahasiswa, Magang, dan *Logbook*. Halaman ini berfungsi untuk membantu dosen wali memantau, mengelola, dan mendukung proses magang mahasiswa secara lebih terstruktur dan efisien.

Berikut ini adalah Gambar 4.19 yang menunjukkan pengkodean untuk halaman *Dashboard Admin*.

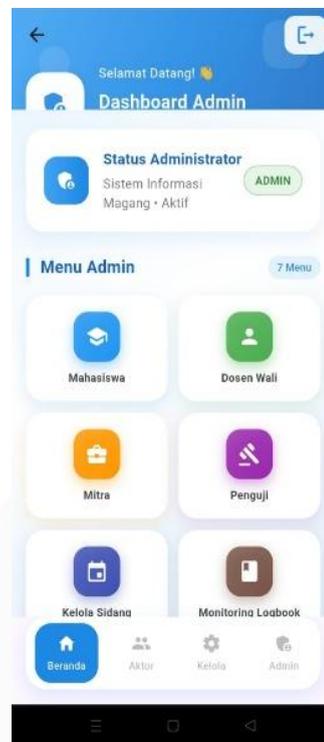
```
Text(
  'Sistem Informasi Magang + Aktif',
  style: TextStyle(
    fontSize: 14,
    color: Colors.grey[600],
  ), // TextStyle
), // Text
), // Column
), // Expanded
Container(
  padding: EdgeInsets.symmetric(horizontal: 12, vertical: 6),
  decoration: BoxDecoration(
    color: Colors.green[50],
    borderRadius: BorderRadius.circular(20),
    border: Border.all(color: Colors.green[200] ?? Colors.green.shade200),
  ), // BoxDecoration
  child: Text(
    'ADMIN',
    style: TextStyle(
      fontSize: 12,
      fontWeight: FontWeight.bold,
      color: Colors.green[700],
    ), // TextStyle
  ), // Text
), // Container
), // Row
), // Container
); // Transform.scale
); // AnimatedBuilder
}

final menuItems = [
  {'title': 'Mahasiswa', 'icon': Icons.school_rounded, 'color': Colors.blue, 'page': StudentManagementPage()},
  {'title': 'Dosen Wali', 'icon': Icons.person_rounded, 'color': Colors.green, 'page': DosenWaliManagementPage()},
  {'title': 'Mitra', 'icon': Icons.business_center_rounded, 'color': Colors.orange, 'page': MitraManagementPage()},
  {'title': 'Penguji', 'icon': Icons.gavel_rounded, 'color': Colors.purple, 'page': PengujiManagementPage()},
  {'title': 'Jadwal Sidang Magang', 'icon': Icons.event_rounded, 'color': Colors.indigo, 'page': SidangManagementPage()},
  {'title': 'Monitoring Logbook', 'icon': Icons.book_rounded, 'color': Colors.brown, 'page': ListLogbookAdmin()},
  {'title': 'Monitoring Absensi', 'icon': Icons.checklist_rounded, 'color': Colors.pink, 'page': AbsensiManagementPage()},
];
```

Gambar 4. 19 Kode Tampilan *Dashboard Admin*

Pada Gambar 4.19 menunjukkan Halaman informasi status akun admin yang ditampilkan menggunakan Container dengan dekorasi berwarna hijau muda, teks “ADMIN” ditulis dengan gaya tebal dan warna hijau tua untuk menandakan status aktif. Selanjutnya, bagian menu dibuat dengan daftar yang berisi beberapa pilihan seperti Mahasiswa, Dosen Wali, Mitra, Penguji, dan lainnya. Setiap item menu memiliki ikon, warna, dan halaman tujuan yang berbeda. Tampilan menu ini kemungkinan disusun dalam bentuk grid atau kartu, sehingga tampil rapi seperti di gambar. Intinya, kombinasi Container, TextStyle, dan daftar menu tersebut menghasilkan antarmuka dashboard yang informatif dan interaktif.

Berikut ini adalah Gambar 4.20 yang menunjukkan Tampilan *Software Dashboard* admin



Gambar 4. 20 Tampilan *Software (Dashboard Admin)*

Pada Gambar 4.20 menunjukkan Halaman *dashboard* untuk Administrator. Fungsinya adalah memberikan kontrol penuh kepada admin untuk mengelola seluruh data dan aktivitas dalam sistem magang. Halaman ini menampilkan status akun admin yang aktif, serta menyediakan menu administrasi yang lebih lengkap. Menu-menu yang tersedia meliputi pengelolaan data mahasiswa, dosen wali, mitra, penguji, pengelolaan sidang, hingga fitur monitoring *logbook*. Hal ini memungkinkan admin memantau keseluruhan proses magang, memastikan data selalu terbaru, dan memfasilitasi koordinasi antar pihak yang terlibat. Dengan tampilan yang terstruktur dan ikon yang memudahkan identifikasi, *dashboard* admin berfungsi sebagai pusat kendali untuk menjaga kelancaran dan keakuratan pengelolaan sistem informasi magang.

