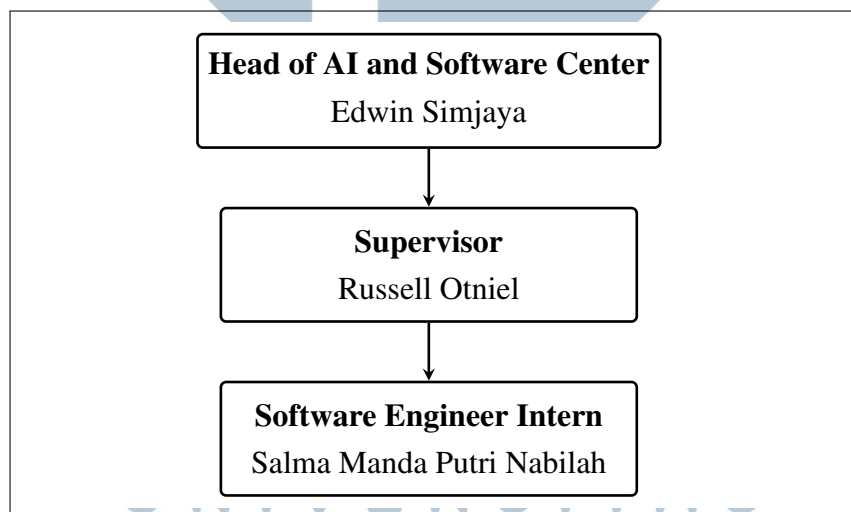


BAB 3

PELAKSANAAN KERJA MAGANG

3.1 Kedudukan dan Koordinasi

Program magang dilaksanakan di lingkungan departemen Corporate Digital Technology (CDT) PT Kalbe Farma Tbk dengan penempatan pada posisi Software Engineer Intern. Posisi ini bertanggung jawab untuk membantu proses perancangan dan pembangunan aplikasi yang digunakan untuk mendukung kebutuhan operasional internal perusahaan. Proses analisis dan pengembangan aplikasi dilakukan melalui koordinasi intensif bersama pengguna dari unit terkait, sehingga solusi yang dihasilkan selaras dengan kebutuhan bisnis dan alur kerja di lapangan. Seluruh kegiatan magang berada di bawah bimbingan seorang mentor yang menjabat sebagai Software Engineer Officer, dengan struktur kedudukan dan jalur koordinasi yang ditunjukkan pada Gambar 3.1.



Gambar 3.1. Struktur Kedudukan dan Koordinasi Magang

3.2 Tugas yang Dilakukan

Sebagai Software Engineer Intern di PT Kalbe Farma Tbk, peran utama adalah mendukung rangkaian aktivitas eksplorasi, perancangan, serta pengembangan aplikasi web untuk sistem Regulatory Information Management (RIM). Sistem ini dikembangkan untuk membantu tim Regulatory Affairs (RA) dalam mengelola dokumen regulasi secara terpusat, tersusun, dan efisien.

Pengembangan RIM melibatkan pembuatan berbagai fitur inti yang dibutuhkan pengguna, di antaranya modul *Dashboard* dan modul *Documents* yang berfokus pada pemantauan dan pengelolaan dokumen. Teknologi utama yang digunakan meliputi Next.js sebagai kerangka kerja antarmuka dan Supabase untuk layanan basis data serta penyimpanan berkas.

Seluruh proses dikerjakan secara kolaboratif dengan tim Project Management, tim UI/UX, dan perwakilan Regulatory Affairs, dimulai dari pengumpulan kebutuhan (*requirement gathering*), perancangan fitur, implementasi, hingga pengujian fungsional dan *debugging* untuk memastikan stabilitas aplikasi. Kegiatan rutin seperti diskusi harian, rapat evaluasi mingguan, dan penyusunan dokumentasi menjadi bagian dari siklus pengembangan perangkat lunak yang dijalani selama magang.

Secara garis besar, pengalaman magang mencakup tahapan awal siklus hidup pengembangan aplikasi web, mulai dari analisis kebutuhan, pembuatan Proof of Concept (PoC), implementasi fitur, hingga uji coba bersama pengguna akhir. Selain keterampilan teknis, kemampuan komunikasi, kerja sama tim, dan pemecahan masalah juga terasah dalam konteks kerja profesional. Tabel 3.1 menyajikan ringkasan tugas mingguan selama program magang.

Tabel 3.1. Pekerjaan Mingguan Selama Magang

Minggu ke-	Pekerjaan yang Dilakukan
1	Mendesain struktur awal <i>header</i> dan <i>top navigation</i> pada halaman <i>Dashboard</i> , serta mulai membangun komponen <i>top bar Dashboard</i> .
2	Menyelesaikan pembangunan area <i>header Dashboard</i> dan mulai membangun area bawah <i>Dashboard</i> , termasuk eksplorasi paket untuk pembuatan grafik berbasis data yang dikumpulkan.
3	Pembuatan area bawah <i>Dashboard</i> beserta tampilan grafik, serta merancang antarmuka <i>Dashboard Settings</i> yang mengatur konfigurasi tampilan <i>Dashboard</i> .
4	Mengimplementasikan fungsi-fungsi pada <i>Dashboard Settings</i> dan melakukan perbaikan serta pembaruan tampilan <i>Dashboard</i> berdasarkan desain Figma terbaru.

Minggu ke-	Pekerjaan yang Dilakukan
5	Memulai membuat modul <i>Documents</i> dengan membangun komponen tabel dokumen (<i>document table</i>) dan menyusun struktur tampilan daftar dokumen.
6	Melanjutkan pembangunan tabel dokumen dan mengembangkan komponen <i>action list</i> pada halaman <i>Documents</i> untuk mendukung berbagai aksi terhadap dokumen.
7	Menyempurnakan tabel dokumen, memperbaiki umpan balik saat pemilihan baris tabel, serta membangun komponen aksi <i>check-in</i> pada halaman <i>Documents</i> .
8	Membuat komponen aksi <i>related document</i> pada halaman <i>Documents</i> , melakukan <i>refactor</i> tampilan tabel dokumen, serta mulai mengambil data dokumen langsung dari basis data.
9	Memperbaiki dan memperbarui halaman <i>RnD Documents</i> , termasuk uji coba integrasi dengan API perusahaan untuk dokumen RnD serta penyempurnaan tampilan dan fungsinya.
10	Melakukan <i>refactor</i> lanjutan pada tabel dokumen dan mekanisme pengambilan data, mempelajari modul proyek <i>handoff</i> , serta memperbaiki komponen dialog <i>check-in</i> dan tampilan <i>related documents</i> .
11	Mengikuti kegiatan <i>pre-IQ</i> proyek <i>handoff</i> di Kalbe Site serta melanjutkan penyempurnaan tampilan dan fungsionalitas <i>related documents</i> pada daftar aksi dokumen.
12	Membuat modul impor dokumen untuk <i>RA Documents</i> , mencakup pembuatan dialog unggah file, dialog pemilihan tipe dokumen, serta konfigurasi tabel pendukung di basis data.
13	Membuat dialog pengisian metadata impor dokumen, membuat fungsi untuk mengunggah file ke <i>cloud</i> dan menyimpan metadata ke basis data, serta memperbaiki struktur pohon tipe dokumen.
14	Menambahkan fitur pencarian pada dialog tipe dokumen, memperbaiki validasi formulir, dan melakukan pengujian modul impor dokumen RA termasuk dukungan kegiatan <i>RIM Integration Unit Test meeting</i> .

Minggu ke-	Pekerjaan yang Dilakukan
15	Melanjutkan <i>refactor</i> tabel dokumen dan mekanisme pengambilan data dari basis data, serta mempelajari perubahan validasi pada modul proyek <i>handoff</i> .
16	Membuat dialog <i>Responsible User</i> pada modul impor dokumen untuk mengatur penetapan peran pengguna, membangun dialog pemilihan tipe dokumen pada modul pembuatan dokumen RA, serta mengikuti rapat rutin mingguan/dua mingguan.
17	Membuat dialog metadata dokumen pada modul pembuatan dokumen RA yang memuat isian informasi umum dokumen, serta mengikuti rapat koordinasi terkait perkembangan modul.
18	Membangun dialog <i>Responsible User</i> pada modul pembuatan dokumen RA untuk konfigurasi peran sekaligus mengikuti rapat koordinasi mingguan.
19	Membangun dialog <i>Related Documents</i> pada modul pembuatan dokumen RA untuk mengelola keterkaitan antar dokumen regulasi yang saling berhubungan.

3.3 Uraian Pelaksanaan Magang

Pelaksanaan kegiatan magang dilakukan dengan menyelesaikan rangkaian tugas yang diberikan oleh mentor dan tim pengembang sesuai dengan prioritas proyek yang sedang berjalan. Selama kurang lebih enam bulan, fokus utama diarahkan pada keterlibatan langsung dalam proses pengembangan dan implementasi aplikasi web sistem RIM.

Aktivitas yang dilakukan mencakup identifikasi kebutuhan teknis, perancangan solusi fungsional, pembangunan komponen antarmuka, hingga integrasi dan pengujian fitur-fitur utama dalam sistem. Tugas-tugas tersebut dijalankan melalui pendekatan kolaboratif dengan tim UI/UX, Project Management, dan Regulatory Affairs agar hasil pengembangan sesuai dengan konteks penggunaan aktual di lapangan.

Setiap tahapan pengembangan diatur secara berurutan, dimulai dari analisis dan desain, dilanjutkan dengan implementasi dan pengujian, kemudian penyempurnaan berdasarkan umpan balik pengguna. Pendekatan ini bertujuan

untuk memastikan sistem RIM yang dibangun benar-benar mampu mendukung pengelolaan dokumen regulasi secara terpusat dan sesuai kebutuhan bisnis perusahaan.

Kebutuhan pengguna pada proyek ini dirumuskan dengan mengacu pada dokumen User Requirement Specification (URS) yang disusun melalui proses kolaboratif bersama pihak-pihak terkait. URS terlebih dahulu diawali dengan pelaksanaan studi kelayakan (*feasibility study*) untuk memastikan rancangan aplikasi selaras dengan kondisi dan kebutuhan nyata. Dokumen URS kemudian digunakan sebagai acuan untuk menentukan fitur-fitur kunci yang harus tersedia, merancang alur kerja sistem, serta menetapkan batasan dan arah pengembangan berikutnya. Berikut adalah fitur utama yang diperlukan pada modul *Documents* dan modul *Dashboard*.

