LAPORAN MILESTONE 4

“Sistem Bukti Dukung Administrasi BPS DKI Jakarta”

Disusun oleh:

3SI2 / Kelompok 3

Imalia Rosyida	(222112106)
Agape Bagus Rega Anggara	(222212455)
Blessy Munthia Purba	(222212537)
Muhammad Ilham Cahyanto	(222212756)
Reykhan Kharis Alexandro Batubara	(222212844)
Yulia Dhita Cahya Kumala	(222212925)

PROGRAM STUDI D-IV KOMPUTASI STATISTIK

POLITEKNIK STATISTIKA STIS

2025

DAFTAR ISI

DAFTAR ISI....1
BAB I....3
KONTEKS BISNIS....3
1.1 Tinjauan Organisasi....3
1.2 Masalah dan Peluang Bisnis....4
1.3 Dampak Bisnis....6
BAB II....9
RUANG LINGKUP PROYEK....9
2.1 Lingkup Solusi....9
2.1.1 Lingkup Solusi Bisnis - Process....9
2.1.2 Lingkup Solusi Bisnis - People....9
2.1.3 Lingkup Solusi Bisnis - Technology....10
2.2 Kebutuhan....10
2.2.1 Kebutuhan Bisnis....10
2.2.2 Kebutuhan Pemangku Kepentingan....11
2.3 Tinjauan Proyek....11
2.3.1 Deskripsi Proyek....11
2.3.2 Objektif....12
2.3.3 Pemangku Kepentingan Proyek....14
2.3.4 Kinerja Proyek....14
2.3.5 Asumsi Proyek....15
2.3.6 Batasan Proyek....15
2.3.7 Project Milestone....16
BAB III....17
MODEL KEBUTUHAN....17
3.1 Kebutuhan Fungsional dan Non Fungsional....17
3.1.1 Kebutuhan Fungsional....17
3.1.2 Kebutuhan Non-Fungsional....20
3.2 Skenario....20
3.2.1 Skenario Operator (Pegawai BPS Provinsi DKI Jakarta)....20
3.2.2 Skenario Tim Keuangan....22
3.3 Implementasi Proses Bisnis....23
3.3.1 Operator (Pegawai BPS Provinsi DKI Jakarta)....24
3.3.2 Tim Keuangan....26
3.4 Use Case dan Use Case Description....29
3.4.1 Use Case....29
3.4.2 Use Case Description:....29
3.5 Rancangan Basis Data....47
3.5.1 Database bda....48
3.5.2 Tabel akun belanja....48
3.5.3 Tabel akun file keuangan....48
3.5.4 Tabel akun file operator....48
3.5.5 Tabel form pengajuan....49
3.5.6 Tabel kegiatan....49
3.5.7 Tabel komponen....49
3.5.8 Tabel kro....49

3.5.9 Tabel output....50
3.5.10 Tabel pegawai....50
3.5.11 Tabel sub komponen....50
3.5.12 Tabel users....50
3.5.13 Tabel file buku panduan....51
3.5.14 Tabel file upload keuangan....51
3.5.15 Tabel file upload operator....51
3.5.16 Tabel form keuangan....51
3.5.17 Tabel jenis file keuangan....52
3.5.18 Tabel jenis file operator....52
3.5.19 Tabel role....52
3.5.20 Tabel status pengajuan....52
3.6 UI/UX Wireframe dan Mock-up:....52
3.6.1 Landing page....53
3.6.2 Login....54
3.6.3 Operator....55
3.6.4 Tim Keuangan....61
3.6.5 Admin....65
3.7 Arsitektur System....71
3.7.1 Rancangan Arsitektur Sistem....71
3.7.2 Penjelasan Rancangan Arsitektur Sistem....71
BAB IV....73
DESAIN PROGRAM....73
4.1 User Access Matrix....73
4.2 Desain Program....74
4.3 Identifikasi Resiko Sistem, Sisi Keamanan, Performa, dan Keandalan....94
BAB V....99
IMPLEMENTASI SISTEM....99
5.1 Hasil Implementasi Sistem....99
5.2 Akun untuk Masuk ke Sistem....99
BAB VI....100
PENGUJIAN SISTEM....100
6.1 Penjelasan terkait Pengujian Sistem....100
6.1.1 Scenario Testing Sistem....101
6.1.2 Instrumen atau Kuesioner yang Digunakan....104
6.1.3 Analisis dan Hasil Kuesioner Testing....105
6.1.4 Review Client yang Melakukan Testing....108
BAB VII....113
KESIMPULAN....113
LAMPIRAN....114
Tautan web....114
Link git-stis....114
Link record zoom Milestone 2....114
Link record zoom Milestone 3....114
Link record zoom Milestone 4....114

BAB I
KONTEKS BISNIS

1.1 Tinjauan Organisasi

Badan Pusat Statistik (BPS) adalah lembaga pemerintah non-kementerian yang bertanggung jawab dalam menyediakan data dan informasi statistik untuk mendukung perencanaan, pelaksanaan, dan evaluasi kebijakan pemerintah. BPS DKI Jakarta sebagai bagian dari struktur BPS nasional memiliki misi untuk menyediakan data dan informasi statistik yang andal, akurat, dan tepat waktu guna memenuhi kebutuhan pemerintah daerah serta masyarakat DKI Jakarta. Selain itu, BPS DKI Jakarta juga berperan dalam mendukung pelaksanaan kebijakan statistik nasional yang diimplementasikan di wilayahnya.

Visi strategis BPS DKI Jakarta adalah "Menjadi penyedia data statistik yang berkualitas untuk mendukung perencanaan pembangunan yang lebih baik di DKI Jakarta." Dalam upaya mencapai visi tersebut, BPS DKI Jakarta memiliki beberapa sasaran dan tujuan layanan strategis, antara lain: (1) Meningkatkan kualitas dan akurasi data yang dikumpulkan; (2) Memperkuat peran statistik dalam pengambilan keputusan publik; dan (3) Mengoptimalkan pemanfaatan data dan informasi statistik untuk mendukung pembangunan daerah yang berkelanjutan. Visi dan misi ini sejalan dengan misi BPS nasional, yang menekankan pentingnya data berkualitas dalam pengambilan kebijakan dan perencanaan pembangunan.

Saat ini, aktivitas dan layanan utama BPS DKI Jakarta meliputi pengumpulan data melalui survei dan sensus, penyusunan laporan statistik, serta penyediaan layanan konsultasi statistik kepada pemerintah daerah, lembaga swasta, dan masyarakat umum. BPS DKI Jakarta memiliki pemangku kepentingan utama seperti pemerintah provinsi DKI Jakarta, kementerian dan lembaga pemerintah lainnya, serta masyarakat luas yang memanfaatkan data statistik dalam perencanaan dan pengambilan keputusan. Untuk mengoptimalkan layanan, BPS DKI Jakarta juga bekerja sama dengan perguruan tinggi dan lembaga riset lainnya dalam pengembangan metodologi dan analisis statistik.

Struktur organisasi BPS DKI Jakarta terdiri dari beberapa bidang yang menangani berbagai fungsi utama, seperti pengumpulan data, analisis statistik, dan layanan informasi publik. Selain itu, ada pula dukungan dari unit manajemen sumber daya yang menangani administrasi, sumber daya manusia, serta keuangan. Setiap bidang dipimpin oleh seorang kepala bidang yang bertanggung jawab langsung kepada Kepala BPS DKI Jakarta.

Dalam hal kapasitas yang ada, BPS DKI Jakarta memiliki sumber daya manusia yang terdiri dari pegawai dengan latar belakang pendidikan di bidang statistik dan ilmu terkait lainnya. Sumber daya keuangan diperoleh dari anggaran pemerintah pusat yang dialokasikan untuk mendukung kegiatan operasional dan pengembangan layanan statistik. Meskipun demikian, untuk memenuhi tuntutan dan meningkatkan efisiensi layanan, BPS DKI Jakarta memerlukan pengembangan sistem informasi yang lebih modern, salah satunya adalah sistem yang diusulkan dalam proyek ini, yaitu "Sistem Bukti Dukung Administrasi."

1.2 Masalah dan Peluang Bisnis

Masalah

Permasalahan utama yang muncul dalam proses bisnis program Pembangunan Sistem Bukti Dukung Administrasi yang selama ini dilakukan masih dilakukan secara konvensional dengan google drive dan tidak memiliki sistem keamanannya. Dokumen-dokumen yang bisa diakses, diupload, maupun dihapus oleh semua orang menjadi salah satu permasalahan. Untuk mempermudah kegiatan administrasi keuangan dan menghemat waktu pengecekan berkas maka dibutuhkan sistem yang lebih terorganisir menimbang banyaknya kegiatan-kegiatan akun belanja di administrasi keuangan BPS DKI Jakarta.

Harapan

Sistem yang diusulkan diharapkan dapat mengatasi permasalahan dengan menyediakan platform digital untuk mendukung proses administrasi bukti dukung secara terintegrasi dan terpusat. Sistem ini akan memfasilitasi penyimpanan, pengolahan, dan akses dokumen secara efisien, serta meningkatkan kolaborasi antar unit kerja melalui fitur berbagi informasi dan pemantauan proses secara real-time. Dengan adanya sistem ini, diharapkan proses validasi dan verifikasi data menjadi

lebih cepat, transparan, dan akurat, serta mendukung pengambilan keputusan yang lebih efektif.

