p-ISSN : 2406-7733
e-ISSN : 2597-9922

IMPLEMENTASI *AGILE SCRUM* DAN PEMODELAN UML DALAM PENGEMBANGAN APLIKASI LAYANAN PENGADUAN MAHASISWA MENGGUNAKAN TRELLO

Cahyani Budihartanti[1], Muhammad Jidan[2], Aulia Adrina Azzahra[3], Juan Cesare Zidane[4], Andini Putri Wulandari[5].
Program Studi Sistem Informasi, Fakultas Teknologi Informasi, Universitas Nusa Mandiri[1]
Program Studi Sistem Informasi, Fakultas Teknik dan Informatika, Universitas Bina Sarana Informatika[2,3,4,5]
E-mail: *cahyani.cbh@nusamandiri.ac.id[1], muhammadjidan952@gmail.com[2], auliaadrina19@gmail.com[3], zidan.yrk105@gmail.com[4], diniputrii08@gmail.com[5].

Abstrak - Pengelolaan fasilitas dan pelayanan akademik di lingkungan kampus masih menghadapi berbagai kendala akibat sistem pelaporan keluhan yang bersifat konvensional dan belum terintegrasi. Kondisi tersebut menyebabkan mahasiswa mengalami kesulitan dalam memantau status laporan, lambatnya respons penanganan, serta kurangnya transparansi layanan. Penelitian ini bertujuan untuk merancang dan mengembangkan aplikasi layanan pengaduan mahasiswa berbasis *mobile* yang terintegrasi guna meningkatkan efisiensi dan transparansi pelayanan. Metode yang digunakan dalam pengembangan sistem adalah *Agile Scrum* dengan dukungan Trello sebagai alat manajemen proyek serta *Unified Modeling Language* (UML) sebagai pemodelan arsitektur sistem. Pendekatan Scrum yang bersifat iteratif memungkinkan proses pengembangan berjalan lebih adaptif melalui beberapa tahapan sprint. Hasil penelitian menunjukkan bahwa aplikasi pengaduan berhasil diimplementasikan dengan lima fitur utama, yaitu distribusi laporan otomatis kepada petugas aktif, fitur pelaporan anonim, informasi tingkat prioritas laporan, sistem penilaian kinerja menggunakan rating bintang, serta penguncian fitur chat secara otomatis pada laporan yang ditolak. Implementasi sistem informasi terpadu ini memberikan dampak berupa percepatan proses penanganan keluhan dan peningkatan kualitas pelayanan fasilitas di lingkungan kampus.

Kata Kunci: *Agile Scrum*, Aplikasi *Mobile*, Layanan Pengaduan, Trello, UML.

I. PENDAHULUAN

Pesatnya perkembangan teknologi informasi mendorong berbagai sektor, termasuk perguruan tinggi, untuk beralih ke sistem digital. Kampus tidak hanya menjadi pusat kegiatan akademik, tetapi juga ruang interaksi antara mahasiswa dan petugas yang rentan terhadap berbagai permasalahan, mulai dari kerusakan fasilitas, kebersihan, hingga kejadian yang membutuhkan penanganan segera (Sianipar et al., 2025).

Saat ini, pelaporan kerusakan fasilitas maupun pengaduan mahasiswa umumnya masih dilakukan melalui saluran komunikasi yang tidak terintegrasi, seperti media sosial atau aplikasi pesan instan. Akibatnya, laporan sering tidak tertanggapi, terlewat, atau tidak mendapat respons yang jelas. Kondisi ini menyulitkan mahasiswa dalam memantau status laporan mereka, sekaligus mengurangi efektivitas dan kepuasan layanan kampus. Teknologi *mobile* dinilai mampu menjadi solusi melalui sistem pengaduan daring yang memungkinkan pelaporan, pelacakan, dan penyelesaian aduan dilakukan kapan saja dan di mana saja (Saragih et al., 2025).

Selain itu, sistem yang ada belum mampu mengelompokkan laporan berdasarkan tingkat urgensi, belum menjamin anonimitas pelapor, masih mengandalkan distribusi laporan secara manual, serta belum menyediakan fitur penilaian kinerja petugas. Untuk mengatasi permasalahan tersebut, diperlukan sistem pengaduan berbasis mobile yang efektif, efisien, dan terintegrasi.

Pengembangan sistem ini menggunakan metode *Agile* dengan pendekatan *Scrum* yang bersifat iteratif dan adaptif (Ramadhani et al., 2026). Agile merupakan metode pengembangan perangkat lunak yang menekankan proses berulang dalam siklus kecil sehingga fleksibel terhadap perubahan kebutuhan (Ardytia et al., 2021). *Scrum* sebagai kerangka kerja Agile mewujudkan prinsip-prinsip tersebut melalui serangkaian sprint, yang masing-masing mencakup *sprint planning, daily scrum, sprint review,* dan *sprint retrospective* dengan durasi maksimal 30 hari. Sebagai alat manajemen proyek, digunakan Trello yang mendukung kolaborasi tim sekaligus pencatatan *product backlog* dan *sprint backlog*, serta dapat diakses melalui web maupun aplikasi *mobile* (Ardytia et al., 2021).

