

## PENERAPAN DOCKER PLATFORM UNTUK INTEGRASI SISTEM INFORMASI SKRIPSI DAN SISTEM INFORMASI PENGECEKAN JUDUL SKRIPSI BERBASIS WEB PADA PROGRAM STUDI INFORMATIKA

Rizky Adi Ryanto<sup>1</sup>, M.Julkarnain<sup>2\*</sup>

<sup>1,2</sup>Informatika, Universitas Teknologi Sumbawa

<sup>1,2</sup>email:adiryantorizky140820@gmail.com, m.julkarnain@uts.ac.id\*

**Abstrak:** Penelitian ini membahas integrasi sistem informasi skripsi dan sistem pengecekan judul skripsi pada Program Studi Informatika di Universitas Teknologi Sumbawa menggunakan platform Docker. Tujuan utamanya adalah meningkatkan efisiensi operasional serta memperbaiki pengalaman pengguna, khususnya bagi mahasiswa dan staf pengajar. Melalui pendekatan Software Development Life Cycle (SDLC) dan metode kualitatif, penelitian ini mencakup berbagai langkah, mulai dari analisis sistem yang berjalan dan usulan hingga implementasi menggunakan Docker sebagai platform containerisasi. Integrasi ini membawa perubahan signifikan, menyatukan proses pengecekan judul skripsi dan pendaftaran skripsi dalam satu platform terintegrasi. Hasilnya menunjukkan peningkatan efisiensi administratif, memudahkan akses, serta penggunaan yang lebih terfokus. Meskipun demikian, penelitian ini juga mendorong pengembangan lebih lanjut untuk pemeliharaan sistem, keamanan data, dan perbaikan sesuai dengan umpan balik pengguna guna meningkatkan fungsionalitas serta ketahanan sistem di masa mendatang.

**Kata Kunci :** Integrasi Sistem, Metode Kualitatif, Platform Docker, Software Development Life Cycle (SDLC)

**Abstract:** This research explores the integration of the thesis information system and thesis title checking system in the Informatics Study Program at the University of Technology Sumbawa using the Docker platform. The main aim is to enhance operational efficiency and improve user experience, particularly for students and faculty members. Employing the Software Development Life Cycle (SDLC) approach and qualitative methods, this study encompasses various stages, ranging from analyzing the current system and proposing solutions to implementing them using Docker as a containerization platform. This integration brings about significant changes by consolidating the thesis title checking process and thesis registration into a unified, integrated platform. The results indicate an enhancement in administrative efficiency, facilitating access, and providing more focused usability. Nevertheless, the research also advocates for further development to maintain the system, ensure data security, and incorporate improvements based on user feedback to enhance system functionality and resilience in the future.

**Keywords :** System Integration, Qualitative Methodology, Docker Platform, Software Development Life Cycle (SDLC)

### PENDAHULUAN

Era digital telah membawa perubahan besar dalam berbagai sektor, termasuk pendidikan tinggi. Di masa kini, universitas di seluruh dunia mengalami transformasi dalam pengelolaan proses akademis dan administratif mereka, berkat kemajuan teknologi informasi. Penggunaan sistem informasi berbasis teknologi tidak hanya meningkatkan efisiensi operasional tetapi juga memperkaya pengalaman belajar dan mengajar. Dalam konteks ini, pengelolaan skripsi sebagai bagian penting dari program pendidikan tinggi, menjadi salah satu area yang mendapat perhatian khusus [1].

Di Universitas Teknologi Sumbawa khususnya di program studi Informatika, Fakultas Rekayasa Sistem, proses pengelolaan skripsi mencakup berbagai aspek, mulai dari pendaftaran judul, pembimbingan, hingga penilaian akhir. Setiap aspek ini memerlukan koordinasi dan pengelolaan data yang tepat. Di sisi lain, pengecekan judul skripsi merupakan langkah penting untuk menjamin originalitas dan menghindari duplikasi topik. Namun, tantangan muncul ketika sistem pengelolaan skripsi dan sistem pengecekan judul skripsi beroperasi secara

terpisah. Hal ini mengarah pada masalah seperti redundansi data, kesulitan dalam sinkronisasi informasi, dan peningkatan beban kerja bagi staf dan mahasiswa [2].

Dari hasil wawancara penulis dengan Sekretaris Program Studi Informatika, diketahui saat ini proses pendaftaran serta penjadwalan seminar proposal dan sidang tugas akhir masih dilakukan secara konvensional. Sementara itu di program studi informatika sendiri sudah memiliki dua sistem sebelumnya yaitu sistem pengecekan judul skripsi dan sistem pendaftaran serta penjadwalan tugas akhir namun belum di integrasikan serta di implementasikan ke server sehingga proses pendaftaran saat ini masih dilakukan secara konvensional.

Docker, sebagai platform containerisasi yang inovatif, menawarkan solusi yang efektif untuk masalah integrasi ini. Platform ini memungkinkan aplikasi untuk dijalankan dalam container yang terisolasi, memastikan konsistensi lingkungan operasional tanpa memperhatikan perbedaan infrastruktur[3]. Penggunaan Docker dalam konteks akademik, khususnya untuk integrasi sistem

informasi, dapat memberikan sejumlah keuntungan. Ini termasuk peningkatan efisiensi dalam deployment aplikasi, kemudahan dalam manajemen versi, dan pengurangan konflik sistem yang berpotensi terjadi karena perbedaan lingkungan pengembangan dan produksi.

Lebih lanjut, penerapan Docker dalam integrasi Sistem Informasi Skripsi dan Sistem Pengecekan Judul Skripsi berbasis *web* di Program Studi Informatika diharapkan dapat meningkatkan efisiensi operasional. Dengan mengintegrasikan kedua sistem ini, diharapkan dapat terjadi peningkatan dalam akurasi data, pengurangan waktu dalam proses administratif, dan pengalaman pengguna yang lebih efektif bagi mahasiswa dan staf pengajar. Selain itu, penelitian ini juga berupaya untuk mengeksplorasi potensi Docker dalam memfasilitasi skenario penggunaan yang lebih luas di lingkungan akademik.