Untuk meningkatkan efisiensi proses bisnis program Pembangunan Sistem Bukti Dukung Administrasi, maka diharapkan terdapat sistem yang memiliki kemampuan berikut:

1. Dapat mewujudkan sinergi ASN yang adaptif dan kolaboratif di BPS Provinsi DKI Jakarta, dengan terus berinovasi dalam penatausahaan dokumen dan arsip keuangan.
2. Dapat menyediakan pencadangan data yang tersusun secara efektif, efisien, dan sistematis.
3. Dapat menerapkan praktik peduli lingkungan (tidak perlu print berkas lagi), memiliki akses security dan proses bisnis yang berkelanjutan.

Peluang

Berdasarkan analisis masalah dan uraian hasil yang diharapkan, berikut ialah aspek-aspek yang dapat dinilai sebagai suatu pengembangan peluang bisnis:

a. **Time (Waktu)**

   Penerapan sistem ini dapat menghemat waktu dalam proses administrasi dan pemrosesan dokumen dengan mengotomatisasi dan memusatkan proses yang sebelumnya dilakukan secara manual.

b. **Cost (Biaya)**

   Sistem ini berpotensi mengurangi biaya operasional yang dihabiskan untuk pencetakan dan distribusi dokumen fisik, serta menghemat sumber daya yang sebelumnya dialokasikan untuk proses manual.

c. **Speed (Kecepatan)**

   Sistem ini akan mempercepat proses verifikasi dan validasi dokumen serta koordinasi antar unit kerja, sehingga meningkatkan efisiensi dan responsivitas layanan administrasi.

d. **Transparency (Keterbukaan)**

   Dengan adanya sistem ini, seluruh proses dan status administrasi bukti dukung dapat dipantau secara real-time, memungkinkan pemantauan dan penelusuran yang lebih akurat dan terbuka bagi seluruh pemangku kepentingan.

1.3 Dampak Bisnis

Proyek "Sistem Bukti Dukung Administrasi BPS DKI Jakarta" akan berdampak pada satuan kerja dalam beberapa aspek, seperti alat, proses, peran dan tanggung jawab, serta perangkat keras/perangkat lunak. Berikut penjelasannya:

1. **Alat**: Pengembangan sistem ini akan mengubah tata kelola operasional dalam pengolahan dan penyimpanan dokumen administrasi bukti dukung dari yang sebelumnya dikelola secara manual menjadi terkomputerisasi dan terpusat. Sistem ini memungkinkan unit-unit kerja terkait di BPS DKI Jakarta untuk mengakses dan memproses informasi secara cepat dan efisien.
2. **Proses:** Implementasi proyek ini akan menyederhanakan dan mengoptimalkan proses pengolahan bukti dukung. Sistem akan mendigitalisasi alur kerja, mulai dari pengumpulan hingga verifikasi dokumen, sehingga dapat mengurangi waktu dan sumber daya yang diperlukan serta meningkatkan transparansi dalam pemantauan proses administrasi.
3. **Peran dan Tanggung Jawab**: Sistem baru ini akan mengubah peran beberapa pihak terkait di BPS DKI Jakarta. Staf yang sebelumnya bertanggung jawab atas pengelolaan dokumen secara manual akan dialihkan perannya menjadi pengelola dan pengguna sistem digital. Selain itu, tanggung jawab untuk memastikan keakuratan dan validitas data akan lebih mudah dipantau dan dikoordinasikan antar unit kerja melalui fitur-fitur yang disediakan oleh sistem.
4. **Perangkat Keras / Perangkat Lunak:** Penerapan sistem ini memerlukan infrastruktur teknologi informasi yang memadai, seperti server yang aman dan perangkat lunak manajemen dokumen yang terintegrasi. Selain itu, perangkat lunak berbasis web akan dikembangkan untuk memudahkan akses bagi pengguna di berbagai unit kerja BPS DKI Jakarta.

Dampak - Existing - Berikut merupakan dampak belum diterapkannya proyek terhadap 3 aspek (Manusia, Proses, Teknologi)

Manusia

negatif :

- Staf mengalami kesulitan dalam mengelola dokumen secara efektif karena proses manual yang memakan waktu dan tenaga.

- Koordinasi antar unit kerja menjadi lambat dan kurang efisien karena ketidaktersediaan akses dokumen yang terpusat.

positif :

- -

Proses

negatif :

- Proses pengolahan dan verifikasi dokumen memerlukan waktu yang lama, mengakibatkan keterlambatan dalam pengambilan keputusan.
- Minimnya transparansi dalam alur proses administrasi menghambat pemantauan status dokumen.

positif :

- -

Teknologi

negatif :

- Belum adanya sistem informasi terpusat untuk mengelola administrasi bukti dukung secara digital.

positif :

- -

Dampak - To Be - Berikut merupakan dampak penerapan proyek terhadap 3 aspek (Manusia, Proses, Teknologi)

Manusia

positif :

- Staf dapat lebih mudah mengakses dan mengelola dokumen administrasi melalui sistem yang terintegrasi.
- Peningkatan koordinasi antar unit kerja dalam memantau dan memverifikasi dokumen secara real-time.

negatif :

- Diperlukan pelatihan bagi staf untuk beradaptasi dengan sistem baru yang diterapkan.
- Dibutuhkan tim teknis khusus untuk pemeliharaan dan perbaikan sistem agar dapat beroperasi dengan optimal.

Proses

positif :

- Proses pengelolaan dan verifikasi dokumen menjadi lebih cepat dan efisien, dengan pemantauan status dokumen yang dapat dilakukan secara real-time.
- Sistem ini meningkatkan efisiensi dan transparansi dalam alur kerja administrasi.

negatif :

- -

Teknologi

positif :

- Sistem ini menyediakan dashboard untuk memantau status dokumen secara real-time dan memungkinkan akses yang mudah di berbagai unit kerja.
- Proses administrasi dapat dilakukan secara otomatis dan fleksibel kapanpun dan dimanapun melalui platform web.

negatif :

- -

BAB II
RUANG LINGKUP PROYEK

2.1 Lingkup Solusi

2.1.1 Lingkup Solusi Bisnis - Process

Solusi bisnis	Proses
Fungsional layanan bisnis	Proses yang digunakan saat ini (input data, validasi, checking data) masih terhitung kuno atau dilakukan secara manual dengan mengandalkan Google Sheet. Akan dibentuk suatu sistem yang memanfaatkan web, sehingga semua orang yang berkepentingan bisa mengaksesnya dimana saja dengan melalui mekanisme login yang akan membuat sistem BDA ini aman.
Fungsional layanan data	Saat ini pengelolaan data yang digunakan masih terhitung konvensional dan tidak terlalu aman, karena hanya menggunakan Google Form, Google Drive, dan Google Sheet saja sebagai media pengumpulan dan penyimpanan data-data BDA. Akan dibuatkan suatu sistem baru yang akan mencangkup form, dan penyimpanan data yang terdigitalisasi dengan baik, serta fitur-fitur keamanan yang membuat data akan terlindungi dengan baik, sehingga tidak terjadi kebocoran data dan mempermudah petugas PPK untuk melakukan cross check.

2.1.2 Lingkup Solusi Bisnis - People

Solusi bisnis	Proses
Klien/ Produsen	Penanggung jawab unit kerja PPK BPS Provinsi DKI Jakarta
Pengelola	Unit kerja PPK dan IPDS BPS Provinsi DKI Jakarta
Pengguna	Pegawai BPS DKI Jakarta

2.1.3 Lingkup Solusi Bisnis - Technology

Solusi bisnis	Proses
Alur kerja	Aturan bisnis atau alur data perlu dirancang dengan dasar hukum yang kuat untuk mengatur manajemen operasional dan sistematisasi persiapan, perencanaan, perancangan, pelaksanaan, pemantauan, serta evaluasi kegiatan operasional. Hal ini bertujuan agar aturan tersebut dapat dijadikan pedoman oleh pemangku kepentingan dalam menetapkan kebijakan terkait penggunaan sistem manajemen operasional di masa mendatang.
Fungsional aplikasi	Fitur-fitur dalam sistem manajemen kegiatan operasional dirancang dan dikembangkan berdasarkan kebutuhan bisnis atau pengguna, sehingga mampu memenuhi berbagai keperluan yang terkait dengan perencanaan dan pelaksanaan kegiatan.
Non Fungsional aplikasi	Persyaratan non-fungsional aplikasi sistem dirancang dan dikembangkan sesuai dengan kebutuhan pengguna agar aplikasi dapat beroperasi dengan optimal.
Perawatan aplikasi	Perawatan website dirancang agar sistem manajemen kegiatan operasional dapat terus berkembang seiring dengan peningkatan kebutuhan bisnis.

2.2 Kebutuhan

2.2.1 Kebutuhan Bisnis

Beberapa Business Requirement yang diperlukan untuk dapat mewujudkan business need, dijabarkan sebagai berikut :

1. Perubahan proses yang dimulai pengajuan bukti dukung administrasi, pengecekan bukti dukung administrasi maupun penyetujuan oleh tim keuangan dilakukan menggunakan sistem terintegrasi dan kolaboratif.
2. Karena sifat penyelenggaraan sistem yang kolaboratif namun juga memiliki batasan akses, perlu adanya model autentikasi user yang berwenang dan leveling view dari layanan informasi kegiatan itu sendiri sesuai role user nya.