Berikut ini adalah Gambar 4.21 yang menunjukkan pengkodean untuk halaman *Dashboard Mitra*.

```
idget_buildMenuSection() {
  final menuItems = [
    {'title': 'Data Logbook', 'icon': Icons.book_rounded, 'color': Colors.green, 'route': 'logbook'},
    {'title': 'Data Absensi', 'icon': Icons.fingerprint_rounded, 'color': Colors.red, 'route': 'absensi'},
    {'title': 'Kelola Penilaian', 'icon': Icons.rate_review_rounded, 'color': Colors.orange, 'route': 'penilaian'},
  ];
}

return Column(
  crossAxisAlignment: CrossAxisAlignment.start,
  mainAxisAlignment: MainAxisAlignment.min,
  children: [
    Row(
      children: [
        Container(
          width: 4,
          height: 18,
          decoration: BoxDecoration(
            color: Colors.blue[600],
            borderRadius: BorderRadius.circular(2),
          ), // BoxDecoration
        ), // Container
        SizedBox(width: 10),
        Text(
          'Menu Utama',
          style: TextStyle(
            fontSize: 16,
            fontWeight: FontWeight.bold,
            color: Colors.blue[800],
          ), // TextStyle
        ), // Text
        Spacer(),
        Container(
          padding: EdgeInsets.symmetric(horizontal: 8, vertical: 3),
          decoration: BoxDecoration(
            color: Colors.blue[50],
            borderRadius: BorderRadius.circular(12),
          ), // BoxDecoration
          child: Text(
            '3 Menu',
            style: TextStyle(
              fontSize: 10,
              color: Colors.blue[600],
              fontWeight: FontWeight.w500,
            ), // TextStyle
          ), // Text
        ), // Container
      ],
    ), // Row
  ], // Column
);
```

Gambar 4. 21 Kode Tampilan *Dashboard Mitra*

Pada Gambar 4.21 menunjukkan halaman fungsi `_buildMenuSection()` berperan untuk membuat Menu Utama yang berisi tiga pilihan menu: *Data Logbook*, *Data Absensi*, dan *Kelola Penilaian*. Masing-masing menu ditampilkan sebagai kotak dengan ikon, warna khusus, dan teks judul sesuai isi data yang ditangani. Judul "Menu Utama" ditampilkan bersama dengan indikator jumlah menu, yaitu "3 Menu", di sisi kanan. Elemen visual seperti garis vertikal biru di samping judul dan teks yang dicetak tebal digunakan untuk memperjelas tampilan. Tampilan ini tampak rapi karena menggunakan *Row*, *Column*, dan dekorasi seperti *BoxDecoration* dan *TextStyle*. Kombinasi struktur ini menjadikan antarmuka terlihat profesional dan mudah dipahami oleh pengguna.

Berikut ini adalah Gambar 4.22 yang menunjukkan Tampilan *Software* Mitra



Gambar 4. 22 Tampilan *Software* (*Dashboard* Mitra)

Pada Gambar 4.22 menunjukkan halaman *Dashboard* Mitra yang dirancang untuk supervisor magang dari pihak perusahaan. Di bagian atas, terdapat informasi nama perusahaan dan status pengguna sebagai “Supervisor Magang”. Sistem juga menampilkan status perusahaan yang sudah terverifikasi dan aktif. Di bawahnya ditampilkan data statistik seperti jumlah mahasiswa aktif, jumlah penilaian, dan jumlah *logbook* yang terhubung dengan perusahaan tersebut. Selanjutnya, tersedia tiga menu utama yaitu *Data Logbook*, *Data Absensi*, dan *Kelola Penilaian* yang berfungsi untuk mengelola aktivitas mahasiswa selama magang. Desain antarmuka tampak modern dengan elemen warna biru sebagai warna dominan dan navigasi bawah yang memudahkan pengguna mengakses halaman lain.

Berikut ini adalah Gambar 4.23 yang menunjukkan pengkodean untuk halaman *Dashboard* Mahasiswa.