3.3.1 Pembuatan Modul Documents

A User Requirement

Modul *Documents* merupakan fitur utama dalam sistem Regulatory Information Management (RIM) yang memfasilitasi pengelolaan dokumen regulasi secara terpusat, terdokumentasi, dan terkontrol sepanjang siklus hidupnya. Melalui modul ini, pengguna dapat membuat dokumen baru, mengimpor dokumen yang sudah ada, serta mengelola versi dan status dokumen agar selalu selaras dengan standar dan persyaratan regulator yang berlaku.

A.1 Menampilkan Data Dokumen

Fitur ini digunakan untuk menyajikan daftar dokumen utama yang telah dibuat dalam bentuk tabel. Setiap baris berisi informasi inti yang dibutuhkan agar proses identifikasi dan pemantauan dokumen menjadi lebih mudah. Melalui tampilan ini, pengguna dapat meninjau dan mengelola dokumen yang tersimpan di sistem secara lebih menyeluruh.

A.2 Pembuatan Dokumen

Pembuatan dokumen digunakan untuk membuat dokumen regulasi baru langsung di dalam sistem tanpa bergantung pada dokumen yang sudah ada. Melalui fitur ini, pengguna mengisi formulir yang berisi metadata utama serta informasi

lain yang diperlukan untuk pengelolaan dan penelusuran dokumen. Setelah metadata terisi, pengguna dapat melanjutkan ke tahap pengisian konten, kemudian menyimpan dokumen dalam status awal agar dapat melalui proses *review*, *approval*, dan perubahan status sesuai alur kerja yang telah ditetapkan.

A.3 Impor Dokumen

Fitur impor dokumen digunakan untuk memasukkan dokumen regulasi yang sudah ada ke dalam sistem tanpa perlu membuatnya dari awal. Melalui fitur ini, pengguna dapat mengunggah berkas dari penyimpanan lokal atau sumber lain, kemudian melengkapi metadata penting seperti nama dokumen, tipe dokumen, serta status dokumen. Setelah proses impor selesai, dokumen akan tercatat dan muncul di daftar dokumen sehingga dapat dikelola lebih lanjut, seperti *di-edit*, *di-review*, atau diajukan untuk persetujuan sesuai alur kerja yang berlaku.

B Flowchart Modul Documents

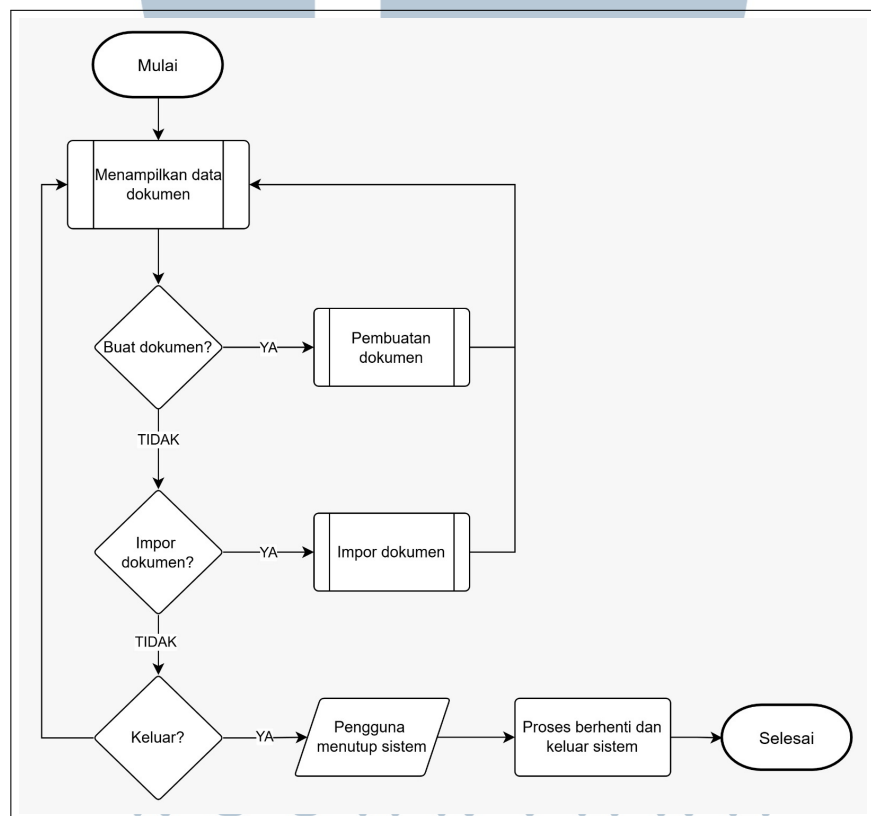
Modul *Documents* memiliki alur utama yang menggambarkan bagaimana pengguna berinteraksi dengan daftar dokumen, mulai dari pembuatan, impor, hingga keluar dari sistem. Alur kerja secara keseluruhan terlihat pada Gambar 3.2 yang menunjukkan rangkaian keputusan dan aksi utama yang dapat dilakukan pengguna di dalam modul *Documents*.

Proses diawali ketika sistem menampilkan data dokumen pada halaman daftar dokumen. Pada tahap ini, pengguna dapat melihat daftar dokumen yang sudah tersedia beserta status terkininya. Selanjutnya, sistem menampilkan keputusan terkait pembuatan dokumen baru. Jika pengguna memilih untuk membuat dokumen, proses berpindah ke sub-proses pembuatan dokumen, di mana pengguna melakukan penyusunan dokumen baru sesuai kebutuhan. Setelah proses pembuatan selesai, alur kembali ke langkah penayangan daftar dokumen sehingga dokumen baru tersebut tampil dalam daftar.

Apabila pengguna tidak melakukan pembuatan dokumen baru, sistem memberikan opsi untuk melakukan impor dokumen. Jika pengguna memilih untuk mengimpor dokumen, sistem menjalankan sub-proses impor dokumen yang memungkinkan pengguna mengunggah dokumen yang sudah ada dan melengkapinya dengan metadata yang diperlukan. Setelah proses impor selesai, alur kembali lagi ke penayangan data dokumen sehingga dokumen hasil impor

langsung terlihat dalam daftar. Bila pengguna tidak memilih opsi impor, sistem melanjutkan ke keputusan terkait penghentian penggunaan modul.

Pada keputusan ini, apabila pengguna memutuskan untuk mengakhiri penggunaan modul, sistem menjalankan langkah pengguna menutup sistem yang dilanjutkan dengan proses penghentian proses dan keluar dari sistem, menandai berakhirnya sesi penggunaan modul *Documents*. Jika pengguna memilih untuk tetap menggunakan modul, alur dialihkan kembali ke langkah awal penayangan data dokumen sehingga pengguna dapat terus melanjutkan aktivitas pengelolaan dokumen tanpa harus keluar dari sistem. Dengan demikian, flowchart ini menggambarkan alur utama yang sederhana namun lengkap untuk mendukung siklus kerja pembuatan, impor, dan penelusuran dokumen di dalam modul *Documents*.



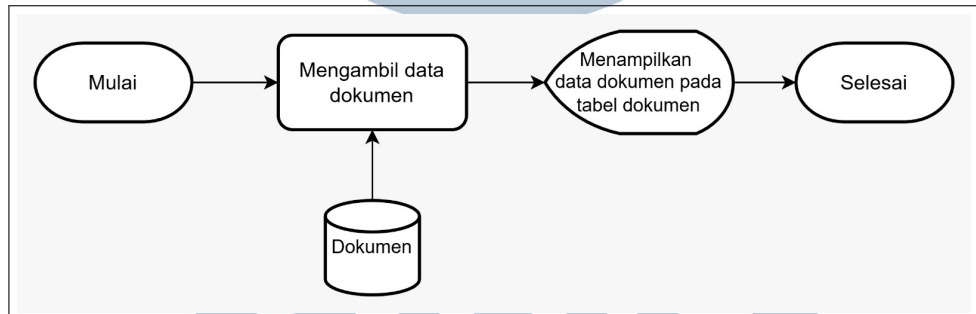
Gambar 3.2. Flowchart Modul Documents

Berikut adalah uraian subproses untuk masing-masing fitur yang tersedia pada modul *Documents*.

B.1 Menampilkan Data Dokumen

Flowchart pada Gambar 3.3 menggambarkan alur sederhana untuk menampilkan data dokumen pada modul *Documents*. Proses diawali ketika pengguna membuka halaman yang berisi tampilan daftar dokumen. Sistem kemudian menjalankan langkah pengambilan data dokumen dari basis data dokumen yang tersimpan pada sistem. Pada tahap ini, sistem melakukan *query* ke penyimpanan data untuk memperoleh kumpulan dokumen beserta atribut yang diperlukan, seperti nama, status, dan informasi pendukung lainnya.

Setelah proses pengambilan data selesai, sistem menampilkan hasilnya dalam bentuk tabel dokumen pada antarmuka pengguna, sehingga setiap baris tabel merepresentasikan satu entitas dokumen. Tampilan tabel ini memungkinkan pengguna untuk melihat, menelusuri, dan melakukan tindakan lanjutan terhadap dokumen-dokumen yang tersedia, misalnya membuka detail atau melakukan pengelolaan lebih lanjut. Setelah data berhasil disajikan kepada pengguna, alur berakhir pada kondisi selesai yang menandakan bahwa proses penarikan dan penayangan data dokumen telah tuntas dilaksanakan.