3. Sistem harus dapat memetakan keterlibatan entitas satuan kerja ke masing-masing kegiatan yang dia ikuti secara efektif.
4. Oleh karena melibatkan pihak ketiga untuk proses pengembangan sistem informasi kegiatan, maka keamanan informasi menjadi pertimbangan penting dalam pengembangan aplikasi.
5. Sistem dapat digunakan selama 24 jam x 7 hari.
6. Sistem harus dapat memiliki model pengembangan secara modular dan tidak mengganggu jalannya layanan ketika terjadi proses enhancement terhadap fitur dan layanan sistem kedepannya.

2.2.2 Kebutuhan Pemangku Kepentingan

1. Adanya mekanisme log aktivitas kegiatan dan registrasi bukti dukung administrasi BPS Provinsi DKI Jakarta
2. Adanya fitur reminder by emailing dari sistem kepada entitas terdaftar terkait kegiatan rapat yang dijadwalkan. (1 week before, 3 days before)
3. Kemudahan penggunaan sistem layanan informasi kegiatan
4. Dapat digunakan di berbagai platform
5. Sistem dapat dikembangkan lebih lanjut dengan tersedianya dokumentasi pengembangan sistem yang baik.

2.3 Tinjauan Proyek

2.3.1 Deskripsi Proyek

Proyek “Bukti Dukung Administrasi (BDA) 2024” di Badan Pusat Statistik (BPS) Provinsi DKI Jakarta bertujuan untuk mengembangkan sistem digital yang aman, efisien, dan ramah lingkungan dalam penatausahaan dokumen administrasi keuangan. Proyek ini akan memfasilitasi proses pencadangan dan pengelolaan data keuangan secara sistematis, serta mendorong kolaborasi antar ASN di lingkungan BPS DKI Jakarta. Sistem ini diharapkan mengatasi tantangan yang ada dalam penggunaan dokumentasi berbasis kertas dan platform seperti Google Drive, yang dinilai kurang aman dan tidak efisien dalam penatausahaan data keuangan.

Proyek ini akan menggunakan pendekatan yang menekankan pada keamanan data, efisiensi penyimpanan, dan keberlanjutan lingkungan. Sistem yang dikembangkan

berbasis web dan akan dirancang dengan arsitektur yang mampu mengelola dokumen administrasi keuangan secara terpusat, aman, dan sesuai dengan prinsip bisnis berkelanjutan. Dalam konteks ini, proyek akan mengintegrasikan praktik teknologi informasi yang peduli lingkungan, seperti pengurangan penggunaan kertas dan optimalisasi sumber daya digital, untuk menciptakan proses bisnis yang lebih hijau.

Proyek "Bukti Dukung Administrasi (BDA) 2024" di BPS DKI Jakarta akan dilaksanakan melalui beberapa langkah yang terstruktur untuk mencapai tujuan yang diinginkan. Langkah pertama adalah analisis kebutuhan dan perencanaan, di mana kebutuhan fungsional sistem akan diidentifikasi, serta dilakukan inventarisasi dokumen keuangan yang sering digunakan dalam kegiatan administrasi. Selanjutnya, pada tahap desain sistem, arsitektur sistem akan dikembangkan dengan fokus pada keamanan, termasuk fitur pencadangan otomatis, enkripsi data, dan pengaturan hak akses yang disesuaikan untuk berbagai level pengguna ASN. Setelah itu, sistem berbasis web akan dikembangkan dan diimplementasikan dengan pengembangan dan implementasi, yang memungkinkan akses dan kolaborasi dokumen secara terintegrasi dengan antarmuka pengguna yang mudah dipahami oleh semua pegawai. Tahap berikutnya adalah uji coba dan validasi, di mana sistem akan diuji untuk memastikan keamanan dan efisiensi penyimpanan, serta memastikan bahwa setiap pengguna dapat mengakses fitur sesuai perannya.

Proyek ini bertujuan untuk mewujudkan penatausahaan dokumen administrasi keuangan yang lebih aman, efektif, dan efisien, serta mendorong kolaborasi adaptif antar ASN dalam pengelolaan dokumen di lingkungan BPS DKI Jakarta. Selain itu, proyek ini juga akan mengimplementasikan praktik ramah lingkungan melalui pengurangan penggunaan kertas dan penerapan proses bisnis yang berkelanjutan. Akhirnya, proyek ini diharapkan mampu membangun sistem pencadangan yang tersusun secara sistematis dengan pengelolaan hak akses yang aman, sehingga dapat melindungi integritas data administrasi keuangan.

2.3.2 Objektif

Proyek “Bukti Dukung Administrasi (BDA) 2024” secara langsung mendukung beberapa tujuan dan sasaran bisnis organisasi BPS DKI Jakarta. Tabel berikut

mencantumkan sasaran dan sasaran bisnis yang didukung oleh Proyek ini dan bagaimana mendukungnya:

Objektif Bisnis	Deskripsi
Efisiensi dan Keamanan Penatausahaan Dokumen Keuangan	Proyek BDA 2024 bertujuan untuk menciptakan sistem pengelolaan dokumen yang lebih efisien dan aman. Sistem ini akan menggantikan metode berbasis kertas dan platform seperti Google Drive dengan solusi digital yang terstruktur. Efisiensi dicapai melalui otomatisasi pencadangan, pengelompokan dokumen, serta fitur pengelolaan akses yang aman.
Mengimplementasikan Praktik Bisnis Berkelanjutan dan Ramah Lingkungan	Proyek ini mendukung kebijakan keberlanjutan BPS dengan mengurangi penggunaan kertas dalam administrasi. Dengan digitalisasi dokumen, BPS DKI Jakarta mengurangi jejak karbon dari produksi dan pembuangan kertas. Sistem ini juga dirancang agar lebih efisien dalam penggunaan energi untuk pemrosesan dokumen digital.
Mendorong Kolaborasi yang Lebih Efektif Antar ASN	Proyek ini akan menyediakan platform kolaborasi terintegrasi yang memungkinkan ASN BPS DKI Jakarta untuk berkolaborasi dengan lebih efektif. Hak akses diatur secara cermat sehingga dokumen dapat dibagikan secara aman dan sesuai peran, mengurangi hambatan komunikasi, dan mempercepat penyelesaian tugas lintas departemen.
Memperkuat Sistem Pencadangan dan Pemulihan Data	Sistem BDA 2024 akan menerapkan pencadangan otomatis dan berkala untuk memastikan keamanan dokumen keuangan penting. Sistem ini juga memiliki fitur pemulihan data, yang memungkinkan pengguna mengakses kembali dokumen yang hilang atau rusak, bahkan dalam situasi darurat atau kerusakan sistem.

Mendukung Transformasi Digital di BPS DKI Jakarta	Proyek ini merupakan bagian dari upaya transformasi digital di BPS DKI Jakarta, dengan menerapkan sistem manajemen dokumen berbasis web. Sistem ini memperkuat digitalisasi administrasi di instansi pemerintah dan selaras dengan kebijakan transformasi digital nasional, dengan target mengintegrasikan 100% dokumen administrasi pada akhir 2024.

2.3.3 Pemangku Kepentingan Proyek

No	Stakeholder (nama jabatan)	Peran dalam proyek
1	Kepala BPS Provinsi DKI Jakarta	Kuasa Pengguna Anggaran
2	Kepala Bagian Umum	Penerima Hasil Pekerjaan
3	PPK	Perencana Kegiatan Pengembangan Sistem
4	Pegawai BPS	Pengguna

2.3.4 Kinerja Proyek

Sumber Daya Utama / Proses / Layanan	Ukuran Kinerja
Aplikasi	Tersedianya aplikasi berbasis web untuk pengelolaan data administrasi yang mempermudah pengumpulan, penyimpanan, dan pengelolaan dokumen secara digital, dengan akses yang cepat.
Pengelolaan Dokumen Berbasis Web	Pengurangan penggunaan kertas hingga 90% dan peningkatan efisiensi akses dokumen sebesar 50%, dengan akses real-time ke dokumen yang dikelola secara terpusat dan aman.
Kolaborasi Antar ASN dalam Pengelolaan Dokumen	Peningkatan produktivitas kolaborasi sebesar 30%, dengan pengurangan waktu penyelesaian dokumen lintas departemen dan pengelolaan tugas bersama secara digital.

Enkripsi dan Manajemen Hak Akses	Pengurangan kebocoran data hingga 95% melalui sistem enkripsi dan pengelolaan hak akses yang membatasi akses dokumen keuangan hanya untuk pengguna yang berwenang.
Pengurangan Kebutuhan Pengelolaan Arsip Manual	Pengurangan hingga 80% pekerjaan manual terkait pengelolaan arsip fisik, dengan pengelolaan dokumen yang sepenuhnya berbasis digital melalui platform BDA 2024.
Peningkatan Keamanan Data	95% peningkatan dalam keamanan sistem melalui enkripsi dan pengelolaan akses, serta pengurangan kejadian pelanggaran data yang melibatkan dokumen keuangan dan administratif.