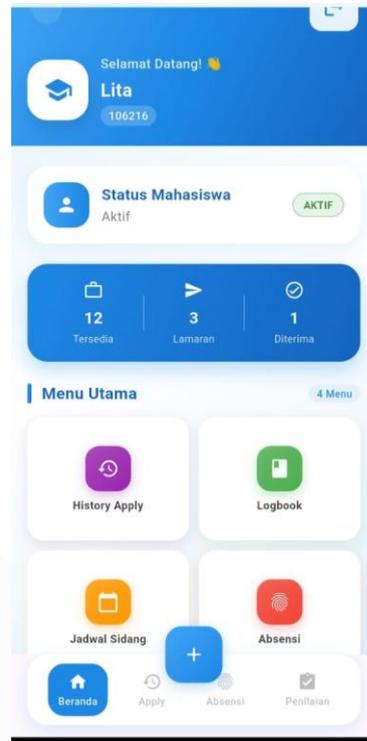
```
Widget buildMenuSection() {
  final menuItems = [
    {'title': 'History Apply', 'icon': Icons.history_rounded, 'color': Colors.purple, 'route': 'apply'},
    {'title': 'Logbook', 'icon': Icons.book_rounded, 'color': Colors.green, 'route': 'logbook'},
    {'title': 'Jadwal Sidang', 'icon': Icons.calendar_today_rounded, 'color': Colors.orange, 'route': 'jadwal'},
    {'title': 'Absensi', 'icon': Icons.fingerprint_rounded, 'color': Colors.red, 'route': 'absensi'},
    {'title': 'Penilaian', 'icon': Icons.assignment_turned_in_rounded, 'color': Colors.red, 'route': 'penilaian'},
  ];

  return Column(
    crossAxisAlignment: CrossAxisAlignment.start,
    mainAxisAlignment: MainAxisAlignment.min,
    children: [
      Row(
        children: [
          Container(
            width: 4,
            height: 20,
            decoration: BoxDecoration(
              color: Colors.blue[600],
              borderRadius: BorderRadius.circular(2),
            ), // BoxDecoration
          ), // Container
          SizedBox(width: 10),
          Expanded(
            child: Text(
              'Menu Utama',
              style: TextStyle(
                fontSize: 16,
                fontWeight: FontWeight.bold,
                color: Colors.blue[800],
              ), // TextStyle
            ), // Text
          ), // Expanded
          Container(
            padding: EdgeInsets.symmetric(horizontal: 8, vertical: 2),
            decoration: BoxDecoration(
              color: Colors.blue[50],
              borderRadius: BorderRadius.circular(15),
            ), // BoxDecoration
            child: Text(
              '4 Menu',
              style: TextStyle(
                fontSize: 10,
                color: Colors.blue[600],
                fontWeight: FontWeight.w500,
              ), // TextStyle
            ), // Text
          ), // Container
        ], // Row
      ), // Row
    ], // Column
  );
}
```

Gambar 4. 23 Kode Tampilan *Dashboard* Mahasiswa

Pada Gambar 4.23 menunjukkan “Menu Utama” pada antarmuka pengguna dengan daftar item menu dalam aplikasi Flutter. Menu ini berisi lima item, yaitu History Apply, Logbook, Jadwal Sidang, Absensi, dan Penilaian, yang masing-masing dilengkapi dengan ikon, warna, dan rute navigasi. Tampilan ini diawali dengan judul "Menu Utama" yang diberi garis biru di sebelah kirinya sebagai penanda visual, serta label jumlah menu seperti “4 Menu” di pojok kanan, yang ditampilkan dalam kotak biru muda. Elemen-elemen UI seperti Row, Text, Container, dan *BoxDecoration* digunakan untuk mengatur tata letak dan gaya visualnya agar tampil rapi dan menarik. Kode ini berfungsi untuk menyusun daftar menu secara terstruktur dan responsif sehingga pengguna mudah mengakses fitur-fitur utama yang tersedia di *aplikasi*.

Berikut ini adalah Gambar 4.24 yang menunjukkan Tampilan Dashboard *Software* Mahasiswa.



Gambar 4. 24 Tampilan *Software* (Dashboard Mahasiswa)

Pada Gambar 4.24 menunjukkan halaman Dashboard Mahasiswa. Halaman ini menyambut pengguna dengan sapaan personal dan menampilkan status mahasiswa yang aktif. Di bagian tengah, disediakan data ringkasan berupa jumlah lowongan magang yang tersedia, jumlah lamaran yang sudah diajukan, dan jumlah lamaran yang diterima. Menu utama terdiri dari *History Apply* untuk melihat riwayat pengajuan tempat magang, *Logbook* untuk mencatat kegiatan harian, *Jadwal Sidang* untuk melihat jadwal presentasi akhir, serta *Absensi* untuk mengisi kehadiran selama magang. Navigasi bawah memudahkan mahasiswa untuk berpindah antara fitur utama seperti *Apply*, *Absensi*, dan *Penilaian* dengan mudah dan cepat.

