

BAB I PENDAHULUAN

1.1 Latar Belakang

Program Merdeka Belajar Kampus Merdeka (MBKM) merupakan salah satu kebijakan strategis nasional yang bertujuan untuk meningkatkan fleksibilitas pembelajaran serta mendorong kemandirian mahasiswa dalam proses pendidikan tinggi. Melalui program ini, mahasiswa diberikan kesempatan untuk mengikuti berbagai aktivitas pembelajaran di luar kelas, seperti pertukaran pelajar, magang, proyek mandiri, maupun kegiatan akademik lainnya yang relevan dengan kebutuhan dunia kerja dan perkembangan industri. Program ini lahir dari kesadaran bahwa pendidikan tinggi perlu bertransformasi agar mampu menghasilkan lulusan yang kompeten, adaptif, dan siap menghadapi tantangan dunia kerja (Kementerian Pendidikan dan Kebudayaan, 2020). Implementasi MBKM menuntut adanya sistem pengelolaan dan *monitoring* yang efektif agar pelaksanaan kegiatan mahasiswa dapat terpantau secara sistematis dan sesuai dengan capaian pembelajaran yang ditetapkan. Program MBKM memberikan mahasiswa kebebasan untuk mengikuti berbagai kegiatan di luar program studi, namun hal ini sekaligus menuntut perguruan tinggi untuk menyediakan sistem pencatatan dan pelaporan yang terstruktur agar capaian kegiatan mahasiswa dapat dipantau dan diakui secara akademik (Halimah & Zulfitria, 2024).

Seiring dengan meningkatnya kompleksitas pelaksanaan MBKM, perguruan tinggi menghadapi tantangan dalam melakukan *monitoring* dan evaluasi aktivitas mahasiswa secara efisien. Proses pelaporan yang masih dilakukan secara manual atau tersebar di berbagai media sering kali menyebabkan keterlambatan informasi, kurangnya transparansi, serta kesulitan dalam melakukan rekapitulasi data secara *real-time*. Kondisi tersebut dapat menghambat proses pengambilan keputusan dan evaluasi kegiatan secara menyeluruh. Oleh karena itu, dibutuhkan sistem informasi yang mampu mengintegrasikan proses pelaporan aktivitas mahasiswa secara terpusat, terstruktur, dan mudah diakses oleh seluruh pihak yang terlibat.

Universitas Muhammadiyah Bone sebagai salah satu institusi pendidikan tinggi yang aktif menerapkan program MBKM telah melakukan berbagai upaya untuk mendukung pelaksanaan kegiatan mahasiswa secara optimal. Berdasarkan hasil observasi dan wawancara yang dilakukan, proses pelaporan aktivitas mahasiswa dalam kegiatan MBKM di Universitas Muhammadiyah Bone masih dilakukan secara manual, di mana mahasiswa menyerahkan laporan dalam bentuk dokumen fisik kepada koordinator program studi yang kemudian harus direkap ulang secara manual oleh staf administrasi. Kondisi ini menyebabkan proses pelaporan menjadi tidak efisien, rentan terhadap kesalahan data, dan mempersulit proses *monitoring* oleh dosen pembimbing maupun pihak universitas. Salah satu langkah strategis yang dapat dilakukan adalah pengembangan aplikasi web pelaporan aktivitas mahasiswa yang dirancang untuk memfasilitasi proses pencatatan, *monitoring*, serta evaluasi kegiatan MBKM secara digital (Yoesdiarti dkk., 2022).

Berbagai penelitian telah menunjukkan pentingnya sistem informasi berbasis web dalam meningkatkan efisiensi administrasi pendidikan tinggi. Danihsan et al. (2025) mengembangkan sistem informasi MBKM berbasis web di Politeknik Manufaktur Negeri

Bangka Belitung yang berhasil meningkatkan efisiensi pengelolaan klaim kredit mahasiswa dengan hasil *User Acceptance Testing* (UAT) mencapai 84,92%. Sementara itu, pengelolaan data kegiatan MBKM yang masih bersifat manual di banyak perguruan tinggi menyebabkan keterlambatan proses verifikasi dan konversi nilai yang berdampak pada penyelesaian studi mahasiswa. Temuan-temuan ini menegaskan urgensi pengembangan sistem informasi yang terintegrasi sebagai solusi atas permasalahan pengelolaan MBKM yang ada. Permasalahan serupa juga ditemukan di Fakultas Ilmu Komputer Universitas Brawijaya, di mana proses konversi mata kuliah dan pengakuan SKS kegiatan MBKM masih dilakukan secara manual dan belum terintegrasi dalam satu sistem, sehingga menyebabkan inefisiensi administrasi yang signifikan (Hafizh dkk., 2024). Kebutuhan serupa juga mendorong Universitas Mulawarman untuk merancang sistem informasi khusus guna mengelola kegiatan MBKM mahasiswanya secara digital, menggantikan proses pencatatan yang sebelumnya masih dilakukan secara manual (Widagdo dkk., 2022).

Pengembangan sistem pelaporan aktivitas mahasiswa tidak hanya berfokus pada implementasi teknologi, tetapi juga harus mempertimbangkan kebutuhan pengguna, kemudahan penggunaan sistem, serta fleksibilitas dalam menyesuaikan perubahan kebutuhan selama proses pengembangan. Oleh karena itu, diperlukan pendekatan pengembangan sistem yang adaptif dan iteratif agar aplikasi dapat terus disempurnakan sesuai kebutuhan pengguna. Metode Agile menjadi salah satu pendekatan yang relevan karena memungkinkan pengembangan sistem secara bertahap melalui siklus iterasi yang melibatkan umpan balik pengguna secara berkelanjutan. Penggabungan metode Agile dengan pendekatan *User-Centered Design* (UCD) dinilai mampu menghasilkan sistem yang tidak hanya fungsional secara teknis, tetapi juga intuitif dan mudah digunakan oleh seluruh pemangku kepentingan. Penerapan metode Agile dalam pengembangan sistem informasi *monitoring* MBKM terbukti menghasilkan aplikasi yang lebih cepat, efisien, dan fleksibel terhadap perubahan kebutuhan pengguna (Santoso dkk., 2022).

Berdasarkan hal tersebut, penelitian ini berfokus pada perancangan dan pembangunan sistem informasi pelaporan aktivitas mahasiswa berbasis web untuk *monitoring* kegiatan MBKM di Universitas Muhammadiyah Bone, yang diberi nama SILASA (Sistem Informasi Laporan Aktivitas Mahasiswa). Sistem ini dikembangkan menggunakan metode pengembangan Agile dengan pendekatan *User-Centered Design* (UCD), dan dibangun menggunakan teknologi Next.js, Express.js, MySQL, dan Tailwind CSS. Penelitian ini diharapkan dapat menghasilkan sistem yang mampu mendukung proses pelaporan dan *monitoring* kegiatan secara efektif, sekaligus memberikan kontribusi dalam pengembangan sistem informasi pendidikan yang adaptif dan berorientasi pada kebutuhan pengguna.

1.2 Rumusan Masalah

Adapun rumusan masalah dalam penelitian ini yaitu:

1. Bagaimana merancang dan membangun sistem pelaporan aktivitas mahasiswa terhadap kegiatan MBKM berbasis web?
2. Bagaimana menguji kinerja dari aplikasi yang dirancang untuk kebutuhan sisi fungsionalitas?

1.3 Batasan Masalah

Berikut ini merupakan beberapa batasan dalam penelitian ini:

1. Sistem aplikasi yang dirancang hanya berfokus pada pelaporan aktivitas mahasiswa. Sistem aplikasi tidak dirancang untuk mengubah data-data pribadi mahasiswa dan dosen.
2. Hasil nilai dari kegiatan MBKM mahasiswa dikeluarkan dalam bentuk sertifikat, tidak langsung memperbarui nilai di sistem informasi akademik kampus.

1.4 Tujuan Penelitian

Adapun tujuan dari penelitian ini dilakukan yaitu:

1. Merancang dan membangun sistem pelaporan aktivitas mahasiswa MBKM berbasis web.
2. Mempermudah mahasiswa dalam melakukan pelaporan terhadap aktivitas dalam kegiatan MBKM-nya.
3. Mempermudah dosen untuk memonitoring kegiatan mahasiswa hingga ke tahap penilaian dan evaluasi.

1.5 Manfaat Penelitian

Adapun manfaat dari penelitian ini yaitu:

1. Membantu mahasiswa untuk melakukan pelaporan selama masa periode pelaksanaan kegiatan MBKM.
2. Membantu dosen untuk melakukan penilaian terhadap aktivitas mahasiswa selama periode pelaksanaan kegiatan MBKM.
3. Memudahkan pihak kampus untuk melihat rekap kegiatan MBKM yang diikuti oleh mahasiswa di kampus tersebut. Hal ini akan memudahkan untuk melihat jenis kegiatan MBKM mana yang paling banyak diminati, paling kurang peminat, dan lain-lainnya.

1.6 Landasan Teori

1.6.1 Sistem Informasi Pelaporan Aktivitas Mahasiswa Berbasis Web

Sistem informasi merupakan sebuah proses mengumpulkan, memproses, menganalisis serta menyebarkan informasi untuk tujuan tertentu yang melibatkan orang, perangkat keras, perangkat lunak maupun basis data (Rahmi dkk., 2023). Dalam konteks perguruan tinggi, sistem informasi digunakan untuk mengelola berbagai proses akademik maupun non-akademik, seperti registrasi mahasiswa, pengelolaan kurikulum, penilaian, dan pelaporan aktivitas mahasiswa. Sistem ini membantu mengurangi pekerjaan manual, meminimalkan kesalahan pencatatan, serta meningkatkan kecepatan dan akurasi penyediaan informasi bagi seluruh pemangku kepentingan.

Sistem informasi pelaporan aktivitas mahasiswa adalah bagian khusus dari sistem informasi akademik yang fokus pada pencatatan, pelacakan, dan pelaporan berbagai kegiatan yang dilakukan mahasiswa, baik di dalam maupun di luar kelas, seperti magang, proyek, penelitian, kegiatan organisasi, dan program MBKM. Informasi yang dicatat mencakup identitas mahasiswa, jenis kegiatan, waktu pelaksanaan, bukti atau

dokumen pendukung, serta status verifikasi oleh dosen pembimbing atau pengelola program. Dengan demikian, sistem ini berperan sebagai media dokumentasi formal yang dapat dijadikan dasar evaluasi, rekognisi kredit, maupun bahan pelaporan institusi ke pihak eksternal.

Sistem informasi pelaporan aktivitas berbasis web dibangun dengan memanfaatkan teknologi web sehingga dapat diakses melalui peramban (*browser*) tanpa memerlukan instalasi aplikasi khusus di perangkat pengguna (Danihsan dkk., 2025). Sistem informasi berbasis *website* memiliki banyak manfaat dan dapat mempermudah kerja manusia, sehingga pengembangan sistem informasi masih terus dilakukan (Rahmi dkk., 2023). Dalam konteks MBKM, sistem berbasis web memungkinkan mahasiswa untuk mengunggah laporan dan bukti kegiatan secara daring, sementara dosen dan pengelola program dapat melakukan pemantauan dan verifikasi secara *real-time* dari mana saja.

Pada program Merdeka Belajar Kampus Merdeka, pelaporan aktivitas mahasiswa menjadi persyaratan penting untuk memastikan kegiatan yang dilakukan benar-benar sejalan dengan capaian pembelajaran dan dapat diakui dalam bentuk kredit semester (Kementerian Pendidikan dan Kebudayaan, 2020). Banyak perguruan tinggi yang masih mengandalkan formulir manual atau pelaporan tersebar di berbagai kanal, sehingga menyulitkan proses *monitoring*, rekapitulasi, dan pelaporan ke lembaga seperti LLDIKTI dan PDDIKTI. Danihsan et al. (2025) membuktikan bahwa sistem informasi MBKM berbasis web mampu meningkatkan efisiensi pengelolaan klaim kredit mahasiswa secara signifikan dengan hasil pengujian penerimaan pengguna yang tinggi. Oleh karena itu, sistem informasi pelaporan aktivitas mahasiswa berbasis web dirancang untuk menyatukan proses tersebut dalam satu *platform* terpusat yang transparan dan terdokumentasi dengan baik.

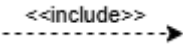
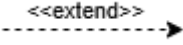
1.6.2 Use Case Diagram

Use Case Diagram adalah bagian dari Unified Modeling Language (UML) yang berfungsi memodelkan fungsionalitas sistem dari perspektif pengguna luar, dengan fokus pada "apa" yang dilakukan sistem, bukan "bagaimana" cara kerjanya. Diagram ini membantu mengkomunikasikan kebutuhan sistem kepada *stakeholder* non-teknis, seperti dosen dan mahasiswa dalam konteks *monitoring* MBKM, sehingga mengurangi kesalahpahaman selama pengembangan (Etriyadi & A'inunisyah, 2023). Tujuannya mencakup identifikasi aktor utama (misalnya Mahasiswa, Dosen, Admin), *use case* (kasus penggunaan) inti (seperti "Melaporkan Aktivitas" atau "Memverifikasi Laporan"), serta relasi seperti *include* (wajib) dan *extend* (opsional) untuk menangani skenario kompleks.

Komponen dasar Use Case Diagram meliputi aktor, *use case*, *system boundary* (batas sistem), dan relasi antar elemen. Aktor digambarkan sebagai figur manusia yang berinteraksi dengan sistem, *use case* sebagai oval yang merepresentasikan fungsi tunggal, sementara *boundary* kotak menandai ruang lingkup sistem. Relasi seperti *<<include>>* digunakan saat satu *use case* memanggil *use case* lain secara wajib (contoh: "Login" termasuk dalam "Melaporkan Aktivitas"), sedangkan *<<extend>>* untuk kondisi opsional seperti "Upload Bukti Tambahan". Dalam sistem pelaporan MBKM, komponen ini memastikan semua alur pengguna tertangkap, mulai dari *input* data hingga

evaluasi (Etrariadi & A'inunisya, 2023). Komponen dasar dalam Use Case Diagram meliputi aktor, *use case*, serta relasi antar elemen yang digunakan untuk menggambarkan interaksi sistem. Simbol dan makna dari masing-masing komponen tersebut ditunjukkan pada Tabel 1.

Tabel 1. Simbol Use Case Diagram dan Keterangannya

Simbol	Nama	Keterangan
	<i>Actor</i>	Pihak yang berinteraksi langsung dengan sistem, seperti mahasiswa, dosen, dan admin.
	<i>Use Case</i>	Fungsi yang disediakan oleh aplikasi.
	<i>Association</i>	Hubungan antara aktor dengan <i>use case</i> dalam sistem.
	<i>Generalization</i>	Hubungan pewarisan antara aktor atau <i>use case</i> yang bersifat umum dan khusus.
	<i>Include</i>	Hubungan di mana suatu <i>use case</i> selalu melibatkan <i>use case</i> yang lain.
	<i>Extend</i>	Hubungan di mana suatu <i>use case</i> dapat menambahkan fungsi lain secara opsional.

Pada sistem informasi pelaporan aktivitas mahasiswa MBKM di Universitas Muhammadiyah Bone, Use Case Diagram dapat menggambarkan aktor Mahasiswa yang menjalankan "Input Aktivitas MBKM" (*include* "Upload Dokumen"), Dosen yang melakukan "Verifikasi Laporan" (*extend* "Berikan Penilaian"), dan Admin yang mengelola "Rekapitulasi Data". Diagram ini mendukung metode Agile dengan memungkinkan iterasi cepat berdasarkan *feedback* (umpan balik) aktor, sehingga memastikan sistem responsif terhadap kebutuhan *monitoring real-time*. Penggunaan Use Case Diagram dalam konteks ini membantu memastikan bahwa seluruh fungsionalitas sistem yang dikembangkan benar-benar sesuai dengan kebutuhan pengguna yang telah diidentifikasi.

Use Case Diagram menyederhanakan analisis kebutuhan, memfasilitasi validasi awal dengan pengguna, dan menjadi dasar untuk diagram lain seperti Activity Diagram. Penggunaan Use Case Diagram dalam pengembangan sistem informasi akademik efektif sebagai alat untuk mendokumentasikan dan mengkomunikasikan kebutuhan sistem antara pengembang dan pemangku kepentingan (Etrariadi & A'inunisya, 2023).

Keunggulannya terletak pada visualisasi non-teknis yang mudah dipahami, ideal untuk proyek Agile seperti skripsi ini. Namun, keterbatasannya adalah tidak menjelaskan urutan proses detail atau *data flow*, sehingga harus dilengkapi model lain seperti Activity Diagram dan ERD.