Gambar 3.3. Flowchart Menampilkan Data Dokumen

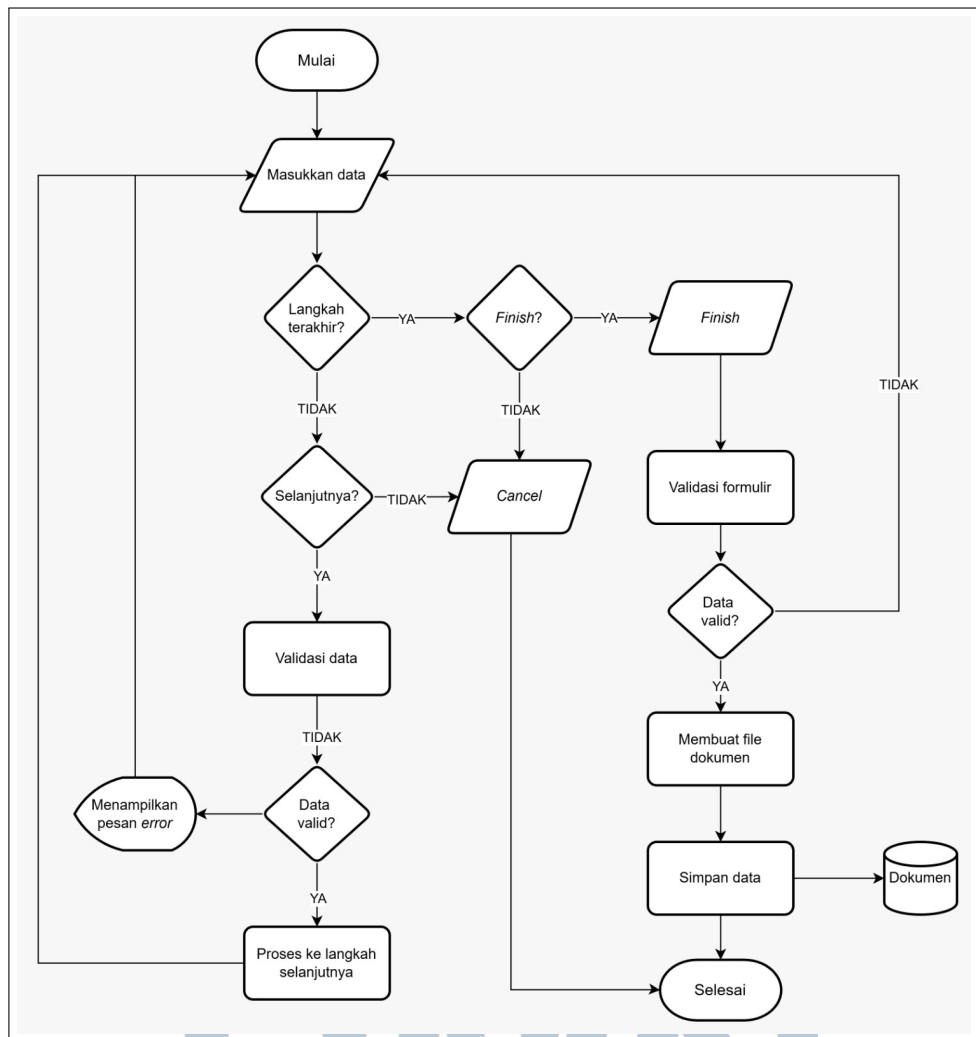
B.2 Pembuatan Dokumen

Flowchart pada Gambar 3.4 menggambarkan alur proses pembuatan dokumen baru pada modul *Documents*. Proses dimulai ketika pengguna memasukkan data ke dalam formulir pembuatan dokumen. Setelah data pada langkah pertama diisi, sistem mengevaluasi apakah langkah yang sedang diisi merupakan langkah terakhir pada rangkaian pengisian formulir. Jika masih terdapat langkah berikutnya, sistem menampilkan bagian formulir selanjutnya agar pengguna dapat melengkapi seluruh informasi yang dibutuhkan sebelum validasi dilakukan.

Pada saat pengguna telah mencapai langkah terakhir, sistem menyediakan opsi untuk menyelesaikan pengisian formulir atau membatalkan proses. Jika proses dibatalkan, alur kembali ke awal sehingga data tidak disimpan. Jika pengguna memilih untuk menyelesaikan pengisian, sistem melakukan validasi formulir secara menyeluruh untuk memastikan bahwa semua data wajib telah terisi dan sesuai dengan aturan yang berlaku. Apabila ditemukan kesalahan atau ketidaksesuaian, sistem menandai data sebagai tidak valid dan menampilkan pesan kesalahan agar pengguna dapat melakukan perbaikan. Jika data dinyatakan valid, sistem melanjutkan proses dengan membuat file dokumen berdasarkan data yang telah di-*input*.

Setelah file dokumen berhasil dibuat, sistem menyimpan data dokumen beserta metadata ke dalam basis data dokumen. Pada bagian alur di sisi kiri, terdapat proses validasi data bertahap yang memastikan setiap langkah pengisian memenuhi persyaratan sebelum melanjutkan ke langkah berikutnya. Jika data yang dimasukkan belum memenuhi ketentuan, sistem akan menampilkan pesan *error* dan mengarahkan pengguna kembali ke formulir untuk melakukan perbaikan. Ketika seluruh tahapan validasi dan penyimpanan selesai dilakukan, alur berakhir pada kondisi selesai yang menyatakan bahwa proses pembuatan dokumen baru telah tuntas dan dokumen sudah tercatat di dalam sistem.





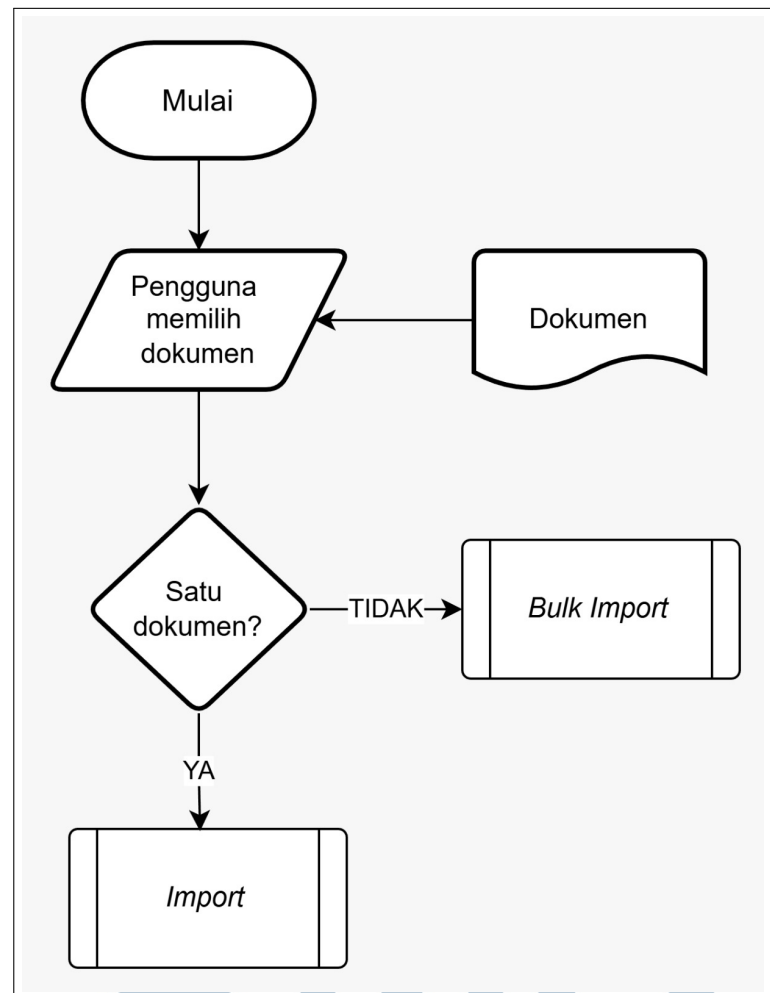
Gambar 3.4. Flowchart Pembuatan Dokumen

B.3 Impor Dokumen

Flowchart pada Gambar 3.5 menampilkan alur utama ketika pengguna melakukan proses impor dokumen ke dalam sistem. Proses diawali saat pengguna memilih satu atau lebih berkas dari penyimpanan lokal untuk diunggah ke sistem. Setelah pemilihan berkas dilakukan, sistem memeriksa jumlah *file* yang dipilih untuk menentukan skema impor yang akan digunakan.

Apabila hanya satu berkas yang dipilih, alur akan dilanjutkan ke subproses *Import* seperti yang terlihat pada Gambar 3.6 yang merepresentasikan prosedur impor dokumen tunggal. Namun, jika pengguna memilih lebih dari satu berkas, sistem akan mengarahkan ke subproses *Bulk Import* yang mengelola impor banyak dokumen sekaligus sebagaimana digambarkan pada Gambar 3.7. Dengan demikian,

alur utama ini bertindak sebagai pengatur awal yang memutuskan apakah proses impor dijalankan dengan pola tunggal atau massal berdasarkan jumlah dokumen yang dipilih.

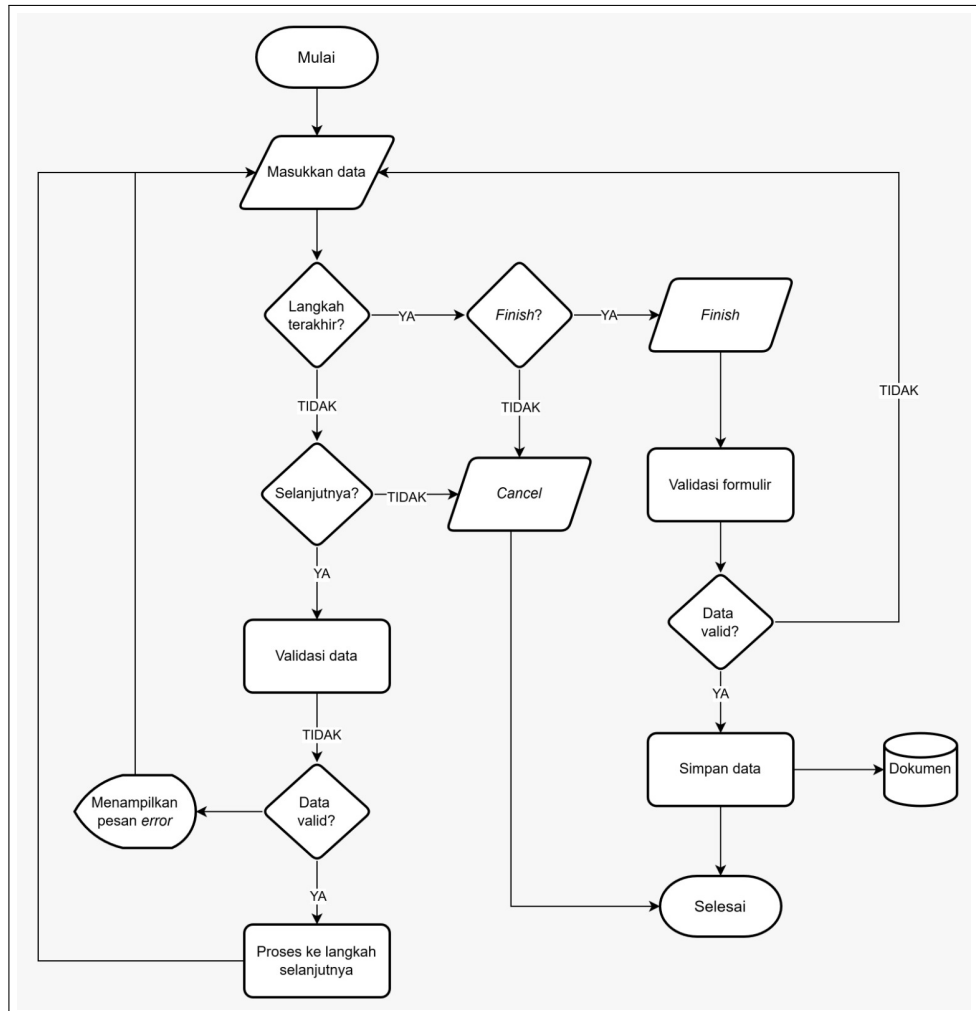


Gambar 3.5. Flowchart Utama Impor Dokumen

Flowchart impor dokumen tunggal pada sub-modul *Import* (Gambar 3.6) menjelaskan langkah-langkah ketika pengguna mengimpor satu dokumen. Proses dimulai dengan pengisian data pada formulir impor yang berisi metadata dokumen, seperti judul, tipe dokumen, produk terkait, serta informasi pendukung lainnya. Setelah formulir diisi, sistem melakukan validasi data untuk memastikan bahwa seluruh *field* wajib telah terisi dengan format yang benar.