Berikut ini adalah Gambar 4.25 yang menunjukkan pengkodean untuk halaman *Dashboard* Penguji.

```
child: LayoutBuilder(
  builder: (context, constraints) {
    return Row(
      mainAxisAlignment: MainAxisAlignment.spaceAround,
      children: [
        SizedBox(width: 56),
        _buildNavItem(Icons.home_rounded, 'Beranda', 0, () {}),
        _buildNavItem(Icons.schedule_outlined, 'Jadwal', 2, () {
          Navigator.push(context, MaterialPageRoute(builder: (context) => const JadwalSidangPage()));
        }),
      ],
    ); // Row
  }, // LayoutBuilder
); // Container
}

Widget _buildNavItem(IconData icon, String label, int index, VoidCallback onTap) {
  final isSelected = _currentIndex == index;
  return GestureDetector(
    onTap: () {
      setState(() {
        _currentIndex = index;
      });
      onTap();
    },
    child: Container(
      padding: EdgeInsets.symmetric(vertical: 8, horizontal: 10),
      decoration: BoxDecoration(
        color: isSelected ? Colors.blue[600] : Colors.transparent,
        borderRadius: BorderRadius.circular(16),
      ), // BoxDecoration
      child: Column(
        mainAxisAlignment: MainAxisAlignment.min,
        children: [
          Icon(
            icon,
            color: isSelected ? Colors.white : Colors.grey[400],
            size: 20,
          ), // Icon
          SizedBox(height: 2),
          Text(
            label,
            style: TextStyle(
              fontSize: 10,
              fontWeight: FontWeight.w500,
              color: isSelected ? Colors.white : Colors.grey[400],
            ), // TextStyle
          ),
        ],
      ),
    ),
  );
}
```

Gambar 4. 25 Kode Tampilan *Dashboard* Penguji

Pada Gambar 4.25 menunjukkan navigasi bawah dalam aplikasi *Flutter*, yang memungkinkan pengguna berpindah antar halaman dengan mengetuk ikon di bagian bawah layar. Komponen *LayoutBuilder* digunakan untuk menyesuaikan tata letak berdasarkan ukuran layar. Di dalamnya, *Row* menampilkan dua menu: Beranda dan Jadwal. Setiap item navigasi dibentuk oleh fungsi *\_buildNavItem*, yang menerima ikon, label teks, indeks posisi, dan aksi (*onTap*) saat menu ditekan. Setiap item ditampilkan dalam *Container* yang akan berubah warna ketika dipilih (*isSelected*). Warna biru tua menandai menu aktif, sementara menu lainnya tetap transparan dan berwarna abu-abu. Saat pengguna menekan menu “Jadwal”, aplikasi akan menavigasi ke halaman *JadwalSidangPage* menggunakan *Navigator.push()*. Secara visual, ikon dan teks menu juga berubah warna sesuai status aktif atau tidak.

Kode ini menciptakan pengalaman navigasi yang sederhana dan intuitif, dengan tampilan yang bersih dan responsif sesuai standar desain modern.

Berikut ini adalah Gambar 4.26 yang menunjukkan Tampilan *Software* Penguji



Gambar 4. 26 Tampilan *Software* (Dashboard Penguji)

Halaman *Dashboard* Dosen Penguji. Halaman ini menampilkan identitas dosen sebagai penguji sidang magang dan menunjukkan status aktif. Di bagian atas disediakan data statistik terkait sidang seperti jumlah sidang hari ini, sidang yang masih menunggu penilaian, dan total sidang yang pernah dilakukan. Menu utama pada halaman ini meliputi Jadwal Sidang untuk melihat daftar sidang yang akan berlangsung, Penilaian Sidang untuk menginput nilai mahasiswa, Laporan Mahasiswa untuk meninjau dokumen laporan, dan Rekap Nilai untuk melihat ringkasan seluruh nilai yang sudah diberikan.

## 4.5 Hasil Evaluasi Sistem

Sistem yang telah dikembangkan kemudian dievaluasi secara menyeluruh dari berbagai aspek, meliputi aspek teknis, performa, dan fungsionalitas. Evaluasi ini bertujuan untuk memastikan bahwa seluruh modul, fitur, serta alur kerja sistem berjalan sesuai dengan desain awal dan benar-benar memenuhi kebutuhan pengguna yang telah diidentifikasi sebelumnya. Proses evaluasi dilakukan secara bertahap, dimulai dari pengujian internal oleh tim pengembang untuk menilai kestabilan sistem, mendeteksi bug, dan memastikan bahwa integrasi antar komponen telah berjalan dengan baik. Selanjutnya dilakukan validasi terhadap alur proses sistem, di mana pengembang memastikan bahwa setiap fitur merespons dengan benar terhadap input pengguna, serta menghasilkan *output* yang sesuai. Selain itu, aspek tampilan juga turut ditinjau ulang, termasuk kesesuaian tata letak antarmuka, kemudahan *navigasi*, dan keakuratan data yang ditampilkan. Hasil evaluasi ini menjadi dasar untuk melakukan perbaikan sebelum sistem diluncurkan kepada pengguna akhir, dengan harapan sistem dapat memberikan pengalaman yang optimal, stabil, dan bebas dari kesalahan fungsi.