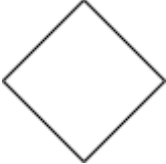
1.6.3 Activity Diagram

Activity Diagram adalah salah satu diagram perilaku UML yang menekankan pada *flow* (alur) aktivitas, mirip *flowchart* tetapi lebih kuat untuk memodelkan proses kompleks dengan cabang kondisional dan paralel. Diagram ini bertujuan mengkomunikasikan logika bisnis kepada *stakeholder* (pemangku kepentingan), mengidentifikasi *bottleneck* (kemacetan) seperti *delay* verifikasi laporan MBKM, dan menjadi dasar implementasi kode (Etrariadi & A'inunisyah, 2023). Dalam konteks Universitas Muhammadiyah Bone, diagram ini memetakan *end-to-end* (ujung ke ujung) proses pelaporan agar efisien dan *traceable* (dapat dilacak). Activity Diagram sangat berguna dalam analisis sistem untuk memvisualisasikan perbedaan antara proses yang sedang berjalan (*as-is*) dengan proses yang diusulkan (*to-be*), sehingga memudahkan identifikasi titik-titik inefisiensi dalam sistem lama.

Activity Diagram memiliki peran strategis dalam pengembangan sistem informasi berbasis Agile karena setiap alur proses yang digambarkan dapat langsung dijadikan acuan dalam penyusunan *user stories* pada product backlog. Dalam konteks sistem SILASA, Activity Diagram digunakan untuk memodelkan alur pelaporan aktivitas mahasiswa secara menyeluruh, mulai dari proses *login*, pengisian data kegiatan, pengunggahan dokumen bukti, hingga tahap verifikasi oleh dosen dan rekapitulasi oleh administrator. Dengan demikian, setiap aktor dalam sistem memiliki alur kerja yang terdefinisi dengan jelas, sehingga proses pengembangan dapat dilakukan secara terstruktur dan meminimalkan risiko kesalahan implementasi (Etrariadi & A'inunisyah, 2023).

Komponen inti mencakup *initial node* (titik hitam tebal), *action* (persegi panjang bulat), *decision node* (belah ketupat), *merge node* (belah ketupat terbalik), *fork/join* untuk paralel, dan *final node* (lingkaran dengan titik). *Guard condition* pada panah *decision* menentukan alur, seperti *[status=approved]* untuk lanjut ke evaluasi. Komponen dasar dalam Activity Diagram meliputi aktor, *use case*, serta relasi antar elemen yang digunakan untuk menggambarkan interaksi sistem. Simbol dan makna dari masing-masing komponen tersebut ditunjukkan pada Tabel 2.

Tabel 2. Simbol Activity Diagram dan Keteranganannya

Simbol	Nama	Keterangan
	<i>Start Point</i>	Menunjukkan awal dari alur aktivitas sistem.
	<i>End Point</i>	Menunjukkan akhir dari alur aktivitas sistem.
	<i>Activities</i>	Aktivitas atau proses yang dilakukan dalam sistem.
	<i>Decision</i>	Percabangan alur berdasarkan kondisi tertentu.
	<i>Swimlane</i>	Pembagian aktivitas berdasarkan peran atau aktor.
	<i>Join</i>	Penggabungan beberapa alur aktivitas menjadi satu.

Notasi UML 2.0 mendukung *swimlanes* (jalur aktivitas) untuk memisahkan tanggung jawab aktor (kolom Mahasiswa/Dosen/Admin), *object flow* (panah bertitik untuk data), dan *time event* (peristiwa waktu) untuk batas waktu MBKM. Alur dimulai dari *initial node*, melalui *actions sequential*, *decision* dengan *guard* [condition], hingga *final node*; *loop* untuk revisi laporan. Struktur ini ideal untuk Agile *sprint*, di mana setiap iterasi menyempurnakan satu alur proses.

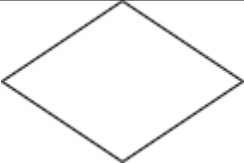
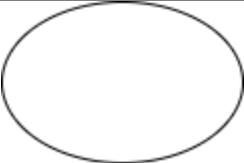
Untuk sistem pelaporan aktivitas, Activity Diagram menggambarkan: Mahasiswa *login* -> Input kegiatan MBKM -> Upload dokumen -> Submit (*decision*: lengkap?) -> Dosen *review* (*fork*: verifikasi + notifikasi) -> [Approved] -> Evaluasi nilai -> Simpan ke rekap & kirim sertifikat. Proses *parallel join* setelah verifikasi memastikan efisiensi *monitoring real-time*. Activity Diagram memfasilitasi simulasi proses sebelum *coding*, deteksi inkonsistensi alur, dan dokumentasi untuk *maintenance* setelah Agile. Keunggulannya pada visualisasi kompleksitas seperti multi-aktor MBKM, tetapi terbatas untuk data-heavy process (lebih cocok DFD) atau *real-time* timing detail. Dalam skripsi, diagram ini divalidasi *user* untuk iterasi desain.

1.6.4 Entity Relationship Diagram

Entity Relationship Diagram (ERD) adalah diagram yang berbentuk notasi grafis yang berada dalam pembuatan *database* menghubungkan antara data satu dan data yang lain (Pulungan dkk., 2023). Tujuannya mencakup identifikasi kebutuhan data, normalisasi untuk menghindari redundansi, dan validasi struktur *database* dengan *stakeholder* seperti dosen pembimbing MBKM. Penggunaan ERD dalam perancangan sistem informasi memastikan bahwa struktur basis data yang dibangun konsisten dan mampu mendukung kebutuhan fungsional sistem. Dalam konteks skripsi ini, ERD memastikan data laporan aktivitas mahasiswa dapat diintegrasikan dengan SIAKAD tanpa inkonsistensi.

ERD terdiri dari entitas (objek utama seperti persegi panjang), atribut (karakteristik seperti oval), relasi (hubungan seperti belah ketupat), dan kardinalitas (1:1, 1:N, N:M). Atribut primer key digarisbawahi untuk identifikasi unik, sementara atribut turunan seperti status verifikasi dihitung dari data lain. Notasi Chen atau Crow's Foot sering digunakan, dengan garis menghubungkan elemen untuk menandakan *flow* data. Elemen utama dalam ERD terdiri dari entitas, atribut, dan relasi yang menunjukkan bagaimana data saling terhubung (Pulungan dkk., 2023). Komponen visual dalam Entity Relationship Diagram terdiri dari berbagai simbol standar yang merepresentasikan entitas, atribut, relasi, serta struktur koneksi antar data. Penjelasan mengenai simbol dan makna dari setiap elemen tersebut dapat dilihat pada Tabel 3.

Tabel 3. Simbol Entity Relationship Diagram (ERD) dan Keterangannya

Simbol	Nama	Keterangan
	Entitas	Objek atau data utama yang disimpan dalam sistem.
	Relasi	Hubungan antara satu entitas dengan entitas lainnya.
	Atribut	Karakteristik atau informasi yang dimiliki oleh entitas.
	Spesialisasi	Hubungan pewarisan dari entitas umum ke entitas khusus.
	Connection	Garis penghubung antara entitas, relasi, dan atribut.

ERD dibagi menjadi konseptual (tingkat tinggi, fokus entitas-relasi), logis (menambahkan tipe data), dan fisik (skema *database* spesifik seperti MySQL). Notasi Crow's Foot populer karena sederhana, dengan garis bercabang untuk "many" dan lingkaran untuk "optional". Untuk sistem MBKM, ERD konseptual mengidentifikasi entitas Mahasiswa (atribut: NIM, nama, prodi), Kegiatan (jenis MBKM, durasi), dan Laporan (status: draft/verified).

Dalam sistem ini, ERD memodelkan relasi 1:N antara Dosen (pembimbing) dan Laporan (yang diverifikasi), serta N:M antara Mahasiswa dan Kegiatan (via tabel junction untuk magang/proyek MBKM). Hubungan ini mendukung *query monitoring real-time*, seperti rekap kegiatan per prodi, sesuai Agile iteration. Normalisasi hingga 3NF mencegah anomali saat update status laporan. ERD memfasilitasi komunikasi tim pengembang, mengurangi kesalahan desain, dan menjadi *blueprint* untuk Entity Framework atau ORM. Namun, tidak menangani aturan bisnis kompleks atau performa *query*, sehingga dilengkapi DFD. Pada skripsi Agile ini, ERD iteratif disesuaikan berdasarkan *feedback* pengguna.

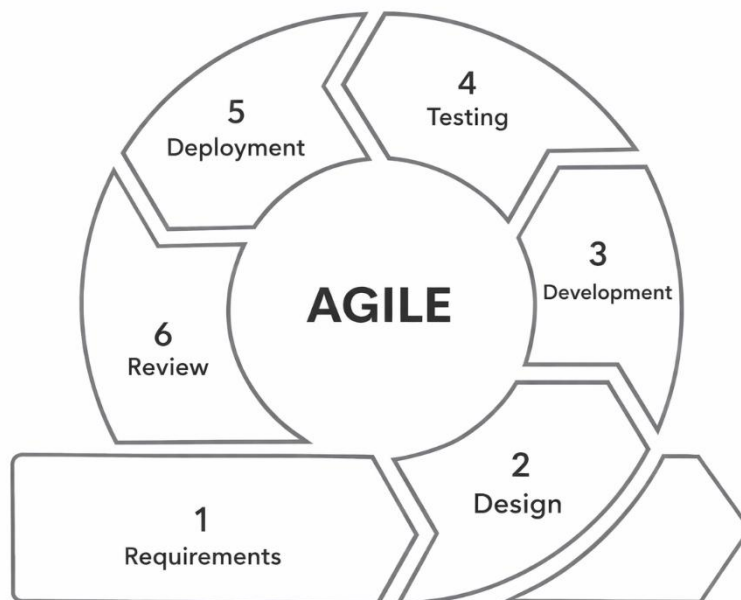
1.6.5 Metode Pengembangan Agile

Agile didefinisikan dalam Agile Manifesto (2001) sebagai prioritas individu dan interaksi di atas proses/*tools*, *software* working di atas dokumentasi komprehensif, kolaborasi customer di atas negosiasi kontrak, serta respons terhadap perubahan di atas mengikuti rencana. Empat nilai inti ini didukung 12 prinsip, seperti pengiriman *software* berfungsi rutin, sambutan perubahan requirement meski late *development*, dan refleksi rutin untuk peningkatan efektivitas. Dalam skripsi ini, Agile memungkinkan iterasi cepat berdasarkan *feedback* mahasiswa/dosen MBKM, berbeda dengan Waterfall yang linear dan kaku. Metode Agile terbukti efektif dalam memastikan kesesuaian sistem dengan kebutuhan pengguna yang terus berkembang, khususnya dalam pengembangan sistem informasi berbasis web (Nova dkk., 2022).

Scrum menjadi *framework* paling populer dengan peran (Product Owner, Scrum Master, Development Team), artefak (Product Backlog, *Sprint* Backlog, Increment), dan event (*Sprint* Planning, Daily Scrum, Review, Retrospective). *Sprint* tipikal 2-4 minggu menghasilkan *potentially shippable product*, ideal untuk prototipe *dashboard monitoring* MBKM. Kanban fokus visualisasi *workflow* via board (To Do, In Progress, Done) untuk limit WIP, cocok mengelola *maintenance* (pemeliharaan) laporan pasca launch. Penelitian oleh Etrariadi & A'inunisya (2023) membuktikan bahwa pengembangan sistem informasi menggunakan metode Agile Scrum menghasilkan produk yang lebih responsif terhadap kebutuhan pengguna dibandingkan metode pengembangan konvensional. Lifecycle dimulai requirement gathering (*user stories* di backlog), dipecah ke *sprint planning* -> *development* -> *testing* -> *review* -> *retrospective*. *User stories* ditulis format "As a [user], I want [feature] so that [benefit]", diprioritaskan MoSCoW (Must, Should, Could, Won't). Daily standup (3 pertanyaan: yesterday, today, blocker) memastikan transparansi. Velocity (story points selesai/*sprint*) mengukur produktivitas tim Agile skripsi ini. Pendekatan ini memastikan bahwa setiap fitur yang dikembangkan benar-benar dibutuhkan dan divalidasi oleh pengguna melalui proses review berkala (Nova dkk., 2022).

Dalam proyek ini, Agile diterapkan via *sprint* 2-minggu: *Sprint 1* (login + input laporan), *Sprint 2* (verifikasi dosen + notifikasi), *Sprint 3* (dashboard rekap + evaluasi). Demo *sprint* ke user (mahasiswa/dosen Bone) memvalidasi fitur, retrospective sesuaikan backlog untuk MBKM compliance. Untuk memberikan gambaran visual mengenai tahapan pengembangan sistem yang digunakan dalam penelitian ini, siklus metodologi Agile ditampilkan pada Gambar 1.

Berdasarkan Gambar 1, proses pengembangan dilakukan secara iteratif melalui beberapa tahapan utama yaitu requirements, design, development, *testing*, deployment, dan review yang berlangsung secara berulang. Setiap tahap saling terhubung untuk memastikan sistem SILASA dapat dikembangkan secara bertahap melalui *sprint*, dimulai dari pengumpulan kebutuhan pengguna, perancangan model sistem, implementasi fitur, pengujian, hingga evaluasi berdasarkan feedback stakeholder. Nova dkk. (2022) melalui Systematic Literature Review menyimpulkan bahwa model Scrum merupakan model Agile yang paling banyak digunakan dalam pengembangan sistem informasi berbasis *website* karena mengutamakan kecepatan dan fleksibilitas, sementara Etrariadi & A'inunisyia (2023) membuktikan bahwa pendekatan ini menghasilkan produk yang lebih responsif terhadap kebutuhan pengguna. Dengan demikian, pengembangan sistem SILASA menjadi lebih adaptif terhadap perubahan kebutuhan dan meningkatkan kualitas sistem secara berkelanjutan.



Gambar 1. Siklus Pengembangan Sistem Menggunakan Metodologi Agile

1.6.6 User-Centered Design

User-Centered Design (UCD) adalah pendekatan perancangan yang menempatkan pengguna sebagai pusat perhatian pada seluruh siklus pengembangan, dengan tujuan memastikan bahwa produk akhir sesuai dengan kebutuhan, kemampuan, dan konteks penggunaan mereka. UCD menekankan keterlibatan aktif pengguna melalui wawancara, observasi, *prototype testing*, dan aktivitas lain yang memungkinkan perancang memahami pola pikir dan pola kerja pengguna. Dengan demikian, keputusan desain tidak hanya didasarkan pada asumsi perancang, tetapi juga pada data empiris mengenai perilaku dan harapan pengguna.

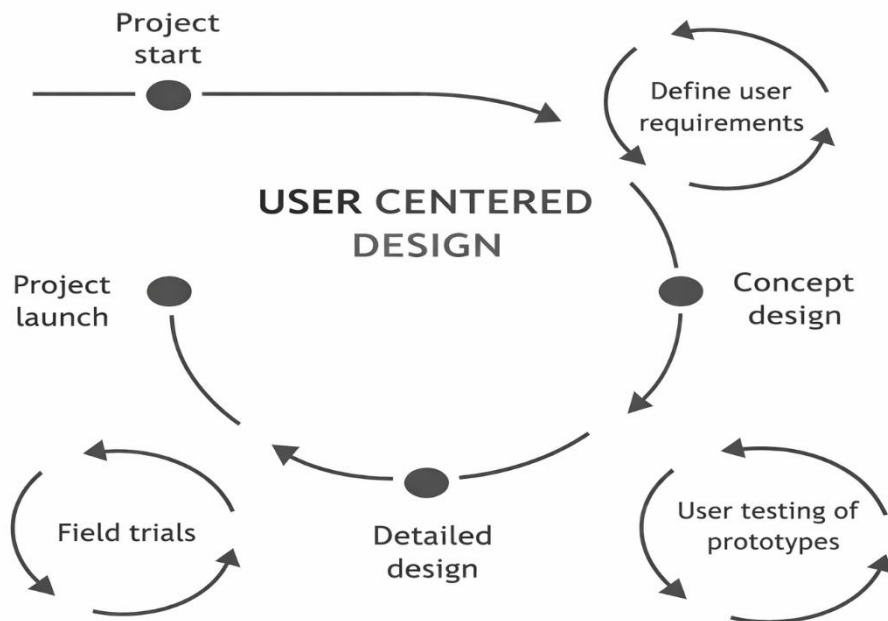
Secara umum, UCD mencakup empat aktivitas utama: memahami dan menentukan konteks penggunaan; menentukan kebutuhan pengguna dan organisasi; menghasilkan solusi desain; serta mengevaluasi desain terhadap kebutuhan tersebut. Aktivitas-aktivitas ini bersifat iteratif, artinya hasil evaluasi dapat mengarahkan perbaikan atau perubahan pada rancangan sehingga produk terus disempurnakan hingga mencapai tingkat kualitas yang diinginkan. Dalam pengembangan sistem pelaporan aktivitas mahasiswa MBKM, UCD dapat diwujudkan melalui kegiatan seperti pengumpulan kebutuhan dari mahasiswa dan dosen, pembuatan sketsa dan prototipe antarmuka, lalu pengujian prototipe tersebut kepada calon pengguna untuk memperoleh umpan balik.