Penelitian ini bertujuan membangun aplikasi layanan pengaduan mahasiswa berbasis *mobile* menggunakan pendekatan *Agile Scrum* dan pemodelan UML, dengan mewujudkan lima fungsionalitas utama yaitu, distribusi laporan otomatis (*auto-assign*), opsi laporan anonim, penjelasan tingkat prioritas, penilaian *rating* bintang, dan penguncian *chat* pada laporan yang ditolak petugas.

p-ISSN : 2406-7733
e-ISSN : 2597-9922

II. TINJAUAN PUSTAKA

2.1 Sistem Informasi

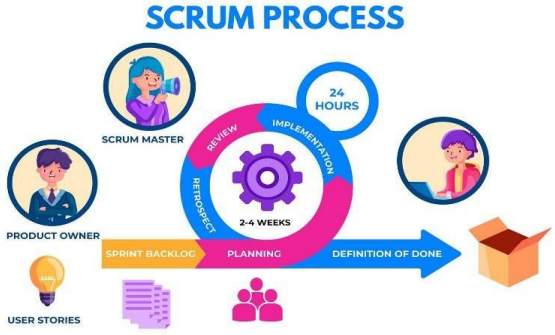
Sistem informasi (SI) memainkan peran penting dalam kehidupan organisasi dan bisnis modern. SI didefinisikan sebagai kumpulan elemen yang saling berinteraksi dalam proses pengolahan, penyimpanan, dan pendistribusian informasi (Approaches et al., 2025), yang berfungsi sebagai landasan dalam pengelolaan informasi secara efektif untuk mendukung pengambilan keputusan, pengendalian, dan operasional organisasi. Penerapannya membantu organisasi meningkatkan efisiensi, akurasi, dan daya saing, serta membuat berbagai aktivitas berjalan lebih terstruktur. Dalam implementasinya, SI memiliki siklus hidup yang mencakup tahap perencanaan, pengembangan, implementasi, operasi, hingga pemeliharaan sistem (Widarti et al., 2024).

2.2 Sistem Layanan Pengaduan Mahasiswa

Pengaduan mahasiswa merupakan mekanisme bagi mahasiswa untuk menyampaikan keluhan terkait pelayanan akademik kampus yang dinilai belum memenuhi standar yang diharapkan (Sunarti, Ridwang, dan Hayat, 2024). Bentuknya dapat berupa aspirasi, komentar, masukan, maupun ulasan sesuai kebutuhan organisasi (Elda Sulistiawati, Heri Mulyono, 2023). Keberadaan mekanisme ini penting bagi universitas sebagai upaya perbaikan dan peningkatan kualitas layanan secara berkelanjutan.

2.3 Metode *Agile*

Agile merupakan salah satu *System Development Life Cycle* (SDLC) yang banyak digunakan dalam pengembangan sistem informasi. Salah satu framework-nya adalah *Scrum*, yakni kerangka kerja yang bertujuan membuat proses pengembangan menjadi transparan, dapat dipantau, dan adaptif terhadap perubahan. *Scrum* mampu menangani permasalahan kompleks yang terus berkembang sambil tetap menghasilkan produk bernilai tinggi secara kreatif dan produktif. Kerangka kerjanya mencakup *Project Management* dan Team Management dengan pembagian peran antara *Product Owner, Developer Team,* dan *Scrum Master* (Abiyyu & Kurniawan, 2023).



Gambar 1. *Scrum Process* (nimblework.com)

2.4 *Scrum*

Scrum adalah cara-cara yang dilakukan untuk memecahkan suatu masalah. Scrum sendiri menggunakan pendekatan dari metode lain yaitu *Agile. Agile* mengacu pada seperangkat metode dan praktik berdasarkan nilai dan prinsip yang diungkapkan dalam *Agile Manifesto*. Hal itu mencakup hal-hal seperti kolaborasi, pengorganisasian yang dilakukan sendiri oleh member tim, dan tim yang lintas fungsi (Magdalena et al., 2023).

2.5 *Unified Modeling Language* (UML)

UML digunakan untuk mendeskripsikan desain awal sistem yang akan dikembangkan. Sebagai alat pemodelan visual yang efektif dalam pengembangan sistem berbasis objek, UML memungkinkan pengembang merepresentasikan konsep sistem dalam format yang standar dan mudah dipahami (Pangestu et al., 2024). Dari berbagai jenis pemodelan yang tersedia, hanya beberapa yang relevan akan dibahas dalam penelitian ini.

2.6 Flutter

Flutter adalah *framework* pengembangan aplikasi mobile *multiplatform* yang memungkinkan pengembang membangun aplikasi hanya dengan satu basis kode (*codebase*) yang dapat dijalankan pada platform Android maupun iOS (Erzed et al., 2022)

2.7 MySQL

MySQL adalah DBMS yang *open source* dengan dua bentuk lisensi, yaitu *Free Software* (perangkat lunak bebas) dan *Shareware* (perangkat lunak bermilik yang penggunaannya terbatas). Jadi MySQL adalah *database server* yang gratis dengan lisensi GNU *General Public License* (GPL) sehingga dapat Anda pakai untuk keperluan pribadi atau komersial tanpa harus membayar lisensi yang ada. Seperti yang sudah disebutkan sebelumnya, MySQL masuk ke dalam jenis RDBMS (*Relational database Management Sistem*). Maka dari itu, istilah semacam baris, kolom, tabel, dipakai pada MySQL. Contohnya di dalam MySQL sebuah *database* terdapat satu atau beberapa tabel (Putri, 2020).

2.8 PHP

PHP (*Hypertext Preprocessor*) adalah bahasa pemrograman *server-side* bersifat *open source* yang disisipkan dalam dokumen HTML untuk pengembangan web. PHP mendukung berbagai web server seperti Apache, IIS, dan Nginx, serta kompatibel dengan berbagai sistem operasi termasuk Linux, Unix, Macintosh, dan Windows. Fleksibilitas ini menjadikan PHP mudah dipelajari dan didukung oleh komunitas pengembang yang luas (Sitorus et al., 2023).

p-ISSN : 2406-7733
e-ISSN : 2597-9922

2.9 *Blackbox Testing*

Blackbox testing adalah proses menguji fungsi perangkat lunak (*software*) atau aplikasi dari sudut pandang pengguna, tanpa mengetahui struktur internal atau desain struktur tersebut. Sederhananya, *blackbox testing* hanya melakukan penilaian dari apakah sistem bisa memberikan *output* atau hasil sesuai dengan input (Hozairi, Buhari, Syariful Alim, 2024).

III. METODE PENELITIAN

Pada pengembangan aplikasi layanan pengaduan mahasiswa ini menggunakan metode agile scrum serta menggunakan Trello sebagai alat kolaborasi tim, Didalam *scrum* terdiri dari beberapa tahapan seperti:

1. *Requirement Analysis*

Tahap ini dilakukan dengan menganalisis kebutuhan sistem melalui identifikasi masalah dan pengumpulan informasi terkait layanan pengaduan di lingkungan kampus. Pengamatan terhadap proses pelaporan yang masih menggunakan media komunikasi tidak terintegrasi menunjukkan bahwa laporan kerap terlewat, tidak terdokumentasi dengan baik, dan sulit dipantau perkembangannya.