Apabila hasil validasi menunjukkan adanya kesalahan atau kekosongan data, sistem menampilkan pesan kesalahan dan mengembalikan pengguna ke formulir untuk melakukan perbaikan. Setelah data dinyatakan valid, sistem menyimpan

informasi dokumen beserta *file* yang diunggah ke dalam basis data dokumen. Alur kemudian berakhir pada kondisi selesai yang menandakan bahwa proses impor dokumen tunggal telah tuntas dan dokumen sudah tercatat dalam sistem.

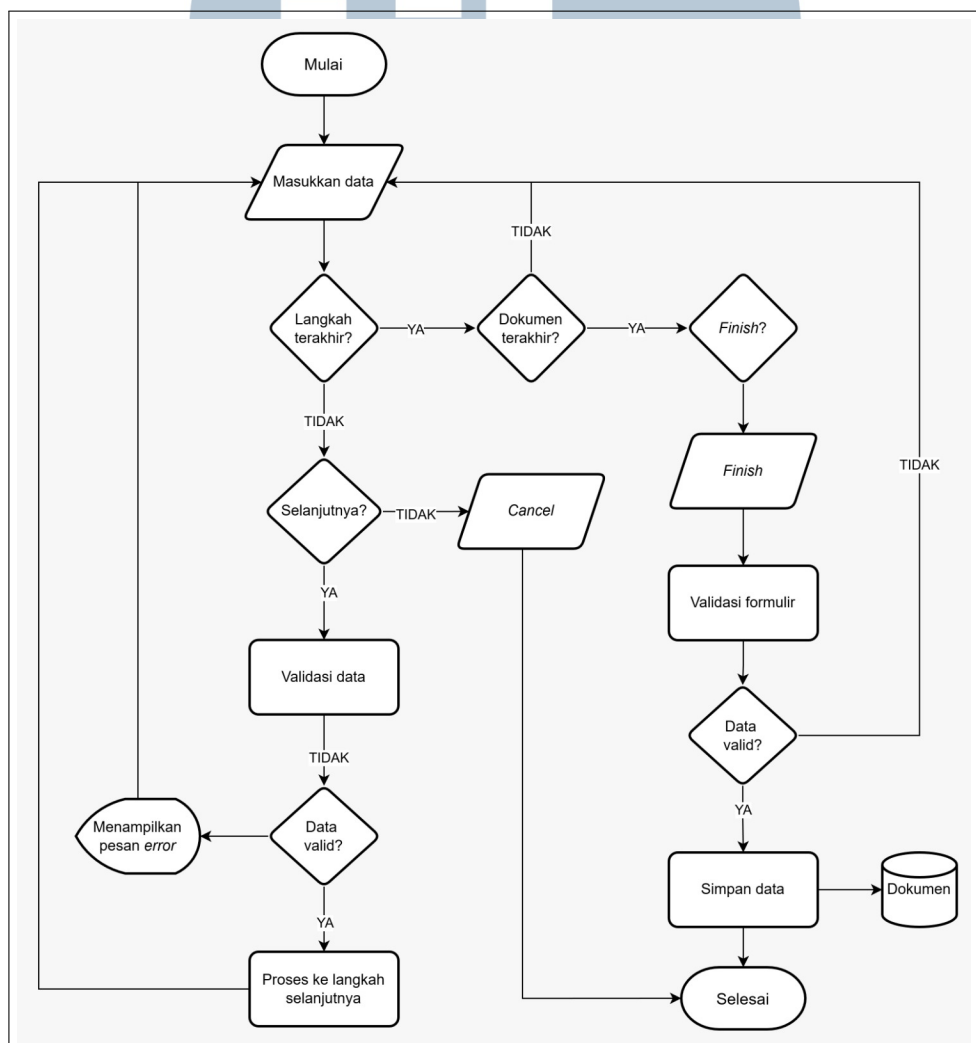


Gambar 3.6. Flowchart Sub-fitur Import

Flowchart *Bulk Import* pada Gambar 3.7 menggambarkan alur ketika pengguna mengimpor lebih dari satu dokumen dalam satu rangkaian proses. Setelah proses dimulai, sistem menampilkan langkah pengisian data untuk dokumen pertama. Setiap kali pengguna mengisi formulir, sistem memeriksa apakah bagian formulir yang sedang diisi merupakan langkah terakhir. Jika belum, sistem akan menampilkan bagian formulir berikutnya hingga seluruh informasi yang dibutuhkan untuk dokumen tersebut lengkap. Setelah itu, sistem menentukan apakah pengguna akan melanjutkan ke dokumen berikutnya atau membatalkan proses.

Jika pengguna melanjutkan, sistem melakukan validasi data untuk dokumen

yang sedang diproses. Data yang tidak valid akan memunculkan pesan kesalahan agar pengguna dapat memperbaiki *input* sebelum melanjutkan. Apabila data valid dan dokumen yang sedang diproses belum merupakan dokumen terakhir dalam kumpulan impor, sistem mengulangi alur pengisian dan validasi untuk dokumen berikutnya hingga seluruh dokumen memperoleh metadata yang lengkap. Setelah semua dokumen selesai diisi dan pengguna memilih untuk mengakhiri proses, sistem menjalankan validasi formulir akhir, menyimpan seluruh data dokumen ke basis data, lalu menutup proses pada kondisi selesai. Dengan alur ini, proses *Bulk Import* memastikan bahwa setiap dokumen yang diimpor secara massal tetap melalui tahapan validasi dan pencatatan yang terkontrol.

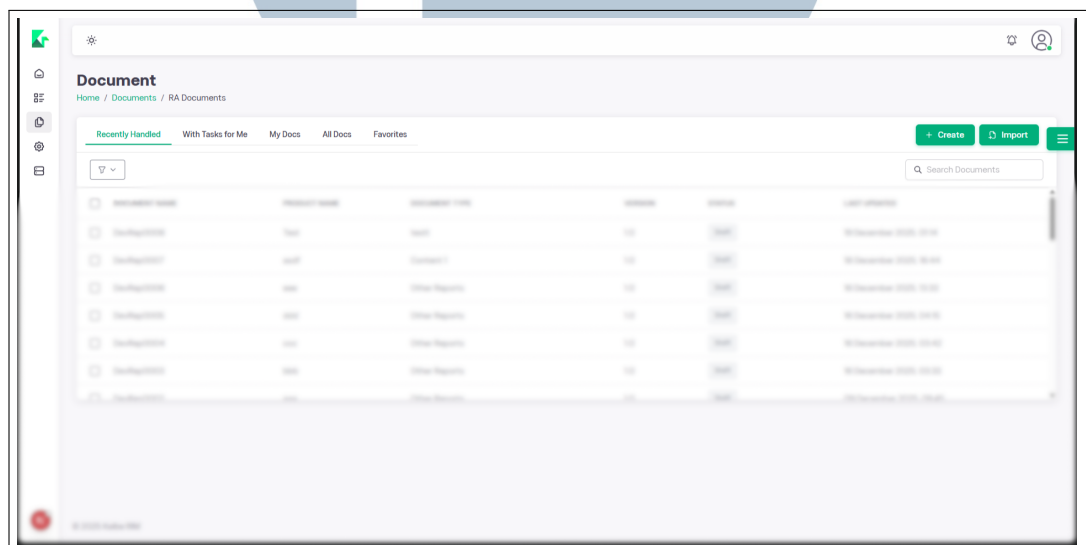


Gambar 3.7. Flowchart Sub-fitur Bulk Import

C Hasil Pembuatan Modul Documents

C.1 Menampilkan Data Dokumen

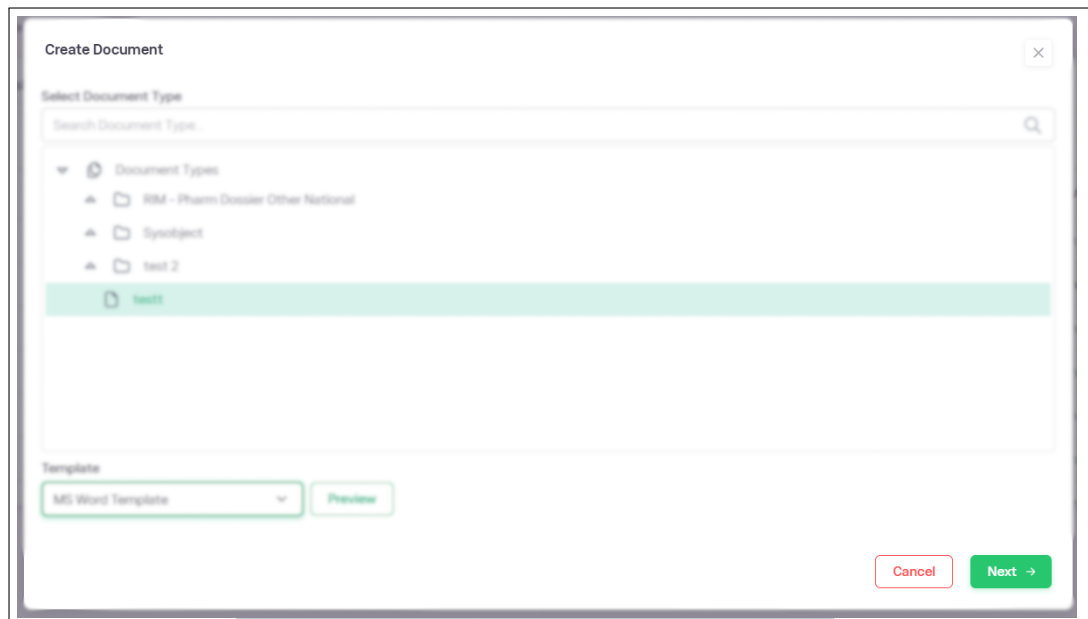
Fitur menampilkan data dokumen digunakan untuk menampilkan daftar dokumen yang telah dibuat pengguna dalam bentuk tabel yang terstruktur. Setiap baris tabel mewakili satu dokumen lengkap dengan informasi kunci yang diperlukan, sehingga memudahkan proses pemantauan dan pengelolaan dokumen di dalam sistem. Antarmuka halaman dilengkapi berbagai komponen pendukung, seperti *tabs* untuk memisahkan dokumen berdasarkan kategori, kolom pencarian untuk memfilter data dengan kata kunci, tombol "Create" untuk menambahkan dokumen baru, serta tombol "Import" untuk memasukkan dokumen yang sudah tersedia, sehingga keseluruhan proses pengelolaan dokumen menjadi lebih terpadu, efisien, tertata, dan mudah diakses oleh pengguna.