Penerapan UCD sangat relevan karena pengguna sistem pelaporan MBKM berasal dari latar belakang yang beragam, memiliki tingkat literasi teknologi yang berbeda, dan menjalankan tugas yang tidak sama (misalnya mahasiswa mengisi laporan, dosen memverifikasi, admin melakukan rekapitulasi). Dengan menempatkan pengguna sebagai fokus, rancangan antarmuka dapat dioptimalkan agar setiap peran memiliki alur kerja yang sederhana dan jelas, sehingga mengurangi resistensi terhadap penggunaan sistem baru. Selain itu, pendekatan UCD mendukung tujuan program MBKM yang mendorong kemandirian mahasiswa, karena sistem yang mudah dipahami akan memudahkan mereka mengelola dan melaporkan aktivitas secara mandiri (Kementerian Pendidikan dan Kebudayaan, 2020).

Penelitian oleh Purbo et al. (2023) menunjukkan bahwa penerapan pendekatan UCD dalam perancangan antarmuka aplikasi menghasilkan sistem yang lebih intuitif dan mudah digunakan oleh pengguna dari berbagai latar belakang. Temuan ini memperkuat relevansi penggunaan UCD dalam pengembangan sistem SILASA yang memiliki beragam kelompok pengguna dengan karakteristik yang berbeda. Untuk memberikan gambaran visual mengenai tahapan pendekatan *User-Centered Design* yang digunakan dalam penelitian ini, proses UCD ditampilkan pada Gambar 2.

Berdasarkan Gambar 2, pendekatan UCD dilakukan secara iteratif melalui beberapa tahapan utama, yaitu identifikasi kebutuhan pengguna (*define user requirements*), perancangan konsep (*concept design*), perancangan detail antarmuka (*detailed design*), pengujian prototipe oleh pengguna (*user testing*), hingga evaluasi melalui *field trials* sebelum sistem diluncurkan. Penerapan metode UCD dalam pengembangan sistem informasi berbasis *web* terbukti menghasilkan antarmuka yang lebih sesuai kebutuhan pengguna karena setiap tahap perancangan selalu melibatkan pengguna secara langsung (Kurnia & Awaludin, 2023). Siklus ini menekankan keterlibatan pengguna secara berkelanjutan sehingga pengembangan sistem SILASA dapat menyesuaikan

kebutuhan nyata mahasiswa, dosen, dan admin sebagai pengguna utama, serta menghasilkan desain antarmuka yang lebih efektif, mudah digunakan, dan sesuai konteks penggunaan.



Gambar 2. Proses User Centered Design (UCD) dalam Perancangan Sistem

1.6.7 MySQL

MySQL adalah sistem manajemen basis data relasional (RDBMS) *open-source* yang banyak digunakan dalam pengembangan aplikasi web. MySQL menggunakan bahasa SQL (Structured Query Language) untuk mengakses dan mengelola data. MySQL pertama kali dibuat pada tahun 1995 oleh MySQL AB Swedia dan diakuisisi Sun Microsystems (kemudian Oracle) pada 2008. Database ini mengikuti standar SQL ANSI dengan *storage engine* pluggable seperti InnoDB (default, ACID-compliant dengan *row-level locking* dan MVCC), MyISAM (read-heavy), dan Memory (in-RAM). Keunggulan MySQL antara lain adalah kecepatan eksekusi *query*, dukungan terhadap transaksi, kemudahan integrasi dengan berbagai bahasa pemrograman dan *framework*, serta ketersediaannya sebagai perangkat lunak gratis. Dalam pengembangan sistem informasi berbasis web, MySQL sering digunakan bersama dengan Node.js dan Express.js untuk membangun *backend* yang handal dan efisien.

Karakteristik utama MySQL meliputi skalabilitas horizontal via replication *master-slave*, *query* optimizer canggih, dan kompatibilitas tinggi dengan *framework* Node.js/Express untuk API MBKM. MySQL menyediakan data types komprehensif: VARCHAR/TEXT untuk deskripsi kegiatan, DATETIME/TIMESTAMP untuk jadwal MBKM, JSON (sejak 5.7) untuk metadata bukti, dan ENUM untuk status laporan ('draft', 'submitted', 'verified', 'evaluated'). Stored *procedures*, *triggers*, dan *views* mendukung

logic *server-side* seperti auto-update rekap prodi. Indexing B-tree dan *full-text search* (InnoDB 5.6+) mempercepat *query monitoring* seperti `SELECT * FROM laporan WHERE status='pending' ORDER BY tanggal DESC`.

Dalam konteks pengembangan sistem SILASA, MySQL dipilih karena kemampuannya dalam mengelola data relasional yang kompleks, seperti hubungan antara mahasiswa, dosen pembimbing, laporan aktivitas, dan kegiatan MBKM dalam satu skema basis data yang terintegrasi. Normalisasi basis data hingga Third Normal Form (3NF) diterapkan untuk mencegah anomali data saat terjadi pembaruan status laporan atau perubahan data mahasiswa. Danihsan et al. (2025) dalam penelitian pengembangan sistem informasi MBKM berbasis web juga menggunakan MySQL sebagai basis data utama yang terbukti mampu menangani volume transaksi data yang tinggi dalam lingkungan perguruan tinggi.

1.6.8 Express.js

Express.js adalah *framework* web minimalis untuk Node.js yang digunakan untuk membangun aplikasi web dan API RESTful. Express.js pertama kali dirilis tahun 2010 oleh TJ Holowaychuk dan kini dipelihara Node.js Foundation. Framework ini berbasis *middleware* pattern di mana setiap *request* melewati stack *middleware* (authentication -> validation -> business logic -> *response*), memungkinkan modularitas tinggi untuk sistem MBKM. Core philosophy "minimal & flexible" hanya menyediakan *routing* dasar, HTTP utilities, dan *middleware* system tanpa opinionated structure. Express.js menyediakan sekumpulan fitur yang memudahkan pengembangan sisi *server* (*backend*), seperti *routing*, *middleware*, dan pengelolaan HTTP *request-response*.

Express mendukung *routing* RESTful dengan method chaining (`app.get()`, `app.post()`, `app.put()`, `app.delete()`), parameter *routing* (`/mahasiswa/:nim`), dan route parameters untuk laporan MBKM (`/laporan/:id/verifikasi`). Middleware system seperti `express.json()`, `express.urlencoded()`, CORS (cors package), dan custom *middleware* autentikasi JWT sangat relevan untuk sistem multi-user (mahasiswa/dosen/admin). Error handling terpusat via `next(err)` memastikan konsistensi *response* format JSON API. Penggunaan Express.js memungkinkan pengembang untuk membangun *backend* yang ringan namun powerful, dengan waktu pengembangan yang lebih singkat dibandingkan membangun dari nol. Framework ini banyak digunakan dalam kombinasi dengan Next.js di sisi *frontend* dan MySQL sebagai basis data (Danihsan dkk., 2025). Fitur-fitur utama *framework* Express.js yang mendukung pengembangan *backend* sistem SILASA beserta fungsi dan contoh implementasinya dirangkum pada Tabel 4.

Tabel 4. Fitur Framework Express.js dan Implementasinya dalam Sistem

Fitur	Fungsi	Contoh Implementasi
Routing	URL mapping	<code>/api/mahasiswa/:nim/laporan</code>
Middleware	Request processing pipeline	JWT auth, validation
Error Handling	Centralized error <i>middleware</i>	404 laporan tidak ditemukan

Express.js berpadu sempurna dengan MySQL via library untuk *query* kompleks rekap MBKM (*pool.query()* dengan parameterized queries), Tailwind CSS via API *response* untuk *frontend* Next.js, dan WebSocket (*socket.io*) untuk *real-time* notifikasi status verifikasi. ORM seperti Sequelize/Prisma atau raw SQL dengan promise mengelola relasi Mahasiswa-Laporan-Kegiatan sesuai ERD sebelumnya. Struktur aplikasi ideal mencakup: *routes/* (modular per fitur), *controllers/* (business logic), *models/* (*database* interaction), *middleware/* (auth/validation), dan *utils/* (helpers). Security: helmet, bcrypt *password* hashing, JWT dengan refresh token, dan *input* sanitization mencegah SQL injection XSS pada *upload* dokumen MBKM.

1.6.9 Tailwind CSS

Tailwind CSS adalah *framework* CSS *utility-first* yang memungkinkan pengembang membangun antarmuka pengguna yang responsif dan modern langsung dari markup HTML tanpa perlu menulis CSS kustom secara terpisah. Pendekatan *utility-first* Tailwind CSS memberikan fleksibilitas tinggi dalam membangun desain antarmuka yang konsisten dan dapat disesuaikan dengan kebutuhan (Danihsan dkk., 2025). Berbeda dengan *framework* tradisional seperti Bootstrap yang menyediakan komponen siap pakai, Tailwind menyediakan class-class kecil yang fokus pada properti tunggal seperti *bg-blue-500*, *p-4*, *flex*, atau *grid-cols-3*, sehingga memberikan fleksibilitas tinggi dalam menciptakan desain yang unik dan konsisten. Pendekatan ini mengurangi kompleksitas CSS, meminimalkan ukuran bundle akhir, dan memudahkan *maintenance* karena seluruh styling terkonsolidasi dalam satu tempat.

Salah satu keunggulan Tailwind adalah kemampuan konfigurasi yang luas melalui file *tailwind.config.js*, di mana pengembang dapat menyesuaikan skema warna, spacing, font size, *breakpoints* responsif, dan tema khusus sesuai kebutuhan proyek. Dalam konteks sistem pelaporan aktivitas mahasiswa MBKM, Tailwind dapat digunakan untuk styling *dashboard monitoring*, form *input* kegiatan, tabel rekapitulasi, serta notifikasi status dengan konsistensi visual tinggi. Fitur Just-in-Time (JIT) mode pada Tailwind CSS versi terbaru juga menghasilkan CSS yang di-generate secara *on-demand*, sehingga ukuran file CSS tetap minimal meskipun menggunakan banyak variasi class.

Tailwind mendukung desain responsif melalui prefix seperti *sm:*, *md:*, *lg:*, dan *xl:* yang secara otomatis menerapkan breakpoint berbeda pada berbagai ukuran layar, sehingga satu set class dapat menghasilkan tampilan optimal pada desktop, tablet, dan mobile. Selain itu, Tailwind menyediakan plugin ecosystem yang kaya, seperti *@tailwindcss/forms* untuk styling form yang konsisten, *@tailwindcss/typography* untuk konten rich text, dan *@tailwindcss/aspect-ratio* untuk *layout* media, yang semuanya relevan untuk aplikasi pelaporan aktivitas mahasiswa. Penerapan Tailwind pada sistem informasi pelaporan aktivitas mahasiswa memungkinkan pembuatan antarmuka yang responsif dan konsisten dengan effort minimal, sekaligus mendukung rapid prototyping yang diperlukan dalam metodologi *User-Centered Design*. Perbandingan karakteristik antara Tailwind CSS dan *framework* CSS berbasis komponen ditampilkan pada Tabel 5 guna menunjukkan alasan pemilihan Tailwind CSS dalam pengembangan antarmuka sistem SILASA.

Tabel 5. Perbandingan Tailwind CSS dengan Framework Komponen Bootstrap

Aspek	Tailwind CSS	Bootstrap / Framework Komponen
Filosofi	Utility- <i>first</i> , atomic classes	Pre-built components
Fleksibilitas	Sangat tinggi, fully customizable	Terbatas oleh komponen bawaan
Ukuran Bundle	Minimal (sekitar 10KB setelah purge)	Besar (sekitar 140KB)
Learning Curve	Sedang (memorize <i>utility</i> classes)	Mudah (HTML attributes)
Maintenance	Mudah (styling langsung di HTML)	Sulit (override CSS bawaan)

1.6.10 Next.js

Next.js adalah *framework* React full-stack yang dibangun di atas Node.js, dirancang untuk pengembangan aplikasi web modern dengan fitur *server-side rendering* (SSR), *static site generation* (SSG), *incremental static regeneration* (ISR), dan API routes dalam satu *platform* terintegrasi. Next.js dikembangkan oleh Vercel dan menyediakan arsitektur *file-based routing* otomatis, di mana struktur folder *pages/* atau *app/* secara langsung diterjemahkan menjadi rute aplikasi tanpa konfigurasi *routing* manual, sehingga mempercepat pengembangan aplikasi berskala besar. Framework ini juga otomatis mengoptimalkan bundling, *code splitting*, dan *image optimization*, menghasilkan performa *loading* yang superior dibandingkan aplikasi React vanilla.

Salah satu fitur unggulan Next.js adalah *hybrid rendering* yang memungkinkan halaman menggunakan kombinasi SSR untuk konten dinamis (seperti *dashboard monitoring real-time* MBKM), SSG untuk halaman statis (seperti panduan pelaporan), dan CSR untuk interaksi *frontend* yang intensif. App Router pada Next.js 13+ memperkenalkan konsep Server Components dan Client Components yang memisahkan logika *server* dari interaktivitas *client* secara eksplisit, mengurangi JavaScript yang dikirim ke *browser* hingga 50-70% sambil mempertahankan kemampuan *hydration* interaktif. Dalam sistem pelaporan aktivitas mahasiswa, Server Components dapat digunakan untuk fetch data kegiatan MBKM dari *database* secara aman di *server-side*, sementara Client Components menangani form *input* dan *real-time* updates.

Next.js menyediakan *built-in support* untuk Tailwind CSS, TypeScript, ESLint, dan *testing framework* secara *out-of-the-box*, mengurangi *boilerplate setup* dan memungkinkan *developer* fokus pada *business logic*. Fitur *Image component* otomatis melakukan *lazy loading*, dan WebP/AVIF *format conversion*, yang sangat berguna untuk menampilkan bukti foto kegiatan MBKM tanpa mengorbankan performa. Dalam pengembangan sistem informasi modern, Next.js menjadi pilihan populer untuk membangun *frontend* yang berinteraksi dengan API *backend* yang dibangun menggunakan Express.js. Ringkasan fitur utama *framework* Next.js yang digunakan dalam pengembangan aplikasi serta manfaat implementasinya terhadap sistem pelaporan aktivitas mahasiswa ditampilkan pada Tabel 6.

Tabel 6. Fitur Next.js dan Manfaatnya dalam Pengembangan Aplikasi

Fitur	Deskripsi	Manfaat untuk Aplikasi
App Router	File-based <i>routing</i> + Server/Client Components	Routing otomatis untuk <i>dashboard</i> , form laporan
Hybrid Rendering	SSR + SSG + ISR + CSR	Dashboard <i>real-time</i> + panduan statis
Image Optimization	Auto WebP, <i>lazy loading</i> , responsive	Bukti kegiatan cepat <i>loading</i>
API Routes	Backend endpoints dalam Next.js	Verifikasi laporan, autentikasi
Built-in Tailwind	Zero-config Tailwind CSS integration	Styling responsive konsisten

1.6.11 Blackbox Testing

Blackbox *testing* adalah metode pengujian perangkat lunak yang digunakan untuk menilai sejauh mana suatu sistem berfungsi sesuai dengan kebutuhan yang telah ditentukan, dengan cara memeriksa *output* yang dihasilkan sistem berdasarkan *input* yang diberikan tanpa memperhatikan struktur internal atau kode program. Melalui blackbox *testing*, penguji dapat mengidentifikasi kesalahan-kesalahan pada fungsionalitas sistem, seperti fitur yang tidak berjalan sesuai harapan, kesalahan pada antarmuka, atau *output* yang tidak sesuai dengan spesifikasi (Hidayat & Suryayusra, 2022). Temuan ini kemudian digunakan sebagai dasar untuk memperbaiki sistem sebelum diluncurkan kepada pengguna. Pengujian ini dilakukan dengan mensimulasikan interaksi pengguna terhadap sistem untuk memastikan setiap fitur berjalan dengan benar tanpa error (Intyanto dkk., 2021).