Dari analisis tersebut, diperoleh dua kategori kebutuhan. Kebutuhan fungsional mencakup pembuatan laporan pengaduan, pengelolaan status laporan, distribusi laporan otomatis berdasarkan kategori, fitur laporan anonim, fitur *chat* pengaduan, dan sistem penilaian kinerja petugas. Sementara kebutuhan non-fungsional meliputi kemudahan penggunaan, keamanan data, performa sistem, dan aksesibilitas melalui perangkat *mobile*.

2. *Product Backlog*

Penentuan product backlog adalah daftar fitur dan kebutuhan pengembangan aplikasi yang disusun berdasarkan tingkat prioritas, berfungsi sebagai landasan utama dalam perencanaan sprint. Daftar ini bersifat fleksibel sehingga dapat diperbarui sesuai *feedback* pengguna maupun perubahan kebutuhan selama pengembangan. Dalam metode Agile Scrum, *backlog* berperan penting dalam membantu tim menentukan prioritas pengembangan secara adaptif.

3. *Sprint planning*

Sprint planning merupakan langkah pembuka setiap siklus sprint, di mana tim Scrum merancang rencana kerja berdasarkan item prioritas dalam *product backlog*. Pada tahap ini, tim menetapkan *user stories* yang akan diimplementasikan, yakni gambaran singkat fungsionalitas yang dibutuhkan pengguna berdasarkan pemahaman terhadap kebutuhan pengguna akhir.

4. *Sprint* (Pelaksanaan iterasi)

Fase ini merupakan tahap eksekusi yang mencakup pengembangan *front-end*, *back-end*, dan basis data sesuai *sprint backlog*, dilakukan secara kolaboratif, iteratif, dan inkremental dengan menerapkan *continuous integration*, pengujian, serta dokumentasi untuk menjaga kualitas perangkat lunak. Selama pelaksanaan, tim mengadakan *daily scrum* untuk memantau progres dan mengatasi hambatan yang muncul.

5. *Sprint Review*

Setelah satu sprint berakhir, tim melakukan demonstrasi dan peninjauan modul aplikasi yang telah dibangun kepada *Product Owner* dan *Scrum Master*. Fitur-fitur yang dikembangkan diuji untuk memastikan sistem berjalan dengan baik, bebas dari *bug*, dan memenuhi *acceptance criteria* yang ditetapkan di awal perencanaan. Tahap ini juga bertujuan mengevaluasi kemajuan yang dicapai serta mendiskusikan masukan untuk perbaikan pada sprint berikutnya.

6. *Sprint Retrospective*

Sprint Retrospective merupakan evaluasi internal tim di akhir setiap siklus iterasi, yang diikuti seluruh anggota tim untuk meninjau proses kerja, mengidentifikasi hal yang berjalan baik, tantangan yang dihadapi, serta merumuskan perbaikan untuk sprint berikutnya. Tujuannya adalah menilai efektivitas proses pengembangan sekaligus menjadi bahan pembelajaran untuk iterasi mendatang.

IV. HASIL DAN PEMBAHASAN

Pengembangan aplikasi ini menggunakan metode Agile Scrum yang didukung Trello sebagai alat kolaborasi tim. Trello berfungsi menjaga keterbukaan informasi antar anggota tim sekaligus sebagai alat inspeksi kemajuan pengembangan berdasarkan tujuan yang telah disepakati. Pemeriksaan dilakukan secara berkala untuk mencegah penyimpangan, sekaligus memungkinkan tim beradaptasi secara cepat apabila terjadi perubahan kebutuhan. Studi kasus penelitian ini berfokus pada manajemen proyek perancangan dan pembuatan aplikasi layanan pengaduan mahasiswa berbasis mobile dengan tahapan pengembangan sebagai berikut.

4.1 *Requirement Analysis*

Berdasarkan pengumpulan data yang telah dilakukan, tahapan selanjutnya difokuskan pada analisis terhadap prosedur pelaporan keluhan dan pengelolaan fasilitas yang selama ini berjalan di lingkungan kampus. Penggalian informasi ini bertujuan untuk mendapatkan data serta fakta pendukung yang relevan, yang kemudian ditindaklanjuti melalui kegiatan diskusi dan analisis

p-ISSN : 2406-7733
e-ISSN : 2597-9922

bersama pihak-pihak terkait. Hasil dari tahapan tersebut digunakan untuk merumuskan spesifikasi dan kebutuhan sistem aplikasi layanan pengaduan mahasiswa berbasis *mobile* yang akan dikembangkan.

Unit fundamental dari metode perancangan ini adalah pembentukan sebuah *Scrum Team*. Tim ini terdiri dari satu *Scrum Master*, satu *Product Owner*, serta *Developer Team* (tim pengembang) yang disusun berdasarkan spesifikasi keterampilan teknis yang dibutuhkan selama proses pengembangan. Berikut merupakan daftar pembagian peran di dalam *Scrum Team* pada proyek ini:

Tabel 1. *Scrum Team*

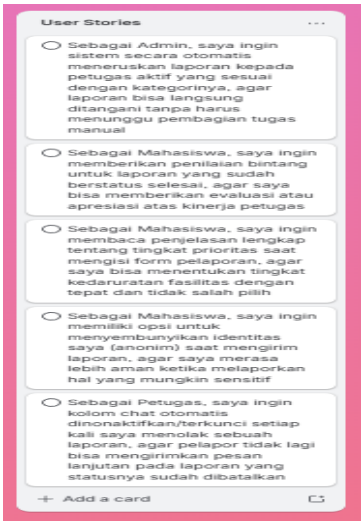
Nama Anggota	Peran
Juan Cesare Zidane	*Product Owner*
Aulia Adrina Azzahra	*Scrum Master*
Muhammad Jidan	*Developer*
Andini Putri Wulandari	*Developer*

Selanjutnya disusun *user stories* berdasarkan kebutuhan yang telah diidentifikasi bersama *Product Owner*, dengan hasil sebagaimana ditampilkan pada tabel berikut.

