

BAB III

PEMBAHASAN

3.1 Tahap Perencanaan

Tahap perencanaan diawali dengan analisa, baik analisa pada proses bisnis maupun kebutuhan. Analisis proses bisnis bertujuan untuk memahami alur kerja yang mengadaptasi proses bisnis di lapangan, yang selanjutnya akan diimplementasikan ke aplikasi SMARD. Proses analisis ini diharapkan menghasilkan sistem yang dapat meningkatkan efisiensi dalam pencatatan, pelacakan, dan evaluasi arahan yang diberikan oleh Dirjen. Sedangkan analisis kebutuhan bertujuan untuk mengidentifikasi kebutuhan pengguna dan bagaimana sistem dapat memenuhi tersebut. Analisis tersebut akan didetailkan pada sub bab di bawah ini.

3.1.1 Kebutuhan Fungsional

Kebutuhan fungsional meliputi deskripsi layanan, fitur, atau fungsi yang disediakan atau diberikan oleh aplikasi SMARD bagi penggunanya.

Tabel III.1 Kebutuhan Fungsional

Fungsi	Deskripsi
Merekam arahan Dirjen	Analisis Tata Usaha Direktur Jenderal (selanjutnya disebut Analis) dapat melakukan perekaman terkait agenda pertemuan/rapat, baik jenis kegiatan, tema kegiatan, tanggal, dan nomor undangan kegiatan tersebut. Selanjutnya Analis merekam arahan yang disampaikan oleh Dirjen dalam kegiatan tersebut beserta usulan Unit Penanggung Jawab arahan, tanggal mulai dan tanggal akhir tindak lanjut.
Persetujuan rekaman arahan Dirjen	Kepala Subbagian Tata Usaha Direktur Jenderal (selanjutnya disebut Kasubbag TUD) dapat melakukan verifikasi dan persetujuan rekaman arahan yang telah diajukan oleh Analis. Apabila ada hal yang tidak sesuai, rekaman dikembalikan ke Analis untuk kemudian dihapus. Apabila terdapat beberapa isian yang kurang sesuai, Kasubbag TUD dapat melakukan perbaikan langsung atau dikembalikan ke Analis. Terakhir, apabila seluruh

Fungsi	Deskripsi
	isian telah sesuai, maka arahan dapat diteruskan ke Unit Penanggung Jawab.
Menindaklanjuti arahan Dirjen	PIC Unit Eselon II dapat merekam tindak lanjut dan melampirkan dokumen pendukung apabila tersedia dari arahan sesuai tenggat waktu yang telah ditentukan sebelumnya oleh Subbagian TUD.
Pemeriksaan tindak lanjut arahan Dirjen	Analisis melakukan pemeriksaan kelengkapan isian tindak lanjut dan dokumen pendukung apabila tersedia dari tindak lanjut yang diajukan oleh PIC Unit Eselon II. Apabila terdapat ketidaksesuaian secara administratif, Analisis mengembalikan status arahan ke PIC Unit Eselon II. Apabila telah sesuai, status arahan diteruskan ke Kasubbag TUD untuk diverifikasi dan dilakukan persetujuan apabila hasil pemeriksaan telah sesuai.
Monitoring tindak lanjut arahan Dirjen	Analisis, Kasubbag TUD, dan Unit Kepatuhan Internal dapat melakukan pemantauan terkait status arahan dan hasil tindak lanjut yang telah direkam oleh PIC Unit Eselon II.
Pengaturan hak akses PIC Unit Eselon II	Unit Kepatuhan Internal dapat melakukan pengaturan terkait hak akses pengguna bagi PIC Unit Eselon II. Akses tersebut perlu dibatasi mengingat informasi yang terdapat pada aplikasi SMARD bersifat terbatas.

3.1.2 Kebutuhan Non-Fungsional

Kebutuhan non-fungsional merupakan karakteristik yang dimiliki aplikasi SMARD terkait dengan operasional, kinerja, keamanan dan budaya.

Tabel III.2 Kebutuhan Non-Fungsional

Karakteristik	Deskripsi
Operasional	a. Aplikasi dapat dijalankan pada berbagai <i>browser</i> . b. Dijalankan dalam lingkungan sistem CEISA 4.0.
Kinerja	a. Aplikasi mampu diakses oleh minimal 100 pengguna dalam waktu bersamaan. b. Aplikasi mampu digunakan dalam 24 jam/hari, 365 hari/tahun.
Keamanan	a. Aplikasi dapat diakses melalui intranet oleh pegawai DJBC. b. Pengguna hanya dapat mengakses menu sesuai dengan hak akses yang diberikan.
Budaya dan Politik	Antisipasi kebutuhan kolaborasi dan pertukaran data dengan <i>common applications</i> Kementerian Keuangan lainnya, misalkan HRIS E-Prime (<i>Human Resources Information System</i>).

3.2 Tahap Pemodelan Sistem

Tahap pemodelan sistem merupakan fase yang berfokus pada proses penerjemahan kebutuhan yang telah dianalisis ke dalam bentuk rancangan sistem yang

lebih terstruktur. Seperti dilakukan pendefinisian kebutuhan sistem, perancangan arsitektur sistem, alur proses bisnis, serta hubungan antar komponen.

3.2.1 Tahap Analisis

Fokus utama dari tahap ini adalah mengidentifikasi serta mendefinisikan kebutuhan secara detail, dilakukan melalui pengumpulan data maupun studi dokumen untuk memahami permasalahan dan menentukan solusi yang dibutuhkan. Hasil dari tahap ini adalah spesifikasi kebutuhan sistem yang terstruktur, yang mencakup fungsi utama, alur proses bisnis, serta batasan-batasan yang harus dipenuhi oleh sistem.

3.2.1.1 Skema Kebutuhan Sistem

Skema kebutuhan sistem merupakan gambaran atau struktur yang menjelaskan komponen-komponen teknis dan kebutuhan sistem dalam pengembangan aplikasi. Skema ini menunjukkan bahwa arsitektur sistem dibangun secara modular, *scalable*, dan modern, dengan memanfaatkan kombinasi teknologi. Berikut adalah tabel rincian skema kebutuhan sistem pada SMARD.

Tabel III.3 Skema Kebutuhan

Skema		Teknologi
Backend	Web Service (RESTful)	a. Web service harus menyediakan semua <i>endpoint</i> yang mampu melayani kebutuhan pada aplikasi <i>frontend</i> b. Harus memperhatikan <i>high-performance</i> arsitektur RESTful API c. Dibangun dengan menggunakan bahasa pemrograman Java Spring Boot d. Aplikasi harus siap dipublikasikan ke <i>cluster</i> (<i>cluster-ready</i>)
	API Gateway	c. Menyediakan portal manajemen API d. Menyediakan portal API <i>access management</i> e. Menyediakan fitur pengelolaan API yang dapat mengakomodir <i>Health-Check</i>, <i>Analytics</i> dan <i>Scalable</i>

Skema		Teknologi
Frontend	Web-based application	a. Desain harus memperhatikan konsep <i>responsive-website</i> b. Dibangun dengan menggunakan <i>modern web framework</i> , misalkan: ReactJS c. Mendukung <i>reusable component frontend</i> d. Mendukung <i>progressive web app</i>

3.2.1.2 Spesifikasi Aktor

Berdasarkan analisis kebutuhan fungsional, terdapat 4 (empat) aktor dalam proyek ini, yaitu: Analisis TUD, Kasubbag TUD, PIC Unit Eselon II/III, dan Unit Kepatuhan Internal; yang berinteraksi langsung dengan *Graphic User Interface* (GUI). Selain itu, aplikasi SMARD juga berinteraksi dengan sistem dan modul data pokok pegawai, dan sistem lainnya yang berkomunikasi dengan menggunakan API. Berikut adalah penjabaran dari masing-masing aktor.

Tabel III.4 Spesifikasi Aktor

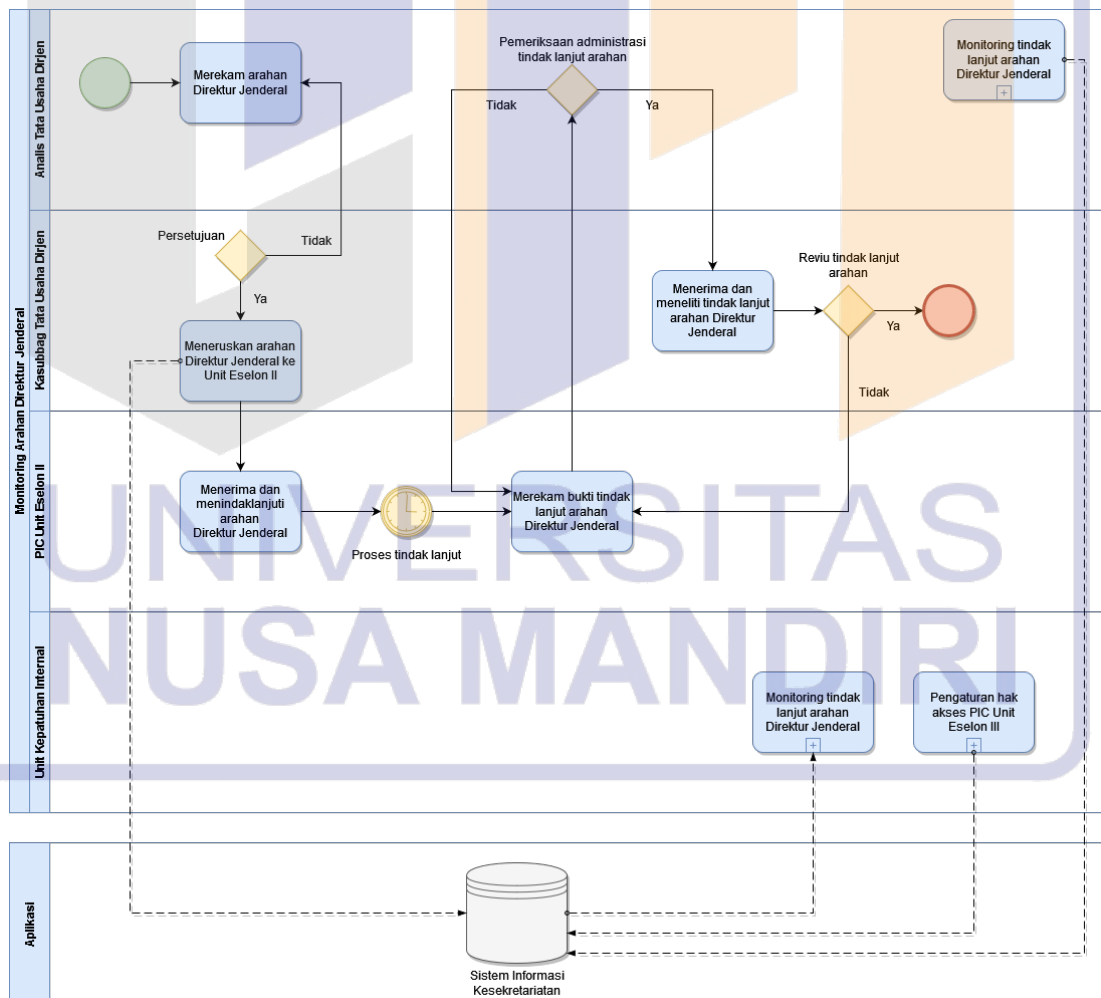
Aktor	Deskripsi
Analisis TUD	Aktor dapat merekam arahan, memeriksa tindak lanjut arahan, dan memantau tindak lanjut arahan Dirjen.
Kasubbag TUD	Aktor dapat melakukan persetujuan rekaman arahan, memeriksa tindak lanjut arahan, dan memantau tindak lanjut arahan Dirjen.
PIC Unit Eselon II/III	Aktor dapat merekam tindak lanjut arahan dan melampirkan dokumen pendukung apabila tersedia.
Unit Kepatuhan Internal	Aktor dapat memantau status arahan dan hasil tindak lanjut yang direkam oleh PIC Unit Eselon II, serta mengatur hak akses pengguna.

3.2.2 Tahap Desain

Kebutuhan yang telah terdefinisi akan diterjemahkan ke dalam bentuk rancangan teknis yang lebih konkret. Rancangan ini dapat meliputi diagram alur (*flowchart*), diagram penggunaan (*use case diagram*), rancangan *database*, hingga desain antarmuka (*mock-up*). Tujuan dari tahap desain adalah memberikan gambaran yang jelas mengenai bagaimana sistem akan dibangun dan dijalankan secara teknis, sehingga dapat menjadi pedoman yang lengkap bagi tahap konstruksi selanjutnya.