Blackbox *testing* dipandang melalui beberapa aspek pengujian utama, yaitu pengujian fungsionalitas, pengujian antarmuka, dan pengujian alur sistem. Pengujian fungsionalitas berkaitan dengan kemampuan sistem menjalankan setiap fitur dengan benar sesuai kebutuhan, pengujian antarmuka berkaitan dengan kesesuaian tampilan dan navigasi sistem, sedangkan pengujian alur sistem berkaitan dengan kelancaran proses dari awal hingga akhir tanpa hambatan. Dalam sistem pelaporan aktivitas mahasiswa MBKM, pengujian fungsionalitas dapat mencakup proses *login*, pengisian laporan, dan pengiriman data, pengujian antarmuka dapat dilihat dari kesesuaian tampilan pada berbagai perangkat, dan pengujian alur sistem dapat dievaluasi dari kelancaran proses monitoring oleh dosen dan admin (Hidayat & Suryayusra, 2022).

Salah satu teknik yang banyak digunakan dalam blackbox *testing* adalah pengujian berbasis skenario (*test case*), yaitu serangkaian kondisi pengujian yang dirancang untuk memverifikasi apakah sistem menghasilkan *output* yang diharapkan. Test case dipilih karena terstruktur, dapat diterapkan pada berbagai jenis sistem, dan memberikan hasil yang reliabel dalam mengidentifikasi kesalahan fungsional. Setiap *test case* terdiri dari kondisi masukan, langkah pengujian, dan hasil yang diharapkan, di mana hasil pengujian dikategorikan sebagai "berhasil" apabila *output* sistem sesuai dengan yang diharapkan, dan "gagal" apabila terdapat ketidaksesuaian (Hidayat & Suryayusra, 2022). Pada

penelitian rancang bangun sistem informasi pelaporan aktivitas mahasiswa MBKM, *blackbox testing* digunakan untuk memvalidasi seluruh fitur yang dikembangkan, kemudian hasil pengujian dibandingkan dengan spesifikasi kebutuhan yang telah ditetapkan.

Hasil *blackbox testing* ini menjadi landasan apakah sistem sudah layak digunakan atau perlu dilakukan perbaikan lebih lanjut. Pengujian menggunakan metode *blackbox* juga dilakukan dalam penelitian oleh Intyanto dkk. (2021) yang menunjukkan bahwa metode ini mampu mengidentifikasi kesalahan fungsional sistem secara reliabel dan konsisten. Indikator pengujian *blackbox* yang digunakan untuk menilai kinerja sistem pelaporan aktivitas mahasiswa dirangkum pada Tabel 7.

Tabel 7. Indikator Evaluasi Kinerja Sistem Pelaporan

Aspek	Deskripsi Singkat	Contoh Indikator dalam Sistem Pelaporan
Fungsionalitas	Kemampuan sistem menjalankan fitur sesuai kebutuhan	Fitur <i>login</i> , pengisian laporan, dan pengiriman data berjalan tanpa error
Antarmuka	Kesesuaian tampilan dan navigasi sistem	Tombol, menu, dan halaman tampil sesuai rancangan
Alur Sistem	Kelancaran proses dari awal hingga akhir	Proses monitoring oleh dosen dan admin berjalan sesuai alur yang ditentukan

BAB II METODE PENELITIAN

2.1 Waktu dan Lokasi Penelitian

Penelitian ini dilaksanakan pada dua lokasi utama yang disesuaikan dengan tahapan pengembangan sistem menggunakan metode Agile. Tahapan awal berupa pengumpulan kebutuhan pengguna dilakukan di Kota Makassar selama tiga hari pada bulan Juli 2024 melalui proses diskusi dan identifikasi kebutuhan sistem bersama pihak terkait. Tahapan selanjutnya meliputi perancangan sistem (*design*), pengembangan aplikasi (*development*), serta pengujian sistem (*testing*) yang dilaksanakan secara iteratif di Kota Makassar mulai bulan Juli hingga akhir Agustus 2024. Proses tersebut dilakukan menggunakan pendekatan *sprint* pada metode Agile, sehingga aktivitas perancangan, implementasi, dan pengujian berjalan secara paralel dan berulang sesuai kebutuhan pengembangan.

Selanjutnya, tahap implementasi berupa pelatihan penggunaan sistem kepada pengguna dilaksanakan pada bulan September 2024 di Hotel Novena Bone, Kabupaten Bone, Sulawesi Selatan. Kegiatan pelatihan ini melibatkan dosen-dosen Universitas Muhammadiyah Bone sebagai pengguna sistem dengan tujuan memperkenalkan fitur aplikasi serta memastikan kesiapan penggunaan sistem pada lingkungan operasional. Keterlibatan langsung pengguna dalam tahap pelatihan ini juga menjadi bagian dari proses evaluasi *User-Centered Design*, di mana umpan balik dari pengguna dikumpulkan untuk penyempurnaan sistem lebih lanjut.

Setelah tahap implementasi, dilakukan fase pemeliharaan sistem (*maintenance*) yang berlangsung mulai bulan September 2024 hingga Agustus 2025. Pada fase ini dilakukan perbaikan bug, penyesuaian fitur, serta peningkatan performa sistem berdasarkan umpan balik pengguna sebagai bagian dari siklus pengembangan Agile yang berkelanjutan. Fase *maintenance* yang cukup panjang ini mencerminkan komitmen untuk terus menyempurnakan sistem berdasarkan kebutuhan nyata pengguna di lapangan, sejalan dengan prinsip-prinsip metode Agile yang menekankan respons terhadap perubahan (Nova dkk., 2022). Adapun jadwal kegiatan penelitian secara rinci dapat dilihat pada Gambar 3 berikut.

NO	TAHAPAN PENELITIAN	Jul 2024				Aug 2024				Sep 2024				Okt 2024 - Aug 2025
		1	2	3	4	1	2	3	4	1	2	3	4	
1.	<i>User Requirements</i>													
2.	<i>Design</i>													
3.	<i>Development</i>													
4.	<i>Testing</i>													
5.	Pelatihan Sistem													
6.	<i>Maintenance Sistem</i>													

Gambar 3. Jadwal Pelaksanaan Penelitian dan Pengembangan Sistem

2.2 User-Centered Design

Penelitian ini menerapkan pendekatan *User-Centered Design* (UCD) sebagai bagian integral dari proses pengembangan sistem informasi pelaporan aktivitas mahasiswa MBKM. Sebagaimana dijelaskan dalam landasan teori pada Bab I, UCD menempatkan pengguna sebagai pusat perhatian pada seluruh siklus pengembangan dengan tujuan memastikan bahwa produk akhir sesuai dengan kebutuhan, kemampuan, dan konteks penggunaan mereka.

Dalam konteks sistem pelaporan aktivitas MBKM di Universitas Muhammadiyah Bone, penerapan UCD menjadi sangat penting mengingat keberagaman karakteristik pengguna yang akan berinteraksi dengan sistem. Terdapat tiga kelompok pengguna utama dengan kebutuhan dan tugas yang berbeda: mahasiswa yang perlu melaporkan aktivitas secara mandiri, dosen pembimbing yang bertugas memverifikasi dan menilai laporan, serta administrator yang mengelola data *master* dan melakukan rekapitulasi kegiatan. Perbedaan latar belakang, tingkat literasi teknologi, dan alur kerja yang dimiliki masing-masing kelompok pengguna ini menjadi pertimbangan utama dalam setiap keputusan desain sistem.

Proses UCD dalam penelitian ini dilaksanakan melalui beberapa tahapan sistematis. Pertama, dilakukan aktivitas memahami dan menentukan konteks penggunaan melalui observasi langsung terhadap proses pelaporan MBKM yang berjalan saat ini di Universitas Muhammadiyah Bone serta wawancara dengan perwakilan dari masing-masing kelompok pengguna untuk mengidentifikasi kendala dan kebutuhan mereka. Kedua, tahap menentukan kebutuhan pengguna dan organisasi dilakukan dengan menganalisis hasil observasi dan wawancara untuk merumuskan kebutuhan fungsional sistem yang responsif terhadap kebutuhan riil di lapangan. Ketiga, tahap menghasilkan solusi desain berupa pembuatan sketsa dan prototipe antarmuka (*wireframe* dan *mockup*) yang kemudian dikembangkan menjadi rancangan UI/UX sistem. Keempat, tahap evaluasi desain dilakukan melalui pengujian prototipe kepada calon pengguna untuk memperoleh umpan balik yang kemudian digunakan sebagai dasar perbaikan desain secara iteratif (Purbo dkk., 2023).

Penerapan UCD dalam penelitian ini sejalan dengan karakteristik metode Agile yang menekankan iterasi cepat dan respons terhadap perubahan kebutuhan. Setiap *sprint* dalam pengembangan Agile akan melibatkan validasi dengan pengguna untuk memastikan fitur yang dikembangkan benar-benar memenuhi ekspektasi dan mudah digunakan. Dengan demikian, pendekatan UCD tidak hanya membantu menghasilkan sistem yang secara fungsional memadai, tetapi juga memastikan bahwa sistem tersebut efektif, efisien, dan memuaskan bagi seluruh pemangku kepentingan dalam ekosistem MBKM di Universitas Muhammadiyah Bone.

2.3 Metode Pengumpulan Data

Pengumpulan data dalam penelitian ini dilakukan melalui tiga metode utama yang saling melengkapi. Pertama, metode wawancara dilakukan dengan berbagai pemangku kepentingan di Universitas Muhammadiyah Bone, meliputi mahasiswa yang telah atau sedang mengikuti program MBKM, dosen pembimbing MBKM, serta administrator/pengelola program MBKM di tingkat universitas. Wawancara bertujuan untuk mengidentifikasi kebutuhan sistem, kendala yang dihadapi dalam proses

pelaporan aktivitas yang berjalan saat ini, serta harapan terhadap fitur-fitur yang akan dikembangkan. Pendekatan wawancara semi-terstruktur digunakan agar responden dapat memberikan informasi yang lebih mendalam dan komprehensif terkait permasalahan yang mereka hadapi.

Kedua, metode observasi dilakukan dengan mengamati secara langsung alur kerja pelaporan aktivitas MBKM yang sedang berjalan, termasuk media dan *tools* yang digunakan, kendala operasional yang terjadi, serta pola interaksi antara mahasiswa, dosen, dan administrator dalam proses *monitoring* dan evaluasi kegiatan. Observasi langsung memungkinkan peneliti untuk mendapatkan gambaran yang akurat mengenai kondisi aktual proses pelaporan, termasuk aspek-aspek yang mungkin tidak terungkap dalam wawancara. Hasil observasi ini kemudian digunakan sebagai bahan untuk merancang alur proses yang lebih efisien pada sistem SILASA.

Ketiga, metode studi literatur dilakukan dengan menelaah dan menganalisis secara sistematis berbagai sumber pustaka yang relevan dengan topik penelitian. Sumber data sekunder yang digunakan meliputi jurnal ilmiah tentang sistem informasi akademik dan program MBKM, buku referensi terkait rekayasa perangkat lunak dan metode Agile, dokumentasi resmi Kemdikbudristek mengenai kebijakan MBKM, penelitian terdahulu dengan tema serupa, serta dokumentasi teknis dari *framework* dan library yang digunakan (Next.js, Express.js, MySQL, dan Tailwind CSS). Seluruh informasi yang terkumpul dari ketiga metode tersebut kemudian disintesis untuk memperkaya kerangka teoretis dan menjadi landasan dalam perancangan serta pengembangan sistem informasi pelaporan aktivitas mahasiswa MBKM.

2.4 Metode Pengembangan Sistem

Metode pengembangan sistem yang digunakan dalam penelitian ini adalah metode Agile dengan *framework* Scrum sebagai pendekatan utama. Pemilihan metode Agile didasarkan pada karakteristik proyek yang membutuhkan fleksibilitas tinggi terhadap perubahan kebutuhan, iterasi pengembangan yang cepat, serta keterlibatan aktif pengguna dalam setiap tahap pengembangan. Sebagaimana dijelaskan dalam landasan teori, metode Agile menekankan pada empat nilai inti yaitu individu dan interaksi di atas proses dan *tools*, *software* yang berfungsi di atas dokumentasi yang komprehensif, kolaborasi dengan customer di atas negosiasi kontrak, serta respons terhadap perubahan di atas mengikuti rencana (Nova dkk., 2022).

Implementasi metode Agile dalam penelitian ini dilakukan melalui *sprint* dengan durasi 2 minggu per *sprint*. Setiap *sprint* mencakup aktivitas-aktivitas sebagai berikut:

1. *Sprint Planning*
Menentukan *user stories* dari *product backlog* yang akan dikerjakan pada *sprint* tersebut berdasarkan prioritas kebutuhan sistem.
2. Daily Standup
Pertemuan koordinasi harian untuk membahas progress pekerjaan, rencana hari ini, dan kendala yang dihadapi.
3. Development
Implementasi fitur-fitur sistem sesuai dengan *user stories* yang telah ditentukan, termasuk coding, *testing*, dan integrasi.

4. *Sprint Review*

Demonstrasi hasil pengembangan kepada *stakeholder* (mahasiswa, dosen pembimbing, dan administrator Universitas Muhammadiyah Bone) untuk mendapatkan *feedback*.

5. *Sprint Retrospective*

Evaluasi proses *sprint* yang telah berjalan untuk mengidentifikasi hal-hal yang berjalan baik dan area yang perlu diperbaiki pada *sprint* berikutnya.

Pendekatan Agile ini diintegrasikan dengan prinsip *User-Centered Design* (UCD) yang telah dijelaskan sebelumnya, di mana setiap iterasi pengembangan selalu melibatkan validasi dengan pengguna untuk memastikan sistem yang dihasilkan tidak hanya memenuhi kebutuhan fungsional, tetapi juga memberikan pengalaman pengguna (*user experience*) yang optimal. Kombinasi antara Agile dan UCD memungkinkan peneliti untuk menghasilkan sistem informasi pelaporan aktivitas mahasiswa MBKM yang responsif terhadap kebutuhan dinamis program MBKM di Universitas Muhammadiyah Bone, sekaligus memastikan kualitas dan kepuasan pengguna yang tinggi.

2.5 Analisis Kebutuhan Sistem

Analisis kebutuhan sistem merupakan tahap penting dalam proses pengembangan untuk mengidentifikasi seluruh sumber daya yang diperlukan agar sistem dapat berfungsi dengan optimal. Kebutuhan sistem dalam penelitian ini dibagi menjadi tiga kategori utama, yaitu kebutuhan perangkat lunak (*software*), perangkat keras (*hardware*), dan pengguna sistem (*user*). Analisis yang komprehensif terhadap ketiga aspek ini memastikan bahwa sistem informasi pelaporan aktivitas mahasiswa MBKM yang dikembangkan dapat diimplementasikan dan dioperasikan dengan baik di lingkungan Universitas Muhammadiyah Bone.

2.5.1 Spesifikasi perangkat keras

Kebutuhan perangkat keras yang digunakan peneliti dalam pengembangan sistem adalah sebagai berikut:

Tabel 8. Spesifikasi Perangkat Keras Pengembangan Sistem

Komponen	Spesifikasi
Perangkat	Laptop Asus ZenBook Pro Duo UX581GV
Processor	Intel Core i7-9750H
RAM	32GB
Penyimpanan	SSD 1TB

2.5.2 Spesifikasi perangkat lunak

Daftar perangkat lunak yang digunakan dalam membangun sistem meliputi:

Tabel 9. Spesifikasi Perangkat Lunak Pengembangan Sistem

Komponen	Spesifikasi
Sistem Operasi	Windows 11
Editor Kode	Visual Studio Code
Bahasa & Runtime	Node.js, Express.js (Backend), Next.js (Frontend)
Basis Data	MySQL
Desain & Styling	Figma, Tailwind CSS
Tools Pengujian	Postman, Google Chrome
Version Control	Git & GitHub

2.5.3 Analisis Pengguna

Sistem informasi pelaporan aktivitas mahasiswa MBKM dirancang untuk melayani lima kategori pengguna dengan peran dan hak akses yang berbeda:

1) Super User / Admin Universitas

Super User atau Admin Universitas memiliki hak akses penuh terhadap seluruh fitur dan data dalam sistem. Peran dan tanggung jawab meliputi:

- Mengelola data *master* seperti semester akademik, program studi, fakultas, dan jenis kegiatan MBKM.
- Mengelola akun pengguna untuk semua *level* (Admin Fakultas, Admin Prodi, Dosen, dan Mahasiswa).
- Melakukan konfigurasi sistem dan pengaturan parameter umum.
- Mengakses seluruh laporan dan rekapitulasi kegiatan MBKM di tingkat universitas.