## 4.6 Hasil Menguji Sistem

Pengujian dilakukan menggunakan dua pendekatan utama:

### 4.6.1 Blackbox Testing

Metode ini digunakan untuk memastikan bahwa sistem memberikan output yang benar berdasarkan input yang diberikan, tanpa memperhatikan *struktur internal* kode. Pengujian dilakukan pada fitur *login*, form *logbook*, upload dokumen, dan jadwal sidang.

Tabel 4. 1 *Blackbox Testing* Simagang

No	Aktor	Fitur yang Diuji	Hasil yang Diharapkan	Hasil Pengujian	Status
1	Mahasiswa	Login & akses dashboard	Mahasiswa dapat masuk dan melihat fitur utama	Sesuai	Valid
2	Mahasiswa	Input logbook harian	Mahasiswa dapat mengisi logbook	Sesuai	Valid
3	Mahasiswa	Unggah dokumen magang	Dokumen tersimpan dan dapat dilihat admin	Sesuai	Valid
4	Dosen Wali	Validasi logbook & dokumen mahasiswa	Data mahasiswa tampil dan dapat divalidasi	Sesuai	Valid
5	Mitra	Melihat data mahasiswa magang	Daftar mahasiswa tampil sesuai data	Sesuai	Valid

Berikut ini adalah Tabel 4.1 yang menunjukkan Tabel uji *Blackbox*

Tabel di atas merupakan hasil dari proses pengujian sistem menggunakan metode *blackbox testing*, yang bertujuan untuk mengevaluasi apakah setiap fitur dalam aplikasi berfungsi sesuai dengan yang diharapkan tanpa melihat struktur internal program. Pengujian ini melibatkan beberapa aktor, yaitu Mahasiswa, Dosen Wali, dan Mitra, yang masing-masing diuji berdasarkan peran dan fitur yang mereka akses. Mahasiswa diuji untuk fitur login dan akses *dashboard*, input *logbook* harian, serta unggah dokumen magang. Hasil pengujian menunjukkan bahwa seluruh fitur tersebut berjalan sesuai, yaitu mahasiswa dapat masuk ke sistem, mengisi logbook, dan mengunggah dokumen yang dapat diakses admin. Dosen Wali diuji dalam fitur validasi data mahasiswa, dan hasilnya menunjukkan

bahwa data tampil dengan benar dan dapat divalidasi. Sementara itu, Mitra diuji untuk fitur melihat data mahasiswa magang dan hasilnya sesuai dengan data yang tersedia di sistem. Semua fitur yang diuji menunjukkan hasil yang sesuai dengan ekspektasi dan dinyatakan valid, artinya fitur tersebut telah berhasil diuji dan siap digunakan secara fungsional. Pengujian ini membuktikan bahwa sistem telah memenuhi kebutuhan dasar pengguna dari berbagai peran dalam sistem informasi magang.

#### 4.6.2 User Acceptance Testing (UAT)

Pengujian *User Acceptance Testing* (UAT) skala Responden diminta menilai kemudahan penggunaan, tampilan, fungsi, dan kepuasan terhadap sistem dengan lima pilihan jawaban: Sangat Setuju (4), Setuju (3), Tidak Setuju (2), dan Sangat Tidak Setuju (1).

Tabel 4. 2 UAT Simagang

Indikator Penelitian	Butir Penelitian	Penelitian			
		1	2	3	4
Aspek fungsional	1. Apakah fitur pengisian logbook harian dapat digunakan dengan baik?				
	2. Apakah fitur absensi magang mahasiswa dapat mencatat kehadiran ?				
	3. Apakah penjadwalan sidang magang dapat diakses dengan mudah oleh mahasiswa?				
Aspek Desain	5. Apakah tampilan antarmuka sistem menarik dan mudah dipahami oleh mahasiswa?				
	6. Apakah navigasi antar menu (logbook, sidang, absensi) mudah digunakan?				
	7. Apakah sistem menyesuaikan dengan berbagai ukuran layar perangkat?				