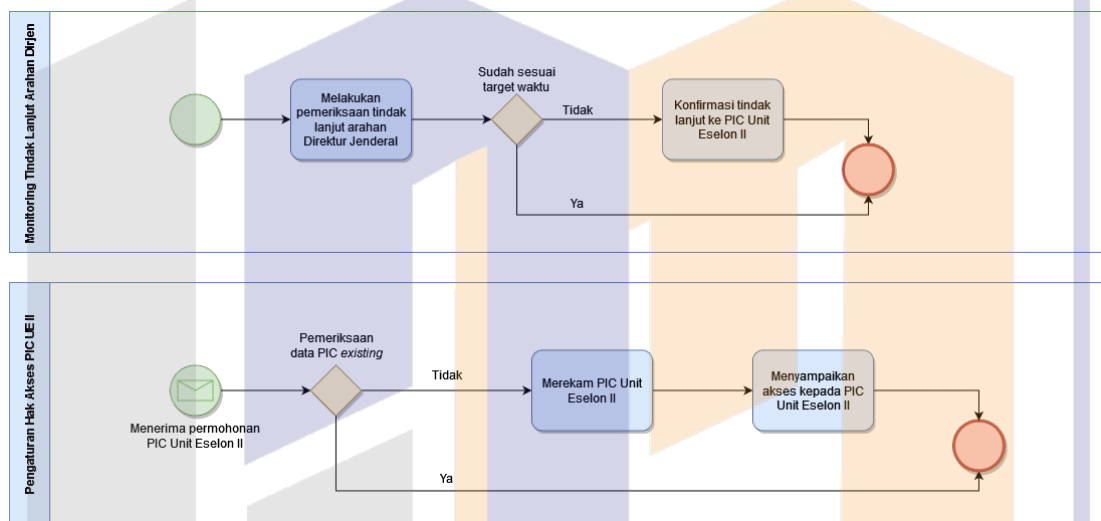
3.2.2.1 Diagram Alur (*Flowchart*)

Proses bisnis arahan Direktur Jenderal diawali oleh Analis Tata Usaha Dirjen yang merekam arahan ke dalam sistem. Arahan tersebut diteruskan oleh Kasubag Tata Usaha Dirjen kepada Unit Eselon II untuk ditindaklanjuti. Hasil tindak lanjut kemudian direkam sebagai bukti. Selanjutnya, PIC Unit Eselon III menerima arahan, melakukan tindak lanjut, serta mencatat bukti pelaksanaannya. Proses ini dimonitor oleh Unit Keasistenan Internal yang juga mengatur hak akses PIC. Seluruh alur didukung oleh aplikasi yang menjadi media pencatatan, tindak lanjut, *monitoring*, dan pengaturan akses sehingga arahan Direktur Jenderal dapat terdokumentasi dan terkontrol dengan baik.



Gambar III.1 Flowchart Fungsi Utama SMARD

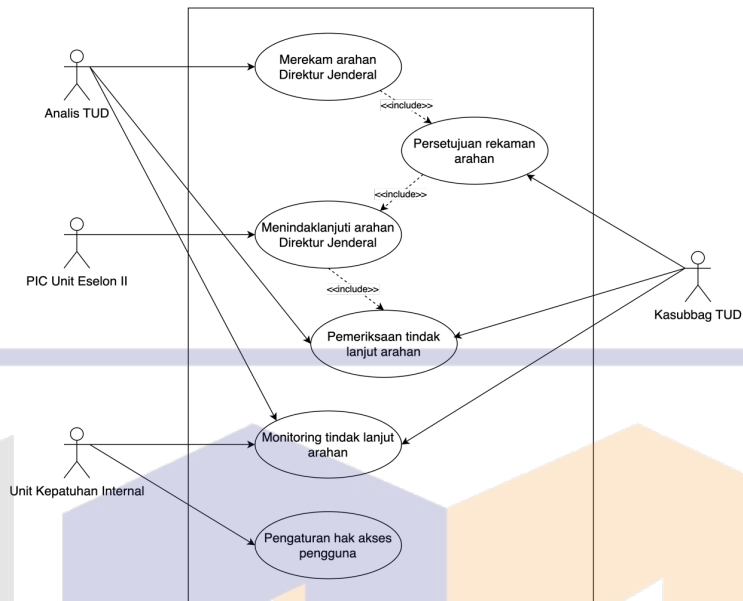
Proses *monitoring* tindak lanjut arahan Dirjen dilakukan dengan memeriksa hasil tindak lanjut yang telah dikerjakan, kemudian memberikan konfirmasi kepada PIC Unit Eselon II sebagai bentuk validasi. Sementara itu, pada proses pengaturan hak akses PIC Unit Eselon II, sistem terlebih dahulu merekam data PIC yang ditunjuk, kemudian menyampaikan hak akses agar PIC dapat melaksanakan tindak lanjut sesuai kewenangannya.



Gambar III.2 Flowchart Menu Monitoring dan Setting Hak Akses

3.2.2.2 Diagram Penggunaan (*Use Case Diagram*)

Use case diagram merupakan salah satu jenis *Unified Modelling Language* (UML) yang digunakan untuk menggambarkan interaksi pengguna dengan sistem. Pada bagian ini menjelaskan *use case diagram* pada aplikasi SMARD dan deskripsi masing-masing *use case* dari gambar di bawah ini.



Gambar III.3 Use Case Diagram SMARD

Berikut adalah detail penjelasan masing-masing *use case*, mulai dari aktor, uraian, kondisi awal, kondisi akhir, alur utama, alur alternatif, dan juga catatan.

Tabel III.5 Deskripsi Use Case: Merekam Arahan

Nama UC	Merekam arahan Direktur Jenderal
Aktor	Analisis Tata Usaha Direktur Jenderal (Analis TUD)
Deskripsi	Aktor merekam agenda pertemuan/rapat (jenis, tema, tanggal, dan nomor undangan kegiatan). Selanjutnya Analis TUD merekam arahan Direktur Jenderal serta usulan Unit Penanggung Jawab arahan, tanggal mulai dan tanggal akhir tindak lanjut.
Kondisi Awal	Aktor ingin merekam arahan Direktur Jenderal.
Kondisi Akhir	Arahan Direktur Jenderal berhasil diajukan ke Kasubbag TUD.
Alur Utama	1. Aktor membuka menu perekaman arahan. 2. Sistem menampilkan form dengan beberapa kolom isian seperti jenis rapat, tema rapat, tanggal rapat, nomor dan tanggal nota dinas/undangan, keterangan dan dokumen pendukung apabila tersedia. 3. Aktor merekam data agenda rapat. 4. Setelah berhasil direkam dan disimpan, sistem menampilkan form untuk merekam arahan dengan beberapa kolom isian seperti keterangan arahan, Unit Penanggung Jawab, waktu mulai, dan batas waktu selesai. 5. Aktor merekam data arahan. 6. Setelah berhasil direkam dan disimpan, aktor mengajukan persetujuan rekaman kepada Kasubbag TUD. 7. Aktor mendapatkan informasi status rekaman arahan (persetujuan Kasubbag TUD/pengembalian).
Alur Alternatif	-
Catatan	Analis TUD dapat memilih lebih dari 1 (satu) Unit Penanggung Jawab.

Tabel III.6 Deskripsi Use Case: Persetujuan Rekaman Arah

Nama UC	Persetujuan rekaman arahan Direktur Jenderal
Aktor	Kepala Subbagian Tata Usaha Direktur Jenderal (Kasubbag TUD)
Deskripsi	Aktor dapat memverifikasi dan menyetujui rekaman arahan yang diajukan oleh Analis TUD. Apabila ada hal yang tidak sesuai, rekaman dapat dikembalikan ke Analis TUD untuk kemudian dihapus atau diubah. Apabila sudah sesuai, dapat disetujui untuk diteruskan ke Unit Penanggung Jawab.
Kondisi Awal	Aktor memverifikasi rekaman arahan.
Kondisi Akhir	Aktor mengembalikan rekaman arahan ke Analis TUD, atau menyetujui dan mengirim arahan tersebut ke Unit Penanggung Jawab.
Alur Utama	1. Aktor membuka menu perekaman arahan. 2. Sistem menampilkan form data arahan. 3. Aktor meneliti rekaman arahan dan/atau dokumen pendukung yang telah direkam oleh Analis TUD. 4. Jika terdapat keterangan arahan yang tidak sesuai, aktor dapat mengubahnya pada form arahan. 5. Aktor menyetujui rekaman arahan tersebut untuk diteruskan ke Unit Penanggung Jawab (dalam hal ini adalah PIC Unit Eselon II).
Alur Alternatif	1. Aktor membuka menu perekaman arahan. 2. Sistem menampilkan form data arahan. 3. Aktor meneliti rekaman arahan dan/atau dokumen pendukung yang telah direkam oleh Analis TUD. 4. Aktor merekam catatan pengembalian arahan ke Analis TUD. 5. Analis TUD menerima informasi pengembalian rekaman arahan untuk perbaikan.
Catatan	-

Tabel III.7 Deskripsi Use Case: Menindaklanjuti Arahan

Nama UC	Menindaklanjuti arahan Direktur Jenderal
Aktor	Unit Penanggung Jawab (PIC Unit Eselon II)
Deskripsi	Aktor merekam tindak lanjut dan melampirkan dokumen pendukung apabila tersedia dari arahan sesuai tenggat waktu yang telah ditentukan oleh Kasubbag TUD.
Kondisi Awal	Aktor merekam tindak lanjut arahan.
Kondisi Akhir	Tindak lanjut berhasil dikirimkan ke Unit Tata Usaha Direktur Jenderal (Unit TUD) untuk diperiksa.
Alur Utama	1. Aktor membuka menu perekaman tindak lanjut arahan. 2. Sistem menampilkan data arahan beserta form tindak lanjut yang harus diisi. 3. Aktor merekam tindak lanjut beserta melampirkan bukti pendukung apabila tersedia. Selanjutnya aktor mengajukan tindak lanjut kepada Unit TUD. 4. Unit TUD menerima informasi tindak lanjut arahan dari PIC Unit Eselon II.
Alur Alternatif	-
Catatan	-

Tabel III.8 Deskripsi Use Case: Pemeriksaan Tindak Lanjut Arahan

Nama UC	Pemeriksaan tindak lanjut arahan Direktur Jenderal
Aktor	Analisis TUD dan Kasubbag TUD
Deskripsi	Aktor (Analisis TUD) memeriksa kelengkapan isian tindak lanjut dan dokumen pendukung apabila tersedia dari tindak lanjut yang diajukan oleh PIC Unit Eselon II. Apabila terdapat ketidaksesuaian secara administratif, aktor (Analisis TUD) mengembalikan status arahan ke PIC Unit Eselon II. Apabila telah sesuai, status arahan diteruskan ke aktor (Kasubbag TUD) untuk diverifikasi dan disetujui.
Kondisi Awal	Aktor memverifikasi dan menyetujui tindak lanjut arahan.
Kondisi Akhir	Aktor selesai memverifikasi dan memberi persetujuan tindak lanjut, atau mengembalikan tindak lanjut ke PIC Unit Eselon II.
Alur Utama	1. Aktor membuka menu detail tindak lanjut arahan. 2. Sistem menampilkan informasi tindak lanjut arahan yang telah direkam oleh PIC Unit Eselon II. 3. Aktor meneliti tindak lanjut dan dokumen pendukung. 4. Aktor merekam catatan persetujuan tindak lanjut arahan. 5. PIC Unit Eselon II menerima informasi bahwa tindak lanjut arahan telah dianggap selesai.
Alur Alternatif	1. Aktor membuka menu detail tindak lanjut arahan. 2. Sistem menampilkan informasi tindak lanjut arahan yang telah direkam oleh PIC Unit Eselon II. 3. Aktor meneliti tindak lanjut dan dokumen pendukung. 4. Aktor merekam catatan pengembalian tindak lanjut arahan. 5. PIC Unit Eselon II menerima informasi pengembalian tindak lanjut arahan untuk perbaikan.
Catatan	Apabila terdapat lebih dari 1 (satu) PIC Unit Eselon II, sistem menampilkan semua capaian/status dari tindak lanjut berdasarkan tiap Unit Eselon II.

