

LAPORAN
REKAYASA PERANGKAT LUNAK SKALABILITAS
IMPLEMENTASI ARSITEKTUR MONOLITIK PADA RANCANGAN BANGUN SISTEM
INFORMASI



DISUSUN OLEH : PARWATI [105841100622]

PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS MUHAMMADIYAH MAKASSAR
2024-2025

Abstrak

Dalam era digital yang semakin maju, sistem informasi memainkan peran penting dalam menjalankan operasional dan mendukung pengambilan keputusan di berbagai organisasi. Penelitian ini bertujuan untuk menganalisis dan mendiskusikan implementasi arsitektur monolitik pada rancang bangun sistem informasi. Arsitektur monolitik merupakan pendekatan pengembangan perangkat lunak yang menggabungkan semua komponen sistem dalam satu entitas tunggal yang utuh. Penelitian ini membahas tentang merancang dan mengimplementasikan arsitektur monolitik dalam rancang bangun sistem informasi . Metode pengembangan yang digunakan adalah metode waterfall. Studi kasus yang digunakan adalah sistem informasi web keuangan FLATS, yang melayani kebutuhan pengelolaan keuangan dalam program beasiswa FLATS. Hasil evaluasi menunjukkan bahwa implementasi arsitektur monolitik pada sistem informasi FLATS telah berhasil. Integrasi semua komponen terkait keuangan ke dalam satu aplikasi monolitik memungkinkan pengelolaan data dan operasional sistem yang lebih efisien. Sistem ini memiliki skalabilitas yang memadai, mampu mengatasi pertumbuhan data dan pengguna dengan baik, serta menjaga performa yang responsif. Dengan pemahaman yang mendalam tentang konsep dan implementasi arsitektur monolitik, penelitian ini memberikan wawasan yang lebih baik dalam merancang dan membangun sistem informasi yang efektif. Kata Kunci— arsitektur monolitik, sistem informasi, rancang bangun, keuangan, FLATS, metode waterfall, skalabilitas.

I. PENDAHULUAN

Dalam era digital yang semakin maju, sistem informasi telah menjadi elemen kritis dalam menjalankan operasional dan mendukung pengambilan keputusan di berbagai organisasi. Arsitektur perangkat lunak memainkan peran penting dalam rancang bangun sistem informasi yang efektif dan efisien. Salah satu pendekatan arsitektur yang umum digunakan adalah arsitektur monolitik. Penelitian ini bertujuan untuk menganalisis dan mendiskusikan implementasi arsitektur monolitik pada rancang bangun sistem informasi.

Menurut O'Connell dan Gallagher (2012)[1], arsitektur monolitik adalah suatu pendekatan pengembangan perangkat lunak yang menggabungkan semua komponen sistem dalam satu entitas tunggal yang utuh. Dalam arsitektur ini, basis data, lapisan logika bisnis, lapisan presentasi, dan lapisan antarmuka pengguna dikemas menjadi satu kesatuan yang terintegrasi. Hal ini berbeda dengan pendekatan arsitektur berbasis layanan atau arsitektur berorientasi mikro yang mengadopsi pendekatan pemecahan sistem menjadi bagian-bagian terpisah.

Penerapan arsitektur monolitik telah mendapatkan perhatian yang signifikan di kalangan profesional IT. Menurut Li et al. (2018)[2], salah satu alasan utama mengapa arsitektur monolitik masih relevan dan digunakan secara luas adalah kemudahan dalam pengembangan dan pemeliharaan sistem. Pengembang hanya perlu fokus pada satu entitas yang terintegrasi, mengurangi kompleksitas yang terlibat dalam mengelola beberapa layanan yang terpisah. Selain itu, arsitektur monolitik juga memberikan keuntungan dalam hal skalabilitas. Menurut Ahmad et al. (2019)[3], karena semua komponen terkait dalam satu entitas, arsitektur monolitik memungkinkan skalabilitas vertikal yang relatif mudah. Penambahan sumber daya dapat dilakukan dengan menambah kekuatan pemrosesan atau memperbesar kapasitas basis data yang ada.

Penelitian ini akan membahas langkah-langkah yang perlu diikuti dalam merancang dan mengimplementasikan arsitektur monolitik dalam rancang bangun sistem informasi. Peneliti juga akan membahas beberapa metode pengembangan yang dapat digunakan, seperti metode waterfall dan metode agile, untuk memandu proses pengembangan sistem informasi secara keseluruhan. Studi kasus yang digunakan dalam penelitian ini adalah sistem informasi web keuangan FLATS. Sistem informasi ini merupakan suatu web yang dapat melayani berbagai kebutuhan pengelolaan keuangan dalam program beasiswa FLATS.