Oleh karena itu, penelitian ini bertujuan untuk tidak hanya mengatasi tantangan integrasi antara dua sistem informasi yang penting ini tetapi juga untuk menunjukkan bagaimana teknologi containerisasi dapat diadopsi secara efektif dalam pengelolaan akademik. Melalui penerapan Docker, penelitian ini diharapkan dapat memberikan wawasan yang berharga mengenai kemudahan integrasi, skalabilitas, dan manajemen sumber daya dalam lingkungan komputasi yang kompleks seperti universitas.

Berdasarkan uraian tersebut maka peneliti memilih judul “Penerapan *Docker Platform* Untuk Integrasi Sistem Informasi Skripsi dan Sistem Informasi Pengecekan Judul Skripsi Berbasis *Web* Pada Program Studi Informatika.”

## TINJAUAN PUSTAKA

Studi sebelumnya memberikan pandangan penting yang menjadi landasan penelitian saat ini. Berikut beberapa riset terkait dalam bidang yang sama: Pertama, Muhammad Fihri, Ridha Muldina Negara, dan Danu Dwi Sanjoyo pada 2019 mengamati performa sistem layanan web pada platform Docker dengan fokus pada throughput, response time, CPU, dan memory utilization. Namun, riset mereka terbatas pada deployment dalam satu jaringan, berbeda dengan penelitian ini yang memperluas deployment ke VPS[4]. Kemudian, M. Hussein Ison & Ricky Eka Putra pada 2021 memfokuskan pemetaan cagar budaya menggunakan teknologi virtualisasi Docker, sementara penelitian ini mengintegrasikan dua sistem dengan Docker sebagai medium[5]. M. Fadlulloh Romadlon Bik pada 2019 memberikan solusi Docker bagi berbagai aplikasi web, berbeda dengan fokus integrasi beberapa aplikasi dalam Docker dalam penelitian ini [6]. Wijaya R, Prasetyo A, Wahyuningsih D, dan lainnya melakukan penelitian penting untuk mengatasi masalah development dan deployment di

website belajarlinux.id dengan fokus pada integrasi antar platform menggunakan Docker dan melakukan testing dengan black box, berbeda dengan fokus penelitian ini pada integrasi antara dua sistem dengan metode SDLC[7]. Nurul Azmi pada 2023 membantu program studi informatika dalam penjadwalan seminar dan skripsi, sedangkan penelitian ini berfokus pada integrasi antara dua sistem informasi dengan metode SDLC[8]. Olvira Nanda pada 2023 mengembangkan sistem pendaftaran skripsi di program studi informatika, di mana fokus penelitian ini adalah pada integrasi sistem tersebut[9]. Wiki Nasmansyah pada 2023 mengimplementasikan algoritma genetika untuk penjadwalan otomatis seminar dan sidang skripsi, berbeda dengan fokus khusus penjadwalan otomatis dalam penelitian ini[10]. Sheva Ricko Apriansyah pada 2023 menerapkan algoritma winnowing dalam mendeteksi kemiripan judul skripsi, sedangkan fokus penelitian ini adalah pada penggunaan algoritma winnowing untuk pengecekan judul skripsi[11].

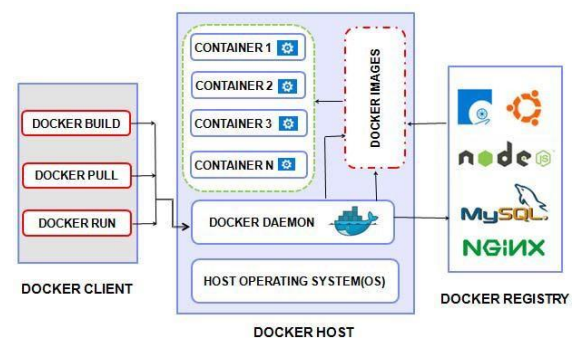
Adapun landasan teori berupa rangkuman teori-teori yang diambil dari pustaka yang mendukung penelitian ini adalah sebagai berikut:

### Virtual Private Server (VPS)

Sebuah server fisik yang terbagi menjadi beberapa server virtual, beroperasi secara virtual tanpa fisik. Setiap VPS memiliki Full Root Access, Sistem Operasi, dan pengaturan sendiri. Dibandingkan shared hosting, VPS bekerja mandiri dengan resources seperti CPU dan RAM yang terisolasi, tidak saling mempengaruhi pada satu server [12]

### Docker

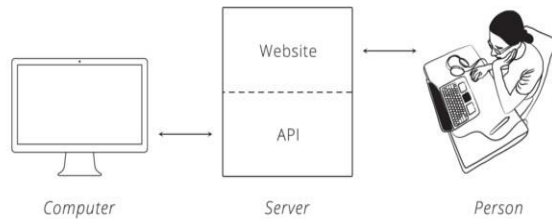
Merupakan aplikasi open source untuk pengembangan, distribusi, dan eksekusi perangkat lunak yang dapat dilihat pada Gambar 1. Menyediakan containerisasi ringan dengan fitur Linux container (LXC) seperti cgroups dan namespaces untuk mengatur sumber daya dan isolasi proses. Docker memiliki docker dan docker hub, yang berfungsi sebagai platform containerization dan layanan Software-as-a-Service (SaaS) untuk berbagi dan mengelola docker container[4]