Tabel 2. *User Stories*

User Role	Deskripsi	Hasil Akhir
Admin	Sistem dapat mendistribusikan laporan secara otomatis kepada petugas aktif sesuai kategori laporan.	Fitur distribusi laporan otomatis berhasil diimplementasikan berdasarkan kategori dan status aktif petugas.
Mahasiswa	Mahasiswa dapat memberikan penilaian pelayanan petugas setelah laporan selesai diproses.	Berhasil menyediakan fitur rating pelayanan menggunakan bintang.
Mahasiswa	Mahasiswa dapat melihat penjelasan tingkat prioritas saat membuat laporan.	Berhasil menampilkan deskripsi prioritas laporan pada form pengaduan.
Mahasiswa	Mahasiswa dapat mengirim laporan secara anonim untuk menjaga privasi.	Berhasil menyediakan fitur pengiriman laporan anonim.
Petugas	Chat pengaduan otomatis terkunci ketika laporan ditolak.	Berhasil mengunci fitur chat setelah laporan ditolak oleh petugas.

Seluruh *user stories* didokumentasikan ke dalam papan kerja Trello untuk memudahkan kolaborasi tim secara digital, sebagaimana terlihat pada gambar 2.



Gambar 2. Implementasi *User Stories* pada Trello

4.2 *Product Backlog*

Dalam pengembangan aplikasi layanan pengaduan mahasiswa, terdapat beberapa requirement sistem yang telah diidentifikasi. Selanjutnya, requirement tersebut disusun ke dalam *product backlog* seperti pada tabel 3.

Tabel 3. *Product Backlog*

Product Backlog	*Sprint* Ke
Sebagai Admin, saya ingin sistem secara otomatis meneruskan laporan kepada petugas aktif yang sesuai dengan kategorinya, agar laporan bisa langsung ditangani tanpa harus menunggu pembagian tugas manual.	1
Sebagai Mahasiswa, saya ingin memberikan penilaian bintang untuk laporan yang sudah berstatus selesai, agar saya bisa memberikan evaluasi atau apresiasi atas kinerja petugas.	1
Sebagai Mahasiswa, saya ingin membaca penjelasan lengkap tentang tingkat prioritas saat mengisi form pelaporan, agar saya bisa menentukan tingkat kedaruratan fasilitas dengan tepat dan tidak salah pilih.	2
Sebagai Mahasiswa, saya ingin memiliki opsi untuk menyembunyikan identitas saya (anonim) saat mengirim laporan, agar saya merasa lebih aman ketika melaporkan hal yang mungkin sensitif.	2
Sebagai Petugas, saya ingin kolom chat otomatis dinonaktifkan/terkunci setiap kali saya menolak sebuah laporan, agar pelapor tidak lagi bisa mengirimkan pesan lanjutan pada laporan yang statusnya sudah dibatalkan.	2

4.3 *Sprint Planning*

Penyusunan *sprint planning* dilakukan dengan mempertimbangkan kondisi dan kebutuhan *sprint*

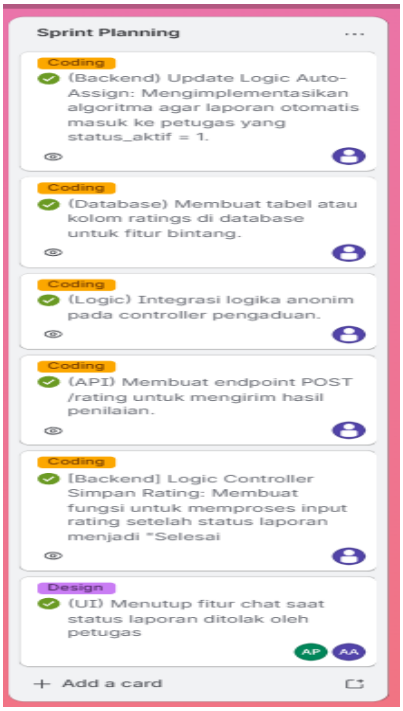
p-ISSN : 2406-7733
e-ISSN : 2597-9922

yang sedang berjalan. Pada proses pengembangan ini, proyek masih berada pada tahap sprint pertama seperti tabel 4.

Tabel 4. *Sprint Planning*

Sprint Planning
Rancangan UI saat menutup fitur chat dalam keadaan status laporan ditolak oleh petugas
Membuat tabel atau kolom ratings di database (untuk fitur bintang).
Membuat endpoint POST /rating untuk mengirim hasil penilaian.
Integrasi logika anonim pada controller pengaduan.
Mengimplementasikan algoritma agar laporan otomatis masuk ke petugas yang aktif.
Membuat fungsi untuk memproses input rating setelah status laporan menjadi "Selesai".

Seluruh tugas teknis dikelola secara digital menggunakan Trello, di mana setiap tugas direpresentasikan sebagai kartu (*card*) yang diberi label kategori pekerjaan seperti *Coding* atau *Design* dan ditugaskan kepada anggota tim yang bertanggung jawab. Dokumentasi rencana sprint pada papan kerja Trello dapat dilihat pada gambar berikut.



Gambar 3. *Sprint Planning* pada Trello

4.4 *Sprint* (Pelaksanaan Iterasi)

Pada fase *sprint*, tim mulai mengerjakan tugas-tugas teknis secara nyata. Setiap tugas yang direpresentasikan sebagai kartu (*card*) di Trello dipindahkan dari kolom *Sprint Planning* ke kolom *In Progress* sebagai penanda bahwa pengerjaan sedang berlangsung secara aktif, sebagaimana ditunjukkan pada Gambar 4.