2.3.5 Asumsi Proyek

- Semua ASN dan pegawai yang terlibat akan dilatih
- Pendanaan tersedia untuk pelatihan dan pengembangan kapasitas
- Pendanaan tersedia untuk perangkat keras dan perangkat lunak
- Konektivitas internet yang memadai di seluruh kantor BPS DKI Jakarta
- Dukungan penuh dari pimpinan dan manajer proyek
- Proyek memiliki dukungan eksekutif dari manajemen puncak BPS
- Data administrasi keuangan yang ada dapat diintegrasikan ke dalam sistem baru
- Pengguna akan menerima dan beradaptasi dengan sistem baru
- Perangkat keamanan siber yang memadai tersedia untuk menjaga integritas data:
- Waktu dan jadwal proyek sesuai dengan target yang ditentukan

2.3.6 Batasan Proyek

- *Migrasi data administrasi dan keuangan dari sistem lama ke sistem baru dapat menghadapi tantangan teknis dan operasional, terutama dalam memastikan semua data dimigrasikan dengan aman dan tanpa kehilangan.*
- *Beberapa ASN mungkin menghadapi resistensi atau kesulitan dalam beradaptasi dengan sistem manajemen dokumen yang sepenuhnya digital, sehingga memerlukan upaya lebih dalam hal perubahan budaya kerja.*

- *Sistem baru mungkin menghadapi keterbatasan dalam hal kemampuan untuk menangani volume data besar, terutama jika volume dokumen keuangan yang dikelola terus meningkat secara signifikan.*
- *Proyek harus mematuhi semua regulasi terkait keamanan data dan privasi, yang bisa memperketat persyaratan implementasi dan memerlukan penyesuaian sistem lebih lanjut.*

2.3.7 Project Milestone

Kegiatan ini berlangsung selama 14 (empat belas) minggu kalander, yang akan dikerjakan selama akhir September 2024 - akhir Januari 2025 (jadwal mulai dapat berubah sesuai pelaksanaan kegiatan pengembangan sistem).

No	Kegiatan	M1-M2	M3	M4-M12	M13	M14
1.	Menyusun Project Plan					
2.	Menyusun SRS					
3.	Membuat Sistem Keseluruhan					
4.	Pelaksanaan Testing Sistem					
5.	Penyerahan Hasil					

BAB III

MODEL KEBUTUHAN

3.1 Kebutuhan Fungsional dan Non Fungsional

3.1.1 Kebutuhan Fungsional

Kode	Modul	Kode Fungsional	Kebutuhan Fungsional	Deskripsi
AL	Autentikasi dan Login	AL01	Autentikasi dan Otorisasi (Login)	Pegawai BPS Provinsi DKI Jakarta dapat melakukan login menggunakan email bps dan password kemudian dilakukan pengecekan apakah pengguna valid atau tidak
		AL02	Membatasi dan membagi akses pada masing-masing role	Setiap role memiliki batasan akses data yang berbeda. Role "operator" memiliki akses hanya pada pengajuan dan dokumen miliknya. Role "tim keuangan" memiliki akses pada semua pengajuan dan dokumen yang diupload oleh semua operator. Role "admin" memiliki akses pada semua database dan memiliki hak untuk mengelola aplikasi.
PF	Profile	PF01	Menyediakan fitur profile.	Pegawai BPS Provinsi DKI Jakarta dapat melihat nama dan role nya sebagai apa
		PF02	Menyediakan fitur keluar/logout	Pegawai BPS Provinsi DKI Jakarta dapat melakukan aksi mengeluarkan akun.
RA	Role-Admin	RA02	Manage User	Pejabat Pengelola Keuangan (PPK) BPS Provinsi DKI Jakarta dapat menambah akun pengguna, mengubah akun pengguna, hingga mengatur role

				pengguna
		RA03	Manage MAK	Pejabat Pengelola Keuangan (PPK) BPS Provinsi DKI Jakarta memiliki akses ke semua data dalam database dan dapat menonaktifkan data.
FP	Form Pengajuan	FP01	Menginput rincian kegiatan	Pegawai BPS Provinsi DKI Jakarta mengisi form pengajuan dengan menginput rincian kegiatan.
MO	Monitoring Operator (Pegawai BPS Provinsi DKI Jakarta)	MO01	View pengajuan	Pegawai BPS Provinsi DKI Jakarta dapat menampilkan detail kegiatan dari form pengajuan.
		MO02	Edit pengajuan	Pegawai BPS Provinsi DKI Jakarta dapat mengedit pengajuan pada form pengajuan dengan isian yang telah diisikan jika status berubah dari "pengecekan dokumen" menjadi "entri operator".
		MO03	Hapus pengajuan	Pegawai BPS Provinsi DKI Jakarta dapat menghapus pengajuan yang telah diajukan.
		MO04	Monitoring file	Pegawai BPS Provinsi DKI Jakarta melakukan upload file.- file yang dibutuhkan pada pengajuan.
		MO05	Fitur status	Pegawai BPS Provinsi DKI Jakarta dapat melihat progress dari masing masing pengajuan yang telah dilakukan, di status: Entri operator, Pengecekan Dokumen, Proses Pembayaran, dan Selesai dari tim keuangan.
		MO06	Fitur filter dan	Pegawai BPS Provinsi DKI Jakarta

			pencarian	dapat menampilkan halaman tabel sesuai filter yang diterapkan, yang mana filter yang disediakan filter tahun dan akun belanja dari semua pengajuan. Selain itu operator dapat melakukan pencarian berdasarkan query.
MK	Monitoring Keuangan	MK01	View pengajuan	Tim keuangan dapat menampilkan detail kegiatan yang diajukan.
		MK02	Monitoring file	Tim keuangan memeriksa dokumen yang diupload oleh operator dan memberikan persetujuan (Approve atau Reject pengajuan)
		MK03	Fitur status	Tim keuangan dapat melihat status pengajuan yang diajukan operator, yakni Approved dan Rejected
		MK04	Form Tim Keuangan	Tim keuangan mengisi form khusus tim keuangan dan mengupload berkas berkas yang diperlukan.
		MK05	Fitur filter dan pencarian	Tim Keuangan dapat menampilkan halaman tabel sesuai filter yang diterapkan, yang mana filter yang disediakan filter tahun dan akun belanja dari semua pengajuan. Selain itu juga dapat melakukan pencarian berdasarkan query.
DW	Download	DW01	Download rincian kegiatan	Pegawai BPS Provinsi DKI Jakarta dapat mengunduh tabel rincian kegiatan yang telah diajukan.

3.1.2 Kebutuhan Non-Fungsional

Kode	Modul	Kebutuhan Non-fungsional	Deskripsi
KJ	Kinerja	Kinerja Sistem	Sistem dibuild menggunakan PHP dengan framework laravel dan database MySQL. Sistem akan diuji dengan Performance Testing.
KM	Keamanan	Sistem	Sistem dapat mencegah serangan dan akses luar yang tidak diizinkan.
KG	Kegunaan (usability)	Antarmuka	Sistem memiliki tampilan yang mudah dimengerti dan dioperasikan oleh semua pengguna. Sistem akan duji dengan Usability Testing.

3.2 Skenario

3.2.1 Skenario Operator (Pegawai BPS Provinsi DKI Jakarta)

a. Operator (Pegawai BPS Provinsi DKI Jakarta) membuka browser dan mengetik URL web BDA
b. Browser akan menampilkan halaman utama web dan melakukan login dengan akun BPS
c. Operator (Pegawai BPS Provinsi DKI Jakarta) berhasil login dan berada di halaman dashboard operator yang memiliki fitur:
   - Form Pengajuan
     1) Operator (Pegawai BPS Provinsi DKI Jakarta) memilih menu pengajuan form
     2) Web akan menampilkan halaman form isi pengajuan
     3) Operator (Pegawai BPS Provinsi DKI Jakarta) mengisi rincian keterangan kegiatan di form pengajuan
     4) Operator (Pegawai BPS Provinsi DKI Jakarta) melakukan submit pada form pengajuan
     5) Selesai