Gambar 1. Arsitektur Docker

## API

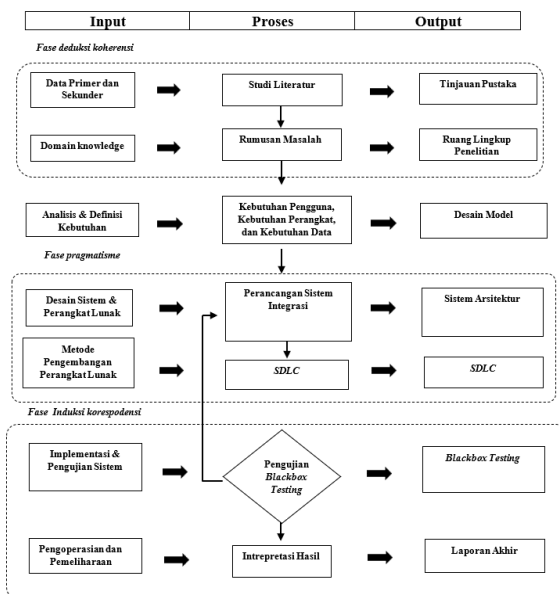
Merupakan antarmuka yang memungkinkan akses terprogram ke fungsi sistem[13]. Representational State Transfer (REST) adalah gaya arsitektur pengembangan API yang menggunakan HTTP untuk komunikasi data sebagaimana tersaji pada Gambar 2 [5]



Gambar 2. Ilustrasi API

## METODE

Dalam Penelitian ini, penulis menggunakan dua metode yaitu metode pengumpulan data dan metode pengembangan perangkat lunak. Jenis penelitian yang digunakan yaitu penelitian kualitatif. Adapun tahapan– tahapan dalam penelitian ini dapat dilihat pada diagram alur berikut:

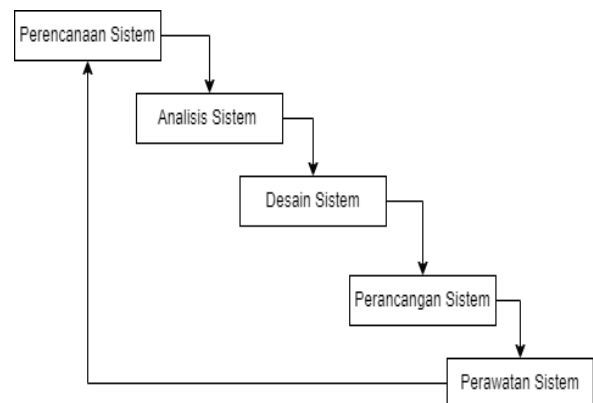


Gambar 3. Metode Penelitian

Berdasarkan dari Gambar 3 dimana Metode pengumpulan data yang digunakan dalam penelitian ini bersifat kualitatif dan meliputi beberapa pendekatan. Pertama, observasi digunakan untuk mencatat kebutuhan yang diperlukan dalam menerapkan platform Docker untuk mengintegrasikan sistem informasi skripsi dan sistem pengecekan judul skripsi di program studi Informatika UTS. Kedua, wawancara dilakukan dengan sekretaris Prodi Informatika UTS, guna mendapatkan data langsung melalui tanya jawab. Selain itu, dokumentasi digunakan untuk mengumpulkan informasi dan catatan penting terkait sistem informasi skripsi dan pengecekan judul skripsi

di program studi Informatika. Terakhir, studi literatur menjadi pendekatan yang dilakukan untuk merujuk pada referensi-referensi terkait dari buku, e-book, skripsi, artikel, dan sumber-sumber internet yang relevan dengan topik penelitian. Pendekatan ini bertujuan untuk mendukung pemikiran serta mendapatkan informasi yang relevan dalam penyusunan program dan laporan penelitian [14]

Dalam penelitian ini, metode pengembangan perangkat lunak yang diterapkan adalah Software Development Life Cycle (SDLC)[15], yang terdiri dari beberapa tahapan yang bisa dilihat pada Gambar 4.



Gambar 4. Software Development Life Cycle (SDLC)

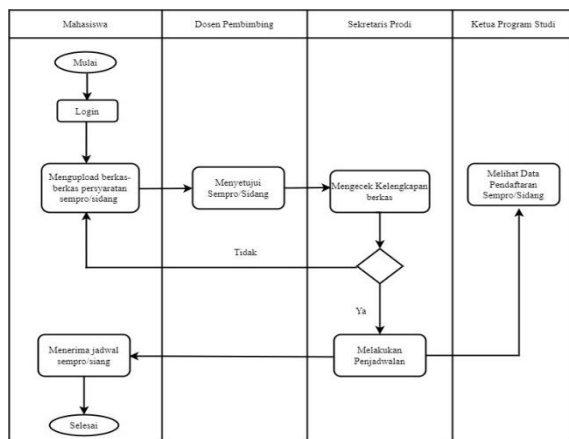
Berdasarkan dari Gambar 4, tahap pertama adalah perencanaan sistem yang melibatkan pengumpulan data melalui observasi di program studi Informatika, wawancara dengan Sekretaris Prodi Informatika dokumentasi terkait penjadwalan pada program studi, dan studi literatur dari berbagai sumber yang relevan. Tahapan yang ke dua adalah Analisis sistem selanjutnya mengungkapkan beberapa hal, termasuk belum diimplementasikannya serta belum terintegrasi dua sistem informasi, perbedaan stack yang digunakan, dan adanya bug di kedua sistem tersebut. Tahap ke tiga yaitu perancangan sistem kemudian memanfaatkan Docker sebagai platform deployment dan REST API sebagai alat untuk integrasi kedua sistem. Dilanjutkan dengan tahap implementasi sistem, yang melibatkan proses pengkodean sistem yang telah dirancang sebelumnya dengan menggunakan Visual Studio Code sebagai text editor, VPS, dan Docker sebagai platform penyebaran sistem[16]. Tahapan terakhir yaitu, tahap perawatan sistem dilakukan untuk memperbaiki bug dan kesalahan yang dilaporkan oleh pengguna selama penggunaan perangkat lunak yang dikembangkan [17].