Tabel III.9 Deskripsi Use Case: Monitoring Tindak Lanjut Arahan

Nama UC	Monitoring tindak lanjut arahan Direktur Jenderal
Aktor	Analisis TUD, Kasubbag TUD, dan Unit Kepatuhan Internal
Deskripsi	Aktor memantau status arahan dan hasil tindak lanjut yang telah direkam oleh PIC Unit Eselon II.
Kondisi Awal	Aktor ingin mengetahui status penyelesaian dan tindak lanjut dari arahan.
Kondisi Akhir	Informasi status penyelesaian dan tindak lanjut dari arahan yang dicari berhasil didapatkan.
Alur Utama	1. Aktor membuka menu <i>monitoring</i> arahan. 2. Aktor dapat melakukan pencarian berdasarkan parameter tertentu, seperti Unit Penanggung Jawab, status arahan, hingga kurun waktu arahan dilaksanakan. 3. Sistem menampilkan daftar arahan sesuai dengan parameter yang dipilih. 4. Aktor memilih salah satu arahan untuk melihat detail arahan beserta tindak lanjutnya. 5. Sistem menampilkan form informasi arahan beserta tindak lanjutnya apabila tersedia.
Alur Alternatif	-
Catatan	-

Tabel III.10 Deskripsi Use Case: Pengaturan Hak Akses Aplikasi

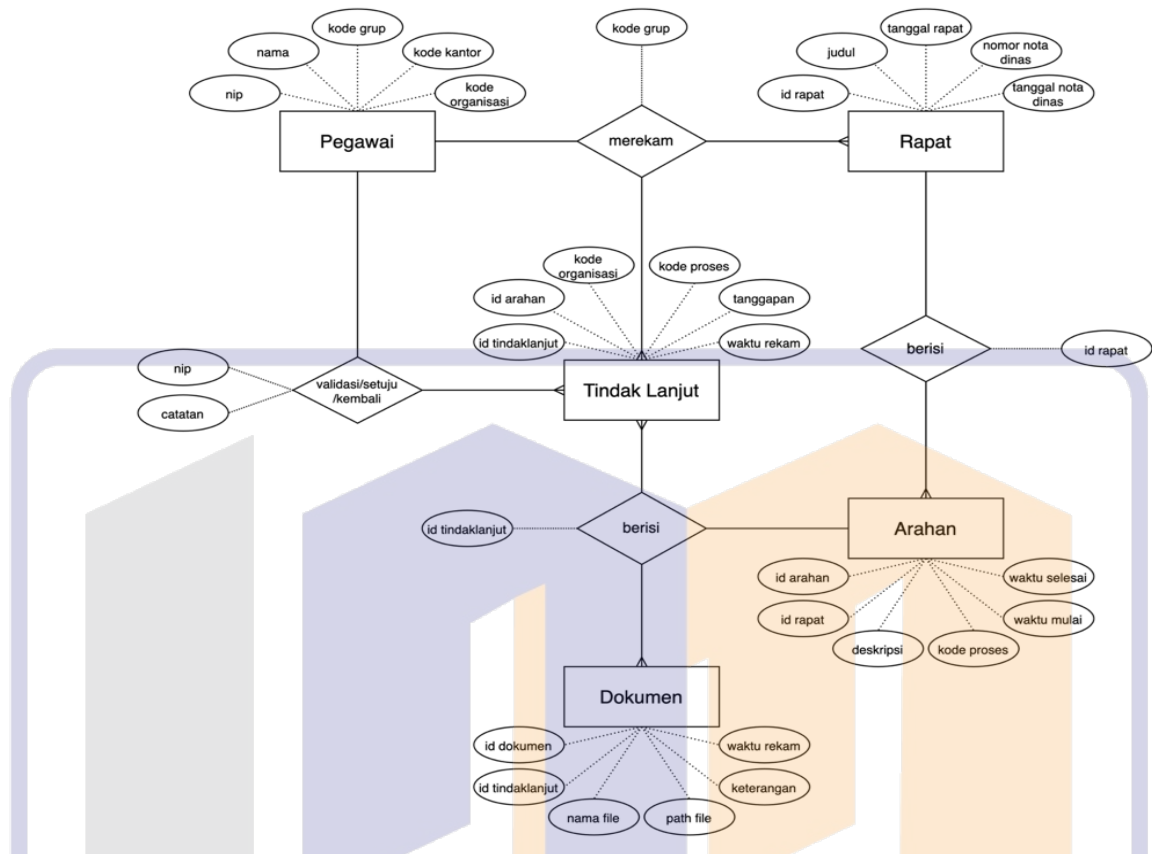
Nama UC	Pengaturan hak akses aplikasi
Aktor	Unit Kepatuhan Internal
Deskripsi	Aktor mengatur/menyetting hak akses pengguna sistem. Akses tersebut perlu dibatasi karena informasi yang terdapat pada aplikasi SMARD bersifat terbatas.
Kondisi Awal	Aktor ingin mengatur hak akses untuk pengguna aplikasi.
Kondisi Akhir	Pengaturan hak akses pengguna dalam aplikasi SMARD berhasil direkam, dan pengguna tersebut dapat mengakses menu pada aplikasi SMARD.
Alur Utama	1. Aktor membuka menu pengaturan hak akses. 2. Sistem menampilkan form dengan beberapa kolom isian seperti nama, Nomor Induk Kependidikan (NIP), dan nama akses/role yang dipilih. 3. Aktor merekam pengaturan hak akses terhadap pegawai yang telah ditentukan dan menyimpannya. 4. Pegawai tersebut dapat mengakses aplikasi SMARD.
Alur Alternatif	-
Catatan	Unit Kepatuhan Internal juga dapat menghapus pengaturan hak akses pengguna, jika pegawai tersebut pindah tempat bertugas.

3.2.2.3 Rancangan Database

Dalam merancang *database*, hal pertama yang dilakukan adalah mengidentifikasi dengan jelas entitas apa saja yang akan digunakan, beserta atribut-atribut yang relevan pada entitas tersebut. Selanjutnya dibuatlah suatu diagram relasi untuk menunjukkan relasi antar entitas/objek.

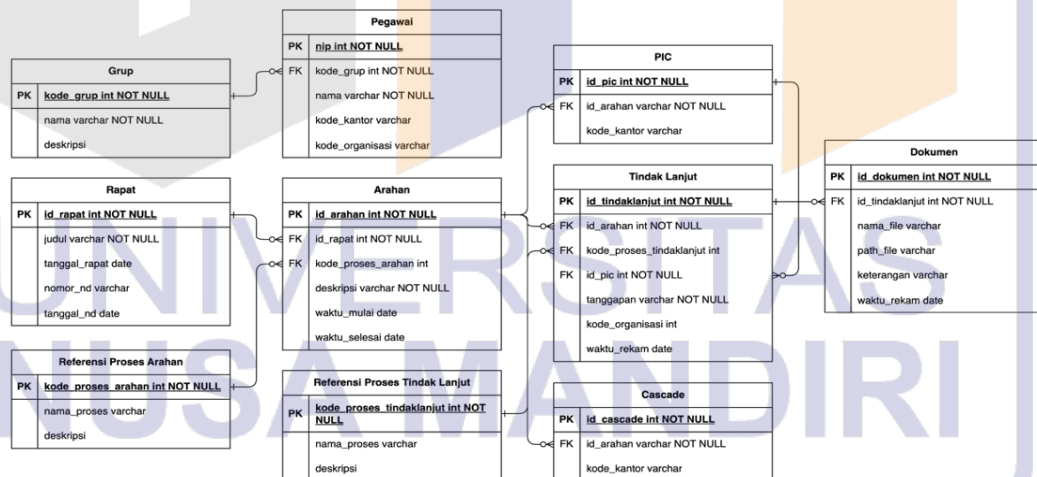
A. Entity Relationship Diagram

Entity Relationship Diagram (ERD) adalah diagram yang digunakan untuk memodelkan struktur data dengan menunjukkan hubungan/relasi antara entitas dan atributnya. Pembuatan ERD harus memperhatikan identifikasi dan normalisasi data untuk mencegah redundansi data dalam *database*.



Gambar III.4 Entity Relationship Diagram

B. Relational Schema



Gambar III.5 Skema Relasi

Pengembangan ERD bisa digunakan untuk membuat diagram lain yang membantu dalam perancangan sistem, salah satunya adalah Skema Relasi (*Relational Schema*). Skema relasi *database* adalah desain atau struktur tabel

dalam sebuah *database* yang menunjukkan bagaimana tabel-tabel di dalamnya saling berhubungan melalui *primary key* (PK) dan *foreign key* (FK). Skema ini berfungsi mengubah ERD menjadi model yang lebih teknis dalam bentuk tabel *database* yang siap diimplementasikan. Dalam skema ini berisi nama tabel, *primary key* (PK), *foreign key* (FK), dan tipe data untuk setiap kolom.

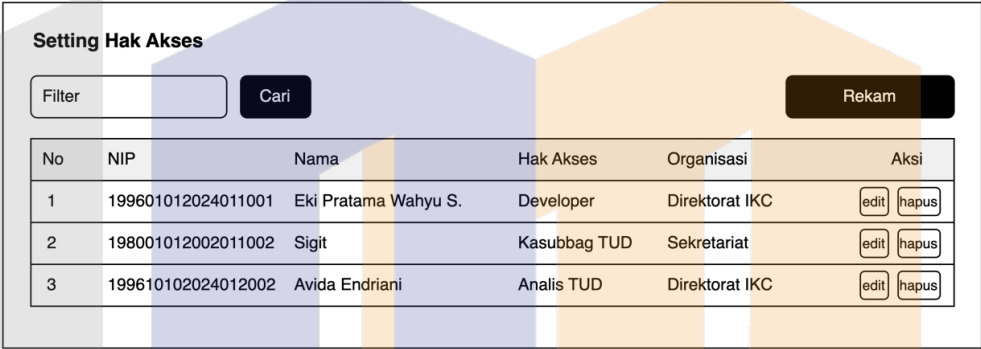
3.2.2.4 Desain Antarmuka (Mock-Up)

Desain antarmuka pengguna (*User Interface/UI*) memegang peran yang sangat penting dalam menciptakan pengalaman pengguna (*User Experience/UX*) yang intuitif, efisien, dan menyenangkan. Antarmuka yang dirancang dengan baik akan secara langsung berdampak pada kemudahan akses, penggunaan sistem, serta minimnya kebutuhan pelatihan bagi pengguna.

Sebagai bagian dari tahap awal perancangan, kami menggunakan *wireframe* untuk memvisualisasikan kerangka dasar antarmuka. *Wireframe* merupakan representasi visual sederhana yang menggambarkan tata letak dan struktur halaman sistem, tanpa memperhatikan aspek estetika seperti warna, icon, atau tipografi. Tujuan utama dari *wireframe* adalah untuk menentukan posisi dan hubungan antar elemen UI, seperti tombol, input form, navigasi, dan area konten, sehingga pengembang dan stakeholder dapat mengevaluasi alur interaksi pengguna sebelum masuk ke tahap implementasi visual. Dalam proses ini, kami menggunakan **draw.io** sebagai alat bantu pembuatan *wireframe* karena kemampuannya yang ringan, fleksibel, serta mendukung kolaborasi. Setiap modul, seperti halaman login, dashboard utama, form pengisian arahan, hingga tampilan daftar rapat, dirancang terlebih dahulu melalui *wireframe* untuk memastikan kesesuaian antara kebutuhan fungsional dan kenyamanan pengguna.

Dengan pendekatan UI/UX berbasis *wireframe* ini, proses pengembangan antarmuka SMARD menjadi lebih terarah dan dapat menyesuaikan dengan masukan pengguna secara iteratif, sehingga menghasilkan sistem yang tidak hanya kuat dari sisi fungsional, tetapi juga mudah digunakan dan responsif terhadap kebutuhan operasional.

A. Menu *Setting* Hak Akses



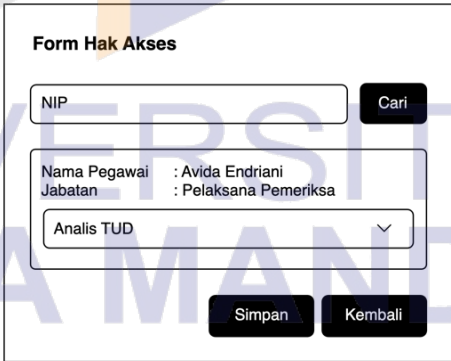
Setting Hak Akses

Filter Cari Rekam

No	NIP	Nama	Hak Akses	Organisasi	Aksi
1	199601012024011001	Eki Pratama Wahyu S.	Developer	Direktorat IKC	edit hapus
2	198001012002011002	Sigit	Kasubbag TUD	Sekretariat	edit hapus
3	199610102024012002	Avida Endriani	Analisis TUD	Direktorat IKC	edit hapus

Gambar III.6 Halaman Utama *Setting* Hak Akses

Halaman utama menampilkan tabel data pegawai yang telah diberikan hak akses penggunaan aplikasi. Terdapat *field* filter untuk mencari data berdasarkan parameter yang diketik, dan tombol rekam untuk menambahkan data akses baru.



Form Hak Akses

NIP Cari

Nama Pegawai : Avida Endriani
Jabatan : Pelaksana Pemeriksa

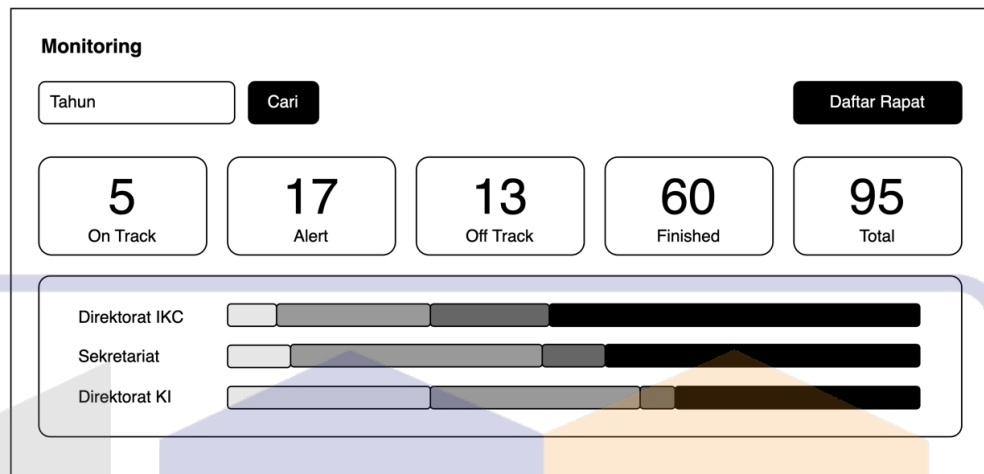
Analisis TUD

Simpan Kembali

Gambar III.7 Form Tambah Hak Akses

Form menambah data hak akses baru meliputi pencarian NIP terlebih dahulu, jika sistem berhasil mencari data pegawai berdasarkan NIP yang ditulis maka nama dan jabatan pegawai akan muncul untuk selanjutnya diberikan jenis hak akses yang diinginkan.