FLATS merupakan program unggulan beasiswa dari Yayasan Kaki Dian Emas yang memberikan beasiswa selama 5 tahun kepada siswa-siswi Kristen pra sejahtera. Melalui studi kasus sistem informasi web keuangan FLATS [4], penelitian ini akan memperlihatkan bagaimana arsitektur monolitik diterapkan dalam suatu sistem informasi dan tantangan yang dihadapi dalam proses implementasinya. Dengan pemahaman yang mendalam tentang konsep dan implementasi arsitektur monolitik, diharapkan pembaca dapat memperoleh wawasan yang lebih baik dalam merancang dan membangun sistem informasi yang efektif.

II. METODE PENELITIAN

Dalam mengimplementasikan arsitektur monolitik pada web keuangan FLATS, dilakukan serangkaian langkah dan proses yang dilakukan untuk mengumpulkan informasi dan mengimplementasikannya. Penelitian ini melakukan pengembangan dengan menerapkan metode waterfall. Metode waterfall adalah pendekatan pengembangan perangkat lunak yang terstruktur dan sekuensial, di mana setiap tahap dilakukan secara berurutan, dan hasil dari satu tahap menjadi input untuk tahap berikutnya. Pendekatan ini sesuai dengan implementasi arsitektur monolitik pada rancang bangun sistem informasi [5].

Metode waterfall terdiri dari beberapa tahap yang harus diselesaikan secara berurutan. Pertama, tahap analisis digunakan untuk mengidentifikasi dan memahami kebutuhan sistem informasi. Pada tahap ini, persyaratan fungsional dan nonfungsional dikumpulkan dan dianalisis untuk membentuk dasar desain sistem. Setelah tahap analisis, langkah selanjutnya adalah tahap desain. Pada tahap ini, desain sistem informasi dibangun berdasarkan persyaratan yang telah dikumpulkan. Desain ini mencakup perencanaan arsitektur monolitik, desain basis data, desain antarmuka pengguna, dan desain logika aplikasi. Setelah desain selesai, tahap implementasi dimulai. Pada tahap ini, kode program dibangun berdasarkan desain yang telah disusun.

Komponen-komponen sistem monolitik diimplementasikan dan diintegrasikan menjadi satu kesatuan yang utuh. Setelah tahap implementasi, dilakukan pengujian sistem informasi untuk memastikan kualitas dan keandalan sistem. Pengujian dilakukan untuk memverifikasi apakah sistem berfungsi sesuai dengan persyaratan yang telah ditetapkan.

Hal ini melibatkan pengujian fungsional, pengujian integrasi, pengujian kinerja, dan pengujian keamanan. Setelah sistem diuji dan terbukti berfungsi dengan baik, tahap pemeliharaan dimulai. Pada tahap ini, sistem informasi secara rutin diperbarui, diperbaiki, dan dipelihara untuk memastikan kelancaran operasional dalam jangka panjang. Metode waterfall memiliki kelebihan dalam menyusun rencana proyek yang jelas, memudahkan pemantauan kemajuan proyek, dan menghasilkan dokumentasi yang lengkap[6].

Tahapan-tahapan metode waterfall adalah sebagai berikut :

A. ANALISIS

Pada tahap analisis, dilakukan pengumpulan dan pemahaman terhadap kebutuhan sistem informasi yang akan dibangun. Langkah ini melibatkan identifikasi kebutuhan fungsional dan nonfungsional, pemodelan proses bisnis, serta analisis kebutuhan data dan pemilihan teknologi yang tepat. Tujuan dari analisis ini adalah untuk mendapatkan pemahaman yang mendalam tentang kebutuhan bisnis FLATS dan merumuskan persyaratan yang jelas dan terukur. Berdasarkan hasil analisis kebutuhan, kebutuhan pengembangan dapat dibedakan menjadi 2 yaitu :

1. KEBUTUHAN PERANGKAT KERAS (HARDWARE)

Kebutuhan perangkat keras (Hardware) yang dibutuhkan untuk tujuan penelitian, adalah sebagai berikut :

No	Nama Kebutuhan	Keterangan
1	Processor	Intel i3 gen 10
2	RAM	12 GB
3	Storage	SSD 256 GB
4	Sistem Operasi	Ubuntu 23.04