## HASIL DAN PEMBAHASAN

Pada bagian Hasil dan Pembahasan terdapat beberapa bagian yaitu sebagai berikut:

### Analisis Sistem Berjalan

Analisis sistem berjalan dalam pada sistem informasi skripsi adalah sebagai berikut:

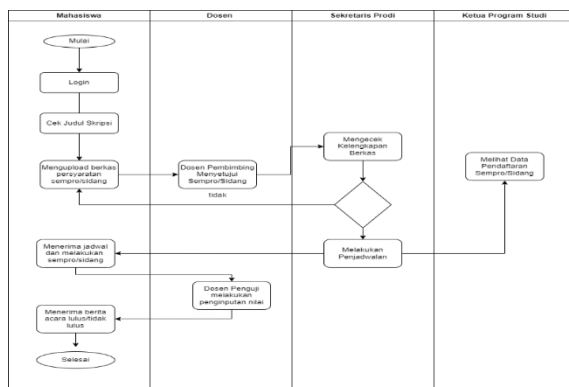


Gambar 5. Analisis Sistem Berjalan

Berdasarkan dari Gambar 5 untuk analisis system yang berjalan pada, prosesnya dimulai ketika mahasiswa melakukan login ke sistem dan mengunggah berkas persyaratan untuk seminar proposal atau sidang. Selanjutnya, dosen pembimbing memiliki peran untuk menyetujui proposal atau sidang yang diajukan oleh mahasiswa. Langkah berikutnya melibatkan peran Sekretaris Program Studi yang bertugas untuk memeriksa kelengkapan berkas yang diunggah. Jika ada kekurangan, berkas tersebut dikembalikan kepada mahasiswa untuk diperbaiki. Namun, jika berkas sudah lengkap, proses dilanjutkan ke tahap penjadwalan seminar atau sidang. Pada tahap ini, Ketua Program Studi memiliki akses untuk melihat data pendaftaran seminar atau sidang yang masuk ke sistem. Setelah semua tahapan telah dilalui, mahasiswa akan menerima jadwal untuk seminar atau sidang, dan proses tersebut dianggap selesai. Dalam keseluruhan sistem ini, setiap tahap memainkan peran kunci dalam mengatur dan melacak progres dari persiapan hingga pelaksanaan seminar proposal atau sidang mahasiswa.

### Analisis Sistem Usulan

Analisis sistem usulan dalam pada sistem informasi skripsi adalah sebagai berikut:

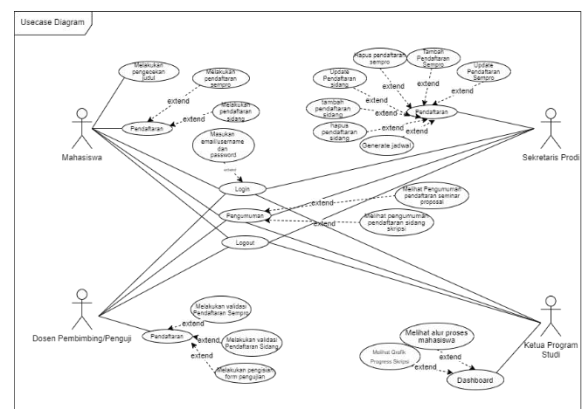


Gambar 6. Analisis Sistem Usulan

Sistem informasi yang diusulkan ini dimulai ketika seorang mahasiswa melakukan login dan mengecek judul skripsi yang telah diajukan. Selanjutnya, mahasiswa mengunggah berkas persyaratan untuk seminar proposal atau sidang. Proses selanjutnya melibatkan dosen pembimbing yang harus menyetujui proposal atau sidang yang diajukan oleh mahasiswa. Setelah itu, sekretaris program studi memeriksa kelengkapan berkas. Jika berkas tidak lengkap, proses akan kembali ke tahap pengunggahan berkas persyaratan oleh mahasiswa. Namun, jika berkas sudah lengkap, sekretaris program studi akan melanjutkan dengan penjadwalan seminar proposal atau sidang. Pada tahap ini, Ketua Program Studi dapat melihat data pendaftaran seminar proposal atau sidang yang masuk. Kemudian, mahasiswa akan menerima jadwal dan melakukan seminar proposal atau sidang. Setelah itu, dosen penguji akan memberikan nilai terhadap seminar atau sidang yang telah dilakukan oleh mahasiswa. Terakhir, mahasiswa akan menerima berita acara mengenai status kelulusannya dan proses tersebut dianggap selesai. Dalam sistem ini, setiap tahap memiliki peran penting dalam mengatur jalannya proses dari persiapan hingga penilaian akhir terkait seminar proposal atau sidang mahasiswa. Sistem ini dirancang untuk memfasilitasi pengelolaan dan pelacakan tahapan-tahapan yang diperlukan dalam proses akademik ini dengan lebih terstruktur dan terorganisir.

### Usecase dan Class Diagram Sistem Informasi Skripsi

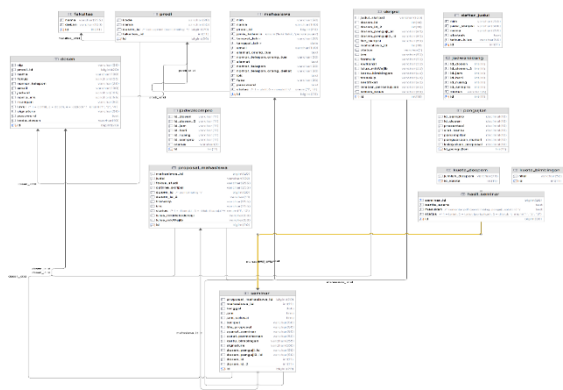
Usecase Diagram dalam pada sistem informasi skripsi adalah sebagai berikut:



Gambar 7. Usecase Diagram Sistem Informasi Skripsi

Use case diagram yang tunjukkan pada Gambar 7 menunjukkan beberapa aktor dan tindakan yang dilakukan oleh setiap aktor dalam sistem. Ada tiga aktor utama: mahasiswa, Dosen Pembimbing/Penguji, Sekretaris Prodi, dan Ketua Program Studi. Mahasiswa memiliki kemampuan untuk melakukan serangkaian tindakan penting.