B. Menu *Monitoring*



Gambar III.8 Halaman Utama *Monitoring*

Dashboard berisi kumpulan data arahan yang ditotal berdasarkan statusnya, dan juga grafik arahan pada masing-masing Unit Eselon II seperti pada gambar di atas.

The Monitoring Daftar Rapat form includes search filters and a table of meeting data. The search filters are:

- Tema Rapat
- Nomor ND
- Tanggal ND
- Tahun Awal
- Tahun Akhir
- Cari

The table displays the following data:

No. ND	Tanggal ND	Perihal ND	Total Arahan	Finished	Outstanding	
No-1/BC/2025	dd-mm-yyyy	Lorem ipsum dolor sit amet	10	8	2	detail

Gambar III.9 Halaman *Monitoring* Rapat

Pengguna dapat mengakses *monitoring* daftar rapat melalui tombol Daftar Rapat pada halaman awal *dashboard*. Halaman ini memuat informasi daftar rapat dengan jumlah arahan beserta statusnya (sedang berjalan dan selesai).

Monitoring Daftar Arahan

Tahun

Cari

<<

Agar Direktorat IKC mempersiapkan pelaksanaan pengembangan sistem CEISA tahap II tahun 2025

Status

dd-mm-yyyy

Waktu Awal

Waktu Akhir

Nama Rekam

PIC

histori

tindak lanjut

detail

dd-mm-yyyy

dd-mm-yyyy

Lorem ipsum

Direktorat IKC

Pelaksanaan Dialog Kinerja Organisasi untuk masing-masing kantor

Status

dd-mm-yyyy

Gambar III.10 Halaman *Monitoring Arahan*

Monitoring daftar arahan juga dapat diakses melalui *card* status arahan pada *dashboard* atau memilih salah satu data rapat pada halaman *monitoring* rapat.

Daftar Arahan

Tahun

Cari

Agar Direktorat IKC mempersiapkan pelaksanaan pengembangan sistem CEISA tahap II tahun 2025

Status

dd-mm-yyyy

Waktu Awal

Waktu Akhir

Nama Rekam

PIC

histori

tindak lanjut

detail

dd-mm-yyyy

dd-mm-yyyy

Lorem ipsum

Direktorat IKC

Pelaksanaan Dialog Kinerja Organisasi untuk masing-masing kantor

Status

dd-mm-yyyy

Detail Tindak Lanjut

Jenis Rapat

Tema Rapat

Tanggal Rapat

Nomor ND

Tanggal ND

Perihal ND

: Lorem ipsum

: Lorem ipsum dolor sit amet

: dd-mm-yyyy

: No-1/BC/2025

: dd-mm-yyyy

: Lorem ipsum dolor sit amet

Arahan

Status

PIC

Waktu Awal

Waktu Akhir

: Agar Direktorat IKC mempersiapkan pelaksanaan pengembangan sistem CEISA tahap II tahun 2025

: Outstanding

: Direktorat IKC

: dd-mm-yyyy

: dd-mm-yyyy

Tanggapan

: Subdirektorat PSI telah melaksanakan pembentukan tim pengembangan dan membuat rancangan SKEP Tim pengembangan

List Dokumen Pelengkap

SKEP Tim Pengembangan

KEP-1/BC/2025

dd-mm-yyyy

Berkas

Gambar III.11 Halaman *Monitoring Detail Tindak Lanjut*

Detail tindak lanjut dapat dimonitor dari daftar arahan. Halaman ini memuat daftar tindak lanjut yang telah direkam PIC terkait.

Histori Proses

No	Nama Proses	Waktu Rekam	Pegawai	Catatan
1	Perekaman Arahan	dd-mm-yyyy	Avida Endriani	sesuai dengan arahan
2	Pengajuan Arahan	dd-mm-yyyy	Avida Endriani	mohon untuk direviu
3	Persetujuan Arahan	dd-mm-yyyy	Eki Pratama	arahan approved

Kembali

Gambar III.12 *Modal Histori Proses*

Dan untuk desain tampilan modal untuk histori proses arahan ataupun tindak lanjut dirancang seperti gambar di atas. Histori proses melekat pada setiap data arahan dan tindak lanjut yang telah direkam.

C. Menu Rekam Arahan

Halaman utama pada menu ini menampilkan form filterisasi data, tombol rekam, dan daftar rapat. Pada tiap data rapat, terdapat tombol edit, hapus, dan lihat detail.

Daftar Rapat

Tema Rapat

Nomor ND Tanggal ND

Tahun Awal Tahun Akhir Cari

Rekam

Judul Rapat	Rincian rapat	No. ND	Tanggal ND	Perihal ND	
		No-1/BC/2025	dd-mm-yyyy	Lorem ipsum dolor sit amet	edit hapus detail
Judul Rapat	Rincian rapat				dd-mm-yyyy

Gambar III.13 Halaman Utama Rekam Arahan: Daftar Rapat

Jika tombol lihat detail diklik, akan berpindah ke halaman daftar arahan seperti gambar di bawah ini. Daftar arahan menampilkan informasi arahan secara singkat, dan terdapat tombol edit, hapus, tambah PIC, histori, dan kirim/ajukan arahan.

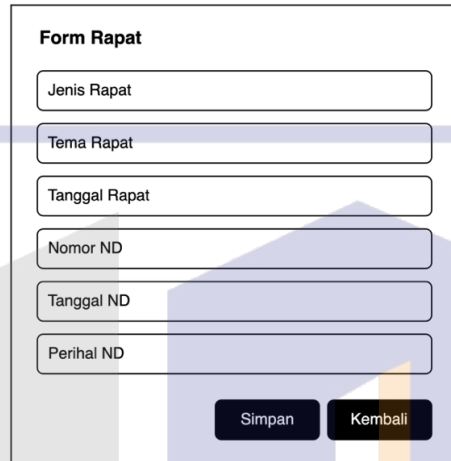
Daftar Arahan

Tahun Cari << Rekam

Agar Direktorat IKC mempersiapkan pelaksanaan pengembangan sistem CEISA tahap II tahun 2025				Status
dd-mm-yyyy				
Waktu Awal	Waktu Akhir	Nama Rekam	PIC	
dd-mm-yyyy	dd-mm-yyyy	Lorem ipsum	Direktorat IKC	edit hapus pic histori kirim
Pelaksanaan Dialog Kinerja Organisasi untuk masing-masing kantor				Status
dd-mm-yyyy				

Gambar III.14 Halaman Daftar Arahan

Desain tampilan form tambah rapat dan arahan bisa dilihat pada Gambar III.15 dan Gambar III.16 di bawah ini. Form tambah data ditampilkan dalam bentuk *drawer* ataupun modal *pop-up*.



Form Rapat

Jenis Rapat

Tema Rapat

Tanggal Rapat

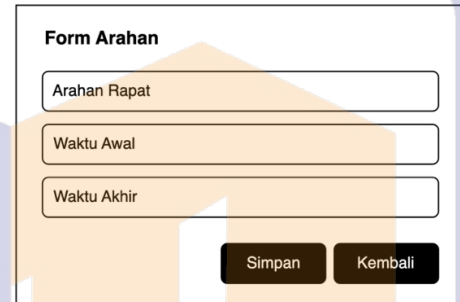
Nomor ND

Tanggal ND

Perihal ND

Simpan Kembali

Gambar III.15 Form Tambah Rapat



Form Arahan

Arahan Rapat

Waktu Awal

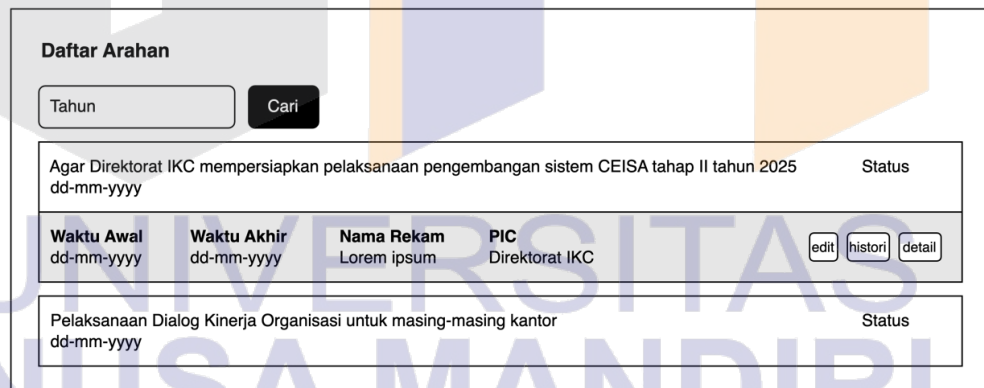
Waktu Akhir

Simpan Kembali

Gambar III.16 Form Tambah Arahan

D. Menu Persetujuan Arahan

Menu persetujuan arahan bertujuan untuk menyetujui arahan yang telah direkam oleh Analis untuk didistribusikan ke PIC terkait.



Daftar Arahan

Tahun Cari

Agar Direktorat IKC mempersiapkan pelaksanaan pengembangan sistem CEISA tahap II tahun 2025				Status
dd-mm-yyyy				
Waktu Awal	Waktu Akhir	Nama Rekam	PIC	
dd-mm-yyyy	dd-mm-yyyy	Lorem ipsum	Direktorat IKC	edit histori detail
Pelaksanaan Dialog Kinerja Organisasi untuk masing-masing kantor				Status
dd-mm-yyyy				

Gambar III.17 Halaman Utama Persetujuan Arahan

Halaman awal menu ini memuat daftar arahan yang telah direkam oleh Analis.

Terdapat tombol edit, histori arahan, dan detail arahan pada tiap baris data arahan.

Pengguna juga dapat mencari data berdasarkan tahun

Daftar Arahan

Agar Direktorat IKC mempersiapkan pelaksanaan pengembangan sistem CEISA tahap II tahun 2025
dd-mm-yyyy

Waktu Awal	Waktu Akhir	Nama
dd-mm-yyyy	dd-mm-yyyy	dd-mm-yyyy
Pelaksanaan Dialog Kinerja Organisasi untuk masing-masing kantor dd-mm-yyyy		

Detail Arahan

Jenis Rapat : Lorem ipsum
Tema Rapat : Lorem ipsum dolor sit amet
Tanggal Rapat : dd-mm-yyyy
Nomor ND : No-1/BC/2025
Tanggal ND : dd-mm-yyyy
Perihal ND : Lorem ipsum dolor sit amet

Arahan : Agar Direktorat IKC mempersiapkan pelaksanaan pengembangan sistem CEISA tahap II tahun 2025
Status : Outstanding
PIC : Direktorat IKC
Waktu Awal : dd-mm-yyyy
Waktu Akhir : dd-mm-yyyy

Gambar III.18 Drawer Detail Arahan

Kasubbag TUD dapat meneruskan atau mengembalikan arahan dengan disertai catatan proses.

E. Menu Tindak Lanjut Arahan

Halaman utama pada menu ini berisi daftar arahan yang ditugaskan kepada pengguna (PIC Unit Eselon).

Daftar Arahan

Agar Direktorat IKC mempersiapkan pelaksanaan pengembangan sistem CEISA tahap II tahun 2025
dd-mm-yyyy

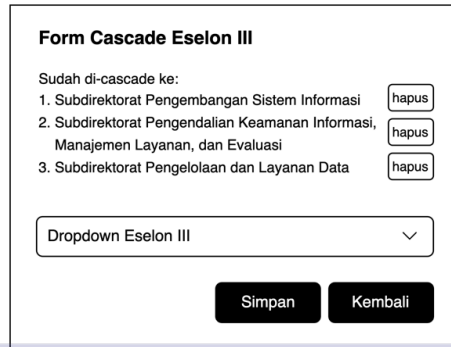
Status

Waktu Awal	Waktu Akhir	Nama Rekam	PIC	
dd-mm-yyyy	dd-mm-yyyy	dd-mm-yyyy	dd-mm-yyyy	
Pelaksanaan Dialog Kinerja Organisasi untuk masing-masing kantor dd-mm-yyyy				Status

Gambar III.19 Halaman Utama Tindak Lanjut Arahan

Tiap baris arahan terdapat tombol *cascade*, tindak lanjut, dan detail arahan.

Terdapat *form* filter arahan dengan berbagai parameter pencarian.