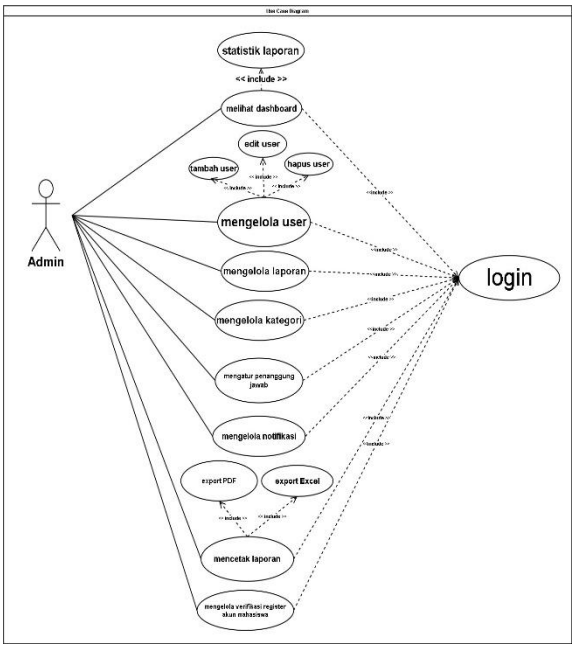


Gambar 4. Sprint pada Trello

4.4.1 Implementasi Pemodelan Arsitektur Sistem (UML & ERD)

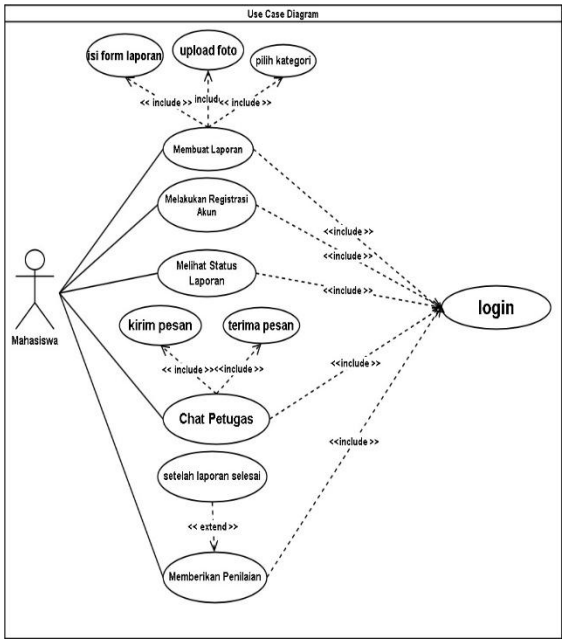
a. *Use Case Diagram*

Use Case Diagram memodelkan fungsionalitas utama sistem dengan pengguna. Sistem ini melibatkan tiga aktor utama, yaitu Mahasiswa, Petugas, dan Admin, yang masing-masing memiliki batasan hak akses spesifik setelah melakukan proses autentikasi (*login*).

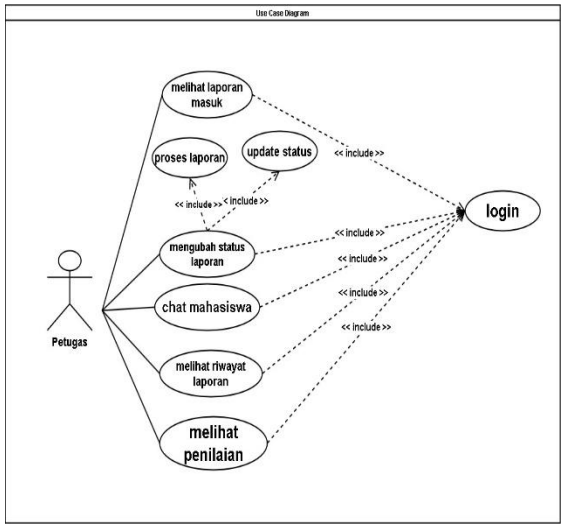


Gambar 5. *Use Case* Admin

p-ISSN : 2406-7733
e-ISSN : 2597-9922



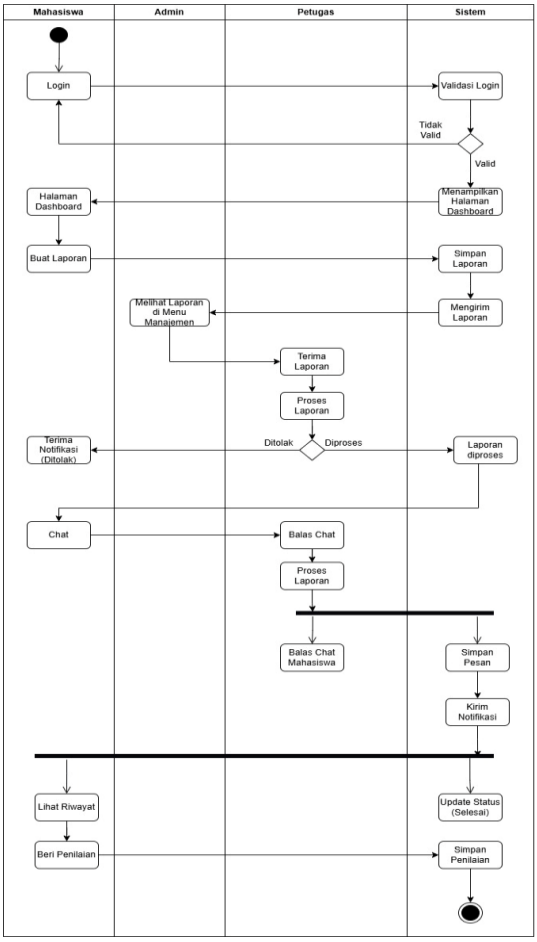
Gambar 6. *Use Case* Mahasiswa



Gambar 7. *Use Case* Petugas

b. *Activity Diagram*

Activity Diagram menggambarkan alur kerja dinamis aplikasi dari sudut pandang pengguna. Setelah proses validasi *login* berhasil, sistem akan mendeteksi hak akses (*role*) pengguna dan secara otomatis mengarahkannya ke *dashboard* masing-masing untuk memilih berbagai menu layanan yang tersedia.