2. KEBUTUHAN PERANGKAT LUNAK (SOFTWARE)

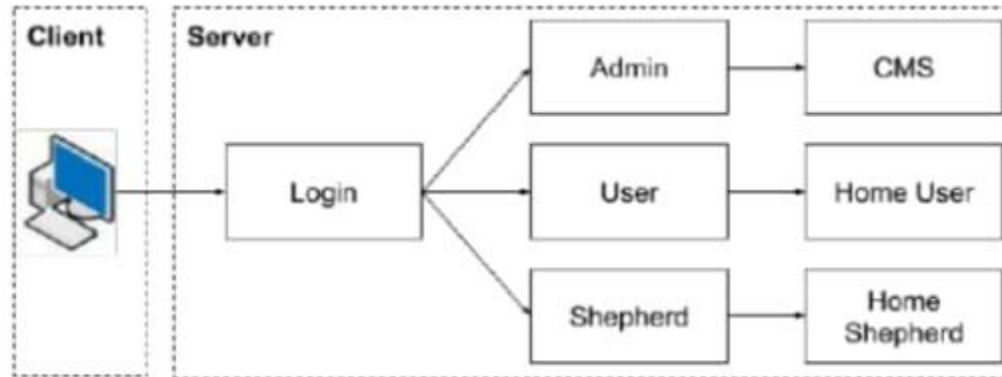
Kebutuhan perangkat lunak (Software) yang dibutuhkan untuk tujuan penelitian, adalah sebagai berikut :

No	Nama Kebutuhan	Keterangan
1	Bahasa Pemrograman	PHP
2	Data Base	MySQL
3	Arsitektur	Monolitik
4	Code Editor	VS Code
5	Web Browser	Google Chrome

B. DESAIN

Setelah tahap analisis, langkah selanjutnya adalah merancang arsitektur sistem informasi. Desain ini melibatkan pemodelan struktur sistem, perancangan basis data, desain antarmuka pengguna, dan penentuan alur logika aplikasi. Dalam konteks implementasi arsitektur monolitik pada web keuangan FLATS, desain ini mencakup pemilihan teknologi, pemilihan bahasa pemrograman, dan desain tata letak aplikasi. Tujuan dari tahap ini adalah untuk merancang solusi yang sesuai dengan kebutuhan dan memastikan skalabilitas, keamanan, dan kinerja sistem.

Pemodelan struktur sistem yang akan diterapkan dalam sistem informasi web keuangan FLATS dapat dilihat dalam gambar berikut ini :



Dalam gambar di atas, menunjukkan desain arsitektur monolitik yang diterapkan dalam sistem informasi. Layanan yang terdapat dalam gambar terintegrasi dalam satu penyimpanan untuk dapat saling berkomunikasi dan menggunakan sumber daya yang sama.

C. IMPLEMENTASI

Setelah desain selesai, langkah berikutnya adalah implementasi sistem informasi. Pada tahap ini, peneliti akan mengkodekan logika bisnis, membangun struktur basis data, mengimplementasikan antarmuka pengguna, dan mengintegrasikan komponen-komponen sistem menjadi satu kesatuan yang utuh. Implementasi arsitektur monolitik pada web keuangan FLATS melibatkan pengembangan modulmodul aplikasi yang saling terhubung dalam satu aplikasi tunggal. Tujuan dari tahap ini adalah untuk menghasilkan sistem informasi yang berfungsi sesuai dengan desain yang telah dirancang sebelumnya.

D. PENGUJIAN

Setelah implementasi, tahap pengujian menjadi penting untuk memastikan kualitas dan keandalan sistem informasi. Pengujian dilakukan untuk memverifikasi apakah sistem berfungsi sesuai dengan persyaratan yang telah ditetapkan, serta untuk mengidentifikasi dan memperbaiki potensi kesalahan atau kekurangan. Pengujian yang relevan dalam konteks web keuangan FLATS dapat meliputi pengujian fungsional, pengujian integrasi, pengujian kinerja, dan pengujian keamanan. Tujuan dari tahap ini adalah untuk memastikan bahwa sistem informasi web keuangan FLATS siap untuk digunakan secara produksi.