Gambar 3.8. Antarmuka Halaman Document

C.2 Pembuatan Dokumen

Fitur pembuatan dokumen memfasilitasi penyusunan dokumen baru melalui serangkaian langkah pengisian formulir yang tersusun secara berurutan. Gambar 3.9 menampilkan langkah awal, ketika pengguna diminta untuk melengkapi sejumlah data terlebih dahulu sebelum dapat berpindah ke tahap selanjutnya. Setelah tombol "Next" ditekan, sistem menjalankan proses validasi terhadap data yang telah di-*input* oleh pengguna.



Gambar 3.9. Antarmuka fitur Pembuatan Dokumen pada tahap pertama

Antarmuka pada Gambar 3.10 dan Gambar 3.11 menunjukkan tahap berikutnya, di mana pengguna mengisi metadata dasar dokumen serta parameter utama lain yang diperlukan sebagai identitas dokumen. Tahap ini membantu memastikan bahwa seluruh informasi pendukung sudah terekam sebelum dokumen masuk ke proses selanjutnya.

Gambar 3.10. Antarmuka fitur Pembuatan Dokumen pada tahap kedua bagian satu

Gambar 3.11. Antarmuka fitur Pembuatan Dokumen pada tahap kedua bagian dua

Gambar 3.12 merupakan tahap ketiga yang menampilkan pengaturan peran pengguna dalam siklus dokumen. Pada tahap ini, pengguna memilih individu yang akan mengisi masing-masing peran melalui daftar pengguna yang tersedia, sehingga alur *review* dan persetujuan dapat berjalan sesuai dengan matriks peran yang telah ditetapkan di sistem.

Gambar 3.12. Antarmuka fitur Pembuatan Dokumen pada tahap ketiga

Gambar 3.13 merupakan tahap akhir pembuatan dokumen yang menampilkan ringkasan dokumen beserta daftar *file* yang akan menjadi isi

dokumen. Dari layar ini, pengguna dapat mengatur relasi dokumen dan kemudian menekan tombol "Finish" untuk menyelesaikan proses pembuatan. Setelah langkah ini, dokumen baru tercatat di sistem dengan status awal sesuai pengaturan siklus hidup.

Create - Imported eCTD documents

Search Find Related Docs

NAME	TITLE	OWNER NAME	MODIFIED
DevReg001	Registration Dossier	John Doe Capuchino, S.Parm.	07/05/2025
DevReg001 CURRENT Effective ANDOTEN 30 MG FILM COATED TABLET			

Name	Size	Link Account	Type	Modified

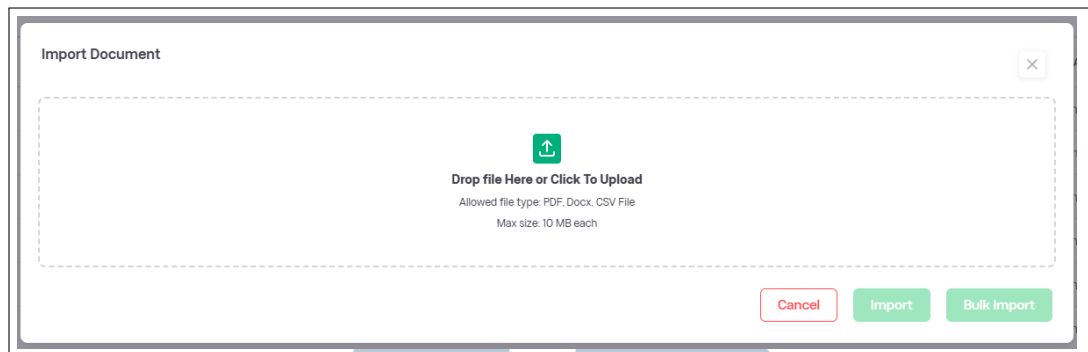
Relation Type: Custom Input Description: Description

Relation Direction: ← Previous Close Finish

Gambar 3.13. Antarmuka fitur Pembuatan Dokumen pada tahap keempat

C.2 Impor Dokumen

Fitur impor dokumen dijalankan melalui serangkaian langkah formulir yang alurnya serupa dengan proses pembuatan dokumen, namun dimulai dari berkas yang sudah tersedia. Pada langkah awal yang diperlihatkan pada Gambar 3.14, sistem menampilkan dialog *Import Document* yang memuat area unggah berkas. Pengguna dapat menyeret dan meletakkan (*drag and drop*) *file* ke area tersebut atau memilihnya langsung dari penyimpanan lokal, dengan jenis berkas yang didukung seperti PDF, DOCX, atau CSV serta batas ukuran tertentu untuk setiap *file*. Di bagian bawah dialog disediakan tombol "Import" untuk menjalankan impor satu dokumen, tombol "Bulk Import" untuk memproses banyak dokumen sekaligus, dan tombol "Cancel" apabila pengguna ingin menghentikan proses impor.



Gambar 3.14. Antarmuka fitur Impor dokumen pada tahap pertama

Pada langkah kedua yang diperlihatkan pada Gambar 3.15, sistem kembali menampilkan layar *Import Document* dengan formulir yang harus diisi oleh pengguna sebelum dapat melanjutkan ke tahap berikutnya. Setelah seluruh data yang dibutuhkan terisi, pengguna dapat menekan tombol "Next" untuk melanjutkan proses atau memilih tombol "Cancel" jika ingin menghentikan impor. Ketika tombol "Next" diklik, sistem akan menjalankan proses validasi terhadap data yang telah di-*input*.



Gambar 3.15. Antarmuka fitur Impor dokumen pada tahap kedua

Pada tahap selanjutnya, antarmuka tahap ketiga terlihat di Gambar 3.16 dan Gambar 3.17 menampilkan formulir metadata dasar. Pada layar ini, pengguna melengkapi informasi seperti yang diperlukan oleh sistem. Data ini digunakan

untuk mengklasifikasikan dokumen dan mengaitkannya dengan konteks regulatori yang tepat. Tombol "Next" di sudut kanan bawah digunakan untuk melanjutkan ke tahap berikutnya, sedangkan "Previous" dan "Cancel" memungkinkan pengguna meninjau atau membatalkan pengisian.

Import Document - [DRAFT V4] URS RM.pdf

General Product Info

Title*

Published Title*

Language Code* Document Code* Registration Office*

Document Type* Section Code*

Country* Company*

Description

← Previous Cancel Next →

Gambar 3.16. Antarmuka fitur Impor dokumen pada tahap ketiga bagian satu

Import Document - [DRAFT V4] URS RM.pdf

General **Product Info**

Product Area*

Generic Name*

Brand Name*

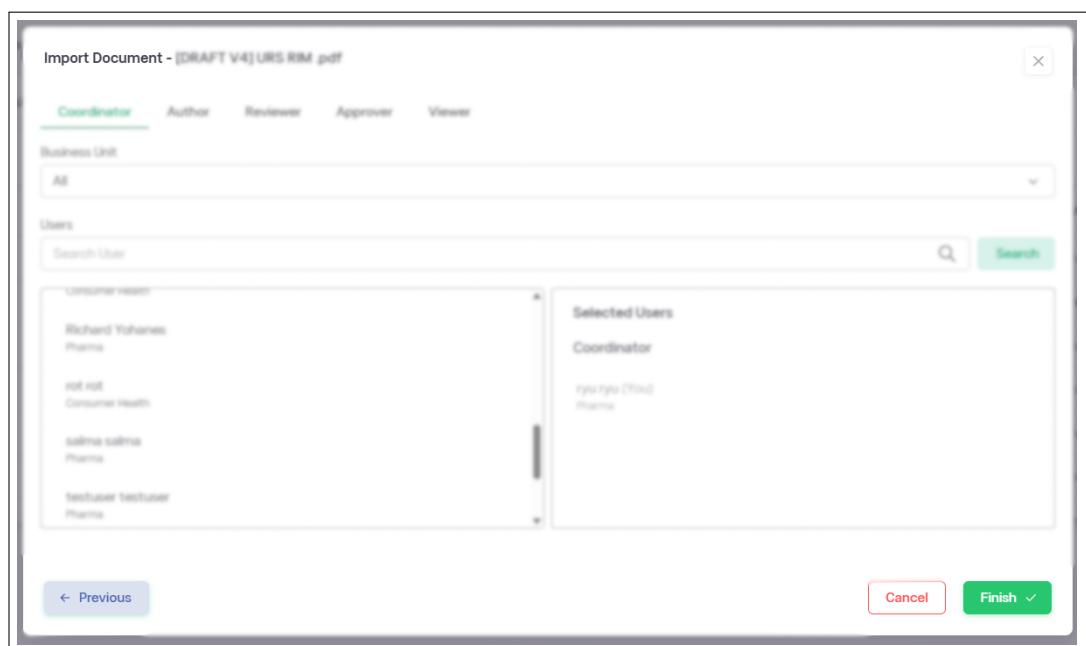
Document Date* mm/dd/yyyy

← Previous Cancel Next →

Gambar 3.17. Antarmuka fitur Impor dokumen pada tahap ketiga bagian dua

Gambar 3.18 merepresentasikan tahap ketiga pada proses impor yang berfokus pada penetapan peran pengguna terhadap dokumen impor. Pada tahap

ini, pengguna menentukan siapa saja yang akan mengemban peran tertentu dengan memilih pengguna dari daftar yang tersedia, sehingga proses peninjauan dan persetujuan dokumen hasil impor dapat berlangsung sesuai dengan matriks peran yang berlaku di dalam sistem.