2) Admin Fakultas

Admin Fakultas memiliki hak akses terbatas pada data dan kegiatan di lingkup fakultas yang dikelolanya. Peran dan tanggung jawab meliputi:

- Mengelola data fakultas yang menjadi tanggung jawabnya.
- Memantau kegiatan MBKM mahasiswa di seluruh program studi dalam fakultasnya.
- Mengakses laporan dan rekapitulasi kegiatan MBKM di tingkat fakultas.
- Melakukan verifikasi data mahasiswa dan dosen di lingkup fakultas.
- Berkoordinasi dengan Admin Prodi dalam pelaksanaan program MBKM.

3) Admin Prodi

Admin Prodi memiliki hak akses terbatas pada data dan kegiatan di lingkup program studi yang dikelolanya. Peran dan tanggung jawab meliputi:

- Mengelola data program studi yang menjadi tanggung jawabnya.
- Menetapkan dosen pembimbing/verifikator untuk mahasiswa di prodinya.
- Memantau kegiatan MBKM mahasiswa dalam program studinya.
- Mengakses laporan dan rekapitulasi kegiatan MBKM di tingkat program studi.
- Memverifikasi kelengkapan dokumen dan persyaratan kegiatan MBKM mahasiswa.
- Mengelola usulan kegiatan MBKM dan RKL dari mahasiswa.

4) Dosen

Dosen berperan sebagai pembimbing dan verifikator kegiatan MBKM mahasiswa. Peran dan tanggung jawab meliputi:

- Memverifikasi laporan aktivitas yang *disubmit* oleh mahasiswa bimbingannya.
- Memberikan penilaian terhadap kegiatan MBKM yang dilakukan mahasiswa berdasarkan rubrik penilaian yang telah ditetapkan.
- Melakukan *monitoring* progress mahasiswa dalam menjalankan kegiatan MBKM.
- Memberikan *feedback* dan catatan perbaikan pada laporan mahasiswa.
- Menyetujui atau menolak usulan kegiatan MBKM mahasiswa.
- Melakukan konversi nilai kegiatan MBKM ke mata kuliah sesuai dengan capaian pembelajaran.

5) Mahasiswa

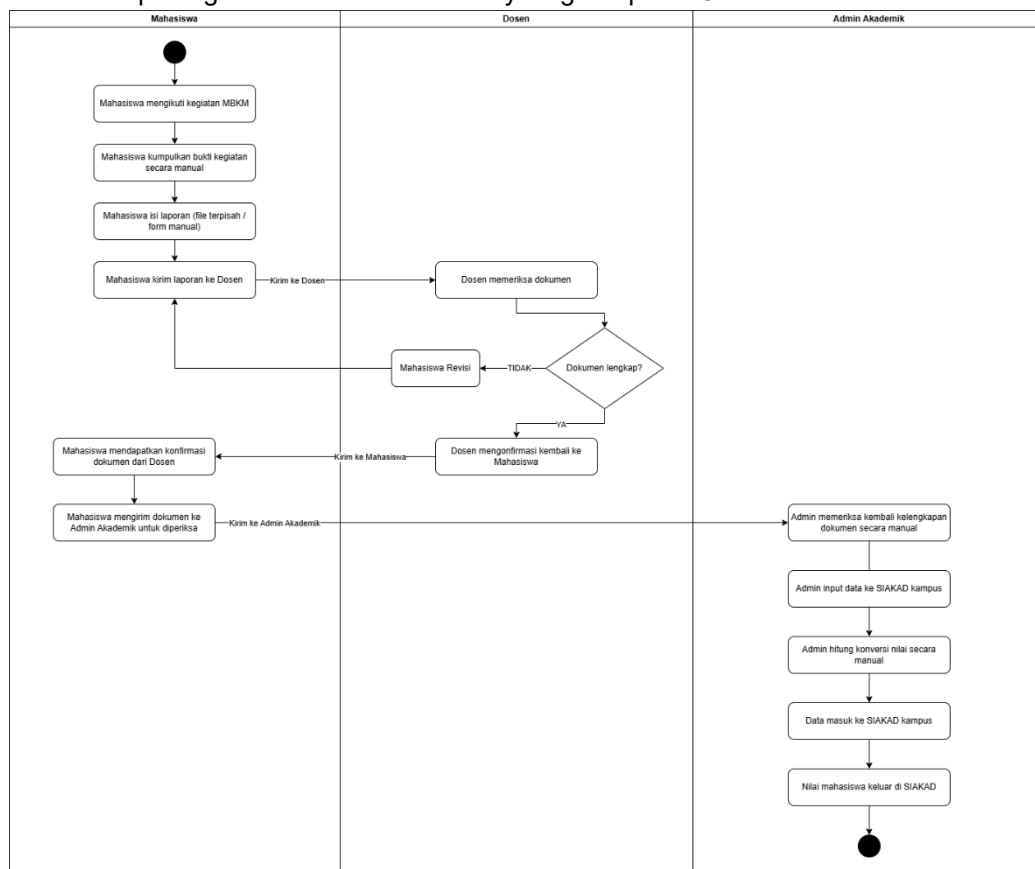
Mahasiswa adalah pengguna utama sistem yang melaporkan aktivitas kegiatan MBKM mereka. Peran dan tanggung jawab meliputi:

- Mendaftarkan diri dan membuat akun pada sistem.
- Mengajukan usulan kegiatan MBKM (baik MBKM maupun RKL) yang akan diikuti.
- Melaporkan aktivitas kegiatan MBKM secara berkala sesuai dengan tahapan yang telah ditetapkan.
- Mengunggah bukti-bukti pendukung (dokumen, foto, sertifikat) untuk setiap aktivitas.
- Mengisi *logbook* atau jurnal kegiatan harian selama pelaksanaan MBKM.
- Melihat status verifikasi dan *feedback* dari dosen pembimbing.
- Mengakses riwayat kegiatan dan nilai yang telah diperoleh.
- Menerima notifikasi terkait status laporan dan pengingat deadline.
- Mengunduh sertifikat atau bukti penyelesaian kegiatan MBKM.

2.6 Analisis Masalah

Berdasarkan hasil wawancara dan observasi yang dilakukan di Universitas Muhammadiyah Bone, ditemukan beberapa permasalahan signifikan dalam proses pelaporan dan *monitoring* kegiatan MBKM yang masih dilakukan secara manual. Mahasiswa mengalami kesulitan dalam melaporkan aktivitas mereka karena tidak adanya *platform* terpusat yang dapat diakses kapan saja dan di mana saja, sehingga

proses pelaporan sering kali dilakukan melalui berbagai media yang berbeda seperti *email*, Whatsapp, atau formulir kertas yang harus diserahkan langsung. Kondisi ini menyebabkan inkonsistensi format laporan, risiko kehilangan dokumen, serta sulitnya pelacakan status verifikasi laporan. Dari sisi dosen pembimbing, proses verifikasi dan penilaian menjadi tidak efisien karena harus mengelola dokumen dari berbagai sumber, kesulitan dalam memantau progress mahasiswa secara *real-time*, dan memerlukan waktu yang lama untuk memberikan *feedback* kepada mahasiswa. Berdasarkan kondisi tersebut, alur proses pelaporan dan *monitoring* kegiatan MBKM sebelum adanya sistem SILASA dapat digambarkan melalui activity diagram pada Gambar 4.

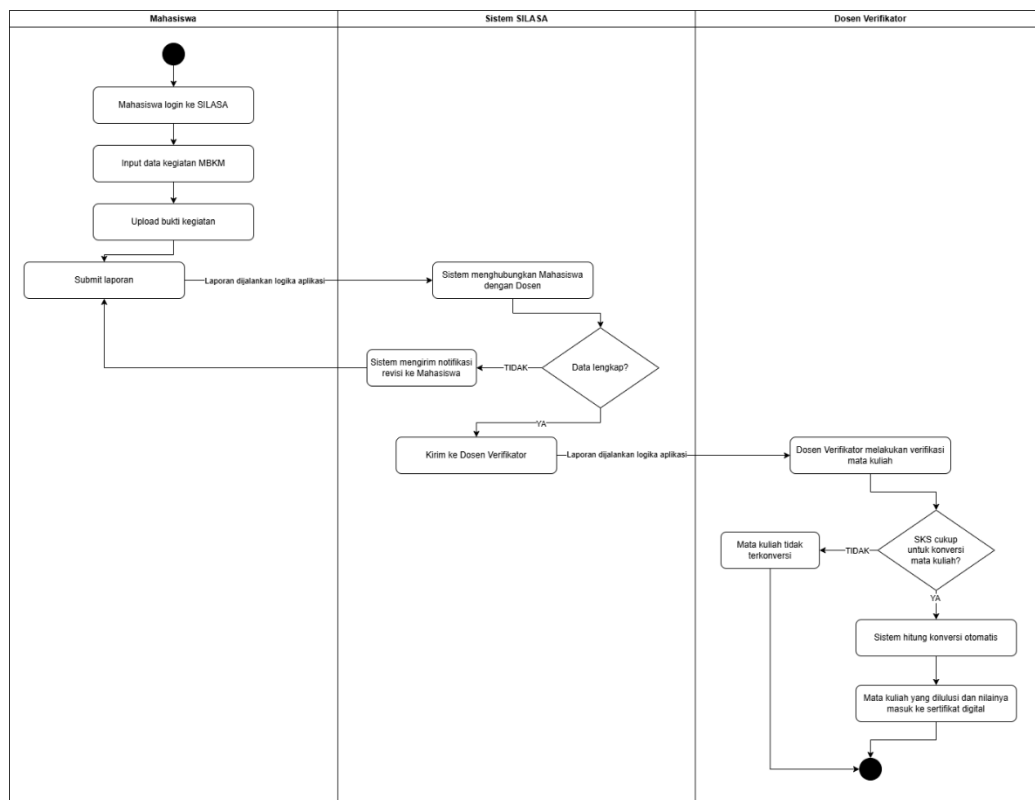


Gambar 4. Alur Proses Pelaporan dan Monitoring Kegiatan MBKM Versi Manual

Pada Gambar 4 dapat dilihat bahwa proses pelaporan kegiatan MBKM masih dilakukan secara manual dimulai dari mahasiswa mengikuti kegiatan hingga pengumpulan bukti kegiatan secara terpisah. Mahasiswa kemudian mengirimkan laporan kepada dosen untuk dilakukan pemeriksaan dokumen. Proses verifikasi dilakukan secara manual sehingga apabila dokumen dinyatakan belum lengkap, mahasiswa harus melakukan revisi dan mengirim ulang laporan. Selanjutnya, laporan yang telah disetujui diteruskan kepada admin akademik untuk dilakukan *input* data dan perhitungan konversi nilai secara manual. Alur tersebut menyebabkan proses menjadi tidak efisien, tidak *real-time*, serta berpotensi menimbulkan kesalahan pengolahan data.

Permasalahan juga terjadi di tingkat administrasi universitas, fakultas, dan program studi dalam hal pengelolaan data dan rekapitulasi kegiatan MBKM. Administrator menghadapi kendala dalam mengumpulkan data kegiatan mahasiswa dari berbagai program studi, kesulitan dalam membuat laporan yang akurat dan tepat waktu untuk keperluan pelaporan ke LLDIKTI dan PDDIKTI, serta tidak tersedianya *dashboard monitoring* yang dapat memberikan gambaran *real-time* mengenai partisipasi mahasiswa dalam program MBKM. Ketidadaan sistem informasi yang terintegrasi juga berdampak pada transparansi dan akuntabilitas pelaksanaan program MBKM, di mana pemangku kepentingan sulit untuk memantau capaian program dan mengidentifikasi kendala yang terjadi di lapangan. Hal ini sejalan dengan temuan penelitian yang menyatakan bahwa *monitoring* dan evaluasi aktivitas mahasiswa masih menjadi tantangan signifikan bagi banyak lembaga pendidikan, terutama dalam hal transparansi dan ketersediaan data *real-time*.

Permasalahan-permasalahan tersebut menunjukkan urgensi untuk mengembangkan sistem informasi pelaporan aktivitas mahasiswa berbasis web yang dapat mengintegrasikan seluruh proses pelaporan, verifikasi, penilaian, dan *monitoring* kegiatan MBKM dalam satu *platform* terpusat. Sistem yang akan dikembangkan diharapkan dapat mengatasi kendala-kendala yang ada dengan menyediakan antarmuka yang *user-friendly* bagi mahasiswa untuk melaporkan aktivitas, *dashboard monitoring* bagi dosen untuk melakukan verifikasi dan penilaian, serta fitur rekapitulasi dan analisis data bagi administrator untuk mendukung pengambilan keputusan dan pelaporan institusional. Dengan demikian, sistem ini tidak hanya bertujuan untuk meningkatkan efisiensi operasional, tetapi juga untuk meningkatkan transparansi, akuntabilitas, dan kualitas pelaksanaan program MBKM di Universitas Muhammadiyah Bone (Danihsan dkk., 2025). Sebagai solusi terhadap permasalahan yang telah diidentifikasi, dirancang sistem SILASA yang mengintegrasikan proses pelaporan, verifikasi, dan *monitoring* kegiatan MBKM dalam satu *platform* berbasis web. Alur proses yang diusulkan dapat dilihat pada activity diagram pada Gambar 5.



Gambar 5. Rancangan SILASA sebagai Proses Pelaporan dan Monitoring Kegiatan MBKM

Pada Gambar 5 dapat dilihat bahwa sistem SILASA memungkinkan mahasiswa untuk melakukan pelaporan kegiatan secara langsung melalui *platform* terpusat dengan melakukan *login*, *menginput* data kegiatan, serta mengunggah bukti kegiatan. Sistem secara otomatis melakukan validasi data sebelum laporan diteruskan kepada dosen verifikator. Proses verifikasi dilakukan secara terstruktur sehingga apabila data belum lengkap, sistem akan mengirimkan notifikasi revisi kepada mahasiswa. Jika laporan disetujui, sistem secara otomatis melakukan perhitungan konversi mata kuliah serta menyimpan hasilnya dalam sistem digital. Dengan adanya alur ini, proses *monitoring* menjadi lebih efisien, transparan, dan *real-time* dibandingkan proses sebelumnya.

2.7 Tahapan Penelitian

Tahapan penelitian ini disusun secara sistematis untuk memastikan proses rancang bangun sistem informasi pelaporan aktivitas MBKM di Universitas Muhammadiyah Bone berjalan sesuai dengan tujuan yang telah ditetapkan. Proses penelitian diawali dengan tahap pra-pengembangan, yang meliputi identifikasi masalah melalui observasi lapangan dan wawancara untuk memahami kendala *monitoring* kegiatan mahasiswa secara manual, serta studi literatur guna memperkuat landasan teori terkait sistem informasi dan manajemen MBKM. Studi literatur yang dilakukan mengacu pada penelitian-penelitian terdahulu yang relevan, seperti penelitian Danihsan et al. (2025) tentang sistem informasi

MBKM berbasis web dan penelitian Nova et al. (2022) tentang penerapan metode Agile dalam pengembangan sistem informasi.

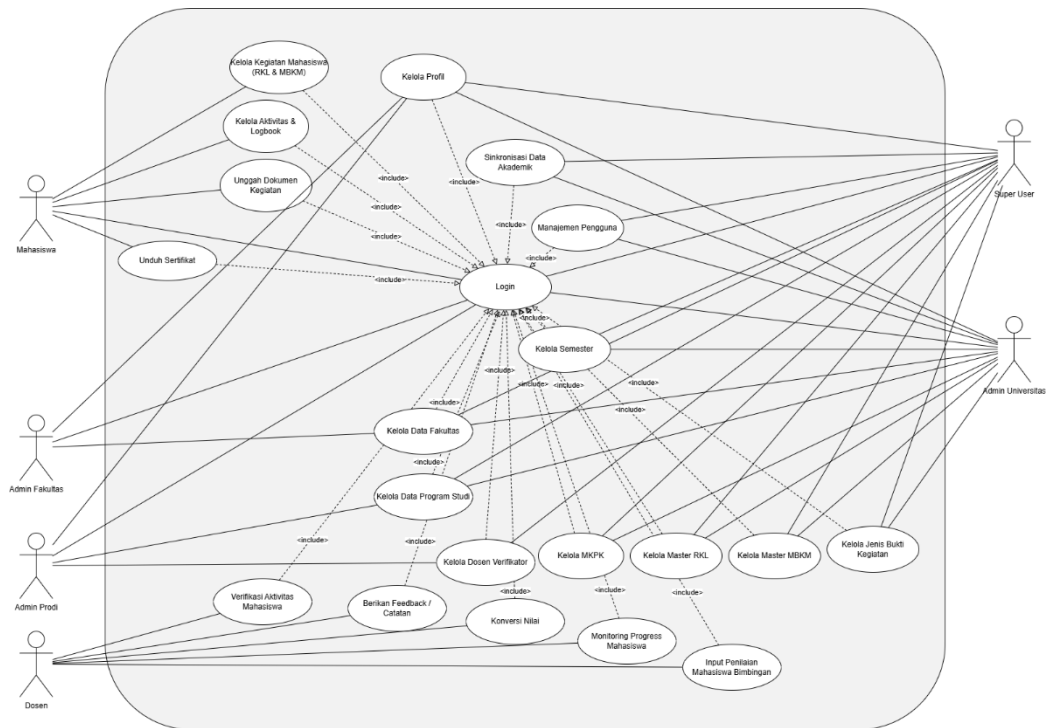
Selanjutnya, dilakukan analisis kebutuhan fungsional dan non-fungsional yang melibatkan pemangku kepentingan untuk memastikan sistem yang dirancang mampu menjawab tantangan administrasi dan transparansi data di tingkat universitas. Tahapan analisis kebutuhan ini dilakukan dengan menggunakan pendekatan *User-Centered Design* yang melibatkan seluruh kelompok pengguna secara aktif, sehingga kebutuhan yang dirumuskan benar-benar mencerminkan kebutuhan riil di lapangan (Purbo dkk., 2023).