Pengujian performa dan penggunaan sumber daya yang digunakan selama pengujian, diukur menggunakan Apache JMeter. Apache JMeter adalah sebuah alat bantu pengujian performa aplikasi yang dirancang untuk menguji performa dan mengukur beban kerja pada aplikasi web dan layanan jaringan. Dalam konteks pengujian performa aplikasi web keuangan Flats, Anda bermaksud menggunakan JMeter untuk mengukur dan mengevaluasi responsibilitas, kinerja, dan keandalan aplikasi tersebut. JMeter menyediakan fungsionalitas yang luas untuk melakukan pengujian performa. Alat ini dapat merekam skenario interaksi pengguna dengan aplikasi web, memungkinkan pengguna untuk membuat skrip pengujian yang dapat diputar ulang dengan berbagai tingkat beban dan konfigurasi. Dalam pengujian performa, JMeter dapat mensimulasikan akses bersamaan oleh banyak pengguna, memantau dan mengumpulkan data statistik, serta memberikan laporan dan grafik untuk menganalisis hasil pengujian[7].

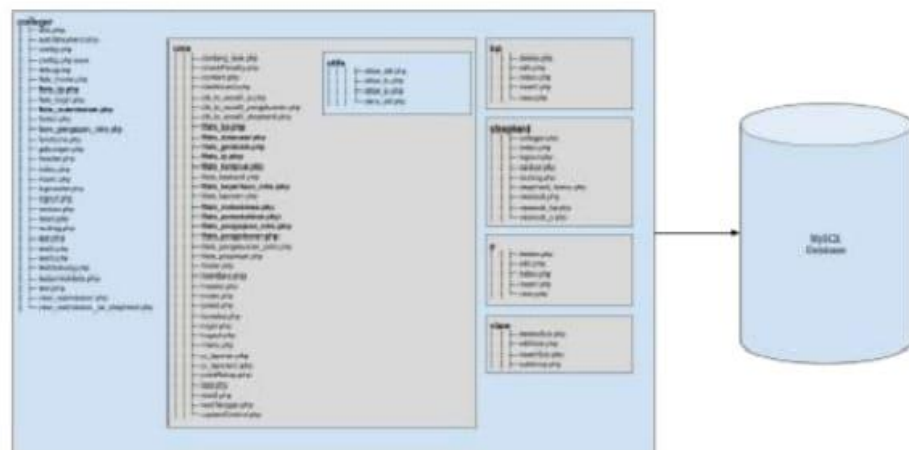
E. PEMELIHARAAN

Setelah sistem diimplementasikan dan diuji, tahap pemeliharaan menjadi penting untuk memastikan kelancaran operasional dan pembaruan sistem secara berkala. Pemeliharaan meliputi perbaikan bug, peningkatan fitur, penyesuaian kebutuhan, serta penanganan masalah yang muncul dalam penggunaan sistem. Tujuan dari tahap ini adalah untuk menjaga kualitas dan kinerja sistem informasi web keuangan FLATS dalam jangka panjang.

III. HASIL DAN PEMBAHASAN

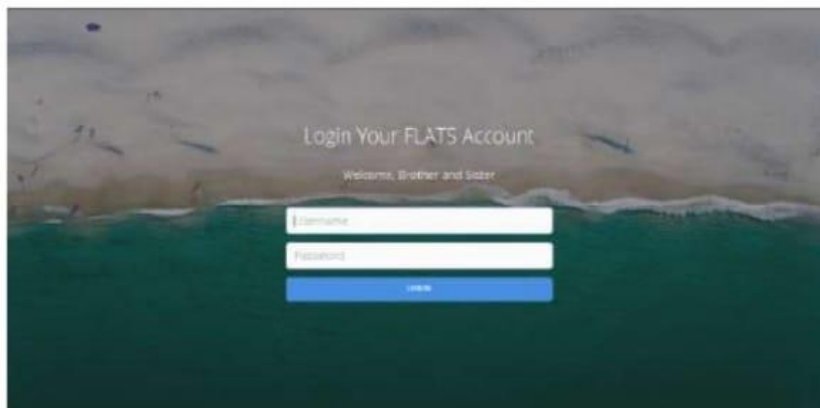
A. IMPLEMENTASI ARSITEKTUR MONOLITIK

Hasil evaluasi menunjukkan bahwa implementasi arsitektur monolitik pada sistem informasi FLATS telah berhasil. Arsitektur monolitik mampu memenuhi kebutuhan fungsional dan nonfungsional sistem dengan baik. Integrasi semua komponen terkait keuangan ke dalam satu aplikasi monolitik memungkinkan pengelolaan data dan operasional sistem yang lebih efisien. Manfaat yang diperoleh dari penggunaan arsitektur monolitik meliputi skalabilitas yang memadai untuk menangani pertumbuhan data dan pengguna, serta kinerja yang responsif dalam menjalankan operasi pengelolaan keuangan.