Form Cascade Eselon III

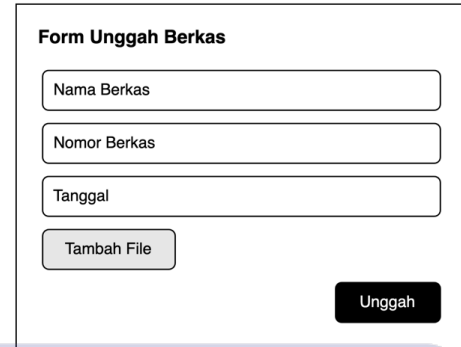
Sudah di-cascade ke:

1. Subdirektorat Pengembangan Sistem Informasi hapus
2. Subdirektorat Pengendalian Keamanan Informasi, Manajemen Layanan, dan Evaluasi hapus
3. Subdirektorat Pengelolaan dan Layanan Data hapus

Dropdown Eselon III ▼

Simpan Kembali

Gambar III.20 Form Tambah *Cascade*



Form Unggah Berkas

Nama Berkas

Nomor Berkas

Tanggal

Tambah File

Unggah

Gambar III.21 Form Tambah Berkas

Form tambah *cascade* ditujukan untuk PIC Unit Eselon II meneruskan arahan kepada Unit Eselon III di bawahnya, jika dalam kondisi arahan perlu pendetailan hingga unit vertikal. Dan setiap tindak lanjut yang direkam akan memerlukan unggah berkas/dokumen pendukung. Desain form unggah dapat dilihat pada gambar di atas.



Daftar Arahan

Tema Rapat	
Arahan	
Tahun Awal	Ta
Perihal ND	
Agar Direktorat IKC mempersiapkan pelaks dd-mm-yyyy	
Waktu Awal dd-mm-yyyy	Waktu Akhir dd-mm-yyyy
Pelaksanaan Dialog Kinerja Organisasi unt dd-mm-yyyy	

Detail Tindak Lanjut

1. Telah dilaksanakan rapat pembahasan pengembangan sistem, dengan terbitnya notulasi rapat upload edit hapus kirim
2. Subdirektorat PSI telah melaksanakan pembentukan tim pengembangan dan membuat rancangan SKEP Tim Pengembangan upload edit hapus kirim

Tanggapan...

Rekam

List Dokumen Pelengkap

SKEP Tim Pengembangan KEP-1/BC/2025 dd-mm-yyyy	Download
--	-----------------------

Gambar III.22 *Drawer* Detail Tindak Lanjut

Tampilan detail tindak lanjut berisi daftar tindak lanjut yang telah direkam oleh PIC, form perekaman tindak lanjut, dan daftar berkas/dokumen pelengkap pada masing-masing tindak lanjut.

F. Menu Verifikasi Analis

Verifikasi Tindak Lanjut - Analis

Tahun Cari

Telah dilaksanakan rapat pembahasan pengembangan sistem, dengan diterbitkannya notulasi rapat dd-mm-yyyy hh:mm:ss

Subdirektorat PSI telah melaksanakan pembentukan tim pengembangan dan membuat rancangan SKEP Tim Pengembangan dd-mm-yyyy hh:mm:ss

Gambar III.23 Halaman Utama Verifikasi Analis

Menu ini memuat daftar tindak lanjut yang direkam oleh PIC dan telah diteruskan ke Analis. Terdapat informasi uraian tindak lanjut, waktu perekaman, dan tombol kirim (diteruskan ke Kasubbag TUD).

Verifikasi Tindak Lanjut - Analis

Tahun Cari

Telah dilaksanakan rapat pembahasan pengembangan sistem, dengan diterbitkannya notulasi rapat

Subdirektorat PSI telah melaksanakan pembentukan tim pengembangan dan membuat rancangan SKEP Tim Pengembangan

Detail Tindak Lanjut

Jenis Rapat : Lorem ipsum
Tema Rapat : Lorem ipsum dolor sit amet
Tanggal Rapat : dd-mm-yyyy
Nomor ND : No-1/BC/2025
Tanggal ND : dd-mm-yyyy
Perihal ND : Lorem ipsum dolor sit amet

Arahan : Agar Direktorat IKC mempersiapkan pelaksanaan pengembangan sistem CEISA tahap II tahun 2025
Status : Outstanding
PIC : Direktorat IKC
Waktu Awal : dd-mm-yyyy
Waktu Akhir : dd-mm-yyyy

Tanggapan : Subdirektorat PSI telah melaksanakan pembentukan tim pengembangan dan membuat rancangan SKEP Tim pengembangan

List Dokumen Pelengkap

SKEP Tim Pengembangan
KEP-1/BC/2025
dd-mm-yyyy

Catatan Proses...

Gambar III.24 Drawer Detail Tindak Lanjut dan Form Persetujuan

Saat akan menyetujui/mengembalikan tindak lanjut, pengguna akan memperoleh informasi detail dari tindak lanjut tersebut, termasuk berkas/dokumen pelengkap yang telah diunggah oleh PIC.

G. Menu Verifikasi Kasubbag TUD

Verifikasi Tindak Lanjut - Kasubbag TUD

Telah dilaksanakan rapat pembahasan pengembangan sistem, dengan diterbitkannya notulasi rapat

dd-mm-yyyy hh:mm:ss

Subdirektorat PSI telah melaksanakan pembentukan tim pengembangan dan membuat rancangan SKEP Tim Pengembangan

dd-mm-yyyy hh:mm:ss

Gambar III.25 Halaman Utama Verifikasi Kasubbag TUD

Menu ini sebagai penentu arahan akan terselesaikan atau belum. Kasubbag TUD sebagai pengguna terakhir yang menentukan kesesuaian tindak lanjut dengan arahan yang diberikan. Untuk prosesnya hamper sama dengan menu sebelumnya (Verifikasi Analisis).

Verifikasi Tindak Lanjut - Kasubbag

Telah dilaksanakan rapat pembahasan pengembangan sistem, dengan diterbitkannya notulasi rapat

dd-mm-yyyy hh:mm:ss

Subdirektorat PSI telah melaksanakan pembentukan tim pengembangan dan membuat rancangan SKEP Tim Pengembangan

dd-mm-yyyy hh:mm:ss

Detail Tindak Lanjut

Jenis Rapat : Lorem ipsum
Tema Rapat : Lorem ipsum dolor sit amet
Tanggal Rapat : dd-mm-yyyy
Nomor ND : No-1/BC/2025
Tanggal ND : dd-mm-yyyy
Perihal ND : Lorem ipsum dolor sit amet

Arahan : Agar Direktorat IKC mempersiapkan pelaksanaan pengembangan sistem CEISA tahap II tahun 2025
Status : Outstanding
PIC : Direktorat IKC
Waktu Awal : dd-mm-yyyy
Waktu Akhir : dd-mm-yyyy

Tanggapan : Subdirektorat PSI telah melaksanakan pembentukan tim pengembangan dan membuat rancangan SKEP Tim pengembangan

List Dokumen Pelengkap

SKEP Tim Pengembangan
KEP-1/BC/2025
dd-mm-yyyy

Catatan Proses...

Gambar III.26 Drawer Detail Tindak Lanjut

41

3.3 Tahap Konstruksi

Tahap konstruksi merupakan proses pembangunan sistem berdasarkan hasil analisis dan perancangan yang telah dilakukan sebelumnya. Pada tahap ini, seluruh rancangan sistem diimplementasikan ke dalam bentuk aplikasi nyata menggunakan bahasa pemrograman Java, *framework* Spring Boot dan ReactJS, serta *database* PostgreSQL. Proses konstruksi meliputi pembuatan program dan dilanjutkan dengan pengujian untuk memastikan aplikasi berjalan sesuai kebutuhan.

3.3.1 Tahap Pembuatan Program

Tahap pembuatan program pada aplikasi SMARD dilakukan dengan mengembangkan modul-modul sesuai dengan kebutuhan pengguna. Proses pembangunan sistem menggunakan pendekatan *microservices* dan *microfrontend*.

3.3.1.1 Persiapan Database

Lingkungan *database* yang tersedia untuk proyek ini menggunakan PostgreSQL. Dalam pengimplementasian *database* PostgreSQL, terdapat tipe data UUID (*Universally Unique Identifier*) yang digunakan untuk menyimpan identitas unik. memiliki panjang tetap 128-bit (16 byte). UUID ini berfungsi sebagai pengganti integer (serial) untuk id *primary key*, karena lebih unik dan menghindari masalah duplikasi id. Fungsi `uuid_generate_v4()` juga diaplikasikan pada UUID untuk menghasilkan kumpulan karakter acak (lebih aman).

Selain tipe data, pada PostgreSQL juga terdapat istilah *collation*, yaitu aturan yang menentukan cara membandingkan dan mengurutkan teks berdasarkan bahasa atau aturan tertentu. Dengan *collation*, PostgreSQL dapat menangani perbedaan huruf besar-kecil (*case sensitivity*), aksen (diakritik), dan urutan alfabet sesuai dengan

kebutuhan. *Collation* dengan aturan *default* berarti sistem akan mengadaptasi aturan bawaan untuk mengurutkan dan membandingkan teks dalam *database*.

Misalnya, saat akan mengurutkan data dengan `ORDER BY`, PostgreSQL perlu tahu apakah huruf besar lebih dulu dari huruf kecil, bagaimana menangani aksen (é, è), atau bagaimana mengurutkan teks dalam berbagai bahasa. Contoh lain dalam *collation* adalah sebagai berikut.

Tabel III.11 Penggunaan *Collation*

Daftar Nama	Collation C (Default ASCII)	Collation en_US.UTF-8 (US English)
Alice álvaro Bob bob	Alice Bob bob álvaro -- (Huruf dengan aksen diurutkan setelah huruf biasa)	Alice álvaro -- (Aksen dianggap sama dengan huruf biasa) Bob bob

A. Tabel Referensi

Persiapan *database* dilakukan dengan langkah awal, menentukan suatu entitas masuk ke dalam kategori tabel referensi atau tabel master. Tabel referensi berfungsi untuk menyimpan daftar pilihan atau kategori yang akan digunakan tabel lain. Karakteristik yang dimiliki tabel referensi ini adalah data di dalamnya bersifat tetap atau jarang berubah, biasanya dihubungkan dengan tabel lain melalui *foreign key*. Dalam proyek ini, terdapat beberapa tabel referensi, yaitu:

Tabel III.12 Tabel Referensi: Proses Arahan

Column Name	Data Type	Not Null	Collation
kode_proses_arahan	int4	√	
nama_proses	varchar	√	Default
keterangan_kirim	varchar		Default
keterangan_terima	varchar		Default
deskripsi	varchar		Default
flag_aktif	bool		

Tabel Proses Arahan digunakan sebagai referensi jenis proses pada arahan.

Tabel III.13 Tabel Referensi: Proses Tindak Lanjut

Column Name	Data Type	Not Null	Collation
kode_proses_tindaklanjut	int4	√	
nama_proses	varchar	√	Default
keterangan_kirim	varchar		Default
keterangan_terima	varchar		Default
deskripsi	varchar		Default
flag_aktif	bool		

Digunakan sebagai referensi jenis proses pada tindak lanjut.

Tabel III.14 Tabel Referensi: Grup

Column Name	Data Type	Not Null	Collation
kode_grup	int4	√	
nama_grup	varchar	√	Default
deskripsi	varchar		Default
flag_aktif	bool		

Digunakan sebagai referensi jenis grup/hak akses aplikasi.

B. Tabel Master

Biasa dikenal dengan tabel data, berfungsi untuk menyimpan data atau entitas yang bersifat static dan tidak sering berubah. Tabel *master* memiliki karakteristik berisi informasi dasar yang digunakan oleh tabel lain, tidak bergantung pada tabel transaksi, dan biasanya data di dalamnya diubah dalam jangka waktu panjang.

Tabel III.15 Tabel Master: Pegawai

Column Name	Data Type	Not Null	Collation
id_pegawai	uuid	√	
nip	varchar	√	Default
nama	varchar	√	Default
kode_grup	int4	√	
kode_organisasi	varchar		Default

Digunakan sebagai tabel penyimpanan data pegawai yang telah diberi hak akses.

Tabel III.16 Tabel Master: Dokumen

Column Name	Data Type	Not Null	Collation
id_dokumen	uuid	√	
id_tindaklanjut	uuid	√	
nama_dokumen	varchar		Default
nomor_dokumen	varchar		Default
tanggal_dokumen	date		

Column Name	Data Type	Not Null	Collation
keterangan	varchar		Default
nama_file	varchar		Default
url_path	varchar		Default
nip_rekam	varchar		Default
waktu_rekam	date		

Digunakan sebagai tabel penyimpanan berkas/dokumen pelengkap tindak lanjut.

Tabel III.17 Tabel Master: Rapat

Column Name	Data Type	Not Null	Collation
id_rapat	uuid	√	
jenis_rapat	varchar	√	Default
tema_rapat	varchar		Default
tanggal_nd	date		
nomor_nd	varchar		Default
tanggal_nd	date		
perihal_nd	varchar		Default
flag_aktif	bool		

Digunakan sebagai tabel penyimpanan data rapat.

Tabel III.18 Tabel Master: Arahan

Column Name	Data Type	Not Null	Collation
id_arahan	uuid	√	
id_rapat	uuid	√	
deskripsi	varchar	√	Default
kode_proses_arahan	int4	√	
waktu_awal	date		
waktu_akhir	date		
waktu_selesai	date		
waktu_rekam	date		
nip_rekam	varchar		Default
flag_setuju	varchar(1)		
flag_outstanding	varchar(1)		Default
flag_selesai	bool		Default
flag_aktif	bool		

Digunakan sebagai tabel penyimpanan data arahan.