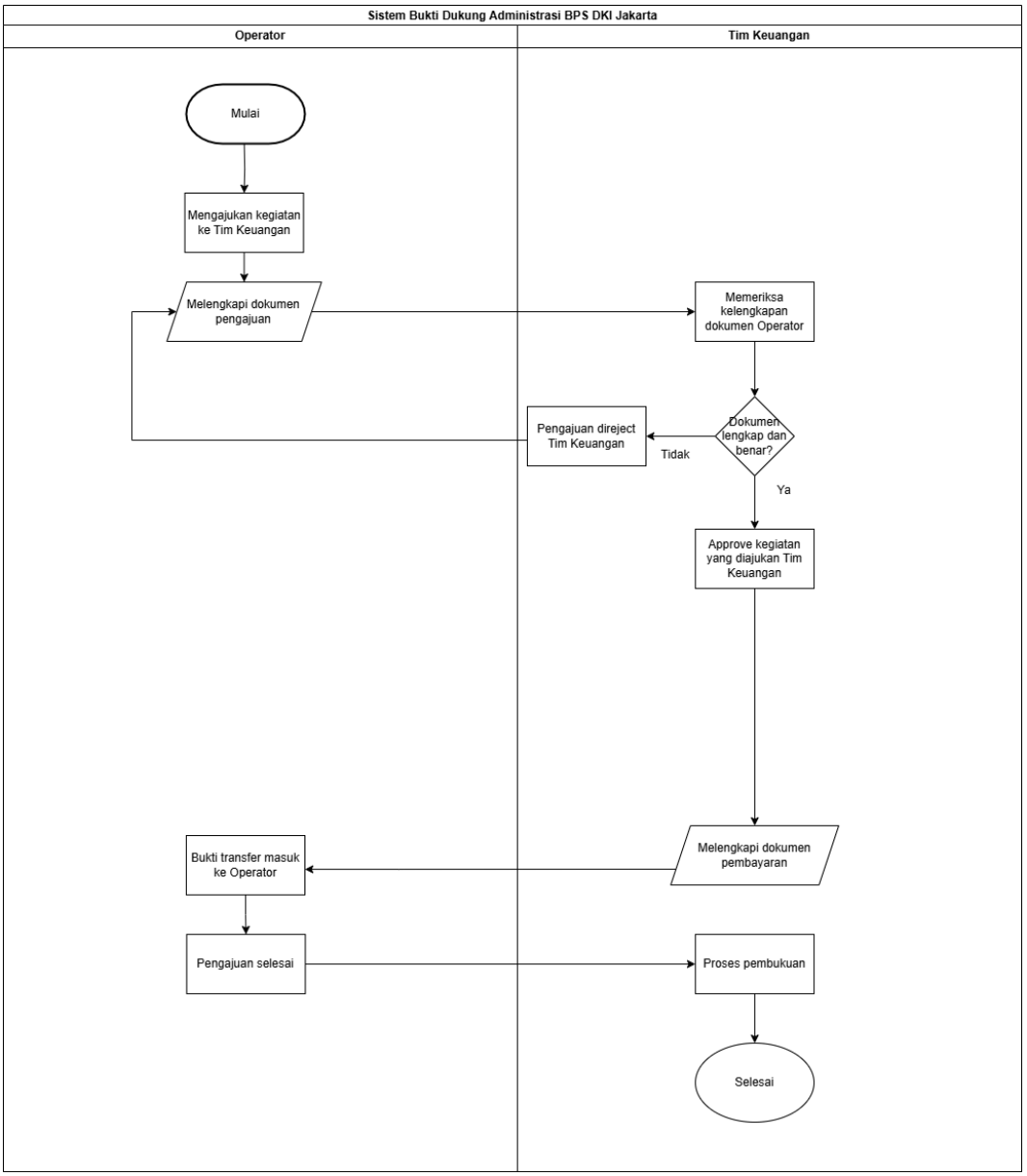
- Monitoring Pengajuan
  1) Operator (Pegawai BPS Provinsi DKI Jakarta) memilih menu monitoring pengajuan
  2) Web akan menampilkan tabel monitoring pengajuan
  3) Pada masing-masing baris pengajuan, operator dapat menggunakan aksi/fitur:
     a) Fitur View pengajuan
        Menampilkan detail kegiatan dari form pengajuan.
     b) Fitur Edit pengajuan
        Operator (Pegawai BPS Provinsi DKI Jakarta) dapat mengedit pengajuan pada form pengajuan dengan isian yang telah diisikan.
     c) Fitur Hapus pengajuan
        Operator (Pegawai BPS Provinsi DKI Jakarta) dapat menghapus pengajuan yang telah diajukan.
     d) Fitur Monitoring file
        - ➔ Browser akan menampilkan halaman tabel upload file.
        - ➔ Operator (Pegawai BPS Provinsi DKI Jakarta) melakukan upload file.-file yang dibutuhkan pada pengajuan.
        - ➔ Operator (Pegawai BPS Provinsi DKI Jakarta) tidak bisa klik kirim jika semua file belum lengkap.
        - ➔ Setelah semua file yang dibutuhkan lengkap, operator dapat mengirim semua dokumen, lalu dikembalikan lagi ke laman tabel monitoring pengajuan
        - ➔ Setelah itu dilakukan pengecekan oleh tim keuangan. Kemudian status akan berubah menjadi pengecekan dokumen.
     e) Fitur Status
        Operator (Pegawai BPS Provinsi DKI Jakarta) dapat melihat progress dari masing masing pengajuan yang telah dilakukan, seperti:
        - Entri Operator (Pegawai BPS Provinsi DKI Jakarta)
        - Pengecekkan Dokumen
        - Rejected Tim Keuangan
        - Proses Pembayaran
        - Selesai
  4) Setelah file-file di kirim pada monitoring file, operator tidak dapat mengedit atau menghapus pengajuan.
  5) Apabila pengajuan ditolak oleh tim keuangan, operator dapat merevisi (mengedit atau menghapus) file-file yang telah diupload sesuai dengan catatan dari tim keuangan.

6) Apabila pengajuan di *approve*, operator akan menerima pemberitahuan berupa *email* dan notifikasi pada web, serta dapat melihat bukti transfer yang dikirim oleh tim keuangan.

3.2.2 Skenario Tim Keuangan

a. Tim Keuangan membuka browser dan mengetik URL web BDA
b. Browser akan menampilkan halaman utama web
c. Tim Keuangan berada di halaman utama dan melakukan login dengan akun BPS
d. Tim Keuangan berhasil login dan berada di halaman dashboard tim keuangan yang memiliki fitur:
   - Monitoring Pengajuan
     1. Tim Keuangan memilih menu monitoring pengajuan
     2. Web menampilkan Tabel Monitoring Pengajuan
     3. Pada masing-masing baris tabel pengajuan, tim keuangan dapat menggunakan aksi/fitur:
        a. View pengajuan
           Menampilkan detail pengajuan yang telah diajukan operator
        b. Monitoring file
           Web Menampilkan halaman monitoring file yang diupload operator, lalu tim keuangan juga dapat mengisi form dan mengupload file dari tim keuangan sesuai dengan bagian kegiatan tim keuangan, lalu mengirim bukti transfer kepada operator.
        c. Isi keterangan
           Mengisi catatan terkait pengajuan jika pengajuan di reject.
        d. Selesai, kembali ke tabel monitoring pengajuan lalu dapat meng-approve atau reject pengajuan.
     4. Fitur Approve:
        Jika sudah sesuai dengan kriteria, tim keuangan meng-*approve* file, jika belum maka akan di-reject dan dikirim catatan ke pengupload.
     5. Selesai
   - Fitur filter:
     Memilih filter tanggal dan akun belanja dari semua pengajuan.
   - Download Permintaan
     Filter berdasar tanggal
     Mendownload tabel pengajuan untuk dokumentasi tabel di monitoring file.
     Berupa file excel dari form pengajuan yang telah diajukan.

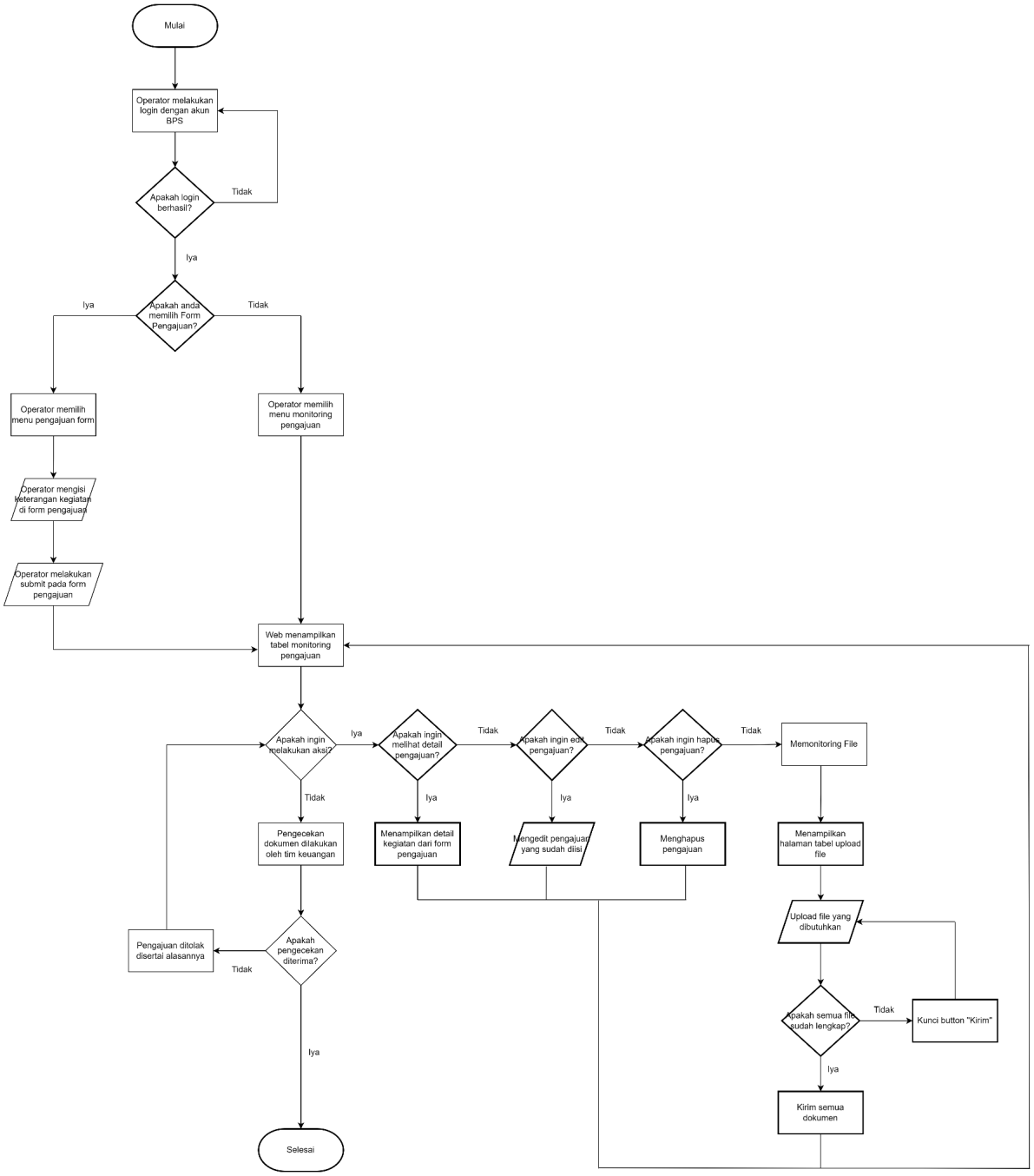
3.3 Implementasi Proses Bisnis



Berikut proses bisnis jika sesuai usernya, dimana kondisi hierarki user sebagai berikut:

- Tim keuangan (bisa menjadi operator)
- Operator (Pegawai BPS Provinsi DKI Jakarta) (yang mengajukan form permintaan/pengajuan).