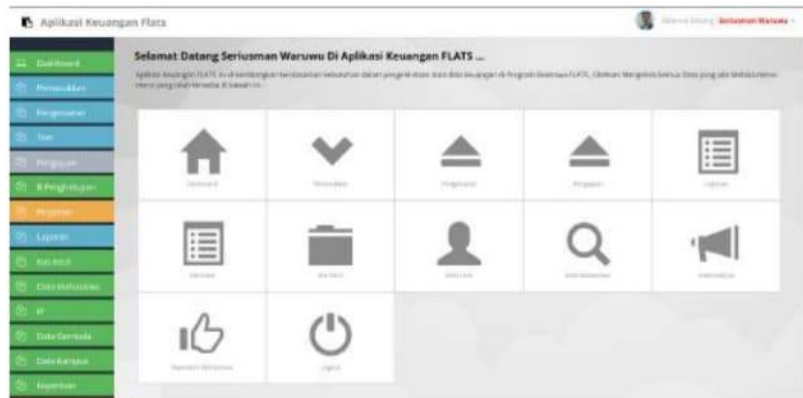
Gambar di atas menunjukkan hasil dari implementasi dari arsitektur monolitik dalam rancang bangun web keuangan FLATS. Dalam gambar tersebut menunjukkan bahwa terdapat berbagai layanan yang tersimpan dalam satu lokasi dan saling terintegrasi. Selain itu juga terdapat pembagian role yang dapat membedakan tipe pengguna dan menampilkan antarmuka yang sesuai dengan tipe pengguna. Sistem informasi yang diterapkan dalam web keuangan FLATS, menggunakan satu database yang sama untuk semua layanan.

Hal ini tentu memberikan manfaat yang memudahkan pengembang untuk menyesuaikan dengan kebutuhan karena menggunakan sumber daya yang sama. Untuk dapat mengakses sistem, diperlukan langkah verifikasi dan validasi keamanan melalui halaman login. Halaman login akan memastikan ketersediaan data pengguna di database serta menentukan role yang dimiliki oleh akun yang akan memasuki sistem. Hasil implementasi halaman login, dapat dilihat dalam gambar berikut ini :

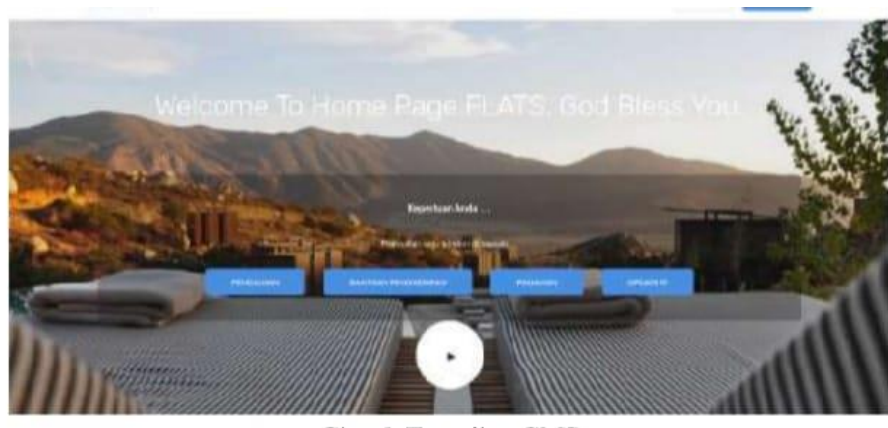


Gambar di atas menunjukkan hasil implementasi dari halaman login. Pengguna dapat melakukan akses fitur sistem melalui memberikan username dan password yang valid, agar sistem dapat mengenali pengguna maupun role yang dimiliki pengguna. Halaman ini melakukan serangkaian proses pengamanan fitur dari pengguna yang tidak terdaftar di dalam sistem, sehingga sistem dapat menjamin keamanan data.