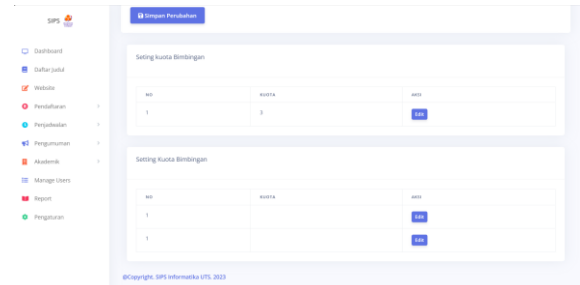
Pertama, mereka dapat melakukan pengecekan judul skripsi, yang merupakan langkah awal dalam pendaftaran seminar proposal atau sidang skripsi. Selanjutnya, mahasiswa dapat melakukan pendaftaran seminar proposal dan pendaftaran sidang. Tindakan ini menggambarkan keterlibatan aktif mahasiswa dalam proses akademik mereka, memberi mereka kemampuan untuk mengelola dan mengatur tahapan penting dalam perjalanan skripsi mereka. Dosen Pembimbing/Penguji juga memiliki peran kunci dalam sistem ini. Mereka dapat melakukan validasi terhadap pendaftaran seminar proposal dan sidang skripsi oleh mahasiswa. Selain itu, mereka bertanggung jawab untuk melakukan pengisian form pengujian terkait seminar proposal atau sidang skripsi mahasiswa. Tindakan ini menunjukkan peran penting dosen dalam memastikan kualitas dan kelayakan seminar proposal serta sidang skripsi mahasiswa. Pada sisi administrasi, Sekretaris Prodi memiliki tanggung jawab untuk mengelola pendaftaran. Mereka dapat melakukan update terhadap pendaftaran seminar proposal dan sidang, baik dalam hal menambah, menghapus, atau melakukan update terhadap data-data yang relevan. Kemampuan ini memungkinkan mereka untuk memastikan integritas dan kelengkapan informasi yang diperlukan dalam proses administrasi. Ketua Program Studi memiliki peran dalam pemantauan keseluruhan proses akademik. Mereka dapat melihat dashboard yang memberikan gambaran tentang alur proses yang dijalani oleh mahasiswa, serta melihat grafik progress skripsi secara keseluruhan. Hal ini memungkinkan mereka untuk memantau kemajuan mahasiswa secara menyeluruh dan memahami dinamika yang terjadi di dalam lingkungan akademik. Kombinasi dari fungsi-fungsi ini memungkinkan aktor-aktor terlibat dalam mengelola tahapan-tahapan penting dalam perjalanan skripsi mahasiswa. Dengan adanya peran dan fungsionalitas yang didefinisikan dalam use case diagram ini, sistem mampu memberikan dukungan yang komprehensif dan terorganisir dalam pengelolaan skripsi mahasiswa. Sedangkan untuk Class Diagram untuk system informasi skripsi bisa di lihat pada Gambar 8 dibawah ini



Gambar 8. Class Diagram Sistem Informasi Skripsi

### Setting kuota pembimbing dan bimbingan

Hasil Implementasi dari Setting kuota pembimbing dan bimbingan dalam pada sistem informasi skripsi adalah sebagai berikut:

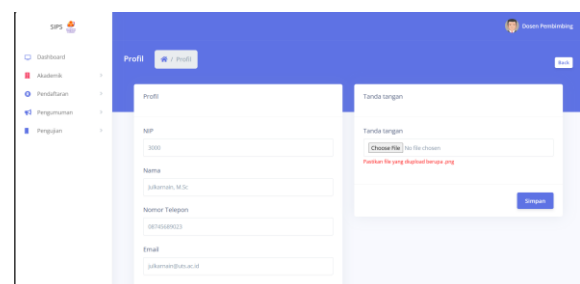


Gambar 9. Setting Kuota Pembimbing dan Bimbingan

Berdasarkan Gambar 8, Dimana menampilkan halaman pengaturan untuk mengatur kuota pembimbing dan bimbingan. Halaman ini mencakup beberapa bidang, seperti bidang kuota dan aksi. Untuk mengedit kuota dosen pembimbing, sekretaris prodi dapat mengeklik tombol edit di bagian aksi. Ini akan memunculkan jendela pop-up yang memungkinkan untuk mengubah kuota dosen pembimbing dan kuota bimbingan. Setelah itu, sekretaris prodi dapat menyimpan perubahan tersebut dengan menekan tombol simpan. Pengaturan ini memungkinkan sekretaris prodi untuk dengan mudah mengelola dan mengubah kuota-kota terkait dengan bimbingan. Proses ini memberikan kemudahan dalam menyesuaikan kuota dosen pembimbing dan kuota bimbingan sesuai dengan kebutuhan program studi. Dengan fitur ini, penyuntingan kuota dapat dilakukan dengan cepat dan efisien, memastikan kelancaran proses administrasi terkait bimbingan akademik.