Memasuki tahap inti, penelitian ini menerapkan siklus pengembangan Agile yang bersifat iteratif dan inkremental, di mana proses desain, pengkodean, dan pengujian dilakukan dalam beberapa iterasi (*sprint*) yang fleksibel terhadap perubahan kebutuhan pengguna. Selaras dengan pendekatan *User-Centered Design* (UCD), setiap hasil rancangan antarmuka dan fitur sistem akan dikembangkan menggunakan teknologi Next.js dan MySQL, kemudian langsung dievaluasi untuk mendapatkan umpan balik pengguna. Tahapan ini diakhiri dengan pengujian sistem menggunakan metode Blackbox Testing untuk memastikan seluruh fungsionalitas sistem berjalan sesuai dengan kebutuhan yang telah ditetapkan (Intyanto dkk., 2021), yang kemudian diikuti dengan penyusunan laporan akhir skripsi sebagai dokumentasi formal dari seluruh hasil penelitian dan implementasi yang telah dilakukan.

2.8 Perancangan Sistem

Perancangan sistem merupakan tahapan krusial untuk mentransformasikan hasil analisis kebutuhan ke dalam cetak biru (blueprint) teknis yang terstruktur sebelum memasuki fase pengkodean. Dalam penelitian ini, perancangan dilakukan dengan menggunakan instrumen Unified Modeling Language (UML) untuk memvisualisasikan interaksi dan perilaku sistem secara komprehensif. Penggunaan UML dalam perancangan sistem informasi memudahkan komunikasi antara pengembang dengan pemangku kepentingan serta memastikan bahwa sistem yang dibangun sesuai dengan kebutuhan yang telah diidentifikasi (Etrariadi & A'inunisyah, 2023).

Fokus utama pada bagian ini adalah penggunaan Use Case Diagram, yang berfungsi untuk mendefinisikan peran dari setiap aktor—mulai dari mahasiswa, dosen pembimbing, hingga administrator—serta hubungan fungsional mereka terhadap fitur-fitur utama dalam sistem pelaporan aktivitas MBKM di Universitas Muhammadiyah Bone. Hal ini memastikan bahwa seluruh alur kerja sistem yang dibangun dalam siklus Agile tetap konsisten dengan kebutuhan pengguna yang telah diidentifikasi sebelumnya. Selain Use Case Diagram, Activity Diagram juga digunakan untuk memodelkan alur proses bisnis, dan Entity Relationship Diagram (ERD) digunakan untuk merancang struktur basis data (Pulungan dkk., 2023). Konsistensi antara kebutuhan pengguna dan rancangan sistem divisualisasikan melalui Use Case Diagram yang ditampilkan pada Gambar 6, yang menggambarkan interaksi antara aktor sistem dengan fitur-fitur utama aplikasi SILASA.



Gambar 6. Use Case Diagram Sistem Pelaporan Aktivitas MBKM

2.9 Rancangan User Interface

Rancangan *User Interface* (UI) merupakan perwujudan visual dari seluruh analisis kebutuhan dan prinsip *User-Centered Design* (UCD) yang telah ditetapkan sebelumnya. Tahap ini bertujuan mentransformasikan alur kerja sistem ke dalam bentuk *mockup high-fidelity*, yang memberikan gambaran nyata mengenai tata letak (*layout*), tipografi, warna, dan elemen navigasi sistem SILASA. Rancangan antarmuka mencakup halaman *login*, *dashboard* untuk masing-masing peran pengguna, *form* pelaporan aktivitas MBKM, halaman verifikasi laporan, dan halaman rekap konversi nilai.

Setiap rancangan antarmuka yang dihasilkan kemudian divalidasi melalui proses *user testing* dengan melibatkan perwakilan dari masing-masing kelompok pengguna, sehingga umpan balik yang diperoleh dapat dijadikan dasar perbaikan desain secara iteratif hingga antarmuka memenuhi standar penggunaan yang diharapkan. Dengan pendekatan ini, sistem SILASA diharapkan menghasilkan antarmuka yang tidak hanya estetik secara visual, tetapi juga efektif dan efisien dalam mendukung proses pelaporan dan monitoring kegiatan MBKM di Universitas Muhammadiyah Bone.

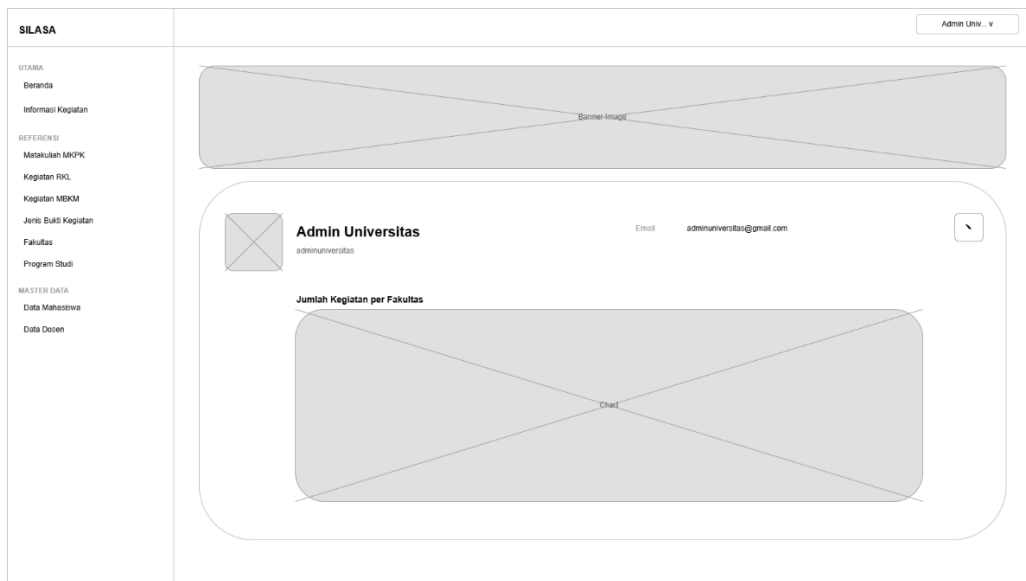
2.9.1 Halaman Login.

Halaman *login* merupakan gerbang utama bagi seluruh aktor sistem untuk melakukan autentikasi identitas sebelum mengakses fitur pelaporan dan *monitoring* MBKM. Pada tampilan ini, pengguna diwajibkan untuk memasukkan alamat *email* dan kata sandi yang valid guna memastikan keamanan data serta kesesuaian hak akses—baik bagi mahasiswa, dosen pembimbing, maupun administrator.

Gambar 7. Mockup Halaman Login

2.9.2 Halaman Beranda.

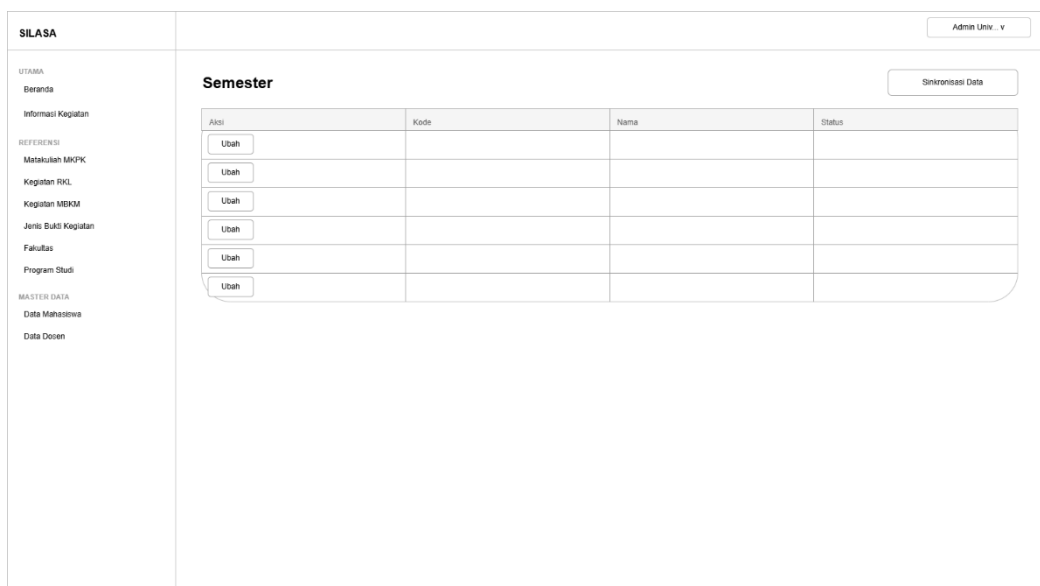
Halaman beranda merupakan antarmuka utama yang pertama kali ditampilkan saat pengguna mengakses sistem informasi pelaporan MBKM. Halaman ini berfungsi sebagai pusat informasi awal yang menyajikan ringkasan mengenai program Merdeka Belajar Kampus Merdeka di Universitas Muhammadiyah Bone, serta menyediakan navigasi utama bagi pengguna untuk masuk ke dalam sistem sesuai dengan peran masing-masing.



Gambar 8. Mockup Halaman Beranda

2.9.3 Halaman Semester

Halaman semester berfungsi sebagai media untuk mengelola dan mengelompokkan data laporan mahasiswa berdasarkan periode tahun ajaran serta semester yang sedang berjalan. Melalui halaman ini, pengguna dapat memilih atau melihat daftar semester aktif guna memastikan bahwa setiap aktivitas MBKM yang dilaporkan tercatat secara terstruktur sesuai dengan linimasa akademik yang berlaku di Universitas Muhammadiyah Bone.



Gambar 9. Mockup Halaman Semester

2.9.4 Pop up Ubah Data Semester.

Pop up ini merupakan jendela dialog yang berfungsi untuk melakukan pembaruan atau modifikasi pada data semester yang telah terdaftar di dalam sistem. Melalui tampilan ini, pengguna dapat mengubah informasi tertentu seperti nama semester atau periode akademik secara cepat dan efisien tanpa harus meninggalkan halaman utama manajemen semester, sehingga memudahkan pemeliharaan data yang tetap akurat dan relevan.

The image shows a web application interface. On the left is a sidebar with the title 'SILASA' and a list of menu items: UTAMA, Beranda, Informasi Kegiatan, REFERENSI, Matakuliah MBKM, Kegiatan RKL, Kegiatan MBKM, Jenis Bukti Kegiatan, Fakultas, Program Studi, MASTER DATA, Data Mahasiswa, and Data Dosen. The main content area is titled 'Semester' and contains a table with columns 'Aksi', 'Kode', 'Nama', and 'Status'. There are 'Ubah' buttons in the 'Aksi' column for each row. A 'Simpan Data' button is in the top right. A 'Ubah Data Semester' pop-up dialog is centered over the table. It has a title bar with a close button 'X'. Inside, there are three input fields: 'Nama Semester' (containing '2025/2026 Ganjil'), 'Kode Semester' (containing '20251'), and 'Status' (containing 'Aktif'). At the bottom are 'Batal' and 'Simpan' buttons.

Gambar 10. Mockup Pop up Ubah Data Semester

2.9.5 Halaman Manajemen Akun Pengguna.

Halaman ini merupakan antarmuka khusus bagi administrator yang berfungsi untuk mengelola seluruh data pengguna sistem, baik mahasiswa maupun dosen pembimbing. Melalui halaman ini, administrator dapat memantau akun yang terdaftar, melakukan aktivasi atau deaktivasi akun, serta memastikan setiap pengguna memiliki peran (*role*) yang tepat sesuai dengan otoritasnya dalam proses pelaporan dan *monitoring* MBKM.

The mockup shows a web application interface for user management. On the left is a sidebar with a 'SILASA' logo and a menu including 'UTAMA', 'Beranda', 'Informasi Kegiatan', 'REFERENSI', 'Matakuliah MKPK', 'Kegiatan RKL', 'Kegiatan MEKM', 'Jenis Buku Kegiatan', 'Fakultas', 'Program Studi', 'MASTER DATA', 'Data Mahasiswa', and 'Data Dosen'. The main content area is titled 'Manajemen Akun Pengguna' and includes a 'Data Baru' button. Below the title is a 'Daftar Akun' section with a search bar 'Cari akun berdasarkan username'. A table lists users with columns for Aksi, Username, Email, Nama, Status, and Keterangan. Each row has 'Ubah' and 'Reset' buttons.

Aksi	Username	Email	Nama	Status	Keterangan
Ubah Reset					
Ubah Reset					
Ubah Reset					
Ubah Reset					
Ubah Reset					
Ubah Reset					

Gambar 11. Mockup Halaman Manajemen Akun Pengguna

2.9.6 Pop up Tambah Akun

Pop up ini merupakan jendela dialog yang digunakan oleh administrator untuk memasukkan data pengguna baru ke dalam sistem secara cepat. Melalui formulir pada jendela ini, administrator dapat menentukan identitas dasar pengguna serta menetapkan hak akses spesifik, sehingga proses registrasi akun mahasiswa maupun dosen dapat dilakukan tanpa harus berpindah halaman secara penuh.

The mockup shows a 'Tambah Akun Mahasiswa' pop-up form overlaid on the 'Manajemen Akun Pengguna' page. The form includes fields for 'Nama' (Masukkan nama user), 'Email' (Masukkan email user), 'Pilih Mahasiswa', 'Username' (Masukkan username), 'Keterangan (Optional)' (Masukkan keterangan), 'Password' (Masukkan password), and 'Konfirmasi Password' (Masukkan password). At the bottom are 'Batal' and 'Simpan' buttons.

Gambar 12. Mockup Pop up Tambah Akun

2.9.7 Halaman Informasi Kegiatan Mahasiswa

Halaman ini menyajikan detail lengkap mengenai aktivitas yang telah dilaksanakan oleh mahasiswa selama mengikuti program MBKM. Antarmuka ini memuat informasi komprehensif seperti deskripsi kegiatan, waktu pelaksanaan, serta lampiran bukti pendukung, yang berfungsi sebagai instrumen bagi dosen pembimbing untuk memantau kualitas dan relevansi kegiatan mahasiswa secara mendalam.

SILASA

Adm Univ - v

UTAMA

- Beranda
- Informasi Kegiatan

REFERENSI

- Matakuliah MKPK
- Kegiatan RKL
- Kegiatan MBKM
- Jenis Bukti Kegiatan
- Fasilitas
- Program Studi

MASTER DATA

- Data Mahasiswa
- Data Dosen

Informasi Kegiatan Mahasiswa

Cari mahasiswa berdasarkan NIM atau nama

Nama	Jurusan	Jumlah Kegiatan
...		
...		
...		
...		
...		
...		
...		
...		
...		
...		

Gambar 13. Mockup Halaman Informasi Kegiatan Mahasiswa

2.9.8 Halaman Daftar Kegiatan Mahasiswa

Halaman ini menampilkan tabulasi data seluruh aktivitas yang telah diinputkan oleh mahasiswa dalam satu periode program MBKM. Antarmuka ini dirancang untuk memudahkan pengguna dalam melihat rangkuman daftar kegiatan secara kronologis, lengkap dengan indikator status verifikasi, sehingga mahasiswa dapat dengan mudah mengidentifikasi laporan mana yang telah disetujui, memerlukan revisi, atau belum diperiksa oleh dosen pembimbing.