Hasil implementasi dari setiap layanan yang dimiliki oleh admin sistem informasi web keuangan FLATS, dapat dilihat dalam gambar berikut ini:



Gambar diatas menunjukkan halaman utama dari CMS (Content Management System), yang merupakan halaman yang memuat berbagai fitur yang dimiliki oleh sistem. Layanan-layanan tersebut dapat diakses oleh pengguna dengan role admin untuk mengelola semua data user maupun permintaan user. Selain role admin, pengguna dengan role user biasa juga memiliki halaman utama yang berbeda. Dapat dilihat melalui gambar berikut ini :



Gambar diatas menunjukkan halaman utama yang dimiliki oleh sistem untuk pengguna dengan role user biasa. Halaman tersebut memuat beberapa fitur yang dimiliki oleh pengguna untuk.

B. ANALISIS PERFORMA SISTEM

Evaluasi performa sistem informasi web keuangan FLATS menunjukkan hasil yang memuaskan. Waktu respon sistem tergolong cepat, dengan rata-rata waktu pemrosesan transaksi di bawah batasan yang ditetapkan. Performa sistem juga stabil dan konsisten saat diuji dengan beban yang tinggi. Hal ini menunjukkan bahwa arsitektur monolitik mampu memberikan kinerja yang baik dalam mengelola transaksi keuangan dan menghasilkan laporan keuangan dengan efisien.

Untuk mendapatkan hasil dari performa sistem informasi web keuangan FLATS, diperlukan pengujian. Oleh karena itu, peneliti menyediakan skenario pengujian untuk mengetahui performa sistem informasi web keuangan FLATS. Skenario pertama, pengujian dilakukan selama 5 menit dengan waktu tunggu user untuk melakukan request adalah 2 detik.

Transaksi yang dilakukan dalam pengujian ini adalah mengirimkan request input data dari salah satu service secara terus menerus dalam jangka waktu 5 menit dan thread yang terus ditingkatkan hingga 7000. Hasil pengujian akan dicatat untuk dilakukan penghitungan rata-rata dari hasil uji. Skenario kedua, adalah pengujian yang bertujuan untuk mengevaluasi kinerja sistem pada beban yang lebih tinggi daripada pengujian pertama. Semua service dipanggil secara bersamaan tanpa mengatur ramp-up khusus apa pun untuk membuat thread berjalan sekaligus tanpa waktu tunggu.

C. KEAMANAN SISTEM INFORMASI

Penggunaan mekanisme keamanan dalam sistem informasi web keuangan FLATS yang diimplementasikan dengan arsitektur monolitik telah efektif dalam melindungi data sensitif. Penggunaan enkripsi data dan mekanisme otentikasi yang kuat mencegah akses yang tidak sah ke sistem. Pengendalian akses berdasarkan peran pengguna juga berhasil diterapkan dengan baik. Sistem memiliki tingkat keamanan yang memadai untuk menjaga kerahasiaan dan integritas data keuangan pengguna.

D. SKALABILITAS DAN KEMUDAHAN PEMELIHARAAN

Sistem informasi web keuangan FLATS dengan arsitektur monolitik terbukti memiliki skalabilitas yang memadai. Saat diuji dengan jumlah data yang lebih besar dan pengguna yang meningkat, sistem tetap menjaga performa yang responsif dan mampu mengatasi beban dengan baik. Pemeliharaan sistem juga relatif mudah dilakukan, dengan kemampuan untuk melakukan perbaikan bug, peningkatan fitur, dan penyesuaian kebutuhan sistem dengan cepat dan efisien.

REFERENSI

- [1] O'Connell, R., & Gallagher, P. (2012). Architecting Applications for the Enterprise. O'Reilly Media.
- [2] Li, Z., Tian, J., & Chen, H. (2018). A Design for a Monolithic Enterprise Application. Journal of Physics: Conference Series, 1080(4), 042025.
- [3] Ahmad, R., Yusof, Y., & Abdullah, A. (2019). Evaluating Monolithic Architecture and Microservices Architecture on a Smart Farm System. International Journal of Engineering and Advanced Technology (IJEAT), 9(1), 4401-4407.
- [4] Flats.id. (2023). Program Beasiswa FLATS. Flats.id. Diakses pada 26 Juni 2023, dari <https://www.flats.id/>
- [5] Pressman, R. S. (2015). Software Engineering: A Practitioner's Approach. McGraw-Hill Education.
- [6] Sommerville, I. (2016). Software Engineering. Pearson Education Limited.
- [7] Apache Software Foundation. (n.d.). Apache JMeter - User's Manual. Diakses pada 27 Juni 2023, dari <https://jmeter.apache.org/usermanual/index.html>