<b>Aspek Teknis</b>	8. Apakah sistem berjalan stabil tanpa error saat mencatat logbook, absensi, dan melihat jadwal sidang?				
	9. Apakah waktu akses sistem dan perpindahan menu cukup cepat?				
<b>Aspek Kegunaan</b>	10. Apakah mahasiswa merasa sistem ini memudahkan proses pelaporan dan absensi?				
	11. Apakah dosen wali dan mitra merasa sistem ini membantu proses pemantauan dan penilaian?				
<b>Aspek Kepuasan Pengguna</b>	12. Apakah user merasa puas dengan fitur logbook, absensi, dan jadwal sidang dalam sistem ini				
	13. Apakah sistem ini dapat menggantikan sistem manual sebelumnya dengan lebih efektif?				

Keterangan :

1. Sangat tidak setuju
2. Tidak setuju
3. Setuju
4. Sangat Setuju

Berdasarkan pengelolaan data dari tanggapan 30 responden melalui kusioner, hasil perhitungan *User Acceptance Test* (UAT).

Tabel 4. 3 Tabel Perhitungan

Point	Jumlah	Skor	% Frekuensi
4	150	600	$\frac{600}{1230} \times 100\%$ = 48,7%
3	210	630	$\frac{630}{1230} \times 100\%$ = 51,3%
2	0	0	
1	0	0	
<b>TOTAL:</b>	360	1230	
<b>MAX:</b>	4x360	1440	
<b>INDEX%</b>	$\frac{1230}{1440} \times 100\%$ = 85,4%		

Berdasarkan hasil rekapitulasi kuesioner yang diperoleh dari total 360 isian responden, diketahui bahwa mayoritas responden memilih poin 3 (Setuju) sebanyak 210 kali (51,3%) dan poin 4 (Sangat Setuju) sebanyak 150 kali (48,7%). Tidak ada responden yang memilih poin 1 (Sangat Tidak Setuju) maupun poin 2 (Tidak Setuju), yang menunjukkan bahwa sistem yang dikembangkan mendapat respon positif dari pengguna. Total skor yang diperoleh dari seluruh responden adalah 1230, dengan skor maksimal yang mungkin dicapai sebesar 1440 (hasil dari 360 responden  $\times$  skor tertinggi 4). Nilai indeks sebesar 85,4% ini menunjukkan bahwa sistem masuk dalam kategori “Sangat Layak Digunakan”. Oleh karena itu, dapat disimpulkan bahwa pengembangan Sistem Pengelolaan Kegiatan Magang Mahasiswa Polman Negeri Babel Berbasis *Aplikasi* dengan Metode *Prototyping* telah berhasil dan sistem dinilai sangat layak untuk digunakan oleh pengguna sesuai dengan kebutuhan dan ekspektasi mereka.

#### 4.7 Penggunaan Sistem

Setelah seluruh tahapan pengujian selesai dan sistem dinyatakan layak, *aplikasi* mulai digunakan oleh mahasiswa, dosen wali, dan admin sebagai media resmi dalam mengelola kegiatan magang. Implementasi dilakukan secara bertahap dengan pendampingan dan pelatihan kepada pengguna, serta penyediaan dokumentasi panduan penggunaan *aplikasi*. Penggunaan sistem ini diharapkan mampu mengatasi permasalahan klasik dalam pengelolaan magang, meningkatkan efisiensi administrasi, dan mempercepat proses koordinasi antar pihak. Sistem juga terbuka untuk dikembangkan lebih lanjut sesuai kebutuhan pengguna di masa depan.



## **BAB V**

### **KESIMPULAN DAN SARAN**

#### **5.1 Kesimpulan**

Berdasarkan hasil pengembangan, implementasi, dan pengujian sistem informasi pengelolaan kegiatan magang mahasiswa berbasis *Android*, maka dapat disimpulkan hal-hal berikut ini:

1. Sistem berhasil dikembangkan untuk mendukung administrasi dan pemantauan magang mahasiswa, sesuai dengan tujuan awal yaitu menggantikan proses manual yang selama ini digunakan. *Aplikasi* ini menyediakan fitur seperti *logbook digital*, unggah dokumen, serta jadwal sidang yang terintegrasi.
2. *Aplikasi* memberikan kemudahan bagi mahasiswa dalam menjalankan proses magang, mulai dari mengakses informasi tempat magang, mencatat aktivitas harian, hingga mengunggah dokumen persyaratan. Semua proses ini kini dapat dilakukan secara digital dan lebih efisien.
3. Metode *prototyping* terbukti efektif diterapkan dalam pengembangan sistem ini, karena mampu menyesuaikan desain dan fitur aplikasi berdasarkan masukan langsung dari pengguna (mahasiswa, dosen wali, admin, dan mitra). Proses iteratif ini membuat aplikasi lebih tepat guna.
4. Pengujian sistem dengan metode *Black Box* dan UAT menunjukkan bahwa aplikasi berjalan sesuai harapan, dengan tingkat kepuasan pengguna sebesar 85,4%. Ini menunjukkan bahwa aplikasi diterima dengan baik oleh pengguna dan mampu menjawab kebutuhan nyata di lapangan.