SILASA

Admin Univ. v

UTAMA
Beranda
Informasi Kegiatan
REFERENSI
Matakuliah MKPK
Kegiatan RKL
Kegiatan MBKM
Jenis Bukti Kegiatan
Fakultas
Program Studi
MASTER DATA
Data Mahasiswa
Data Dosen

Daftar Kegiatan Mahasiswa

Basis	Kegiatan	Status Keanggotaan
...		
...		
...		
...		
...		
...		


FITRI RAMADANI
NIM: 2169010302
Email: -
VERIFIKATOR: SIRWANTI S Pd., M Pd.

Gambar 14. Mockup Halaman Daftar Kegiatan Mahasiswa

2.9.9 Halaman Mata Kuliah MKPK

Halaman ini digunakan untuk mengelola daftar Mata Kuliah Pilihan Konsentrasi (MKPK) yang dapat dikonversi dari kegiatan MBKM mahasiswa. Antarmuka ini memuat informasi kode dan nama mata kuliah guna memastikan sinkronisasi antara aktivitas lapangan dengan kurikulum akademik di Universitas Muhammadiyah Bone.

SILASA

Admin Univ. v

UTAMA
Beranda
Informasi Kegiatan
REFERENSI
Matakuliah MKPK
Kegiatan RKL
Kegiatan MBKM
Jenis Bukti Kegiatan
Fakultas
Program Studi
MASTER DATA
Data Mahasiswa
Data Dosen

Mata Kuliah MKPK

+ Tambah Mata Kuliah Baru

Cari matakuliah berdasarkan nama

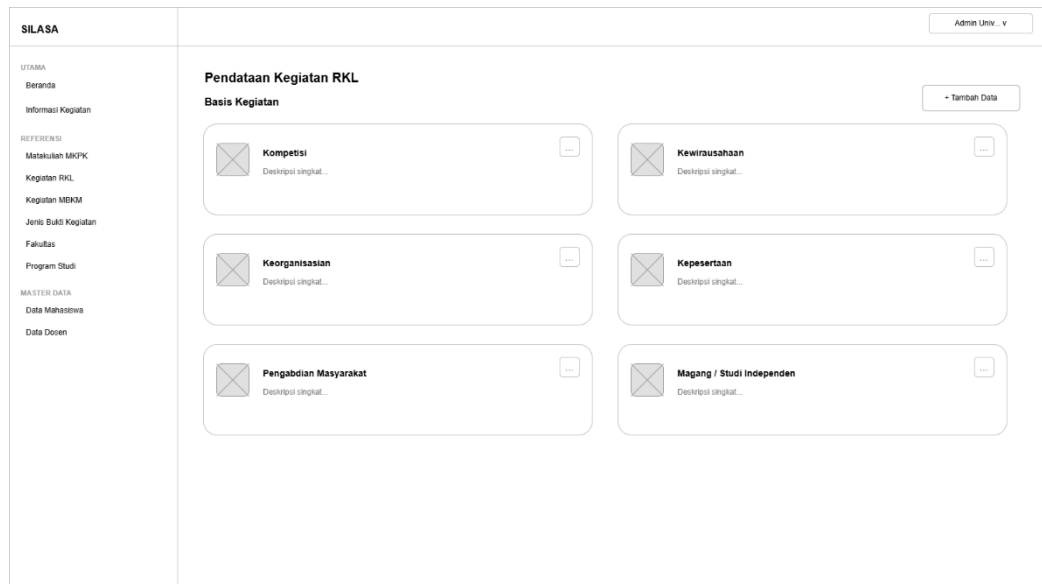
Kategori: Semua v

Aksi	Kode	Nama Mata Kuliah	SKS	Kategori
Ubah				
Ubah				
Ubah				
Ubah				
Ubah				
Ubah				
Ubah				

Gambar 15. Mockup Halaman Mata Kuliah MKPK

2.9.10 Halaman Basis Kegiatan RKL

Halaman ini berfungsi untuk mengelola data Rencana Kegiatan Laporan (RKL) yang menjadi acuan utama mahasiswa dalam menjalankan program MBKM. Antarmuka ini memungkinkan admin atau dosen untuk menetapkan parameter kegiatan yang harus dipenuhi, sehingga proses *monitoring* laporan memiliki standar evaluasi yang jelas dan terukur.



Gambar 16. Mockup Halaman Basis Kegiatan RKL

2.9.11 Halaman Aktivitas Kegiatan RKL

Halaman ini menyajikan rincian poin-poin aktivitas yang harus dilaksanakan mahasiswa berdasarkan rencana yang telah ditetapkan pada RKL. Antarmuka ini berfungsi sebagai panduan operasional bagi mahasiswa dalam mengisi laporan harian agar tetap sinkron dengan target kompetensi yang ingin dicapai selama program MBKM.

Pemilihan Mahasiswa Berprestasi (PILMAPRES)

Kategori

Aktivitas

Mata Kuliah

Nilai

Pilih Aktivitas: Proposal / Sewasi Fakultas

+ Tambah Aktivitas

Ubah Aktivitas

Hapus

+ Tambah Sub Aktivitas

Sub Aktivitas

Aksi	Sub Aktivitas	Durasi (Jam)	Capaian SKS	Status	Dokumen

Gambar 17. Mockup Halaman Aktivitas Kegiatan RKL

2.9.12 Halaman Mata Kuliah yang Direkognisi Kegiatan RKL

Halaman ini menampilkan daftar mata kuliah yang telah disetujui untuk dikonversi nilainya berdasarkan aktivitas yang dilakukan mahasiswa dalam RKL. Antarmuka ini memberikan transparansi bagi mahasiswa mengenai pemetaan kegiatan lapangan ke dalam bobot SKS akademik secara tepat.

Pemilihan Mahasiswa Berprestasi (PILMAPRES)

Ubah

Hapus

Kategori

Aktivitas

Mata Kuliah

Nilai

+ Tambah Mata Kuliah ke Kegiatan

	Kode	Nama Mata Kuliah	SKS	Kategori

Gambar 18. Mockup Halaman Mata Kuliah yang Direkognisi Kegiatan RKL

2.9.13 Halaman Nilai Kegiatan RKL

Halaman ini digunakan untuk mengelola dan menampilkan perolehan nilai mahasiswa berdasarkan hasil evaluasi aktivitas yang telah dilakukan. Antarmuka ini memfasilitasi dosen atau admin dalam *menginput* nilai secara objektif sesuai dengan parameter penilaian yang telah ditentukan dalam RKL.

The mockup shows a web interface for 'Pemilihan Mahasiswa Berprestasi (PILMAPRES)'. It features a sidebar on the left and a main content area. The main area has a header with 'Ubah' and 'Hapus' buttons. Below the header are filter buttons: 'Kategori', 'Aktifitas', 'Mata Kuliah', and 'Nilai'. A '+ Tambah Penilaian' button is also present. The main section is titled 'Rubrik Penilaian' and contains a table with three columns: 'Aksi', 'Butir', and 'Nilai'. The table has five empty rows for data entry.

Aksi	Butir	Nilai

Gambar 19. Mockup Halaman Nilai Kegiatan RKL

2.9.14 Halaman Kegiatan MBKM

Halaman ini menyajikan daftar seluruh program MBKM yang tersedia dan dapat diikuti oleh mahasiswa di Universitas Muhammadiyah Bone. Antarmuka ini dirancang untuk memberikan gambaran umum mengenai jenis kegiatan, mitra penyelenggara, serta periode pelaksanaan guna memudahkan mahasiswa dalam memilih program yang sesuai dengan rencana studi mereka.

Gambar 20. Mockup Halaman Kegiatan MBKM

2.9.15 Halaman Aktivitas Kegiatan MBKM

Halaman ini digunakan oleh mahasiswa untuk mencatat seluruh daftar log aktivitas harian atau mingguan yang dilakukan selama mengikuti program MBKM. Antarmuka ini dirancang agar mahasiswa dapat mendokumentasikan setiap tugas yang dikerjakan secara kronologis guna memenuhi syarat pelaporan administrasi akademik.

Aksi	Sub Aktifitas	Durasi (Jam)	Capaian SKS	Status	Dokumen

Gambar 21. Mockup Halaman Aktivitas Kegiatan MBKM

2.9.16 Halaman Mata Kuliah yang Direkognisi Kegiatan MBKM

Halaman ini menyajikan daftar mata kuliah program studi yang setara dan dapat diakui nilainya melalui partisipasi mahasiswa dalam program MBKM. Antarmuka ini membantu memastikan bahwa seluruh capaian pembelajaran di luar kampus dapat terkonversi secara tepat ke dalam struktur kurikulum akademik mahasiswa.

Magang Badan Usaha Milik Negara (BUMN)

Aktivitas **Mata Kuliah** Nilai

Ubah Hapus

[+ Tambah Mata Kuliah ke Kegiatan](#)

Mata Kuliah yang Direkognisi

Kode	Nama Mata Kuliah	SKS	Kategori

Gambar 22. Mockup Halaman Mata Kuliah yang Direkognisi Kegiatan MBKM

2.9.17 Halaman Rubrik Penilaian Kegiatan MBKM

Halaman ini digunakan untuk menyusun kriteria dan bobot penilaian yang spesifik bagi setiap program MBKM. Antarmuka ini memungkinkan admin atau dosen untuk menentukan butir-butir penilaian secara mendetail, sehingga proses pemberian nilai akhir kepada mahasiswa menjadi lebih terukur, transparan, dan sesuai dengan standar kompetensi yang ditetapkan.

Magang Badan Usaha Milik Negara (BUMN)

Asistias

Mata Kuliah

Nilai

Ubah

Hapus

+ Tambah Rubrik Penilaian

Rubrik Penilaian

Aksi	Butir	Bobot	Nilai 1	Nilai 2	Nilai 3	Nilai 4	Nilai 5

Gambar 23. Mockup Halaman Rubrik Penilaian Kegiatan MBKM

2.9.18 Halaman Daftar Bukti Kegiatan

Halaman ini menampilkan berbagai jenis dokumen atau berkas pendukung yang wajib diunggah oleh mahasiswa sebagai bukti fisik pelaksanaan kegiatan MBKM. Antarmuka ini memfasilitasi pengguna untuk mengelola standarisasi dokumen—seperti sertifikat, laporan akhir, atau foto kegiatan, sehingga mempermudah proses verifikasi validitas aktivitas oleh pihak universitas.

Daftar Bukti Kegiatan

Bukti Kegiatan

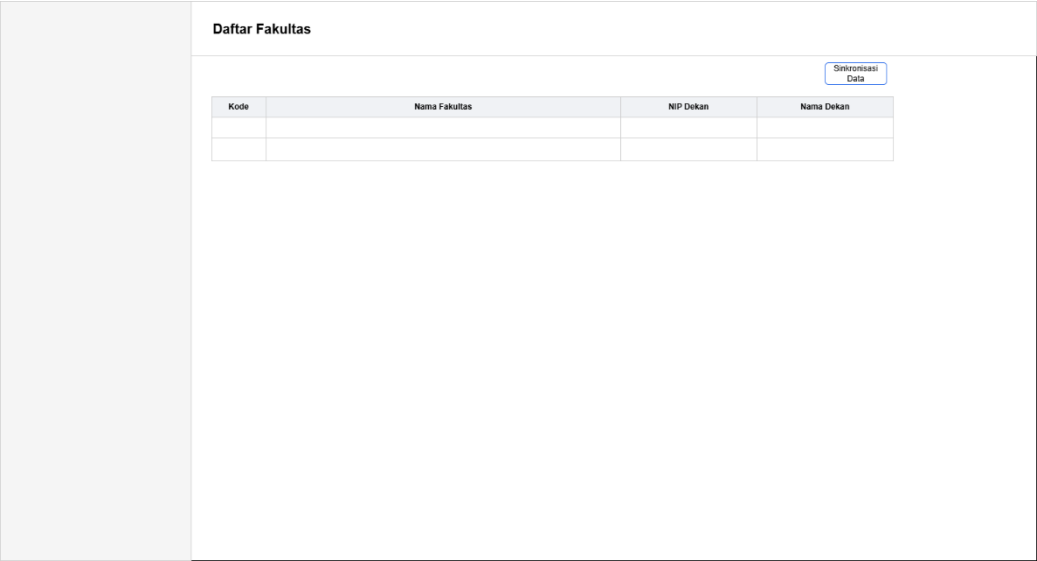
+ Tambah Data

Aksi		Data Bukti

Gambar 24. Mockup Halaman Daftar Bukti Kegiatan

2.9.19 Halaman Daftar Fakultas

Halaman ini memuat daftar seluruh fakultas yang ada di lingkungan universitas. Antarmuka ini berfungsi untuk memetakan data akademik secara makro, sehingga setiap program studi dan kegiatan mahasiswa dapat terorganisir dengan tepat di bawah naungan fakultas masing-masing.



Kode	Nama Fakultas	NIP Dekan	Nama Dekan

Gambar 25. Mockup Halaman Daftar Fakultas

2.9.20 Halaman Daftar Program Studi

Halaman ini menyajikan tabel informasi seluruh program studi yang terdaftar di universitas, mencakup kode dikti, nama prodi, serta identitas ketua program studi. Antarmuka ini memfasilitasi pengorganisasian data akademik yang lebih spesifik, sehingga sistem dapat mengklasifikasikan mahasiswa dan kegiatan MBKM berdasarkan rumpun keilmuan masing-masing secara akurat.

Daftar Program Studi

Sinkronisasi Data

Kode Dikti	Fakultas	Nama Prodi	NIP Ketua Prodi	Nama Ketua Prodi

Gambar 26. Mockup Halaman Daftar Program Studi

2.9.21 Halaman Usulan Kegiatan RKL

Halaman ini digunakan oleh mahasiswa untuk mengajukan rencana aktivitas spesifik dalam kerangka Rekognisi Kegiatan Lampau (RKL). Antarmuka ini menampilkan daftar usulan beserta informasi beban dosen pendamping, sehingga memudahkan koordinasi antara mahasiswa dan pihak program studi dalam memastikan kelayakan usulan kegiatan sebelum dilaksanakan.

User

SILASA

UTAMA

Beranda

KETUA PRODI

Dosen Verifikator

Dosen Pendamping Kegiatan

Kegiatan Mahasiswa

MASTER DATA

Data Mahasiswa

Data Dosen

DOSEN PENDAMPING KEGIATAN

Beban Dosen Pendamping

Aksi	Kode	Nama Mata Kuliah	SKS	Kategori
Ubah				

Previous | 1 | Next

Gambar 27. Mockup Halaman Usulan Kegiatan RKL

2.9.22 Halaman Data Program Studi

Halaman ini menampilkan detail spesifik dari suatu program studi, seperti nama prodi, NIP ketua prodi, dan jenjang pendidikan. Antarmuka ini berfungsi sebagai pusat informasi administratif prodi yang memungkinkan sinkronisasi data kurikulum dan identitas pengelola secara akurat di dalam sistem.

User	SILASA
UTAMA Beranda KETUA PRODI Dosen Verifikator Dosen Pendamping Kegiatan Kegiatan Mahasiswa MASTER DATA Data Mahasiswa Data Dosen	DATA PROGRAM STUDI FAKULTAS KEGURUAN DAN ILMU PENDIDIKAN (FKIP) Nama Prodi Pendidikan Matematika (S1) UBAH NIP Ketua Prodi 019301293129 Jenjang Prodi S-1

Gambar 28. Mockup Halaman Data Program Studi

2.9.23 Halaman Kegiatan Mahasiswa

Halaman ini menampilkan daftar usulan kegiatan yang diajukan oleh mahasiswa dan sedang menunggu proses validasi. Antarmuka ini dirancang khusus untuk memfasilitasi verifikator dalam meninjau rincian pengusul, jenis kegiatan, serta aktivitas yang dilakukan, guna menentukan kelayakan verifikasi laporan secara efisien dalam satu tampilan kartu informasi.

User	SILASA						
UTAMA Beranda Verifikasi Kegiatan KETUA PRODI Kegiatan Mahasiswa VERIFIKATOR Konversi Nilai	DAFTAR KEGIATAN MAHASISWA Kegiatan yang Belum Pernah Diverifikasi <table border="1"> <tr> <td>Pengusul</td> <td>19039123919 ADIRHASRANAS</td> </tr> <tr> <td>Kegiatan</td> <td>Lomba A</td> </tr> <tr> <td>Aktivitas</td> <td>Penggalan Ide/Cagasan</td> </tr> </table> Verifikasi	Pengusul	19039123919 ADIRHASRANAS	Kegiatan	Lomba A	Aktivitas	Penggalan Ide/Cagasan
Pengusul	19039123919 ADIRHASRANAS						
Kegiatan	Lomba A						
Aktivitas	Penggalan Ide/Cagasan						

Gambar 29. Mockup Halaman Kegiatan Mahasiswa

2.9.24 Halaman Daftar Kegiatan Mahasiswa

Halaman ini memuat informasi mengenai daftar mahasiswa beserta jurusan dan total kegiatan yang telah mereka ajukan dalam program MBKM atau RKL. Antarmuka ini dirancang untuk memudahkan verifikator atau pihak prodi dalam memantau keaktifan mahasiswa secara kolektif sebelum meninjau detail aktivitas satu per satu.