Gambar 3.18. Antarmuka fitur Impor dokumen pada tahap keempat

3.3.2 Pembuatan Modul Dashboard

A User Requirement

Modul *Dashboard* berperan sebagai halaman beranda yang menyajikan ringkasan kondisi dan aktivitas terbaru dalam sistem RIM dalam satu tampilan terpusat. Melalui *dashboard*, pengguna dapat melihat status dokumen regulasi, distribusi jumlah dokumen pada setiap tahap siklus, serta aktivitas terkini yang berkaitan dengan tugas masing-masing, seperti permintaan *review* atau persetujuan. Informasi tersebut umumnya ditampilkan dalam bentuk kartu ringkasan, grafik, dan daftar tugas, sehingga pengguna lebih mudah mengenali pekerjaan yang perlu diprioritaskan dan dapat bertindak cepat tanpa harus membuka menu satu per satu. Berikut adalah beberapa fitur yang disediakan pada modul *Dashboard*.

A.1 Menampilkan Ringkasan Dokumen

Fitur ringkasan dokumen pada modul *Dashboard* berfungsi untuk menampilkan gambaran umum status dokumen regulasi dalam sistem secara cepat dan terstruktur. Informasi biasanya disajikan dalam bentuk kartu atau panel yang menampilkan jumlah dokumen pada tiap status sehingga pengguna dapat segera melihat sebaran dan progres dokumen tanpa membuka halaman daftar dokumen secara terperinci. Dengan adanya ringkasan ini, pengguna dapat lebih mudah memantau beban kerja, mengidentifikasi dokumen yang memerlukan tindakan lanjutan, serta memprioritaskan tugas harian berdasarkan kondisi terkini sistem.

A.2 Dashboard Settings

Dashboard Settings merupakan fitur yang memungkinkan pengguna menyesuaikan tampilan dan isi *dashboard* sesuai kebutuhan dan preferensi kerja masing-masing. Melalui fitur ini, pengguna dapat mengatur komponen apa saja yang ingin ditampilkan serta mengaktifkan atau menonaktifkan elemen tertentu agar informasi yang muncul lebih relevan dengan peran dan tanggung jawabnya.

Selain itu, *Dashboard Settings* umumnya menyediakan opsi penyimpanan konfigurasi sebagai *personal view*, sehingga setiap pengguna dapat memiliki tampilan *dashboard* yang konsisten setiap kali mengakses sistem, tanpa memengaruhi tampilan pengguna lain. Hal ini membantu meningkatkan efisiensi kerja karena pengguna dapat segera fokus pada indikator dan tugas yang paling penting bagi aktivitas hariannya.

A.3 Detail Panel

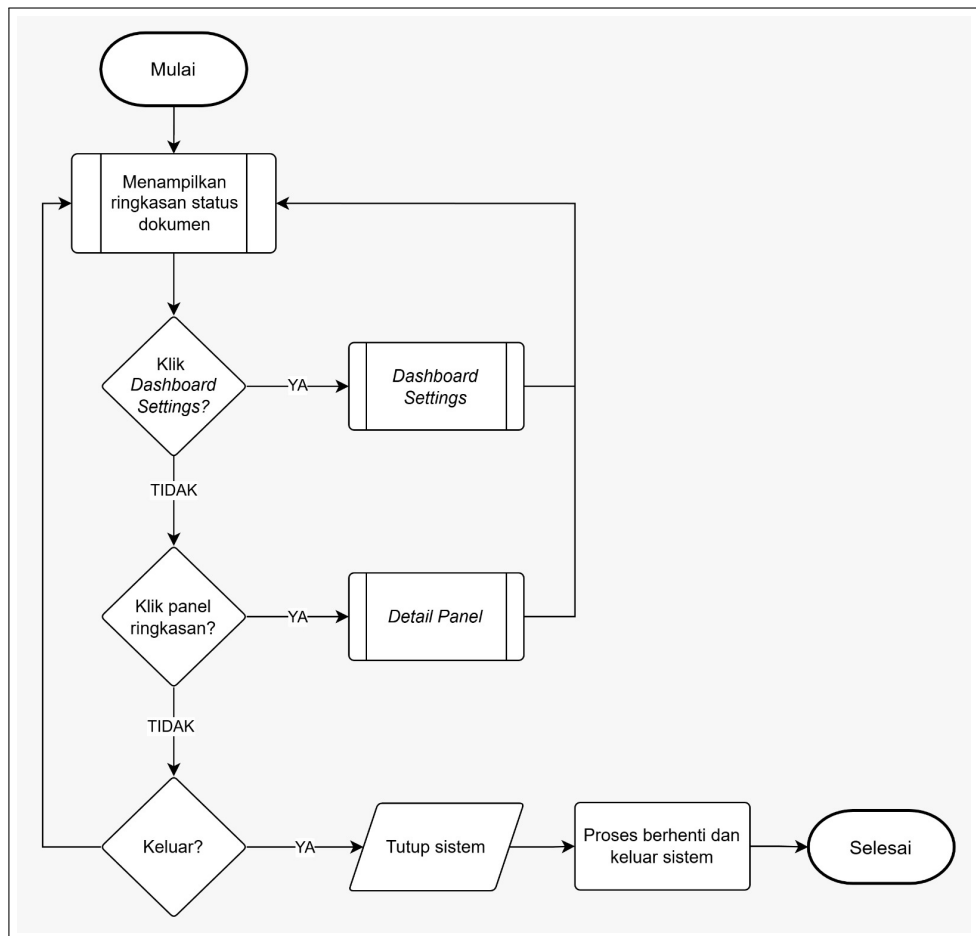
Fitur Detail Panel menyediakan tampilan rincian daftar dokumen secara instan berdasarkan kategori yang dipilih pengguna pada *dashboard*. Saat sebuah panel ringkasan diklik, sistem menampilkan jendela pop up berisi tabel dokumen yang telah terfilter sesuai kategori tersebut dan memuat informasi dasar tiap dokumen. Dari tampilan ini, pengguna dapat langsung meninjau, memilih, dan mengakses dokumen yang diperlukan untuk dikelola lebih lanjut tanpa harus kembali terlebih dahulu ke halaman utama daftar dokumen.

B Flowchart Modul Dashboard

Pada modul *Dashboard*, alur kerja utama digambarkan pada Gambar 3.19 yang menunjukkan rangkaian proses penayangan dan interaksi pengguna terhadap ringkasan status dokumen. Proses dimulai dari subproses penampilan ringkasan pada halaman *dashboard*. Sistem menyajikan informasi utama dalam bentuk panel-panel ringkasan. Setelah tampilan ringkasan muncul, pengguna dapat memilih langkah berikutnya sesuai kebutuhan.

Jika diperlukan penyesuaian konfigurasi tampilan, pengguna dapat mengakses subproses *Dashboard Settings* untuk mengatur elemen ringkasan yang ingin ditampilkan, sebagaimana terlihat pada Gambar 3.20. Selanjutnya, pengguna dapat melihat daftar dokumen secara lebih detail melalui subproses Detail Panel yang aktif ketika salah satu panel ringkasan diklik dan menampilkan tabel dokumen terkait. Apabila pengguna memutuskan untuk mengakhiri sesi, sistem akan menjalankan proses keluar dengan menutup modul dan menghentikan aktivitas yang sedang berlangsung. Secara keseluruhan, alur ini menunjukkan hubungan antar proses utama pada modul *Dashboard* yang bersama-sama mendukung pemantauan, pengaturan tampilan, dan penelusuran detail dokumen secara sistematis.





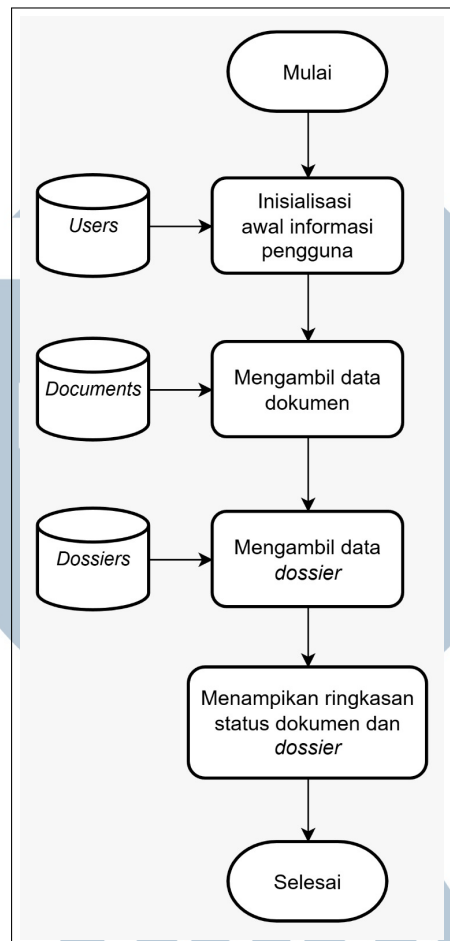
Gambar 3.19. Flowchart Modul Dashboard

Berikut adalah uraian subproses untuk masing-masing fitur yang terdapat pada modul *Dashboard*.

B.1 Menampilkan Ringkasan Dokumen

Fitur menampilkan ringkasan dokumen dan *dossier* digunakan untuk menampilkan informasi agregat kepada pengguna berdasarkan data yang tersimpan di basis data. Gambar 3.20 menunjukkan alur kerja dari fitur tersebut. Proses dimulai dengan pengambilan informasi awal pengguna dari tabel *Users* sebagai dasar penentuan hak akses dan konteks tampilan. Setelah itu, sistem menarik data dokumen dari tabel *documents* serta data *dossier* dari tabel *dossiers* untuk kemudian diolah. Ketika seluruh data telah berhasil dikompilasi dan diproses, sistem menyajikan ringkasan status dokumen dan *dossier* pada *dashboard* dalam bentuk panel-panel ringkasan yang tertata, sehingga pengguna dapat dengan cepat

memperoleh gambaran umum terhadap kondisi dokumen dan *dossier* yang terkelola di sistem.