### Upload Tanda Tangan Dosen

Hasil Implementasi dari Upload tanda tangan dosen dalam pada sistem informasi skripsi adalah sebagai berikut:



Gambar 10. Upload Tanda Tangan Dosen

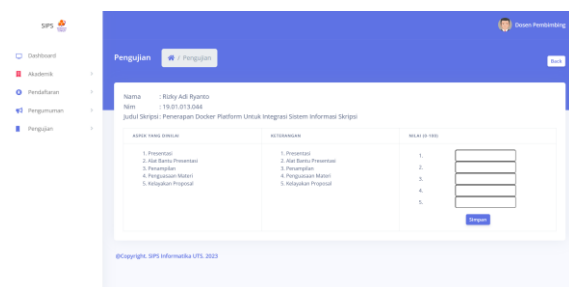
Gambar yang ditampilkan adalah halaman pengaturan yang memungkinkan penyesuaian kuota untuk pembimbing dan bimbingan. Halaman ini memuat beberapa kolom, termasuk kolom kuota dan aksi. Untuk mengubah kuota pembimbing, sekretaris program studi dapat mengklik tombol edit di bagian aksi. Langkah ini akan memunculkan jendela modal yang memungkinkan pengeditan kuota untuk dosen pembimbing serta kuota bimbingan. Setelah



pengaturan selesai, sekretaris dapat menyimpan perubahan dengan menekan tombol save. Proses ini memastikan bahwa kuota yang telah diatur berhasil disimpan. Fitur ini memudahkan pengelolaan kuota dengan memberikan akses kepada sekretaris program studi untuk melakukan penyesuaian sesuai kebutuhan tanpa perlu campur tangan pihak lain. Kemampuan untuk mengubah kuota pembimbing secara langsung melalui tindakan sederhana ini membantu dalam mengelola dan mengalokasikan sumber daya dengan lebih efisien di lingkungan akademis.

### Input Nilai Dosen Penguji

Hasil Implementasi dari Input nilai dosen penguji dalam pada sistem informasi skripsi adalah sebagai berikut:

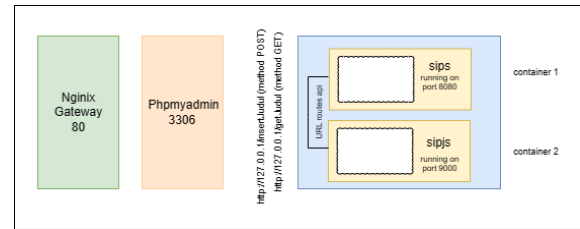


Gambar 11. Input Nilai Dosen Penguji

Gambar tersebut menampilkan halaman yang didedikasikan bagi peran dosen untuk mengunggah tanda tangan, baik sebagai dosen pembimbing maupun dosen penguji. Pada halaman ini, dosen diminta untuk mengunggah tanda tangan dalam format PDF atau gambar dengan menekan tombol "Pilih Berkas". Setelah unggahan tanda tangan selesai, dosen dapat menyimpannya dengan menekan tombol "Simpan". Fitur ini memfasilitasi proses penyimpanan tanda tangan digital secara efisien bagi para dosen, memungkinkan mereka untuk menyediakan tanda tangan mereka sesuai kebutuhan tanpa kesulitan teknis yang berlebihan. Kemampuan untuk mengunggah dalam format yang berbeda memberikan fleksibilitas kepada dosen, sehingga mereka dapat menggunakan format yang paling sesuai dengan tipe tanda tangan yang mereka miliki. Proses ini membantu dalam pengarsipan data penting sekaligus mempercepat berbagai proses administratif yang memerlukan tanda tangan dosen.

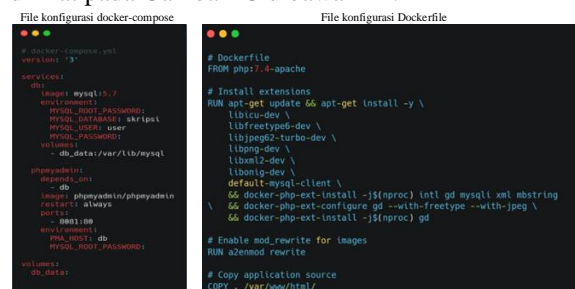
### Integrasi Docker Platform

Implementasi Docker Platform yang direncanakan untuk mengintegrasikan Sistem Informasi Skripsi serta Sistem Informasi Pengecekan Judul Skripsi yang berbasis web di Program Studi Informatika. Dimana Rancangan dapat di lihat pada Gambar 13 dibawah ini.



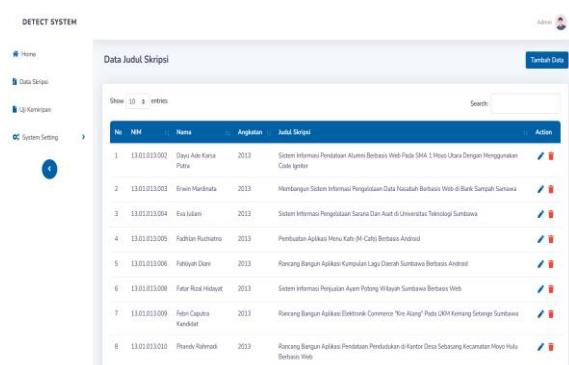
Gambar 12. Arsitektur Integrasi Docker Platform

Dalam penelitian ini, Nginx Gateway 80 berperan sebagai web server yang dioperasikan Docker sebagai reverse proxy. Port 80 dipilih sebagai tempat operasional Nginx dalam rancangan Docker, dimana server ini akan menangani seluruh permintaan HTTP yang masuk melalui port tersebut. Selanjutnya, Phpmyadmin akan digunakan untuk menampung basis data yang tersimpan dalam kontainer Docker. Rencana tersebut juga melibatkan dua kontainer yang beroperasi di port yang berbeda. Kontainer pertama bertujuan untuk menjalankan proyek SIPS pada port 8080, sementara kontainer kedua akan mengeksekusi proyek SIPJS pada port 9000. Keduanya akan terkoneksi melalui URL route API dengan endpoint `http://127.0.0.1/getJudul` yang menggunakan metode GET. Selain itu, endpoint `http://127.0.0.1/insertJudul` berfungsi sebagai endpoint API dengan metode POST. Jika terdapat tambahan seperti `/angkat` dalam endpoint, contohnya `http://127.0.0.1/getJudul/1`, angka 1 akan menjadi parameter yang diperlukan dalam endpoint tersebut. Tidak hanya itu, di dalam kedua kontainer tersebut terdapat file Dockerfile dan docker-compose yang menjadi konfigurasi Docker pada masing-masing kontainer. Rancangan ini penting dalam konteks penelitian karena menyajikan cara terstruktur untuk mengintegrasikan berbagai elemen sistem. Konfigurasi yang disajikan dalam setiap kontainer memungkinkan operasionalitas yang lancar dan efisien. Hal ini sangat krusial dalam memastikan bahwa sistem yang dikembangkan dalam penelitian ini dapat berjalan sesuai kebutuhan proyek serta memberikan manfaat yang diharapkan dalam domain Sistem Informasi Skripsi berbasis web. Untuk File konfigurasi docker-compose dan Dockerfile dapat dilihat pada Gambar 13 dibawah ini.