Tabel III.19 Tabel Master: Catatan Proses

Column Name	Data Type	Not Null	Collation
id_proses_arahan	uuid	√	
id_arahan	uuid	√	
kode_proses_arahan	int4	√	
catatan	varchar		Default
nip_rekam	varchar		Default
waktu_rekam	date		Default

Digunakan sebagai tabel penyimpanan catatan proses arahan.

3.3.1.2 Pengembangan *Backend*

Backend pada SMARD dirancang menggunakan arsitektur RESTful API yang memungkinkan komunikasi fleksibel antara *client* dan server. Berikut akan dibahas mengenai kebutuhan perangkat dan struktur program yang dibutuhkan dalam membangun *backend* pada SMARD.

A. Kebutuhan Perangkat

Untuk pengembangan *backend*, perangkat yang digunakan meliputi:

Tabel III.20 Kebutuhan Perangkat *Backend*

Perangkat Keras		Perangkat Lunak	
Model PC	HP Z4 G4 Workstation	IDE	IntelliJ IDEA
Prosesor	Intel® Xeon® W-2125 CPU @ 4.00GHz 4.01GHz	Plugin	Spring Boot
RAM	32,00 GB		
Harddisk	2 TB		
OS	Windows 11 Pro for Workstations		

B. Struktur Program *Backend*

Proyek ini menerapkan arsitektur *microfrontends* dan *microservices*. Pembangunan *backend* menggunakan Spring Boot, sebuah *framework* dari Java berbasis Spring yang digunakan untuk membangun aplikasi *server-side* secara cepat, mudah, dan efisien. Spring Boot memiliki keunggulan dalam membangun REST API, yaitu tidak memerlukan konfigurasi yang rumit dan pengembangan bisa lebih cepat.

Struktur pengembangan *backend* pada *microservice* dijelaskan menjadi beberapa komponen utama, antara lain:

1. JPA (Java *Persistence* API)

JPA adalah standar spesifikasi Java untuk mengelola relasi *database* menggunakan objek Java (biasanya berupa *class*/entitas). Dalam Spring Boot,

terdapat *class* JPARepository yang menyediakan cara sederhana dan efisien untuk mengakses data dari *database*.

```
public interface TdRapatRepo extends JpaRepository<TdSmardRapat, Long> {  
  
    @Query(value = "select a from TdSmardRapat a where idRapat = ?1")  
    public TdSmardRapat getTdSmardRapatById(UUID idRapat);  
}
```

Gambar III.27 JPARepository pada Spring Boot

Seperti pada gambar di atas, dalam kelas *interface* JPA tersebut menggunakan *custom query* dengan ditandai oleh anotasi `@Query`. Anotasi tersebut memberikan kontrol penuh untuk menuliskan query khusus sesuai kebutuhan pengembang.

2. Model sebagai Entity dan View Layer (MVC)

Penggunaan *model* pada Spring digunakan untuk merepresentasikan sebuah entitas atau tabel pada *database*. *Model* di sini digunakan bersamaan dengan JPA untuk berinteraksi dengan *database*.

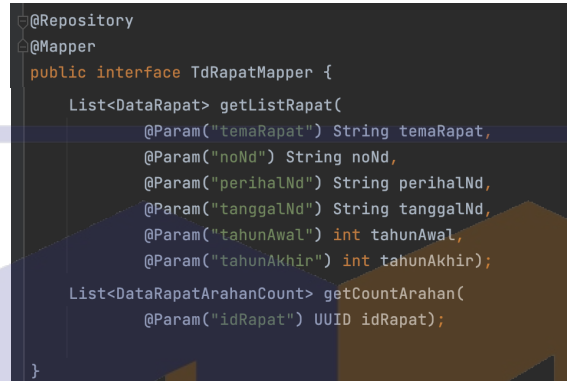
```
@Data  
@NoArgsConstructor  
@AllArgsConstructor  
public class DataRapat {  
    private String idRapat;  
    private String jenisRapat;  
    private String temaRapat;  
    private String tanggalRapat;  
    private String noNd;  
    private String tanggalNd;  
    private String perihalNd;  
    private String flagAktif;  
    private String totalArahan;  
    private String totalFinished;  
    private String totalOutstanding;  
}
```

Gambar III.28 Model Class DataRapat

Model juga dapat digunakan sebagai objek yang dipakai untuk menyimpan data dan akan dikirim ke tampilan (*view*). Biasanya penamaan komponen/variabel kolom disamakan dengan kolom pada *database* untuk memudahkan pengembang dalam memetakan data.

3. *Mapper*

Pada Spring Boot (Java), *mapper* adalah komponen yang berfungsi untuk memetakan atau mengonversi data dari suatu bentuk ke bentuk lainnya.

A screenshot of a code editor showing the definition of the TdRapatMapper interface. The code is in Java and includes annotations for Spring Data JPA. The interface has two methods: getListRapat() and getCountArahan(). The first method has five parameters: temaRapat, noNd, perihalNd, tanggalNd, and tahunAwal, each with a @Param annotation. The second method has one parameter: idRapat, with a @Param annotation. The code is as follows:

```
@Repository
@Mapper
public interface TdRapatMapper {

    List<DataRapat> getListRapat(
        @Param("temaRapat") String temaRapat,
        @Param("noNd") String noNd,
        @Param("perihalNd") String perihalNd,
        @Param("tanggalNd") String tanggalNd,
        @Param("tahunAwal") int tahunAwal,
        @Param("tahunAkhir") int tahunAkhir);

    List<DataRapatArahanCount> getCountArahan(
        @Param("idRapat") UUID idRapat);
}
```

Gambar III.29 Class Mapper TdRapatMapper

Dapat dilihat pada gambar di atas, terdapat sebuah *mapper* dengan nama kelas *interface* TdRapatMapper yang memetakan beberapa method (seperti getListRapat() dan getCountArahan()). Penggunaan @Param("") adalah bentuk representasi data pada *database* (nama kolom) yang dikonversikan ke nama variabel masing-masing.

4. Penggunaan HTTP *Response*

Dalam proses pengiriman permintaan (*request*) dari *client* (ditampung pada *frontend* aplikasi) ke server untuk melakukan operasi tertentu (CRUD), Spring Boot menyediakan cara sederhana untuk menangani HTTP *Request* dengan menggunakan Spring MVC. Gambar III.31 adalah salah satu contoh penggunaan HTTP *Request*, yaitu dengan penggunaan anotasi @GetMapping dan pemanggilan kelas ResponseEntity.

```
@GetMapping(path = "/get-list-rapat", produces = MediaType.APPLICATION_JSON_VALUE)
public ResponseEntity getListRapat(
    @RequestParam(value = "temaRapat", required = false) String temaRapat,
    @RequestParam(value = "noNd", required = false) String noNd,
    @RequestParam(value = "perihalNd", required = false) String perihalNd,
    @RequestParam(value = "tanggalNd", required = false) String tanggalNd,
    @RequestParam(value = "tahunAwal", required = false) String tahunAwal,
    @RequestParam(value = "tahunAkhir", required = false) String tahunAkhir,
    @RequestHeader(value = "Authorization", required = true) String authorizationHeader) {
    MessageModel msg = new MessageModel();
    ResponseEntity validateAuthToken = null;
    LinkedHashMap<String, Object> res = new LinkedHashMap<>();
```

Gambar III.30 Fungsi getListRapat()

Anotasi `@GetMapping` digunakan untuk menangani permintaan HTTP dengan metode GET pada *endpoint* `/get-list-rapat`.

`MediaType.APPLICATION_JSON_VALUE` menunjukkan bahwa *output* yang dihasilkan dari metode ini berbentuk JSON.

Kelas `ResponseEntity` adalah representasi dari HTTP *Response*. Jika sistem telah mengirim *request*, maka data yang didapat dari *database* dengan *query* tertentu akan dikembalikan melalui *response*. Respon yang diberikan berupa HTTP *Status Code* (jika berhasil maka bernilai 200, jika *error* bisa bernilai 404 atau 500), *Body* (data yang dikirim dalam respon), dan *Headers* (informasi tambahan seperti jenis konten, pengaturan *caching*, dan lainnya).

3.3.1.3 Pengembangan Frontend

Frontend pada SMARD dikembangkan menggunakan ReactJS yang memungkinkan pembuatan antarmuka yang modern, interaktif, dan responsif. Berikut akan dibahas mengenai kebutuhan perangkat dan struktur program yang dibutuhkan dalam membangun *frontend* pada SMARD.

A. Kebutuhan Perangkat

Untuk pengembangan *frontend*, perangkat yang digunakan meliputi:

Tabel III.21 Kebutuhan Perangkat *Frontend*

Perangkat Keras		Perangkat Lunak	
Model PC	iMac Retina 5K 27' 2017	IDE	IntelliJ IDEA
Prosesor	3,8 GHz Quad-Core Intel Core i5	Plugin	npm : 8.18.0
RAM	8 GB 2400 MHz DDR4		node : 16.17.0
Harddisk	Macintosh HD 2,12 TB		react : 16.13.1
OS	Ventura 13.0.1		antd : 3.26.9
			axios : 0.19.2

B. Struktur Program *Frontend*

Proyek ini menerapkan arsitektur *microfrontends* dan *microservices*. Pembangunan *frontend* menggunakan ReactJS, sebuah *library* dari JavaScript yang digunakan untuk membangun antarmuka secara modular dan responsif. ReactJS memiliki keunggulan dalam membangun *Single Page Application* (SPA) yang memungkinkan navigasi lebih cepat dan efisien tanpa perlu memuat ulang halaman.

Struktur pengembangan *frontend* pada *microfrontend* dijelaskan menjadi beberapa komponen utama, antara lain:

1. *Routing* Menu pada *Microfrontend*

Routing menu mengacu pada bagaimana navigasi antarmuka pengguna (menu, link, atau halaman) dikelola pada beberapa bagian *frontend* terpisah. Pendekatan umum *routing* menu pada arsitektur *microfrontend* dijelaskan sebagai berikut. Karena kami memakai arsitektur *microfrontend*, maka perlu membuat rute awal pada aplikasi utama. Konfigurasi dilakukan seperti gambar di bawah ini.

```
{
  "hostProduction": "https://mfe-smard.beacukai.go.id/",
  "host": "http://localhost:4000",
  "moduleSlug": "httpsmfesmardbeacukaigoid",
  "path": "/smard/monitoring",
  "title": "Monitoring",
  "isAdditionalPath": true
},
```

Gambar III.31 Konfigurasi Route pada *Microfrontend*

Selanjutnya melakukan konfigurasi pada file `index.js` untuk mengintegrasikan antar *microfrontend* ke dalam *host/container* aplikasi dengan pendekatan *mount-unmount*.

```
window.renderhttpsmfesmardbeacukaigoid = (containerId, props) => {  
  ReactDOM.render(<App {...props} />, document.getElementById(containerId));  
  serviceWorker.unregister();  
};  
window.unmounthttpsmfesmardbeacukaigoid = (containerId) => {  
  return ReactDOM.unmountComponentAtNode(document.getElementById(containerId));  
};
```

Gambar III.32 Konfigurasi ModulSlug SMARD

Fungsi pertama `window.render()` berfungsi untuk melakukan *render/mount* aplikasi (dalam hal ini berbasis React) ke dalam element DOM tertentu yang diberikan oleh *container*.

Fungsi kedua `window.unmount()` berfungsi untuk melakukan *unmount* aplikasi dari elemen DOM saat tidak digunakan lagi, atau saat berpindah halaman/menu.

Fungsi berikutnya adalah bagian dari konfigurasi *routing* di aplikasi React.

```
<Route  
  exact  
  path={`/${smard}/monitoring`}   
  render={props => <Monitoring {...props} {...propsExtra} />}  
/>
```

Gambar III.33 Konfigurasi Route Menu SMARD

Komponen `<Route />` digunakan untuk menentukan rute URL dan komponen mana yang harus di-render saat URL tersebut diakses. Saat URL/path diakses, sistem akan mengeksekusi fungsi di dalam prop `render`, yaitu link pada komponen `<Monitoring />`.

2. Pemanggilan API Service

Integrasi *frontend* dan *backend* dilakukan melalui pemanggilan API adalah proses di mana *frontend* (*client-side*) berkomunikasi dengan *backend* (*server-side*) menggunakan API, biasanya melalui protokol HTTP/HTTPS. *Frontend* membuat permintaan HTTP ke *endpoint* API tertentu (dalam hal ini menggunakan *library* *axios* pada JavaScript).

Seperti pada penggalan kode di bawah ini, berfungsi untuk mengambil daftar rapat dari *endpoint* `/get-list-rapat` secara *asynchronous*. URL HTTP disimpan pada variabel `REACT_APP_SEKRE_SMARD` dengan menambahkan *header* token untuk autentikasi ke API.

```
fetchRapat = async() => {
  const token = await getUserAccessToken()
  this.setState({loading: true})
  await axios.get(`${REACT_APP_SEKRE_SMARD}/get-list-rapat`, {
    headers: {Authorization: `Bearer ${token}`},
    params
  }).then(res => {
    this.setState({loading: false, dataRapat: res.data.data})
  }).catch (e => {
    this.setState({loading: false, error: true})
    Swal.fire({
      title: 'Gagal memuat daftar rapat!',
      text: 'Harap mengecek koneksi dan memuat kembali.',
      icon: 'error',
      timer: 2000
    })
  })
}
```

Gambar III.34 Integrasi API Service

Pemanggilan *endpoint* di atas menggunakan metode GET, dengan *request* dengan parameter yang disimpan pada variabel `params`. Respon yang dihasilkan berbentuk JSON Array, dan disimpan pada variabel *state* `dataRapat`. Dan jika koneksi ke API melempar respon error, sistem akan menampilkan *alert* dengan pesan error gagal memuat.