Rumusan masalah dalam penelitian ini berhasil dijawab secara menyeluruh, baik dari segi pengembangan teknis, penerapan metode, maupun kemudahan akses informasi dan monitoring kegiatan magang.

#### **5.2 Saran**

Berdasarkan hasil pengembangan dan pengujian sistem informasi magang ini, penulis menyarankan agar aplikasi dapat terus dikembangkan dengan menyesuaikan kebutuhan pengguna di lapangan. Tampilan antarmuka perlu dibuat lebih menarik dan modern agar memberikan kenyamanan serta kemudahan bagi pengguna dalam mengoperasikan aplikasi. Selain itu, sistem pencarian dan pengelompokan data seperti *logbook*, jadwal sidang, maupun informasi mahasiswa juga perlu disempurnakan guna mempermudah dosen wali, mitra, dan admin dalam mengakses data secara cepat dan efisien. Agar proses penyampaian informasi dapat berjalan lebih optimal, aplikasi ini juga sebaiknya dilengkapi dengan fitur notifikasi otomatis yang memberikan pemberitahuan kepada setiap aktor terkait perubahan status, jadwal, atau dokumen yang perlu diunggah. agar dilakukan uji coba secara menyeluruh di lingkungan kampus dengan melibatkan lebih banyak pengguna dari berbagai peran. Hal ini bertujuan untuk mengevaluasi stabilitas sistem, mengidentifikasi kebutuhan tambahan, dan memastikan bahwa *aplikasi* benar-benar siap digunakan secara resmi dalam pengelolaan kegiatan magang di Politeknik Manufaktur Negeri Bangka Belitung.

## DAFTAR PUSTAKA

- [1] R. A. Verdinata, "Aplikasi Pengelolaan Magang Berbasis Web Menggunakan Metode Prototyping pada PT. Glory Industrial Semarang," *Web-Based International Management Application Using Prototyping Method at PT. Glory Industrial Semarang*, vol. 1, no. 1, pp. 1–4, 2023.
- [2] U. A. Yusa, *Bangka Belitung Tahun 2022*. 2022.
- [3] I. B. Trisno and Y. Hari, "Desain dan Analisa Sistem Magang di Prodi Teknik Informasi Universitas Widya Kartika Menggunakan UML," *Jurnal Nasional Komputasi dan Teknologi Informasi*, vol. 4, no. 6, pp. 490–500, 2021, doi: 10.32672/jnkti.v4i6.3682.
- [4] P. Kustanto, R. W. P. Pamungkas, and A. Fathurrozi, "Pembangunan Aplikasi E-Magang Perguruan Tinggi dengan Memanfaatkan SDLC Scrum pada Agile Project Management," vol. 2, no. 12, pp. 99–112, 2021.
- [5] I. Keilmuan, "Jurnal Saintiskom," vol. 2, no. 3, pp. 145–154, 2024.
- [6] H. D. Yulianto and R. B. Firdaus, "Perancangan Sistem Informasi Monitoring Magang," *IJIS - Indonesian Journal of Information System*, vol. 6, no. 2, pp. 130–136, 2021, doi: 10.36549/ijis.v6i2.144.
- [7] E. R. Ramadhan, K. Prihandani, A. Voutama, U. Singaperbangsa, and K. Abstract, "Penerapan Metode Agile pada Development Aplikasi Pengelolaan Data Magang Berbasis Web Menggunakan Framework Laravel," *Jurnal Ilmiah Wahana Pendidikan*, vol. 9, no. 7, pp. 144–154, 2023. [Online].
- [8] N. Sopiah and M. Al-Maudina, "Pengembangan Aplikasi Magang Mahasiswa untuk Membantu Proses Administrasi," *Jurnal Ilmiah Matrik*, vol. 24, no. 3, pp. 275–282, 2022. [Online].
- [9] S. D. Asri and K. P. Christiawan, "Rancang Bangun Aplikasi Pendaftaran Peserta Magang Berbasis Web Studi Kasus CV Sinar Pluit," *Information Management for Education and Professional*, vol. 8, no. 1, pp. 51–60, 2023.