3.3.1 Operator (Pegawai BPS Provinsi DKI Jakarta)



1. Mulai:

   Proses dimulai dengan operator melakukan login menggunakan akun BPS yang dimiliki.

2. Proses Login:

   Jika login berhasil, proses dilanjutkan ke langkah berikutnya. Jika tidak, operator harus memperbaiki kredensial atau mencoba kembali.

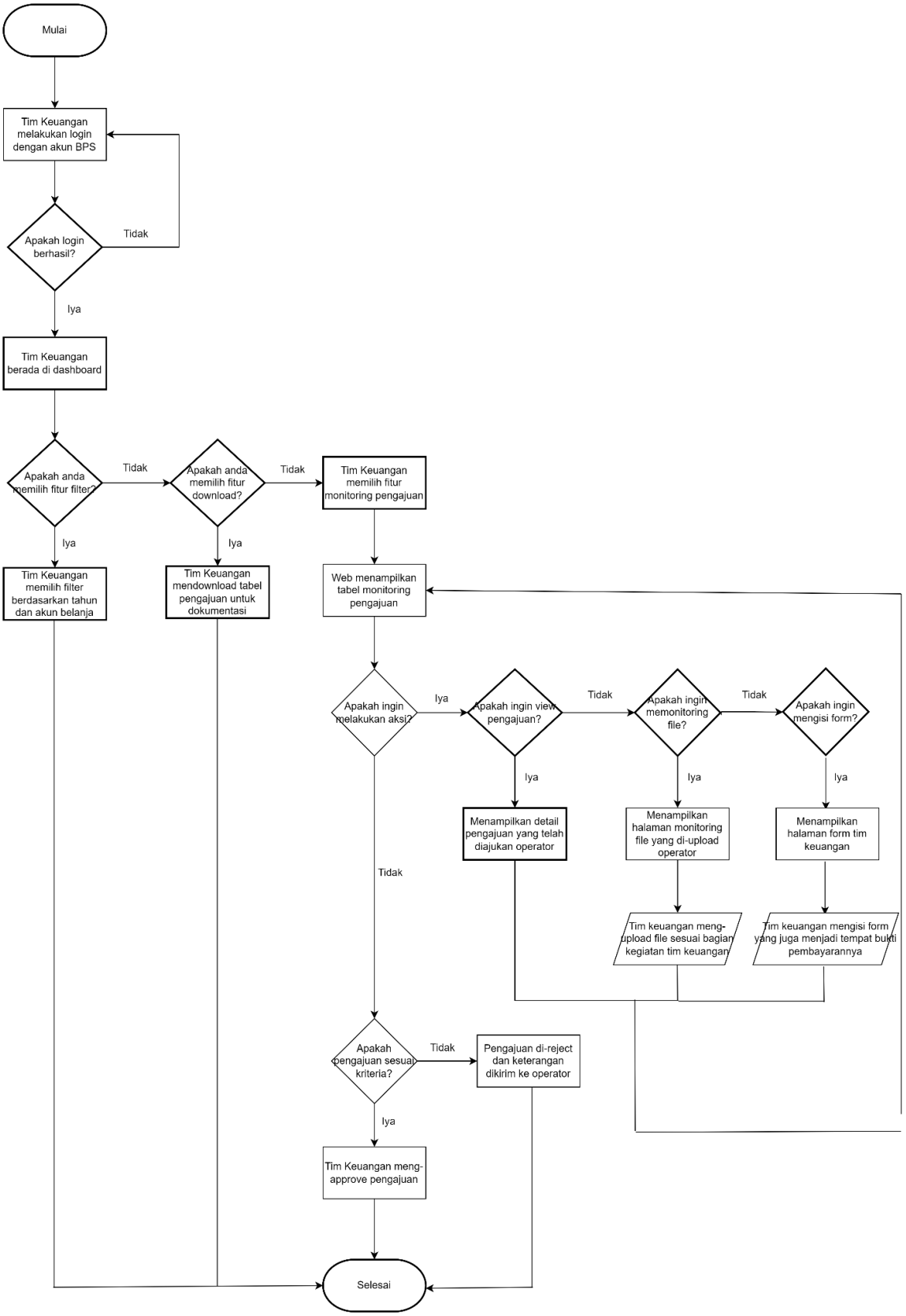
3. Memilih Form Pengajuan atau Monitoring:

1) Setelah login berhasil, operator diberi opsi untuk memilih "Form Pengajuan" atau "Monitoring Pengajuan".
2) Jika operator memilih "Form Pengajuan":
   - ➔ Operator (Pegawai BPS Provinsi DKI Jakarta) akan memilih tipe form dan mengisi data yang dibutuhkan, seperti rincian kegiatan.
   - ➔ Setelah mengisi, operator menekan tombol submit untuk mengirim pengajuan.
3) Jika operator memilih "Monitoring Pengajuan":
   - ➔ Web akan menampilkan tabel berisi daftar pengajuan yang telah dilakukan untuk dipantau.

4. Monitoring Pengajuan:

   Operator (Pegawai BPS Provinsi DKI Jakarta) memiliki beberapa opsi:
   - ➔ Melihat detail pengajuan: Menampilkan detail dari pengajuan yang dipilih.
   - ➔ Mengedit pengajuan: Operator (Pegawai BPS Provinsi DKI Jakarta) dapat mengedit rincian dari pengajuan yang sudah diajukan jika diperlukan.
   - ➔ Menghapus pengajuan: Jika pengajuan perlu dibatalkan, operator dapat menghapusnya.
5. Memantau File:
   - ➔ Operator (Pegawai BPS Provinsi DKI Jakarta) juga dapat memilih untuk mengunggah dokumen yang diperlukan pada halaman file upload.
   - ➔ Setelah dokumen diunggah, sistem akan mengecek apakah semua dokumen yang diperlukan sudah lengkap.
     - Jika sudah lengkap, operator dapat menekan tombol "Kirim" untuk mengirimkan seluruh dokumen yang dibutuhkan.
     - Jika belum lengkap, tombol "Kirim" akan dikunci sampai semua dokumen terpenuhi.
6. Pengecekan oleh Tim Keuangan:
   - ➔ Setelah pengajuan dikirim, dokumen akan dicek oleh tim keuangan. Jika terdapat masalah atau berkas tidak lengkap, pengajuan akan ditolak disertai dengan alasan penolakan.
   - ➔ Jika pengecekan berkas dinyatakan valid, proses akan selesai.

3.3.2 Tim Keuangan



1. Mulai

   Proses dimulai ketika tim keuangan mencoba masuk ke dalam aplikasi.
2. Login ke Sistem

   Tim keuangan melakukan login dengan akun mereka yang terdaftar di BPS.
3. Apakah login berhasil?
   - ➔ Jika iya, tim keuangan diarahkan ke dashboard.
   - ➔ Jika tidak, mereka mungkin diarahkan kembali ke halaman login.
4. Dashboard

   Tim keuangan berada di halaman dashboard, di mana mereka bisa memilih beberapa fitur.
5. Apakah Anda memilih fitur filter?
   - ➔ Jika iya, tim keuangan dapat memilih filter berdasarkan tahun dan akun belanja untuk melihat pengajuan yang lebih spesifik.
   - ➔ Jika tidak, sistem akan bertanya lagi.
6. Apakah Anda memilih fitur download?
   - ➔ Jika iya, tim keuangan mendownload tabel pengajuan untuk keperluan dokumentasi.
   - ➔ Jika tidak, tim keuangan akan memilih fitur monitoring pengajuan.
7. Monitoring Pengajuan

   Web menampilkan tabel pengajuan yang telah diajukan oleh operator untuk ditinjau oleh tim keuangan.
8. Apakah ingin melakukan aksi?
   - ➔ Jika iya, tim keuangan bisa melakukan berbagai aksi seperti meninjau, mengecek, atau memproses pengajuan.
   - ➔ Jika tidak, sistem akan kembali menanyakan pilihan lainnya.
9. Apakah ingin view pengajuan?
   - ➔ Jika iya, tim keuangan akan melihat detail pengajuan yang telah diajukan operator.
   - ➔ Jika tidak, tim keuangan bisa memilih untuk memonitor file yang di-upload operator.
10. Monitoring File

    Web menampilkan halaman monitoring file yang telah di-upload oleh operator.
11. Tim keuangan meng-upload file

    Tim keuangan meng-upload file yang sesuai dengan bagian kegiatan mereka.

12. Pengajuan Diperiksa

    Pengajuan diperiksa apakah sudah sesuai dengan kriteria yang ditetapkan oleh tim keuangan.