3. Pemetaan Respon JSON ke Antarmuka

Agar daftar rapat dapat ditampilkan pada antarmuka pengguna, perlu dilakukan pemetaan respon dari API untuk ditampilkan pada form/komponen UI.

```
this.state.dataRapat.length > 0 ? (
```

Gambar III.35 Pengecekan Array Rapat

Logika awal untuk pengecekan data, apakah respon JSON Array yang disimpan pada `dataRapat` lebih dari 0.

```
<div className="font-cari">
  (menampilkan {this.state.minValue + 1} -
  {this.state.minValue + this.state.dataRapat
  .slice(this.state.minValue, this.state.maxValue).length}
  item dari total {this.state.dataRapat.length} data)
</div>
```

Gambar III.36 *Pagination* pada Daftar Rapat

4. Perintah Menjalankan Program pada Local

Untuk menjalankan program kami menggunakan *library* npm dan mengetikkan perintah seperti gambar di bawah pada terminal.

```
Avida:smard-new hp$ npm install
Avida:smard-new hp$ npm start
```

Gambar III.37 Perintah Menjalankan Program

Jika program berhasil dijalankan, maka akan muncul pesan sukses yang memuat URL pengaksesan program seperti gambar di bawah ini.

```
Compiled successfully!
You can now view mf_sekre_smard in the browser.

Local:      http://localhost:4000
On Your Network:  http://10.102.104.193:4000

Note that the development build is not optimized.
To create a production build, use npm run build.
```

Gambar III.38 Pesan Sukses Menjalankan Program

3.3.2 Tahap Pengujian Aplikasi

Tahap pengujian merupakan tahap akhir dari proses pengembangan sistem, memastikan sistem yang dibangun telah sesuai dengan kebutuhan dan berjalan dengan optimal sebelum produk diterapkan pada lingkungan *production*. Pada bagian ini akan dilakukan pengujian terhadap komponen fungsional (*Unit Testing*).

Tabel III.23 Daftar Fitur Aplikasi

Fitur	Sub Fitur/Proses	Keterangan
Monitoring	Dashboard	Halaman dasbor yang memuat akumulasi data status arahan beserta penampilan grafik arahan berdasarkan kantor yang ditetapkan.
	Daftar Rapat	Memuat daftar rapat beserta status arahan di dalamnya.
	Daftar Arahan	Memuat daftar arahan pada suatu rapat.
Rekam Arahan	Daftar Rapat	Fitur untuk Analis merekam data rapat.
	Daftar Arahan	Merekam arahan dari rapat yang dipilih dan akan diteruskan ke PIC Unit Eselon II.
Persetujuan Arahan		Fitur yang disediakan untuk Kasubbag TUD dalam menyetujui/mengembalikan data rapat yang direkam oleh Analis.
Tindak Lanjut Arahan		Fitur yang disediakan untuk PIC Unit Eselon II maupun Eselon III dalam menindaklanjuti arahan yang ditetapkan pada unit mereka.
Verifikasi Analis		Fitur yang hanya dapat diakses oleh Analis untuk memverifikasi tindaklanjut yang telah direkam oleh PIC unit terkait, yang nantinya akan diteruskan kepada Kasubbag TUD.
Verifikasi Kasubbag TUD		Fitur yang hanya dapat diakses oleh Kasubbag TUD untuk memverifikasi tindaklanjut secara final.
Setting Hak Akses		Fitur yang hanya dapat diakses oleh Unit Kepatuhan Internal dalam manajemen hak akses pengguna aplikasi.

Tabel III.24 Sebaran Role dan Pengaksesan Menu

Role	Menu						
	Monitoring	Rekam Arahan	Persetujuan Arahan	Tindak Lanjut Arahan	Verifikasi Analis	Verifikasi Kasubbag TUD	Setting Hak Akses
Analisis TUD		√			√		
Kasubbag TUD			√			√	
Unit Kepatuhan Internal	√						√
PIC Unit Eselon II				√			
PIC Unit Eselon III				√			

Unit Testing adalah pengujian yang dilakukan terhadap masing-masing komponen dari sistem secara terpisah untuk memastikan bahwa setiap bagian berjalan dengan baik dan sesuai. Pengujian ini dilakukan oleh tim pengembang (*Programmer* dan *Quality Control*) dengan metode *black-box testing*, di mana pengujian berfokus pada *input* dan *output* dengan menghiraukan struktur internal dari kode program.

Tabel III.25 Hasil Unit Testing: Setting Hak Akses

Menu/Halaman	Setting Hak Akses
Deskripsi Pengujian	Menambahkan hak akses pada pegawai
Input	NIP
Output yang Diharapkan	1. Menampilkan daftar pegawai yang telah diberi hak akses 2. Pegawai mendapatkan hak akses untuk mengakses aplikasi
Hasil Uji	<i>Valid</i>

Tabel III.26 Hasil Unit Testing: Monitoring

Menu/Halaman	Monitoring
Deskripsi Pengujian	1. Mencari data arahan berdasarkan parameter pencarian 2. Menampilkan daftar arahan dengan mengklik card status arahan
Input	Tahun awal, tahun akhir, nama kantor
Output yang Diharapkan	1. Data akumulasi dan grafik yang ditampilkan sesuai dengan parameter pencarian 2. Menampilkan daftar arahan sesuai dengan status yang dipilih
Hasil Uji	<i>Valid</i>

Tabel III.27 Hasil Unit Testing: Rekam Arahan

Menu/Halaman	Rekam Arahan
Deskripsi Pengujian	1. Menambahkan data rapat dan arahan baru 2. Menambahkan PIC pada arahan 3. Meneruskan pengajuan arahan ke Kasubbag TUD
Input	Jenis, tema, dan tanggal rapat; nomor, tanggal, dan perihal ND kegiatan; serta nama kantor sebagai PIC
Output yang Diharapkan	1. Data baru tersimpan dan muncul di daftar rapat 2. Menampilkan daftar PIC pada detail arahan 3. Berhasil mengajukan arahan ke Kasubbag TUD
Hasil Uji	<i>Valid</i>

Tabel III.28 Hasil Unit Testing: Persetujuan Arahan

Menu/Halaman	Persetujuan Arahan
Deskripsi Pengujian	Menyetujui/mengembalikan arahan yang telah diajukan oleh Analis
Input	Pesan persetujuan/pengembalian dan tombol aksi persetujuan

Output yang Diharapkan	1. Jika menyetujui, arahan berhasil terdistribusikan ke masing-masing PIC 2. Jika mengembalikan, arahan akan dikembalikan pada Analis untuk diperbaiki lagi
Hasil Uji	<i>Valid</i>

Tabel III.29 Hasil Unit Testing: Tindak Lanjut Arahan

Menu/Halaman	Tindak Lanjut Arahan
Deskripsi Pengujian	1. Merekam tindak lanjut arahan 2. Menambahkan cascade atau PIC Unit Eselon III 3. Mengirim tindak lanjut kepada Analis untuk direviu
Input	Data tindak lanjut, nama unit eselon III sebagai cascade
Output yang Diharapkan	1. Data tindak lanjut berhasil disimpan dan ditampilkan 2. Cascade berhasil disimpan, dan PIC Unit Eselon III mendapat akses untuk merekam tindak lanjut 3. Tindak lanjut berhasil dikirim
Hasil Uji	<i>Valid</i>

Tabel III.30 Hasil Unit Testing: Verifikasi Analis

Menu/Halaman	Verifikasi Analis
Deskripsi Pengujian	Menyetujui/menolak tindak lanjut yang telah diajukan oleh PIC Unit
Input	Pesan persetujuan/penolakan dan tombol aksi persetujuan
Output yang Diharapkan	1. Jika menyetujui, tindak lanjut arahan akan diteruskan ke Kasubbag TUD untuk diverifikasi 2. Jika menolak, tindak lanjut arahan akan dikembalikan pada PIC Unit untuk diperbaiki lagi
Hasil Uji	<i>Valid</i>

Tabel III.31 Hasil Unit Testing: Verifikasi Kasubbag TUD

Menu/Halaman	Kasubbag TUD
Deskripsi Pengujian	Memverifikasi tindak lanjut yang telah disetujui oleh Analis
Input	Pesan persetujuan/penolakan dan tombol aksi persetujuan
Output yang Diharapkan	1. Jika menyetujui, tindak lanjut arahan akan dianggap selesai 2. Jika menolak, tindak lanjut arahan akan dikembalikan pada PIC Unit untuk diperbaiki lagi
Hasil Uji	<i>Valid</i>

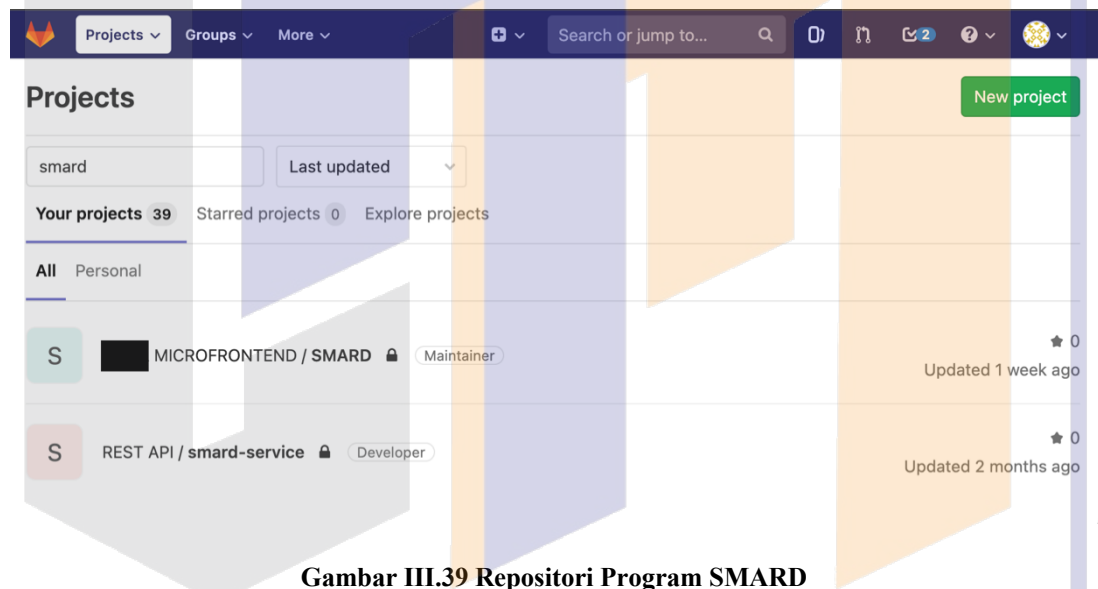
3.4 Tahap Implementasi

Tahap implementasi merupakan proses penerapan aplikasi SMARD ke dalam lingkungan nyata di DJBC. Implementasi dilakukan agar sistem yang telah dibangun dapat digunakan sesuai dengan tujuan, yaitu mendukung pencatatan, penerusan, tindak

lanjut, dan *monitoring* arahan Dirjen secara lebih efektif. Pada tahap ini, dilakukan dua kegiatan utama, yaitu penerapan sistem dan penerimaan sistem.

3.4.1 Tahap Penerapan Sistem

Penerapan sistem SMARD dilakukan dengan memasang aplikasi ke dalam infrastruktur yang dituju. Proses ini meliputi instalasi perangkat lunak pendukung, konfigurasi server, integrasi dengan *database* PostgreSQL, serta penyediaan akses bagi pengguna sesuai dengan peran masing-masing.



Gambar III.39 Repositori Program SMARD

Konfigurasi penerapan kode program ke lingkungan nyata (*production*) dilakukan melalui instalasi otomatis yang diintegrasikan dengan repositori awan (*cloud*) untuk mendapatkan domain *production* yang hanya dapat diakses melalui jaringan intranet.

3.4.2 Tahap Penerimaan Sistem

Selain aspek teknis, pada tahap ini juga dilakukan sosialisasi dan pelatihan penggunaan sistem kepada pengguna agar mereka memahami cara merekam arahan, menindaklanjuti arahan, serta melakukan *monitoring*. Tahap penerimaan sistem merupakan proses evaluasi untuk memastikan bahwa SMARD berjalan sesuai dengan kebutuhan pengguna. Evaluasi dilakukan melalui *User Acceptance Test* (UAT) dengan melibatkan pihak-pihak yang akan menggunakan sistem secara langsung.

UAT merupakan pengujian akhir yang dilakukan oleh pengguna sistem (*end-user*) untuk menilai apakah produk aplikasi siap untuk diimplementasikan. Pengujian dilakukan dengan memberikan akses kepada Subbagian Tata Usaha Direktur Jenderal serta beberapa pengguna lainnya, dan melakukan pengoperasian sistem dengan mengikuti skenario pengujian yang telah diberikan. Hasil pengujian akan menghasilkan dokumen UAT yang memuat informasi tentang fitur apa saja yang diuji beserta respon dari pengguna.