- [10] S. Nasional and U. Desember, *Seminar Nasional UNRIYO*, pp. 159–169, Dec. 2021.
- [11] R. Aditya, V. H. Pranatawijaya, and P. B. A. A. Putra, “Rancang Bangun Aplikasi Monitoring Kegiatan Menggunakan Metode Prototype,” *Journal of Information Technology and Computer Science*, vol. 1, no. 1, pp. 47–57, 2021.
- [12] M. Alda, “Pengembangan Aplikasi Pengolahan Data Siswa Berbasis Android Menggunakan Metode Prototyping,” *Jurnal Manajemen Informatika (JAMIKA)*, vol. 13, no. 1, pp. 11–23, 2023, doi: 10.34010/jamika.v13i1.8216.
- [13] B. B. Apriansah, E. A. Saputra, and others, “Pengujian Aplikasi Absensi dan Kegiatan Magang Mahasiswa Menggunakan Metode Blackbox Testing,” *Jurnal Teknologi Informasi dan ...*, pp. 448–455, 2023. [Online].

## LAMPIRAN

### Lampiran 1 Daftar Riwayat Hidup Penulis

#### DAFTAR RIWAYAT HIDUP

##### 1. Data Pribadi

Nama lengkap : Lita Rahmania  
Tempat & tanggal lahir : Sungailiat, 20 Oktober 2004  
Alamat rumah : Jalan Ambalat 2 Air ruay, Pemali  
Telp : -  
Email : litarahmania10@gmail.com  
Jenis kelamin : Perempuan  
Agama : Islam



##### 2. Riwayat Pendidikan

SD Negeri 25 Sungailiat	2010 - 2016
SMP Negeri 1 Pemali	2016 - 2019
SMK Negeri 1 Sungailiat	2019- 2022
Politeknik Manufaktur Negeri Bangka Belitung	2022 – sekarang

##### 3. Pendidikan Non-Formal

-

Sungailiat, 03 Juli 2025

Lita Rahmania

## LAMPIRAN

### Lampiran 2 Sample Jawaban Kusiner

### Lembar Validasi Ahli Android

Pengembangan Sistem Pengelolaan Kegiatan Magang Mahasiswa Polman Negeri Babel Berbasis Android Dengan Metode Prototype

**NPM \***  
106202

**Nama \***  
Ajeng Oktaviani

**Kelas \***  
3 TRPL

**Program Studi \***  
D4 Teknologi Rekayasa Perangkat Lunak

Aspek fungsional

Apakah fitur pengisian logbook harian dapat digunakan dengan baik? \*

1      2      3      4

Sangat Tidak Setuju                    Sangat Setuju

Aspek Teknis

Apakah sistem berjalan stabil tanpa error saat mencatat logbook, absensi, dan melihat jadwal sidang? \*

	1	2	3	4	
Sangat Tidak Setuju	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	Sangat Setuju

Apakah waktu akses sistem dan perpindahan menu cukup cepat? \*

	1	2	3	4	
Sangat Tidak Setuju	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	Sangat Setuju

Aspek Kegunaan

Apakah mahasiswa merasa sistem ini memudahkan proses pelaporan dan absensi? \*

	1	2	3	4	
Sangat Tidak Setuju	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	Sangat Setuju

Apakah dosen wali dan mitra merasa sistem ini membantu proses pemantauan dan penilaian? \*

	1	2	3	4	
Sangat Tidak Setuju	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	Sangat Setuju

Aspek Kepuasan Pengguna

Apakah user merasa puas dengan fitur logbook, absensi, dan jadwal sidang dalam sistem ini? \*

1      2      3      4

Sangat Tidak Setuju                    Sangat Setuju

Apakah sistem ini dapat menggantikan sistem manual sebelumnya dengan lebih efektif? \*

1      2      3      4

Sangat Tidak Setuju                    Sangat Setuju



# Blackbox Testing Android

Pengembangan Sistem Pengelolaan Kegiatan Magang Mahasiswa Polman Negeri Babel Berbasis *Android* Dengan Metode *Prototype*

NPM \*

1062202

Nama \*

Ajeng Oktaviani

Kelas \*

3 TRPL A

Program Studi \*

Teknologi Rekayasa Perangkat Lunak

Penilaian

## Login & akses dashboard

Mahasiswa dapat masuk dan melihat fitur utama

Hasil Pengujian \*

- Sesuai  
 Tidak Sesuai

Status \*

- Valid  
 Tidak Valid

**Unggah dokumen magang**

Dokumen tersimpan dan dapat dilihat admin

Hasil Pengujian \*

- Sesuai  
 Tidak Sesuai

Status \*

- Valid  
 Tidak Valid