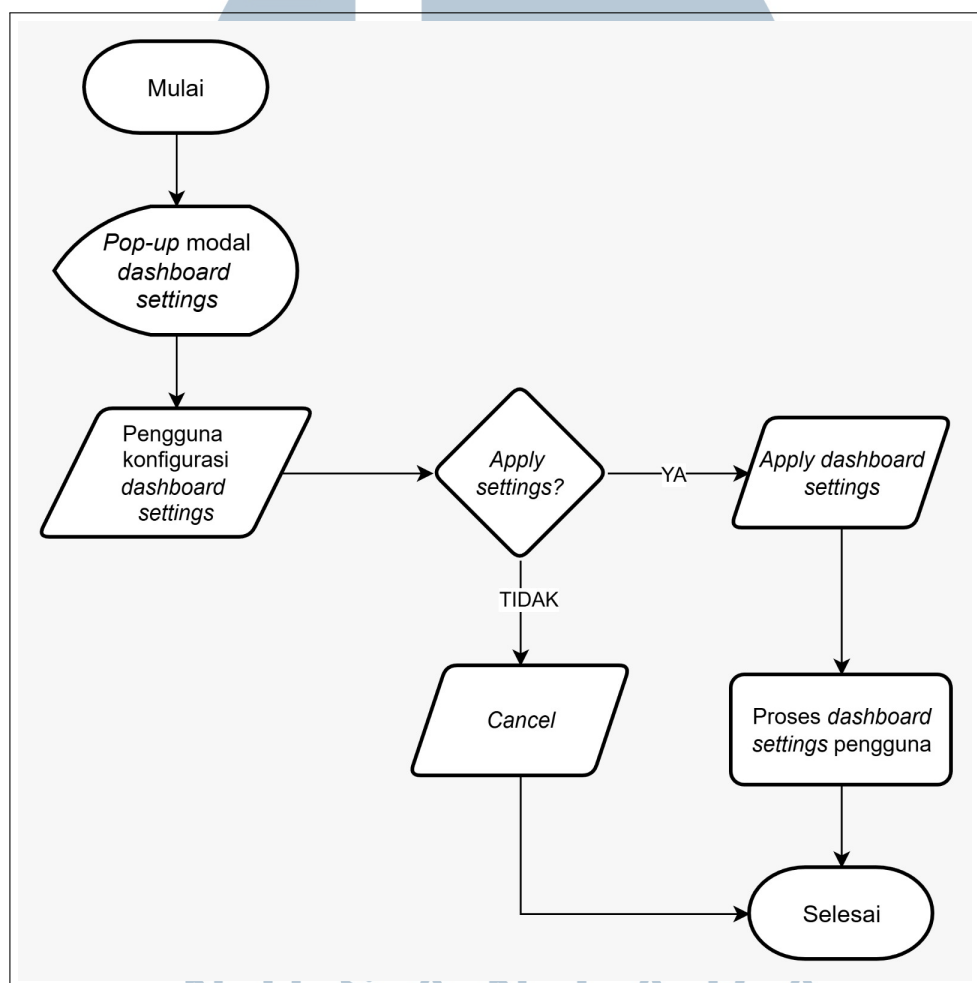
Gambar 3.20. Flowchart Ringkasan

B.2 Dashboard Settings

Pada modul *Dashboard*, alur kerja fitur *Dashboard Settings* digambarkan pada Gambar 3.21 yang menunjukkan urutan proses pengaturan dan penerapan konfigurasi tampilan *dashboard* oleh pengguna. Proses dimulai ketika sistem menampilkan pop up modal *Dashboard Settings* kepada pengguna yang berisi berbagai opsi konfigurasi elemen ringkasan dan preferensi tampilan lain yang dapat disesuaikan. Setelah pop up ditampilkan, pengguna melakukan konfigurasi *Dashboard Settings* sesuai kebutuhan, misalnya memilih panel ringkasan mana yang akan ditampilkan pada *dashboard*.

Berikutnya, sistem menampilkan keputusan apakah pengaturan yang telah

diubah akan diterapkan. Apabila pengguna memilih untuk menerapkan pengaturan, sistem menjalankan proses *apply dashboard settings* dan kemudian memproses serta menyimpan konfigurasi *Dashboard Settings* pengguna sebagai preferensi tampilan yang baru. Sebaliknya, jika pengguna memutuskan untuk tidak menerapkan pengaturan, sistem menjalankan proses "Cancel" yang menutup pop up tanpa menyimpan perubahan konfigurasi. Setelah salah satu jalur tersebut selesai dijalankan, proses pengaturan *Dashboard Settings* pada modul *Dashboard* dinyatakan berakhir.



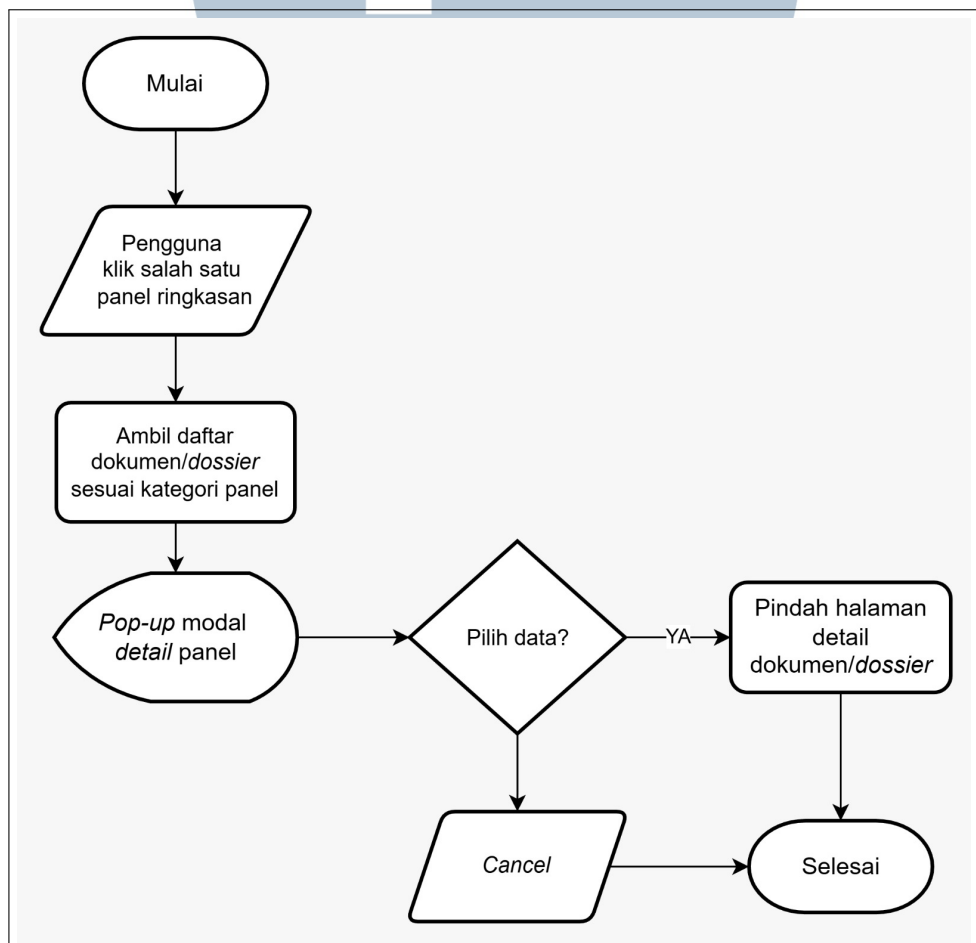
Gambar 3.21. Flowchart Dashboard Settings

B.3 Detail Panel

Pada modul *Dashboard*, alur kerja fitur Detail Panel digambarkan pada Gambar 3.22 yang menunjukkan urutan proses penayangan daftar dokumen atau *dossier* secara rinci berdasarkan kategori panel ringkasan yang dipilih pengguna.

Proses diawali ketika pengguna memilih salah satu panel ringkasan pada *dashboard*. Setelah itu, sistem mengambil daftar dokumen atau *dossier* yang sesuai dengan kategori panel tersebut dari basis data, kemudian menampilkan hasilnya dalam bentuk pop up modal Detail Panel yang berisi daftar data yang relevan.

Pada tahap berikutnya, pengguna dapat menentukan apakah akan menelusuri informasi lebih lanjut dari salah satu entri yang tampil. Jika pengguna memilih satu data pada daftar, sistem akan mengarahkan ke halaman detail dokumen atau *dossier* untuk menampilkan informasi yang lebih lengkap. Sebaliknya, jika pengguna menutup tampilan tanpa melakukan pemilihan, proses dihentikan tanpa perubahan halaman. Setelah salah satu dari kedua kemungkinan tersebut terjadi, interaksi pengguna dengan fitur Detail Panel untuk penelusuran rinci dokumen atau *dossier* dianggap telah berakhir.



Gambar 3.22. Flowchart Detail Panel

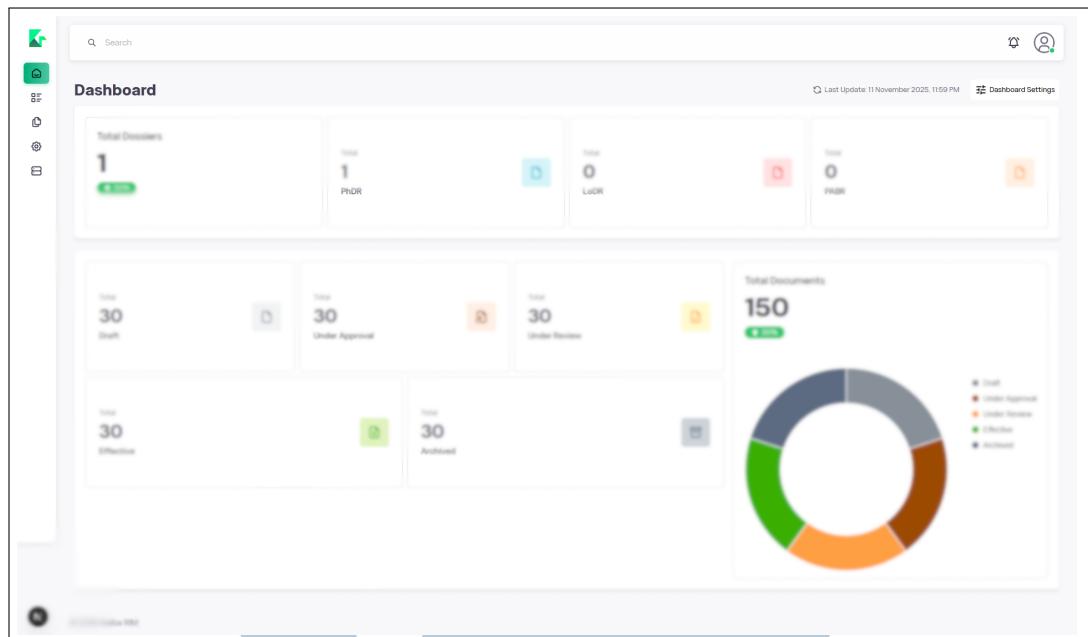
C Hasil Pembuatan Modul Dashboard

C.1 Menampilkan Ringkasan Dokumen

Fitur menampilkan ringkasan dokumen berfungsi untuk menampilkan ringkasan dokumen dan *dossier* dalam satu tampilan terpadu sehingga pengguna dapat memperoleh gambaran umum kondisi data secara cepat. Antarmuka dari fitur ini ditunjukkan pada Gambar 3.23 yang memperlihatkan beberapa panel ringkasan dan grafik distribusi dokumen pada halaman utama *dashboard*. Pada bagian atas, terdapat deretan panel ringkasan untuk total *dossier* beserta kategorinya yang menampilkan angka total dan indikator tren perubahan sehingga pengguna dapat memantau jumlah *dossier* yang tercatat di dalam sistem secara keseluruhan. Di bawahnya, disajikan panel-panel ringkasan status dokumen dengan nilai total yang merepresentasikan jumlah dokumen pada setiap status.