Tabel III.32 Hasil Kasus Uji: SMARD-TC-01

Kode Kasus Uji		SMARD-TC-01		Penguji	Nela Rizky
Deskripsi Uji		Merekam Rapat		Tanggal Uji	25 April
No	Form/Menu	Tahapan Uji	Data Uji	Output	Hasil Uji
1	Perekaman Arahan (Daftar Rapat)	Klik tombol Rekam		Sistem menampilkan form tambah rapat	Valid
		Isikan form perekaman rapat	Jenis rapat Tema rapat Tanggal rapat No. ND Tanggal ND Perihal ND		Valid
		Klik tombol Simpan		Sistem menampilkan <i>pop-up</i> berhasil disimpan	Valid
2	Edit Rapat	Klik tombol 		Sistem menampilkan form edit rapat	Valid
		Klik tombol Update		Sistem menampilkan <i>pop-up</i> berhasil diupdate	Valid

Tabel III.33 Hasil Kasus Uji: SMARD-TC-02

Kode Kasus Uji		SMARD-TC-02		Penguji	Nela Rizky
Deskripsi Uji		Merekam Arahkan		Tanggal Uji	25 April
No	Form/Menu	Tahapan Uji	Data Uji	Output	Hasil Uji
1	Rekam Arahkan (Daftar Rapat) (Daftar Arahkan)	Klik tombol 		Sistem menampilkan daftar arahan	Valid
		Klik tombol Rekam		Sistem menampilkan form tambah arahan	Valid
		Isikan form perekaman arahan	Arahan rapat Waktu mulai Waktu selesai		Valid
		Klik tombol Simpan		Sistem menampilkan <i>pop-up</i> berhasil disimpan	Valid
2	Edit Arahkan	Klik tombol 		Sistem menampilkan form edit arahan	Valid
		Klik tombol Update		Sistem menampilkan <i>pop-up</i> berhasil diupdate	Valid

Tabel III.34 Hasil Kasus Uji: SMARD-TC-03

Kode Kasus Uji		SMARD-TC-03		Penguji	Nela Rizky
Deskripsi Uji		Merekam PIC		Tanggal Uji	25 April
No	Form/Menu	Tahapan Uji	Data Uji	Output	Hasil Uji
1	Rekam Arahkan (Daftar Arahkan)	Klik tombol 		Sistem menampilkan <i>drawer</i> perekaman PIC	Valid
		Isikan form perekaman PIC	Nama kantor		Valid
		Klik tombol Simpan		Sistem menampilkan <i>pop-up</i> berhasil disimpan	Valid

Tabel III.35 Hasil Kasus Uji: SMARD-TC-04

Kode Kasus Uji		SMARD-TC-04		Penguji	Nela Rizky
Deskripsi Uji		Mengajukan Arahan		Tanggal Uji	25 April
No	Form/Menu	Tahapan Uji	Data Uji	Output	Hasil Uji
1	Perekaman Arahkan (Daftar Arahkan)	Klik tombol 		Sistem menampilkan <i>drawer</i> pengajuan arahan	Valid
		Isikan form pengajuan arahan	Catatan proses		Valid
		Klik tombol Simpan		Sistem menampilkan <i>pop-up</i> berhasil disimpan	Valid

Tabel III.36 Hasil Kasus Uji: SMARD-TC-05

Kode Kasus Uji		SMARD-TC-05		Penguji	Nela Rizky
Deskripsi Uji		Menyetujui Rekaman Arahan		Tanggal Uji	25 April
No	Form/Menu	Tahapan Uji	Data Uji	Output	Hasil Uji
1	Persetujuan Arahan	Klik menu Persetujuan Arahan		Sistem menampilkan daftar arahan yang memerlukan persetujuan	Valid
		Klik tombol 		Sistem menampilkan <i>drawer</i> detail arahan	Valid
		Isikan form persetujuan arahan	Catatan proses		Valid
		Klik tombol Teruskan ke PIC		Sistem menampilkan <i>pop-up</i> berhasil diteruskan	Valid

Tabel III.37 Hasil Kasus Uji: SMARD-TC-06

Kode Kasus Uji		SMARD-TC-06		Penguji	Nela Rizky
Deskripsi Uji		Mengembalikan Rekaman Arahan		Tanggal Uji	25 April
No	Form/Menu	Tahapan Uji	Data Uji	Output	Hasil Uji
1	Persetujuan Arahan	Klik menu Persetujuan Arahan		Sistem menampilkan daftar arahan yang memerlukan persetujuan	Valid
		Klik tombol 		Sistem menampilkan <i>pop-up</i> detail arahan	Valid
		Isikan form persetujuan arahan	Catatan proses		Valid
		Klik tombol Kembalikan ke Analisis		Sistem menampilkan <i>pop-up</i> berhasil dikembalikan	Valid

Tabel III.38 Hasil Kasus Uji: SMARD-TC-07

Kode Kasus Uji		SMARD-TC-07		Penguji	Nela Rizky
Deskripsi Uji		Merekam Tindak Lanjut Arahan		Tanggal Uji	25 April
No	Form/Menu	Tahapan Uji	Data Uji	Output	Hasil Uji
1	Tindak Lanjut Arahan	Klik menu Tindak Lanjut Arahan		Sistem menampilkan daftar arahan yang memerlukan persetujuan	Valid
		Klik tombol 		Sistem menampilkan <i>drawer</i> daftar dan form tindak lanjut	Valid

		Isikan form perekaman tindak lanjut	Tindak lanjut arahan		Valid
		Klik tombol Rekam		Sistem menampilkan <i>pop-up</i> berhasil disimpan	Valid
2	Edit Tindak Lanjut	Klik tombol  lalu klik tombol  pada tindak lanjut yang telah direkam		Sistem menampilkan form edit tindak lanjut	Valid
		Klik tombol Update		Sistem menampilkan <i>pop-up</i> berhasil diupdate	Valid
3	Upload Dokumen	Klik tombol 		Sistem menampilkan <i>pop-up</i> form upload	Valid
		Isikan form perekaman dokumen	Nama berkas Nomor berkas Tanggal File		Valid
		Klik tombol Simpan		Sistem menampilkan <i>pop-up</i> berhasil disimpan	Valid

Tabel III.39 Hasil Kasus Uji: SMARD-TC-08

Kode Kasus Uji		SMARD-TC-08		Penguji	Nela Rizky
Deskripsi Uji		Merekam Cascade		Tanggal Uji	25 April
No	Form/Menu	Tahapan Uji	Data Uji	Output	Hasil Uji
1	Tindak Lanjut Arahan	Klik tombol 		Sistem menampilkan <i>drawer</i> perekaman cascade	Valid
		Isikan form perekaman cascade	PIC Unit Eselon III		Valid
		Klik tombol Rekam		Sistem menampilkan <i>pop-up</i> berhasil disimpan	Valid

Tabel III.40 Hasil Kasus Uji: SMARD-TC-09

Kode Kasus Uji		SMARD-TC-09		Penguji	Nela Rizky
Deskripsi Uji		Menyetujui Tindak Lanjut (Analisis)		Tanggal Uji	25 April
No	Form/Menu	Tahapan Uji	Data Uji	Output	Hasil Uji
1	Verifikasi Analisis	Klik tombol 		Sistem menampilkan <i>drawer</i> verifikasi	Valid

		Isikan form verifikasi	Catatan proses		Valid
		Klik tombol Sesuai		Sistem menampilkan <i>pop-up</i> berhasil diteruskan	Valid

Tabel III.41 Hasil Kasus Uji: SMARD-TC-10

Kode Kasus Uji		SMARD-TC-10		Penguji	Nela Rizky
Deskripsi Uji		Mengembalikan Tindak Lanjut (Analisis)		Tanggal Uji	25 April
No	Form/Menu	Tahapan Uji	Data Uji	Output	Hasil Uji
1	Verifikasi Analisis	Klik tombol 		Sistem menampilkan <i>drawer</i> verifikasi	Valid
		Isikan form verifikasi	Catatan proses		Valid
		Klik tombol Kembalikan ke PIC		Sistem menampilkan <i>pop-up</i> berhasil dikembalikan	Valid

Tabel III.42 Hasil Kasus Uji: SMARD-TC-11

Kode Kasus Uji		SMARD-TC-11		Penguji	Nela Rizky
Deskripsi Uji		Menyetujui Tindak Lanjut (Analisis)		Tanggal Uji	25 April
No	Form/Menu	Tahapan Uji	Data Uji	Output	Hasil Uji
1	Verifikasi Kasubag TUD	Klik tombol 		Sistem menampilkan <i>drawer</i> verifikasi	Valid
		Isikan form verifikasi	Catatan proses		Valid
		Klik tombol Setujui		Sistem menampilkan <i>pop-up</i> berhasil diteruskan	Valid

Tabel III.43 Hasil Kasus Uji: SMARD-TC-12

Kode Kasus Uji		SMARD-TC-12		Penguji	Nela Rizky
Deskripsi Uji		Mengembalikan Tindak Lanjut (TUD)		Tanggal Uji	25 April
No	Form/Menu	Tahapan Uji	Data Uji	Output	Hasil Uji
1	Verifikasi Kasubag TUD	Klik tombol 		Sistem menampilkan <i>drawer</i> verifikasi	Valid
		Isikan form verifikasi	Catatan proses		Valid
		Klik tombol Kembalikan		Sistem menampilkan <i>pop-up</i> berhasil dikembalikan	Valid

Tabel III.44 Hasil Kasus Uji: SMARD-TC-13

Kode Kasus Uji		SMARD-TC-13		Penguji	Nela Rizky
Deskripsi Uji		Memfilter Arahkan		Tanggal Uji	25 April
No	Form/Menu	Tahapan Uji	Data Uji	Output	Hasil Uji
1	<i>Monitoring</i>	Klik menu Monitoring		Sistem menampilkan halaman <i>monitoring</i>	Valid
		Isikan parameter pada filter pencarian	Tahun awal Tahun akhir Nama kantor		Valid
		Klik tombol cari		Sistem menampilkan <i>card</i> status arahan dan grafik arahan pada satuan kerja DJBC	Valid

Tabel III.45 Hasil Kasus Uji: SMARD-TC-14

Kode Kasus Uji		SMARD-TC-14		Penguji	Nela Rizky
Deskripsi Uji		Membuka Detail Arahkan		Tanggal Uji	25 April
No	Form/Menu	Tahapan Uji	Data Uji	Output	Hasil Uji
1	<i>Monitoring</i>	Klik menu Monitoring		Sistem menampilkan halaman <i>monitoring</i>	Valid
		Klik salah satu <i>card</i> status arahan		Sistem menampilkan daftar arahan berdasarkan status yang dipilih	Valid

Tabel III.46 Hasil Kasus Uji: SMARD-TC-15

Kode Kasus Uji		SMARD-TC-15		Penguji	Nela Rizky
Deskripsi Uji		Merekam Hak Akses Pegawai		Tanggal Uji	25 April
No	Form/Menu	Tahapan Uji	Data Uji	Output	Hasil Uji
1	<i>Setting</i> Hak Akses	Klik menu Setting Hak Akses		Sistem menampilkan halaman <i>setting</i> hak akses	Valid
		Klik tombol Rekam		Sistem menampilkan form perekaman hak akses	Valid
		Isikan data hak akses pegawai	NIP Hak akses		Valid
		Klik tombol Simpan		Sistem menampilkan <i>pop-up</i> berhasil disimpan	Valid

3.5 Uraian Tugas

Dalam membangun proyek ini, tim pengembangan yang terdiri dari beberapa anggota yang memiliki peran dan tanggung jawab masing-masing untuk memastikan bahwa sistem dikembangkan secara terstruktur, efisien dan sesuai dengan kebutuhan pemilik proses bisnis.

3.5.1 Analis Sistem (System Analyst)

System Analyst selanjutnya disebut SA, bertugas untuk menganalisis kebutuhan dan spesifikasi teknis sistem, serta Menyusun dokumen perancangan sistem untuk disampaikan ke *programmer*. Dalam proyek ini, Avida Endriani bertugas sebagai SA dan bertanggung jawab terhadap perancangan kebutuhan sistem.

3.5.2 Pengembang (Programmer)

Programmer bertugas untuk mengembangkan rancangan sistem menjadi sebuah aplikasi, mulai dari membangun *API services*, manajemen koneksi dengan *database*, membangun antarmuka sistem, hingga memastikan integrasi data pada *backend* dan *frontend* sudah sesuai. Dalam proyek ini, Eki Pratama Wahyu Syahputra bertugas sebagai *Backend Programmer* dan Avida Endriani bertugas sebagai *Frontend Programmer*.

3.5.3 Penguji Sistem (Quality Control)

Quality Control atau selanjutnya disebut QC bertugas untuk melakukan pengujian sistem, mencatat dan melaporkan *bug/error* pada SA/*Programmer*, serta menentukan apakah sistem layak untuk diterapkan ke lingkungan nyata. Dalam proyek ini, Eki Pratama Wahyu Syahputra bertugas sebagai QC.