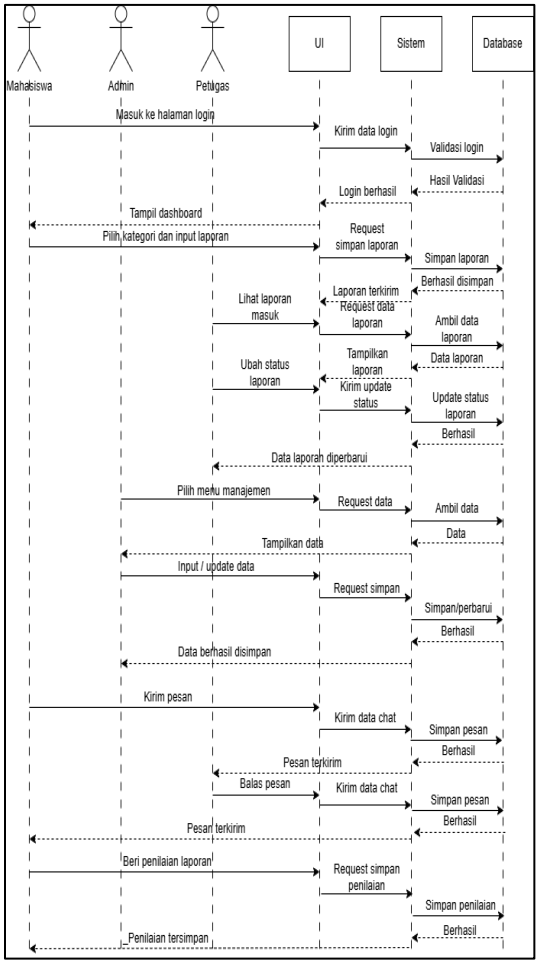


Gambar 8. *Activity Diagram*

c. *Sequence Diagram*

Sequence Diagram menjelaskan urutan interaksi antar objek di dalam sistem secara kronologis. Diagram ini memperlihatkan bagaimana proses pertukaran data terjadi antara lapisan antarmuka (*UI*), pengontrol sistem (*Controller*), dan basis data (*Database*) saat fitur utama seperti pembuatan laporan dan penilaian dijalankan.

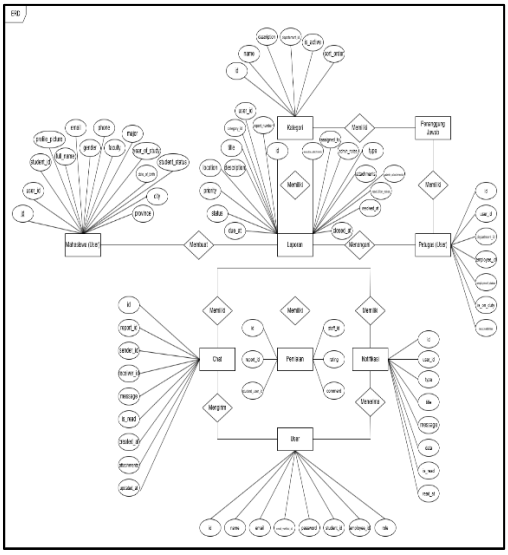
p-ISSN : 2406-7733
e-ISSN : 2597-9922



Gambar 9. *Sequence Diagram*

d. *Entity Relationship Diagram* (ERD)

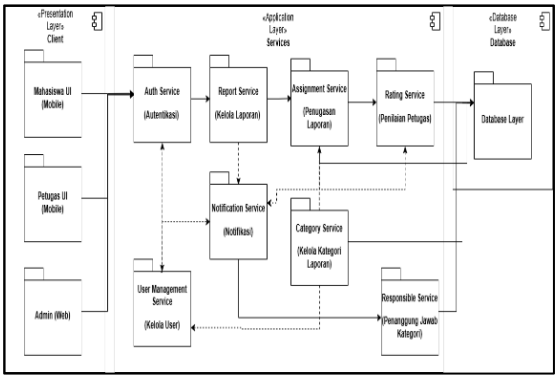
ERD digunakan untuk memvisualisasikan struktur dan relasi basis data (MySQL) yang menyimpan seluruh data operasional. Entitas transaksional utama pada sistem ini adalah Laporan, yang terhubung secara rasional dengan entitas *User* (Mahasiswa, Petugas, Admin), Kategori, *Chat*, dan Penilaian.



Gambar 10. *ERD*

e. *Component Diagram*

Component Diagram memberikan gambaran mengenai struktur perangkat lunak dan ketergantungan antar komponen sistem. Komponen tersebut terbagi ke dalam beberapa lapisan arsitektur, mulai dari lapisan klien (*Presentation Layer*), layanan logika aplikasi (*Application Layer*), hingga basis data (*Database Layer*).

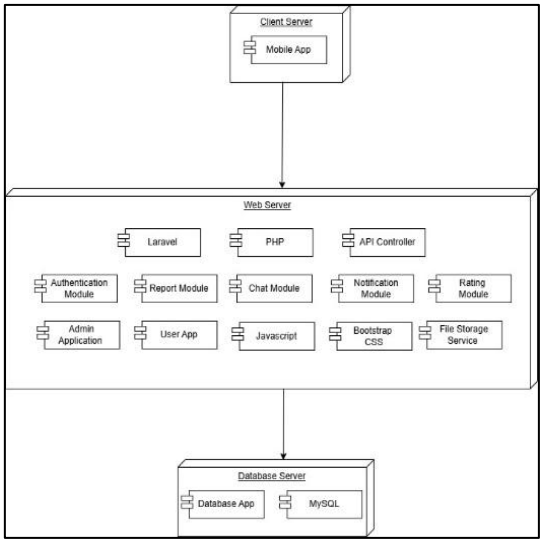


Gambar 11. *Component* Diagram

f. *Deployment Diagram*

Deployment Diagram memodelkan topologi infrastruktur fisik dan jaringan tempat sistem akan diimplementasikan. Diagram ini menunjukkan bagaimana aplikasi *mobile* pada sisi perangkat klien dapat berkomunikasi secara daring dengan *Web Server* (Laravel/PHP) dan *Database Server*.

p-ISSN : 2406-7733
e-ISSN : 2597-9922



Gambar 12. *Deployment Diagram*

4.5 *Sprint Review*

Sprint review dilaksanakan di akhir setiap siklus sprint untuk meninjau seluruh pekerjaan yang telah diselesaikan tim pengembang. Hasil iterasi pertama berupa pembaruan aplikasi mobile yang mencakup beberapa fitur fungsional sesuai *Product Backlog Item*, khususnya fitur distribusi laporan otomatis (*auto-assign*) dan sistem penilaian pelayanan (*rating* bintang).