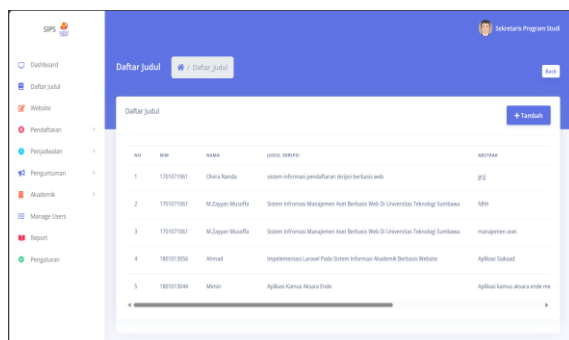
Gambar 13. File Konfigurasi Docker-Compose dan Dockerfile

Sebelumnya di integrasikan menggunakan platform docker sistem pengecekan judul skripsi dan system informasi skripsi memerlukan dua langkah terpisah dimana mahasiswa mengakses situs web sistem pengecekan judul skripsi untuk pengecekan judul, dan menggunakan sistem informasi skripsi untuk pendaftaran. Diperlukan dua akses terpisah karena fungsi pengecekan judul dan pendaftaran terpisah. Mahasiswa harus melakukan langkah pertama untuk mengecek judul, lalu langkah kedua untuk melakukan pendaftaran skripsi. Hal ini memastikan pemisahan fungsi agar proses pendaftaran dan pengecekan judul berjalan secara independen sesuai dengan kebutuhan sistem. Untuk tampilannya Sistem Informasi Pengecekan Judul sebelum di integrasi dapat di lihat pada Gambar 14.



Gambar 14. Sistem Informasi Pengecekan Judul Sebelum di Integrasi

Setelah diintegrasikan melalui Docker Platform, sistem ini menyatukan proses pengecekan judul dan pendaftaran skripsi menjadi satu langkah. Mahasiswa sekarang hanya perlu mengakses satu platform terintegrasi untuk kedua proses tersebut. Penggunaan Docker memungkinkan konsolidasi fungsionalitas yang sebelumnya terpisah menjadi satu, memudahkan akses dan penggunaan bagi mahasiswa. Dengan integrasi ini, proses menjadi lebih efisien dan efektif karena memungkinkan pengguna untuk menyelesaikan proses pengecekan judul dan pendaftaran skripsi dengan langkah yang lebih sederhana dan terpusat dalam satu platform. Untuk tampilannya Sistem Informasi Skripsi sudah di integrasi dapat di lihat pada Gambar 15 dibawah ini.



Gambar 15. Sistem Informasi Pengecekan Judul Sesudah di Integrasi

## Black box testing

Black box testing adalah metode pengujian perangkat lunak yang mengevaluasi fungsionalitas tanpa memerhatikan struktur internal. Pengujian ini memeriksa input dan output sistem tanpa mengetahui kode atau logika internalnya. Dengan simulasi situasi pengguna, tujuannya adalah memvalidasi fungsi sistem sesuai spesifikasi. Uji coba ini mengevaluasi fungsionalitas eksternal, mendeteksi bug, dan memastikan kesesuaian antara kebutuhan pengguna dengan perilaku aplikasi [18]. Black box testing fokus pada interaksi antara sistem dan pengguna serta memastikan bahwa sistem berfungsi dengan benar tanpa memperhatikan bagian internalnya [19]. Adapun hasil pengujian dengan Black box testing dapat di lihat pada table 1 dibawah ini.

Tabel 1. Hasil Black Box Testing

Aksi	Yang Diharapkan	Kesimpulan
Membuka halaman daftar judul skripsi	Menampilkan halaman hasil integrasi	Berhasil
Melakukan Upload tanda tangan dosen	Menyimpan data tanda tangan	Berhasil
Input Nilai Pengujian	Meyimpan data nilai pengujian	Berhasil
Setting kuota dosen pembimbing dan kuota bimbingan	Menyimpan dan membatasi kuota dosen pembimbing dan kuota bimbingan	Berhasil

## KESIMPULAN DAN SARAN

Berdasarkan hasil implementasi dan pengujian yang telah dilakukan pada penelitian ini, integrasi antara Sistem Informasi Skripsi dan Sistem Pengecekan Judul Skripsi menggunakan platform Docker memiliki dampak signifikan dalam meningkatkan efisiensi proses administratif di lingkungan akademik. Integrasi ini berhasil mengkonsolidasikan fungsi-fungsi terpisah dari pengecekan judul skripsi dan pendaftaran skripsi menjadi satu platform terintegrasi, memudahkan akses dan penggunaan bagi mahasiswa. Langkah-langkah implementasi yang dijalankan sesuai dengan Software Development Life Cycle (SDLC) dan melibatkan berbagai metode pengumpulan data serta pengembangan perangkat lunak. Melalui penggunaan Docker sebagai platform containerisasi, penelitian ini memberikan pemahaman yang berharga mengenai kemudahan integrasi, pengelolaan sumber daya, dan efisiensi dalam deployment aplikasi akademik.

Saran yang diberikan adalah perlunya pengembangan lebih lanjut terkait pemeliharaan sistem, termasuk perbaikan bug dan kesalahan yang terdeteksi, serta penyesuaian sesuai feedback pengguna. Diperlukan juga peningkatan keamanan sistem untuk melindungi data sensitif mahasiswa dan pengguna lainnya.