User	SILASA						
UTAMA Beranda Verifikasi Kegiatan KETUA PRODI Kegiatan Mahasiswa VERIFIKATOR Konversi Nilai	INFORMASI KEGIATAN MAHASISWA Daftar Mahasiswa <table border="1"> <thead> <tr> <th>Nama</th> <th>Jurusan</th> <th>Jumlah Kegiatan</th> </tr> </thead> <tbody> <tr> <td colspan="3"> </td> </tr> </tbody> </table> Previous 1 Next	Nama	Jurusan	Jumlah Kegiatan			
Nama	Jurusan	Jumlah Kegiatan					

Gambar 30. Mockup Halaman Daftar Kegiatan Mahasiswa

2.9.25 Halaman Daftar Kegiatan Mahasiswa

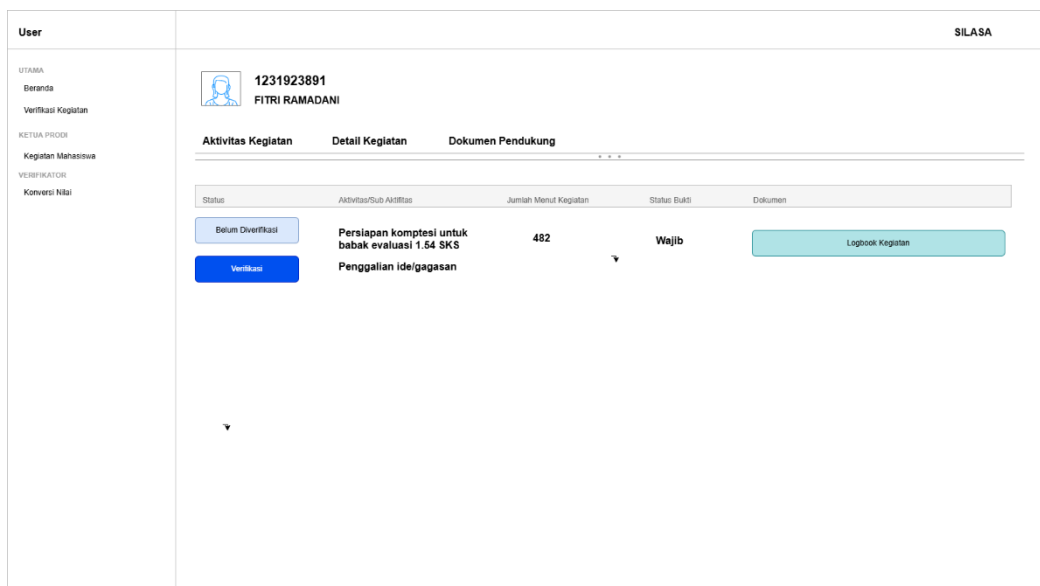
Halaman ini menampilkan ringkasan profil mahasiswa terpilih beserta daftar seluruh usulan kegiatan mereka, baik pada kategori PKL maupun MBKM. Antarmuka ini dilengkapi dengan fitur filter status (seperti Draft, Dikirim, Terlaksana, hingga Selesai) dan kartu informasi kegiatan yang memungkinkan verifikator untuk meninjau status keanggotaan serta memberikan verifikasi pada setiap poin aktivitas secara terstruktur.

The mockup shows a web interface for a student activity registration system. On the left is a sidebar menu with options: UTAMA, Beranda, Verifikasi Kegiatan, KETUA PRODI, Kegiatan Mahasiswa, VERIFIKATOR, and Konversi Nilai. The main content area is titled 'User' and 'SILASA'. It displays the profile of a student named FITRI RAMADANI with ID 1231923891. Below the profile, there are tabs for 'PKL' and 'MBKM'. A filter bar shows various status options: 'Semua' (highlighted), 'Draft', 'Dikirim', 'Terlaksana', 'Direvisi', 'Ditolak', and 'Selesai'. Three activity cards are shown, each with a 'Basis' dropdown, 'Kegiatan' text, 'Status Keanggotaan' (Penggalan Ide/Capaian), and a 'Verifikasi' button.

Gambar 31. Mockup Halaman Daftar Kegiatan Mahasiswa

2.9.26 Halaman Aktivitas Kegiatan RKL

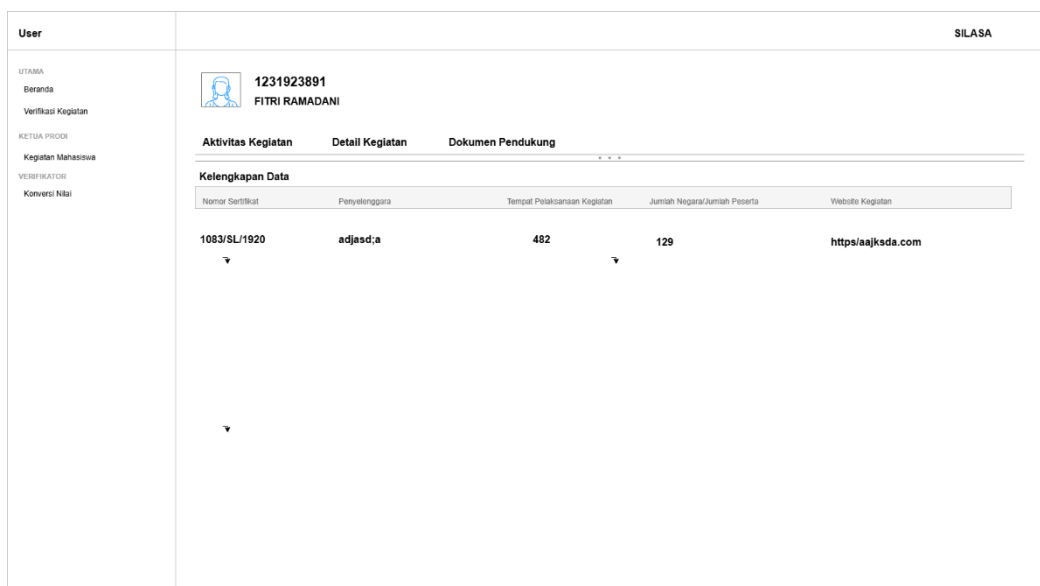
Halaman ini merupakan antarmuka detail yang menampilkan ringkasan aktivitas dan dokumen pendukung mahasiswa dalam program RKL. Antarmuka ini dirancang untuk mempermudah verifikator dalam meninjau progres secara mendalam, mencakup status verifikasi, rincian durasi kegiatan dalam menit, hingga akses langsung ke berkas *logbook* kegiatan mahasiswa guna memastikan validitas setiap klaim aktivitas yang diajukan.



Gambar 32. Mockup Halaman Aktivitas Kegiatan RKL

2.9.27 Halaman Detail Kegiatan

Halaman ini menyajikan informasi komprehensif mengenai kelengkapan data suatu aktivitas yang diikuti mahasiswa, seperti nomor sertifikat, nama penyelenggara, lokasi pelaksanaan, hingga jumlah peserta atau jangkauan negara terlibat. Antarmuka ini dirancang untuk memvalidasi kualitas dan skala kegiatan melalui bukti otentik termasuk tautan situs web kegiatan sehingga pihak verifikator memiliki basis data yang kuat dalam proses penilaian atau rekognisi.



Gambar 33. Mockup Halaman Detail Kegiatan

2.9.28 Halaman Dosen Verifikator

Halaman ini menampilkan daftar kegiatan mahasiswa yang memerlukan penugasan atau tindakan dari dosen verifikator. Antarmuka ini menyajikan informasi dalam bentuk kartu yang memuat detail basis data mahasiswa, nama kegiatan, serta status keanggotaan. Melalui halaman ini, pengguna (seperti Ketua Prodi) dapat memantau kegiatan mana saja yang belum memiliki dosen verifikator dan melakukan aksi verifikasi lebih lanjut untuk memastikan setiap aktivitas mahasiswa tertangani dengan baik.

The mockup shows a web interface for a 'Dosen Verifikator' (Verifier Lecturer) page. On the left is a sidebar menu with the following items: UTAMA, Beranda, KETUA PRODI, Dosen Verifikator (highlighted), Dosen Pendamping Kegiatan, Kegiatan Mahasiswa, Data Program Studi, MASTER DATA, Data Mahasiswa, and Data Dosen. The main content area is titled 'VERIFIKATOR' and contains a section 'BELUM MEMILIKI DOSEN VERIFIKATOR'. This section displays three cards, each representing a student. Each card contains the following information: Basis (19389123919 ADUHASUHAS), Kegiatan (Lomba A), and Status Keanggotaan (Penggalan Ide/Gagasan). Below this information is a blue button labeled 'Verifikasi'.

Gambar 34. Mockup Halaman Dosen Verifikator

2.9.29 Halaman Daftar Mahasiswa untuk Dosen Verifikator

Halaman ini digunakan oleh pihak pengelola (seperti Ketua Prodi) untuk menugaskan mahasiswa kepada dosen verifikator tertentu. Antarmuka ini menampilkan identitas dosen yang bertugas serta tabel kelengkapan data mahasiswa (nama, NIM, dan *email*), yang memungkinkan proses distribusi beban kerja verifikasi aktivitas MBKM atau RKL dilakukan secara terorganisir dan tepat sasaran.

User	SILASA						
UTAMA Beranda Verifikasi Kegiatan KETUA PRODI Kegiatan Mahasiswa VERIFIKATOR Konversi Nilai	 1231923891 SIRWANTI S.PD, M.PD + Tambah Mahasiswa Anda dapat menambahkan mahasiswa pada verifikator Kelengkapan Data <table border="1"> <thead> <tr> <th>Nama</th> <th>NIM</th> <th>Email</th> </tr> </thead> <tbody> <tr> <td colspan="3"> </td> </tr> </tbody> </table>	Nama	NIM	Email			
Nama	NIM	Email					

Gambar 35. Mockup Halaman Daftar Mahasiswa untuk Dosen Verifikator

2.9.30 Halaman Dosen Pendamping Kegiatan

Halaman ini merupakan pusat pengelolaan bagi pihak universitas (seperti Ketua Prodi) untuk memantau beban kerja dan penugasan dosen pendamping dalam kegiatan RKL maupun MBKM. Antarmuka ini menyajikan data dalam bentuk kartu informatif yang menampilkan nama dosen, NIP, serta jumlah mahasiswa yang sedang didampingi, sehingga distribusi pendampingan dapat terpantau secara adil dan transparan bagi seluruh civitas akademika.

User	SILASA
UTAMA Beranda KETUA PRODI Dosen Verifikator Dosen Pendamping Kegiatan Kegiatan Mahasiswa Data Program Studi MASTER DATA Data Mahasiswa Data Dosen	DOSEN PENDAMPING KEGIATAN Beban Dosen Pendamping Usulan Kegiatan RKL Usulan Kegiatan MBKM Tanpa Pendamping Untuk RKL Untuk MBKM Mahasiswa Terdamping 1 Mahasiswa 0 Mahasiswa NIP 121391839813817 Nama KHALID RIJALUDIN S.Pd., M.Pd Mahasiswa Terdamping 1 Mahasiswa 0 Mahasiswa NIP 198381923 Nama Muh Ridwan, S.pd., M.pd Mahasiswa Terdamping 1 Mahasiswa 0 Mahasiswa NIP 138912132 Nama Fahrudi Muhtar S.Pd., M.Pd Mahasiswa Terdamping 1 Mahasiswa 0 Mahasiswa NIP 1280190923 Nama Ansh Hajar S.Pd., M.Pd

Gambar 36. Mockup Halaman Dosen Pendamping Kegiatan

2.9.31 Halaman Nilai Aktivitas MBKM

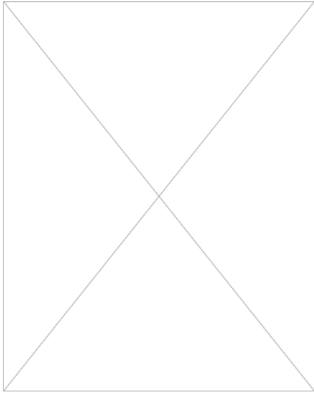
Halaman ini merupakan antarmuka bagi verifikator atau pembimbing akademik untuk memberikan penilaian pada setiap butir aktivitas yang telah diajukan mahasiswa. Antarmuka ini menampilkan deskripsi mendetail dari aktivitas yang dilakukan seperti penyusunan proposal serta menyediakan kolom *input* khusus untuk memberikan bobot dan nilai secara spesifik, sehingga proses penilaian akhir menjadi lebih terstruktur dan transparan berdasarkan bukti kinerja yang ada.

User	SILASA
UTAMA Beranda Verifikasi Kegiatan PEMBIMBING AKADEMIK Kegiatan Mahasiswa VERIFIKATOR Konversi Nilai	 1231923891 FITRI RAMADANI Aktivitas Kegiatan Detail Kegiatan Dokumen Pendukung 1 Penyusunan Proposal Kegiatan Penyusunan Proposal Kegiatan Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum. Bobot Nilai

Gambar 37. Mockup Halaman Nilai Aktivitas MBKM

2.9.32 Halaman Pilih Basis Kegiatan

Halaman ini merupakan antarmuka bagi mahasiswa untuk menentukan dasar atau kategori dari kegiatan yang akan diajukan. Halaman ini menyajikan deskripsi mendetail mengenai ketentuan setiap basis kegiatan serta menyediakan opsi untuk mengunduh rubrik penilaian RKL, sehingga mahasiswa memiliki panduan yang jelas dalam menyusun usulan aktivitas agar sesuai dengan standar rekognisi yang ditetapkan sistem.

User	SILASA	
UTAMA Beranda Kegiatan Mahasiswa Riwayat Kegiatan Info Konvensi Kegiatan Sertifikat	Penyusunan Proposal Kegiatan Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum. Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum. Unduk Rubrik Penilaian RKL	

Gambar 38. Mockup Halaman Pilih Basis Kegiatan

2.9.33 Halaman Aktivitas Kegiatan

Halaman ini berfungsi sebagai antarmuka bagi mahasiswa untuk mengelola rincian aktivitas dari kegiatan yang sedang diikuti (seperti perlombaan atau magang). Mahasiswa dapat menambah sub-aktivitas baru, menentukan durasi waktu dalam menit, serta mengunggah dokumen *logbook* sebagai bukti fisik. Halaman ini juga dilengkapi dengan tombol "Kirim" untuk mengajukan seluruh rangkaian aktivitas tersebut agar dapat diverifikasi oleh dosen pendamping.

User	SILASA															
UTAMA Beranda Kegiatan Mahasiswa Riwayat Kegiatan Info Konvensi Kegiatan Sertifikat	Lomba Inovasi Teknologi Pendidikan Klik Tombol "Kirim" agar diverifikasi dosen pendidikan Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum. Aktifitas Kegiatan + Tambah Aktifitas <table border="1"> <thead> <tr> <th>Status</th> <th>Aktivitas/Sub Aktivitas</th> <th>Jumlah Menit Kegiatan</th> <th>Status Bukti</th> <th>Dokumen</th> </tr> </thead> <tbody> <tr> <td>Draf</td> <td>Persiapan kompetisi untuk babak evaluasi 1.54 SKS</td> <td>482</td> <td>Wajib</td> <td rowspan="2"> Unggah Logbook Kegiatan </td> </tr> <tr> <td>Kirim</td> <td>Penggalan ide/gagasan</td> <td></td> <td></td> </tr> </tbody> </table> Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.		Status	Aktivitas/Sub Aktivitas	Jumlah Menit Kegiatan	Status Bukti	Dokumen	Draf	Persiapan kompetisi untuk babak evaluasi 1.54 SKS	482	Wajib	Unggah Logbook Kegiatan	Kirim	Penggalan ide/gagasan		
Status	Aktivitas/Sub Aktivitas	Jumlah Menit Kegiatan	Status Bukti	Dokumen												
Draf	Persiapan kompetisi untuk babak evaluasi 1.54 SKS	482	Wajib	Unggah Logbook Kegiatan												
Kirim	Penggalan ide/gagasan															

Gambar 39. Mockup Halaman Aktivitas Kegiatan