4.5.1 Implementasi Halaman Pengguna

a. Halaman Form Laporan pada Mahasiswa

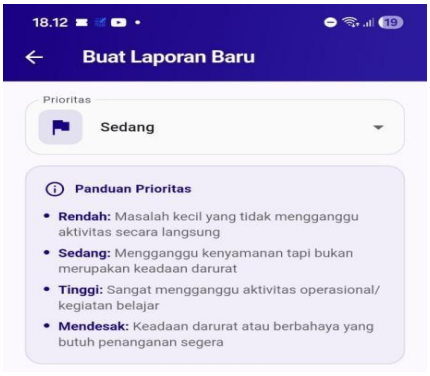
Pada halaman ini mahasiswa bisa membuat laporan dengan mengisi semua *form* laporannya.



Gambar 13. *Form* Laporan

b. Halaman Deskripsi Tingkat Prioritas

Pada halaman pengisian form laporan, terdapat tampilan penjelasan tingkat prioritas secara rinci.



Gambar 14. Deskripsi Prioritas Laporan

c. Halaman Fitur Anonim

Sistem menyediakan opsi pelaporan secara anonim, sebagai upaya menjaga privasi dari pelapor.



Gambar 15. Fitur Anonim

d. Halaman Penilaian Layanan (*Rating*)

Setelah laporan telah selesai ditangani atau status laporannya sudah berubah jadi "Selesai", sistem akan menampilkan opsi penilaian bintang.

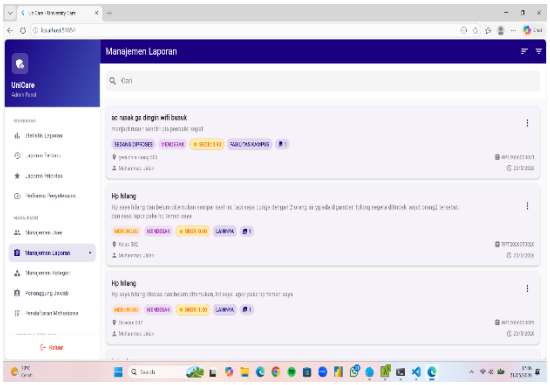


Gambar 16. Memberikan Penilaian

e. Halaman Manajemen Laporan pada Admin

Di halaman ini, admin dapat mengelola langsung laporan yang masuk dan jika terdapat petugas yang aktif maka status akan langsung sedang diproses.

p-ISSN : 2406-7733
e-ISSN : 2597-9922



Gambar 17. Halaman Manajemen Laporan Admin

4.6 *Sprint Retrospective*

Pada akhir siklus *sprint*, seluruh anggota tim mengadakan sesi *Sprint Retrospective* untuk melakukan evaluasi internal terhadap alur kerja yang baru saja dilalui. Melalui sesi ini, tim menganalisis dan mengidentifikasi hal-hal yang sudah berjalan dengan efektif serta membahas kendala teknis atau tantangan yang sempat muncul selama proses pengembangan. Seluruh catatan evaluasi tersebut kemudian dirangkum dan didokumentasikan agar tidak terulang kembali. Hasil dari kegiatan retrospektif ini menjadi bahan pembelajaran sekaligus rekomendasi perbaikan yang berharga, sehingga kinerja dan kolaborasi tim pada iterasi pengembangan (*sprint*) selanjutnya dapat berjalan dengan jauh lebih optimal.

4.7 Pengujian Sistem

Pengujian fungsionalitas aplikasi layanan pengaduan mahasiswa ini dieksekusi menggunakan metode *Blackbox Testing*. Fokus pengujian langsung diarahkan pada evaluasi kecocokan antara interaksi masukan (*input*) pada antarmuka aplikasi dengan keluaran (*output*) sistem. Skenario pengujian menguji fitur-fitur utama aplikasi seperti autentikasi pengguna, pelaporan anonim, sistem distribusi otomatis (*auto-assign*), penguncian fitur *chat*, serta pemberian penilaian (*rating*). Rincian skenario dan hasil pengujian dirangkum pada Tabel 5.

Tabel 5. Hasil Pengujian *Blackbox* Sistem

No	Skenario dan Input	Hasil dan Status
1	Melakukan *login* dengan memasukkan *email* dan *password* yang valid sesuai hak akses (Admin/Mahasiswa/Petugas).	Sistem mengarahkan pengguna ke *dashboard* sesuai hak aksesnya (Berhasil).
2	Melakukan *login* dengan memasukkan *email* atau *password* yang tidak terdaftar/salah.	Sistem menolak akses dan menampilkan pesan *error* (Valid).
3	Membuka form "Buat Laporan Baru" dan menekan pilihan tingkat prioritas laporan.	Sistem menampilkan teks panduan tingkat prioritas (Berhasil).
4	Mengaktifkan opsi "Lapor Secara Anonim" saat mengisi dan mengirimkan form pengaduan.	Laporan terkirim dengan identitas pelapor disembunyikan (Berhasil).
5	Mengirimkan laporan pengaduan baru ke dalam sistem berdasarkan kategori tertentu.	Sistem otomatis meneruskan laporan (*auto-assign*) ke petugas aktif (Berhasil).
6	Petugas menekan opsi tolak laporan pada halaman detail pengaduan mahasiswa.	Sistem otomatis menonaktifkan (*disable*) kolom *chat* (Berhasil).
7	Mahasiswa memberikan penilaian bintang (1-5) pada laporan yang sudah berstatus selesai.	Data penilaian bintang berhasil tersimpan di *database* (Berhasil).
8	Admin membuka menu halaman manajemen laporan atau statistik laporan.	Sistem menampilkan daftar riwayat laporan secara *real-time* (Berhasil).