## DAFTAR PUSTAKA

- [1] Danur Wijayanto, Arizona Firdonsyah, and Faisal Dharma Adhinata, "IMPLEMENTASI CONTINUOUS INTEGRATION/CONTINUOUS DELIVERY MENGGUNAKAN PROCESS MANAGER 2 (Studi Kasus: SIAKAD Akademi Keperawatan Bina Insan)," *Teknika*, vol. 10, no. 3, pp. 181–188, Oct. 2021, doi: 10.34148/teknika.v10i3.400.
- [2] B. Rahman and S. A. Ningsi, "SISTEM INFORMASI MONITORING SKRIPSI MAHASISWA BERBASIS WEBSITE," *Simtek : jurnal sistem informasi dan teknik komputer*, vol. 7, no. 2, 2022, doi: 10.51876/simtek.v7i2.152.
- [3] Restu Agung Parama, "IMPLEMENTASI DOCKER UNTUK PENGELOLAAN APLIKASI SIAKAD PT SENTRA VIDYA UTAMA (SEVIMA)," *Institut Teknologi Sepuluh November*, pp. 1–53, 2022.
- [4] M. Fihri, R. M. Negara, and D. D. Sanjoyo, "IMPLEMENTASI & ANALISIS PERFORMANSI LAYANAN WEB PADA PLATFORM BERBASIS DOCKER IMPLEMENTATION & ANALYSIS OF WEB SERVICE PERFORMANCE BASED ON DOCKER PLATFORM," 2019.
- [5] M. H. Ison and R. E. Putra, "IMPELEMENTASI VIRTUAL SERVER BERBASIS CONTAINER PADA SISTEM INFORMASI GEOGRAFIS CAGAR BUDAYA MOJOKERTO," 2021.
- [6] M. Fadlulloh and R. Bik, "IMPLEMENTASI DOCKER UNTUK PENGELOLAAN BANYAK APLIKASI WEB (Studi Kasus : Jurusan Teknik Informatika UNESA)," 2019.
- [7] R. Wijaya *et al.*, "IMPLEMENTASI CI/CD UNTUK BUIDDAN DEPLOY WEBSITE DENGAN DOCKER RUNNER PADA ORGANISASI BELAJAR LINUX ID," 2021.
- [8] Nurul Azmi, "PENJADWALAN SEMINAR DAN SIDANG TUGAS AKHIR PADA SISTEM INFORMASI PELAYANAN SKRIPSI BERBASIS WEB," 2023.
- [9] Olvira Nanda, "SISTEM INFORMASI PENDAFTARAN SKRIPSI BERBASIS WEBSITE DI PROGRAM STUDI INFORMATIKA UNIVERSITAS TEKNOLOGI SUMBAWA," 2023.
- [10] Wiki Nasmansyah, "PENERAPAN ALGORITMA GENETIKA UNTUK PENJADWALAN OTOMATIS SIDANG SKRIPSI PADA SISTEM INFORMASI PELAYANAN SKRIPSI INFORMATIKA UNIVERSITAS TEKNOLOGI SUMBAWA," 2023, 2023.
- [11] Sheva Ricko Apriansyah, "PENERAPAN ALGORITMA WINNOWER UNTUK MENDETEKSI KEMIRIPAN JUDUL SKRIPSI PADA PROGRAM STUDI INFORMATIKA UNIVERSITAS TEKNOLOGI SUMBAWA," 2023.
- [12] A. Fathiah, "PENGEMBANGAN APLIKASI CLOUD STORAGE DENGAN PEMANFAATAN STORAGE PADA VPS DAN ANDROID INTERFACE (MOBILE CLOUD STORAGE)," *Journal of Informatics and Computer Science*, pp. 1–6, 2019.
- [13] Brian Cooksey, "EBOOK API INDONESIA," 2016. [Online]. Available: <https://zapier.com/learn/apis/>
- [14] H. S. Husni Mubarak, "PENGUJIAN DAN IMPLEMENTASI EFEKTIVITAS SISTEM VIRTUALISASI BERBASIS DOCKER PLATFORM," *urnal InformatikaTerpadu*, vol. 5, pp. 1–6, 2019.
- [15] A. Ikegami, I. D. M. B. A. Darmawan, and I. G. A. G. A. Kadyanan, "IMPLEMENTASI REST API PADA FITUR ADMIN DARI APLIKASI PEMINJAMAN RUANGAN," 2022.
- [16] M. Fihri, R. M. Negara, and D. D. Sanjoyo, "IMPLEMENTASI DAN ANALISIS PERFORMANSI LAYANAN WEB PADA PLATFORM BERBASIS DOCKER IMPLEMENTASI & ANALYSIS OF WEB SERVICE PERFORMANCE BASED ON DOCKER PLATFORM," 2019.
- [17] M. R. Ramadhan, L. E. Nugroho, and S. Sulisty, "PERANCANGAN SISTEM INFORMASI MONITORING SKRIPSI."
- [18] J. Shadiq, A. Safei, R. Wahyudin Ratu Loly, C. sitasi, L. Rwr, and P. Aplikasi Peminjaman Kendaraan Operasional Kantor Menggunakan BlackBox Testing, "INFORMATION MANAGEMENT FOR EDUCATORS AND PROFESSIONALS Pengujian Aplikasi Peminjaman Kendaraan Operasional Kantor Menggunakan BlackBox Testing," *Information Management for Educators and Professionals*, vol. 5, no. 2, pp. 97–110, 2021.
- [19] L. Setiyani, "Techno Xplore Jurnal Ilmu Komputer dan Teknologi Informasi PENGUJIAN SISTEM INFORMASI INVENTORY PADA PERUSAHAAN DISTRIBUTOR FARMASI MENGGUNAKAN METODE BLACK BOX TESTING," 2019.