Selain panel angka, *dashboard* juga dilengkapi dengan grafik berbentuk diagram donat yang menggambarkan proporsi distribusi dokumen berdasarkan statusnya, sehingga informasi menjadi lebih mudah dipahami secara visual. Di bagian kanan atas halaman, tersedia informasi waktu pembaruan terakhir (*Last Update*) yang membantu pengguna mengetahui kekinian data yang ditampilkan, serta tombol "Dashboard Settings" yang dapat digunakan untuk mengatur konfigurasi tampilan panel ringkasan sesuai preferensi masing-masing pengguna. Secara keseluruhan, desain antarmuka *dashboard* ini mendukung aktivitas pemantauan dan evaluasi kondisi dokumen dan *dossier* secara ringkas, informatif, dan mudah diakses dari satu halaman utama.

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A



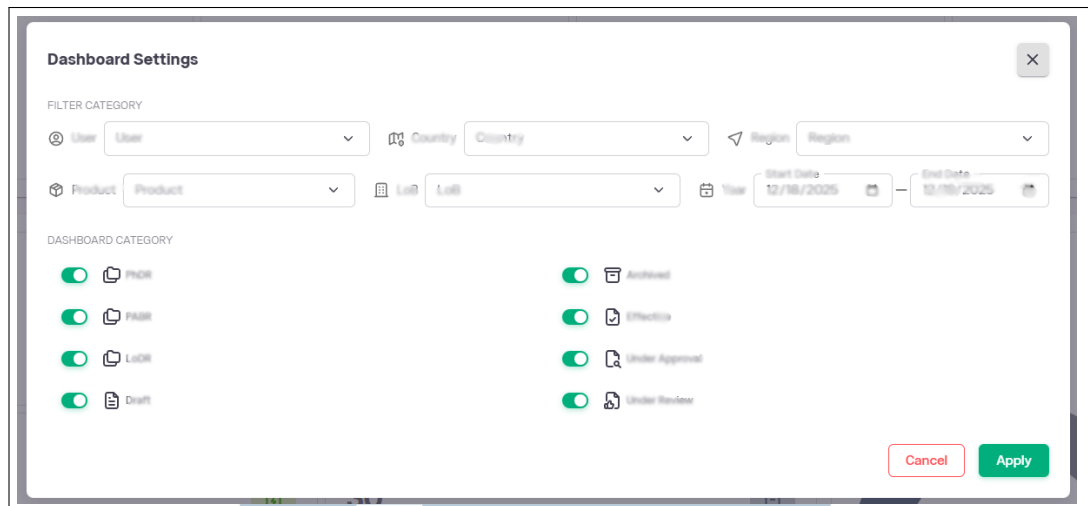
Gambar 3.23. Antarmuka Ringkasan Dokumen

C.2 Dashboard Settings

Fitur *Dashboard Settings* berfungsi untuk mengatur konfigurasi tampilan *dashboard* berdasarkan preferensi dan kebutuhan pengguna. Antarmuka dari fitur ini ditunjukkan pada Gambar 3.24 yang menampilkan sebuah pop up berisi pengaturan kategori filter dan kategori panel ringkasan *dashboard*. Pada bagian *Filter Category*, pengguna dapat memilih beberapa parameter filter yang bertujuan untuk membatasi data dokumen dan *dossier* yang ditampilkan pada *dashboard* sesuai konteks yang diinginkan. Setiap filter disajikan dalam bentuk komponen pilihan (*dropdown*) sehingga pengaturan dapat dilakukan dengan mudah dan konsisten.

Pada bagian *dashboard category*, sistem menyediakan deretan sakelar (*toggle*) untuk mengaktifkan atau menonaktifkan kategori panel ringkasan tertentu, seperti untuk kategori *dossier* atau status dokumen. Melalui pengaturan ini, pengguna dapat menentukan panel mana saja yang akan ditampilkan pada halaman *dashboard*, sehingga tampilan menjadi lebih relevan dan tidak berlebihan. Di sisi kanan bawah antarmuka, terdapat tombol "Cancel" untuk membatalkan perubahan dan menutup pop up tanpa menyimpan pengaturan, serta tombol "Apply" untuk menerapkan konfigurasi yang telah dipilih sebagai preferensi tampilan baru pada *dashboard*. Secara keseluruhan, rancangan antarmuka *Dashboard Settings* ini

mendukung proses kustomisasi tampilan *dashboard* agar lebih fleksibel, terfokus, dan selaras dengan kebutuhan pemantauan pengguna.

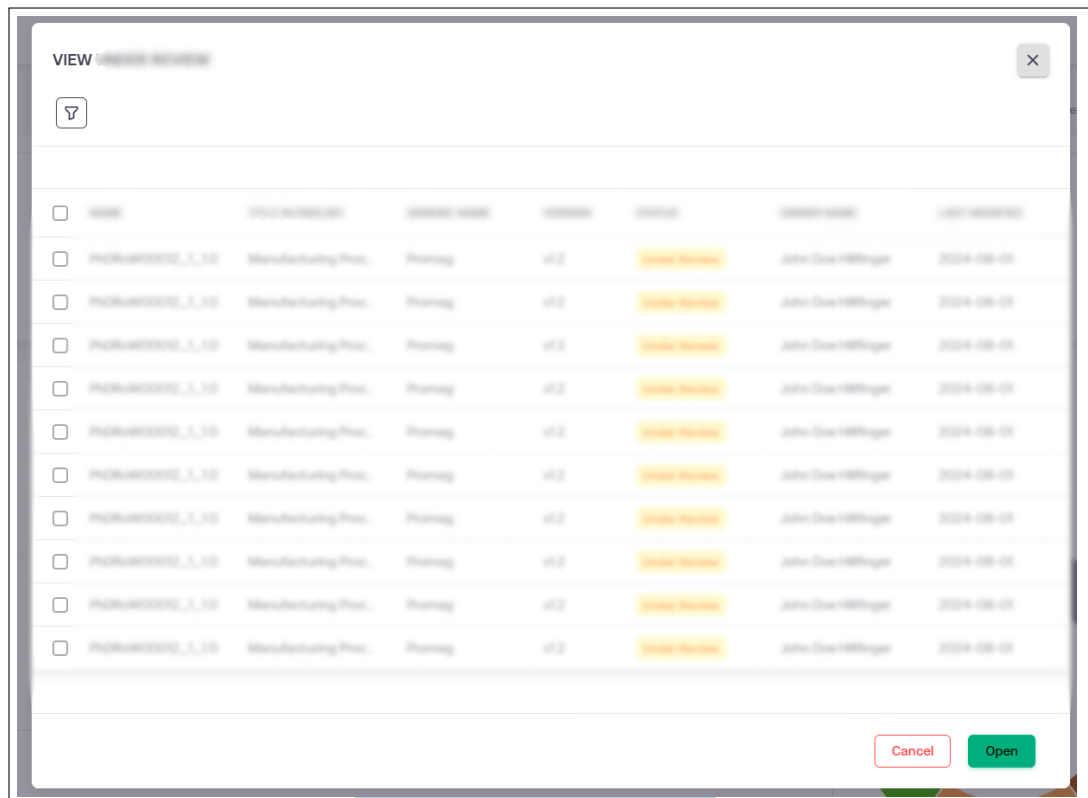


Gambar 3.24. Antarmuka Dashboard Settings

C.3 Detail Panel

Fitur Detail Panel berfungsi untuk menampilkan daftar dokumen secara rinci berdasarkan panel ringkasan yang dipilih pengguna pada halaman *dashboard*. Antarmuka dari fitur ini ditunjukkan pada Gambar 3.25 yang memperlihatkan sebuah pop up berisi tabel daftar dokumen atau *dossier* dengan kategori atau status yang sesuai pilihan pengguna. Pada bagian *header* tabel ditampilkan beberapa kolom informasi penting, sehingga setiap baris merepresentasikan satu entitas dokumen beserta metadata utamanya.

Pada sisi kiri tiap baris tabel tersedia kotak centang (*checkbox*) yang memungkinkan pengguna menandai satu atau beberapa dokumen sekaligus untuk diproses lebih lanjut. Di bagian bawah jendela pop up ditampilkan dua tombol, yaitu "Cancel" untuk menutup tampilan tanpa membuka detail, dan "Open" untuk mengakses halaman detail dokumen yang telah dipilih. Dengan desain antarmuka tersebut, proses meninjau serta memilih dokumen dapat dilakukan secara lebih terstruktur, mudah dipahami, dan efisien.



Gambar 3.25. Antarmuka Detail Panel

3.4 Kendala dan Solusi yang Ditemukan

Selama pelaksanaan magang di PT Kalbe Farma Tbk, terdapat beberapa tantangan yang memengaruhi kelancaran pengerjaan tugas. Berikut adalah ringkasan kendala serta pendekatan solusi yang diterapkan untuk mengatasinya.

3.4.1 Kendala

1. Komunikasi dan koordinasi antar tim tidak selalu berjalan mulus karena perbedaan prioritas pekerjaan dan keterbatasan waktu, sehingga sesi diskusi kebutuhan dan penyesuaian fitur terkadang belum dapat dilakukan secara maksimal.
2. Kebutuhan pengguna (*user requirement*) mengalami perubahan beberapa kali sepanjang proses pengembangan, sehingga rancangan awal dan implementasi fitur yang sudah dikerjakan perlu mengalami penyesuaian kembali.

3.4.2 Solusi

1. Kolaborasi lintas fungsi kemudian dioptimalkan melalui penjadwalan ulang pertemuan rutin dengan agenda yang lebih jelas dan terstruktur, disertai pencatatan notulen serta penyimpanan rekaman rapat pada sistem dokumentasi internal, sehingga alur informasi tetap berkesinambungan meskipun tidak semua anggota tim dapat hadir dalam setiap pertemuan.
2. Perubahan pada *user requirement* dikelola dengan menerapkan mekanisme *change control*, antara lain melalui pembaruan berkala dokumen URS, penetapan prioritas fitur yang perlu dikerjakan terlebih dahulu, serta penyesuaian jadwal dan rencana kerja setiap kali terdapat perubahan yang berdampak besar, sehingga proses pengembangan tetap terarah dan dapat dikendalikan.