V. KESIMPULAN DAN SARAN

Kesimpulan

Berdasarkan hasil penelitian dan pengembangan yang telah dilakukan, dapat disimpulkan bahwa aplikasi layanan pengaduan mahasiswa berbasis mobile berhasil dirancang dan diimplementasikan dengan baik menggunakan metode Agile Scrum serta pemodelan arsitektur UML. Penggunaan Trello sebagai alat manajemen proyek terbukti efektif dalam membantu proses penyusunan kebutuhan fungsional ke dalam *product backlog* dan *sprint backlog*, sehingga kolaborasi tim dan pemantauan tugas dapat dilakukan secara lebih terstruktur serta fleksibel terhadap perubahan kebutuhan pengembangan. Sistem yang dibangun mampu menjawab permasalahan pada proses pelaporan konvensional melalui berbagai fitur yang mendukung transparansi layanan, seperti fitur pelaporan anonim untuk menjaga privasi pengguna, panduan tingkat prioritas kerusakan, sistem penilaian kinerja petugas menggunakan rating, penguncian chat otomatis pada laporan tidak valid, serta sistem distribusi laporan otomatis (*auto-assign*) yang mempercepat proses pembagian tugas kepada petugas terkait. Secara keseluruhan, aplikasi ini berhasil mendigitalisasi proses dokumentasi pengaduan dan meningkatkan efisiensi operasional pelayanan fasilitas kampus.

p-ISSN : 2406-7733
e-ISSN : 2597-9922

Saran

Meskipun sistem telah berhasil dikembangkan, masih terdapat beberapa keterbatasan yang dapat menjadi bahan pengembangan pada penelitian selanjutnya. Saat ini, proses distribusi laporan masih bergantung pada pemilihan kategori secara manual oleh mahasiswa. Oleh karena itu, penelitian berikutnya disarankan untuk mengintegrasikan teknologi *Natural Language Processing* (NLP) atau *Machine Learning*. Dengan penerapan teknologi tersebut, sistem dapat melakukan analisis otomatis terhadap teks deskripsi pengaduan untuk menentukan kategori laporan serta tingkat prioritasnya tanpa memerlukan input manual dari pengguna, sehingga dapat mengurangi kesalahan input dan mempercepat proses penanganan laporan.

DAFTAR PUSTAKA

Abiyyu, F., & Kurniawan, D. (2023). *Implementasi Metode Agile Framework Scrum dalam Pengembangan Sistem Informasi Manajemen Aset Terpadu Universitas Diponegoro Modul Inventarisasi*. *14*(1), 53–69.

Approaches, D., Determine, T. O., & Risks, S. (2025). Threat Modeling Menggunakan Pendekatan Stride dan Dread untuk Mengetahui Risiko dan Mitigasi Keamanan (Threat Modeling using Stride and Dread Approaches to Determine. *7*(1), 142–153.

Ardytia, A., Amarta, F., & Anugrah, I. G. (2021). Implementasi Agile Scrum Dengan Menggunakan Trello Sebagai Manajemen Proyek di PT Andromedia. *4*(6), 528–534.

Elda Sulistiawati, Heri Mulyono, A. Y. P. (2023). Perancangan Sistem Informasi Layanan Pengaduan Badan Eksekutif Mahasiswa Berbasis Web. *3*(1), 1–11.

Erzed, N., Anwar, N., Widodo, A. M., Prasetyo, E., & Juman, K. K. (2022). Implementasi Flutter Pada Aplikasi Presensi Karyawan Berbasis Mobile. *6*(3), 100–106.

Hozairi, Buhari, Syariful Alim, R. (2024). *Pengujian Perangkat Lunak*.

Magdalena, L., Efitra, E., & Safitri, N. (2023). Scrum Agile : Optimalisasi Kualitas Produk Manajemen. PT. Sonpedia Publishing Indonesia.

Pangestu, P. R., Voutama, A., Informasi, S., & Karawang, U. S. (2024). Pemanfaatan UML (Unified Modelling Language) pada Sistem. *8*(6), 11846–11851.

Putri, R. (2020). *Pemrograman Basis Data Menggunakan MYSQL*.

Ramadhani, B., Made, N., Iswari, S., Anugrah, P., Dewi, C., Informasi, S., Primakara, U., Denpasar, K., Questionnaire, U. E., & Primakara, D. U. (2026). Perancangan Sistem Informasi Evaluasi Beasiswa Berbasis Website. *10*(2), 2758–2763.

Saragih, S. N., Hidayah, A. K., & Imanullah, M. (2025). Perancangan Aplikasi Layanan Pengaduan Masyarakat Berbasis Web Menggunakan Metode Agile Development. *9*(6), 9702–9708.

Sianipar, F. D., Arifin, M. H., & Aulia, W. (2025). Perancangan Dan Pengembangan Aplikasi Web Pelaporan Kampus di Universitas Negeri Medan. *11*(1), 90–101.

Sitorus, J. H. P., Sianipar, D. G. A., & Informatika, M. (2023). Sistem Informasi Gereja Berbasis Web Menggunakan PHP dan Database MySQL. *7*(1), 1–13.

Sunarti, Ridwang, dan Hayat, (2024). Klasifikasi Pengaduan Pelayanan Fakultas Teknik Universitas Muhammadiyah Makassar menggunakan Natural Language Processing. *Jurnal Sains dan Teknologi (AJST) 2*(2).

Widarti, E., Joosten, J., Pratiwi, P. Y., Pradnyana, G. A., Indradewi, I. G. A. A. D., Kamilah, N., Bahtiar, A. R., Maysanjaya, I. M. D., Sepriano, S., & Efitra, E. (2024). *Buku Ajar Pengantar Sistem Informasi*. PT. Sonpedia Publishing Indonesia.