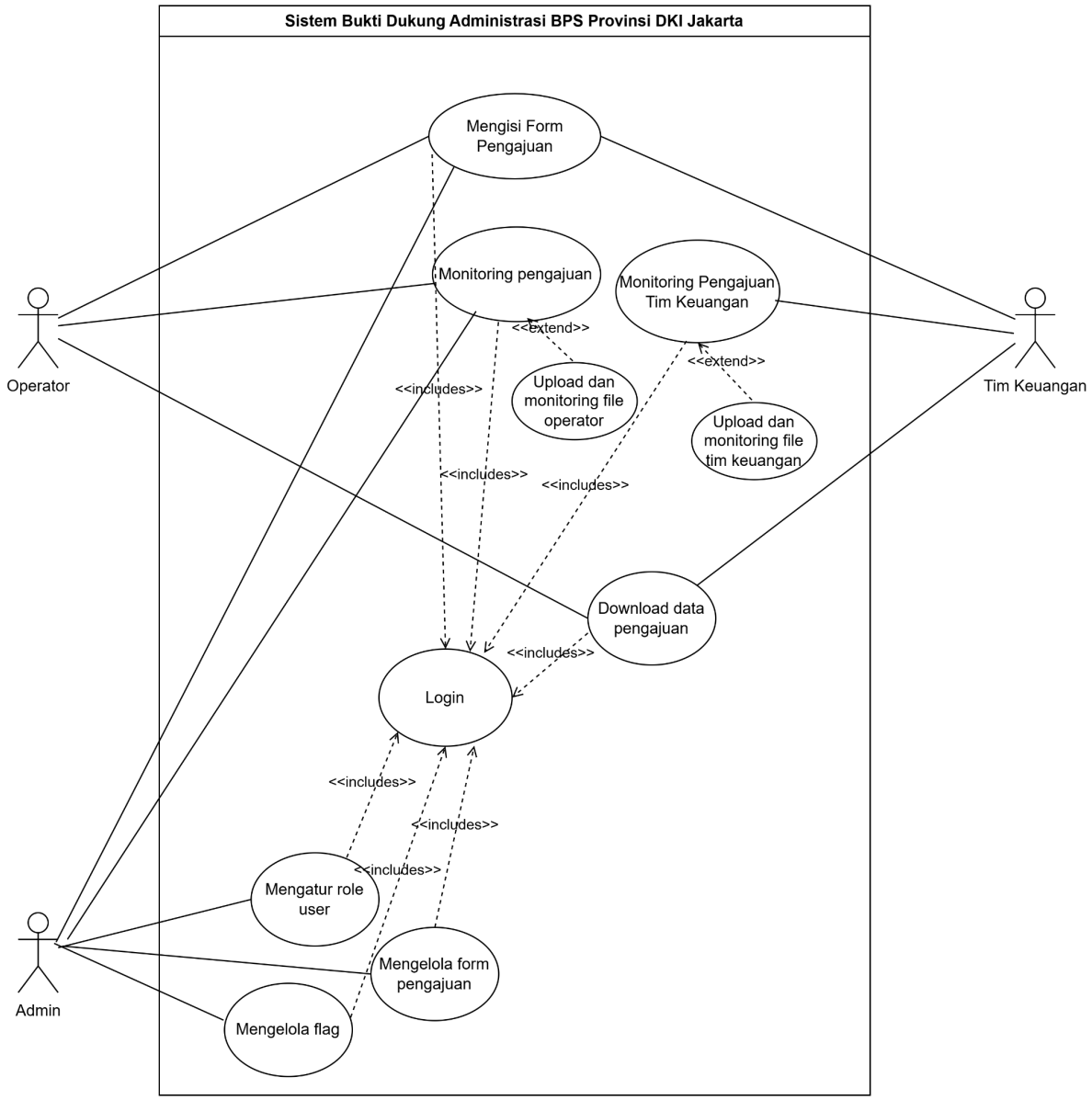
    - ➔ Jika tidak sesuai, pengajuan akan direject dan catatan penolakan dikirim ke pengunggah.
    - ➔ Jika sesuai, tim keuangan meng-approve pengajuan tersebut.

13. Selesai

    Proses berakhir ketika pengajuan disetujui atau ditolak.

3.4 Use Case dan Use Case Description

3.4.1 Use Case



3.4.2 Use Case Description:

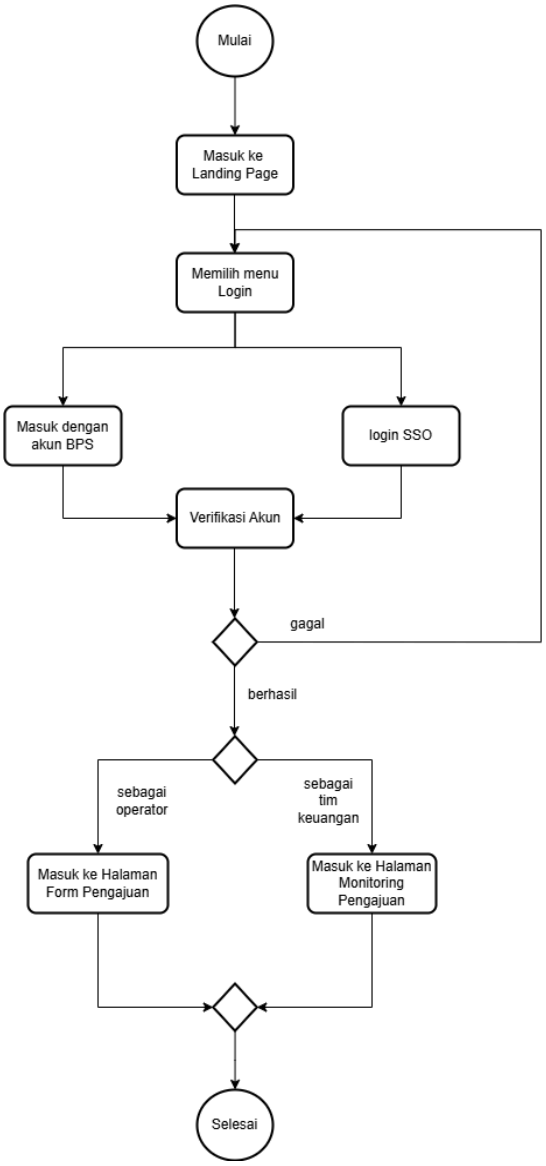
1. Login

Use case name: Login	Importance level: Kritis
Primary Actor	Operator (Pegawai BPS Provinsi DKI Jakarta) - yaitu Pegawai BPS Provinsi DKI Jakarta yang ingin

	mengumpulkan bukti dukung administrasi keuangan atas kegiatan yang dilakukan Tim Keuangan
Stakeholders and Interest	Operator (Pegawai BPS Provinsi DKI Jakarta) - melakukan login sesuai dengan perannya untuk mengajukan bukti dukung administrasi keuangan Tim Keuangan - melakukan monitoring dan pengecekan pada bukti dukung administrasi yang diunggah operator
Brief Description	Use case ini menggambarkan bagaimana proses autentikasi untuk memastikan data aman terhadap akses akses yang tidak diijinkan.
Relationship	-
Normal Flow of Events	1. Operator (Pegawai BPS Provinsi DKI Jakarta) atau Tim Keuangan masuk ke Landing Page 2. Operator (Pegawai BPS Provinsi DKI Jakarta) atau Tim Keuangan memilih menu Login 3. Operator (Pegawai BPS Provinsi DKI Jakarta) atau Tim Keuangan memasukkan akun BPS 4. Sistem memverifikasi akun BPS 5. Jika verifikasi berhasil, operator atau Tim Keuangan akan diarahkan ke halaman form pengajuan 6. Operator (Pegawai BPS Provinsi DKI Jakarta) atau Tim Keuangan mengisi form pengajuan 7. Operator (Pegawai BPS Provinsi DKI Jakarta) atau Tim Keuangan mengirimkan form pengajuan 8. Operator (Pegawai BPS Provinsi DKI Jakarta) atau Tim Keuangan akan diarahkan ke halaman Monitoring Pengajuan
Alternative Flow	1. Operator (Pegawai BPS Provinsi DKI Jakarta) atau Tim Keuangan memasukkan akun BPS yang salah

	2. Sistem akan menampilkan pesan error 3. Operator (Pegawai BPS Provinsi DKI Jakarta) atau Tim Keuangan dapat mencoba lagi memasukkan akun BPS
Exceptional Flows	1. System error saat memverifikasi akun BPS 2. System menampilkan pesan error 3. Operator (Pegawai BPS Provinsi DKI Jakarta) atau Tim Keuangan menghubungi admin

Activity Diagram:

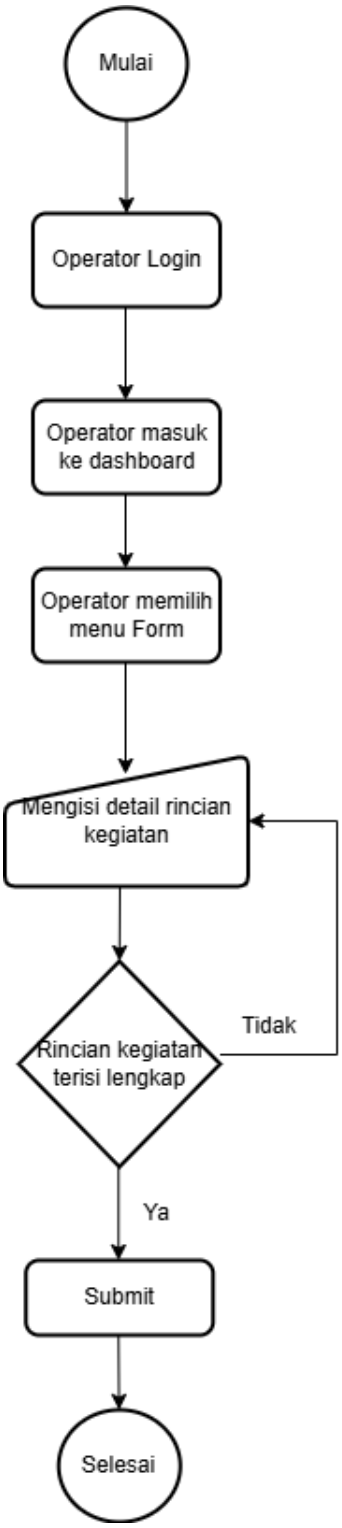


2. Mengisi Form Pengajuan

Use case name: Mengisi Form Pengajuan	Importance level: Kritis
Primary Actor	Operator (Pegawai BPS Provinsi DKI Jakarta) - yaitu Pegawai BPS Provinsi DKI Jakarta yang ingin mengumpulkan bukti dukung administrasi keuangan atas kegiatan yang dilakukan
Stakeholders and Interest	Operator (Pegawai BPS Provinsi DKI Jakarta) - mengisi rincian kegiatan yang dilakukan untuk mengumpulkan bukti dukung administrasi keuangan
Brief Description	Use case ini menggambarkan bagaimana operator mengisi detail kegiatan yang dilakukan agar detail tersebut masuk ke monitoring untuk selanjutnya mengumpulkan dokumen bukti dukung administrasi keuangan yang diperlukan
Relationship	● Include : Login
Normal Flow of Events	1. Mulai: Operator (Pegawai BPS Provinsi DKI Jakarta) memulai sistem dengan login ke akun. 2. Operator (Pegawai BPS Provinsi DKI Jakarta) memilih menu "Form" di dashboard. 3. Operator (Pegawai BPS Provinsi DKI Jakarta) mengisi detail rincian kegiatan yang ingin dicatat. 4. Operator (Pegawai BPS Provinsi DKI Jakarta) memastikan semua informasi rincian kegiatan telah lengkap dan terisi dengan benar. 5. Operator (Pegawai BPS Provinsi DKI Jakarta) mengkonfirmasi semua data sudah benar dan submit data kegiatan. 6. Selesai: Sistem menampilkan pesan konfirmasi bahwa kegiatan berhasil tercatat.
Exceptional Flows	Rincian kegiatan tidak terisi lengkap: Sistem akan

	menampilkan pesan error dan Operator (Pegawai BPS Provinsi DKI Jakarta) perlu kembali mengisi detail rincian kegiatan.

Activity Diagram:

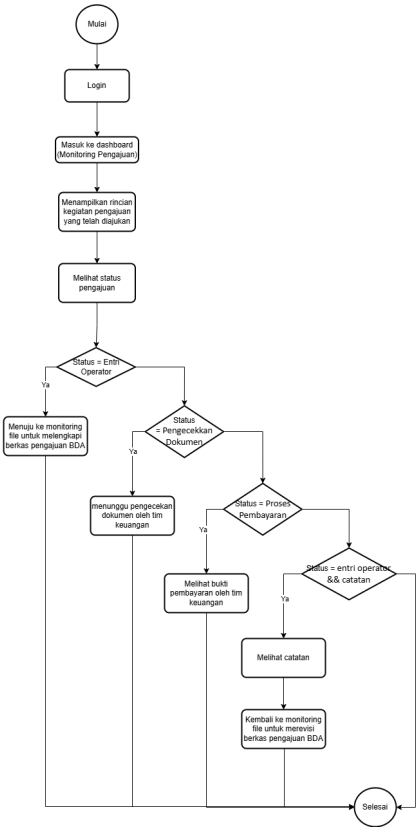


3. Monitoring Pengajuan
    a. Monitoring Pengajuan Operator (Pegawai BPS Provinsi DKI Jakarta)

Use case name: Monitoring Pengajuan Operator (Pegawai BPS Provinsi DKI Jakarta)	Importance level: Kritis
Primary Actor	Operator (Pegawai BPS Provinsi DKI Jakarta) - yaitu Pegawai BPS Provinsi DKI Jakarta yang ingin melihat apakah kegiatan yang diajukan diterima/ditolak oleh tim keuangan.
Stakeholders and Interest	Operator (Pegawai BPS Provinsi DKI Jakarta) - melihat status kegiatan yang diajukan untuk menindaklanjuti proses permintaan.
Brief Description	Use case ini menggambarkan bagaimana operator mengetahui status dari kegiatan yang diajukan dan selanjutnya menindaklanjuti proses (view, edit, hapus, monitoring file) apa yang akan dilakukan dari status (Entri operator, Pengecekan Dokumen, Rejected Tim Keuangan, Proses Pembayaran, dan Selesai).
Relationship	● Include : Login
Normal Flow of Events	1. Mulai: Operator (Pegawai BPS Provinsi DKI Jakarta) memulai sistem dengan login ke akun. 2. Operator (Pegawai BPS Provinsi DKI Jakarta) memilih menu "Monitoring" di dashboard. 3. Operator (Pegawai BPS Provinsi DKI Jakarta) melihat status pada tabel monitoring untuk melakukan proses apa yang akan dilakukan. 4. Setelah menunggu tindakan dari tim keuangan, jika status pengajuan langsung di-approve maka kegiatan selesai. 5. Jika status pengajuan belum selesai/ditolak oleh tim

	keuangan, lihat isi keterangan yang disampaikan tim keuangan. 6. Operator (Pegawai BPS Provinsi DKI Jakarta) merevisi ulang pengajuan sesuai isi keterangan dari tim keuangan (apakah delete, edit pengajuan, atau revisi upload berkas di monitoring file). 7. Operator (Pegawai BPS Provinsi DKI Jakarta) menunggu tindakan dari tim keuangan sampai status kegiatan selesai (tim keuangan menerima permintaan). 8. Selesai: Sistem menampilkan status kegiatan pengajuan selesai.
Exceptional Flows	Operator (Pegawai BPS Provinsi DKI Jakarta) akan terus menindaklanjuti proses sampai kegiatan pengajuannya diterima oleh tim keuangan dan berstatus selesai.

Activity Diagram:

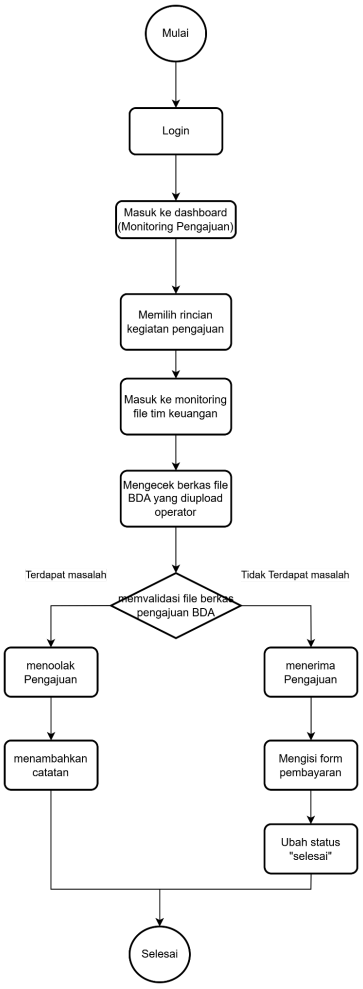


b. Monitoring Pengajuan Tim Keuangan

Use case name: Monitoring Form Pengajuan	Importance level: Kritis
Primary Actor	Tim Keuangan - yaitu Pegawai tim keuangan BPS Provinsi DKI Jakarta yang ingin memonitoring bukti dukung administrasi keuangan atas kegiatan yang diajukan oleh operator
Stakeholders and Interest	Tim keuangan - memeriksa rincian kegiatan pengajuan yang telah diajukan operator
Brief Description	Use case ini menggambarkan bagaimana tim keuangan memeriksa detail kegiatan yang diajukan operator, jika sesuai, tim keuangan mengisi form dari keuangan, lalu mengupload bukti transfer serta meng-*approve* pengajuan yang telah sesuai dengan tim keuangan. Jika belum sesuai, tim keuangan akan me-*reject* pengajuan lalu akan mengirim catatan kepada operator apa yang harus direvisi/salah lalu memunculkan notif di laman operator. Dengan demikian operator dapat menerima catatan dan dapat memperbaiki kesalahannya.
Relationship	● Include : Login
Normal Flow of Events	1. Mulai: Tim Keuangan memulai sistem dengan login ke akun. 2. Tim Keuangan memilih menu "Monitoring Pengajuan" di dashboard. 3. Tim Keuangan memeriksa detail rincian kegiatan yang diajukan. 4. Tim Keuangan meng-*reject* pengajuan jika tidak sesuai dengan kriteria Tim Keuangan, lalu mengirim catatan terkait pengajuan yang tidak sesuai, lalu dikirim notif ke operator. 5. Tim Keuangan mengkonfirmasi pengajuan yang

	sesuai dengan cara mengisi form khusus Tim Keuangan disertakan bukti transfer, lalu meng-*approve* pengajuan. 6. Selesai: Sistem menampilkan pesan konfirmasi bahwa kegiatan berhasil di-*approve*.
Exceptional Flows	1. Form Tim Keuangan tidak terisi lengkap: Sistem akan menampilkan pesan error dan Tim Keuangan perlu kembali mengisi form. 2. Bukti transfer belum diupload: Sistem akan menampilkan pesan error dan Tim Keuangan diingatkan kembali untuk mengupload bukti transfer di monitoring file.

Activity Diagram:

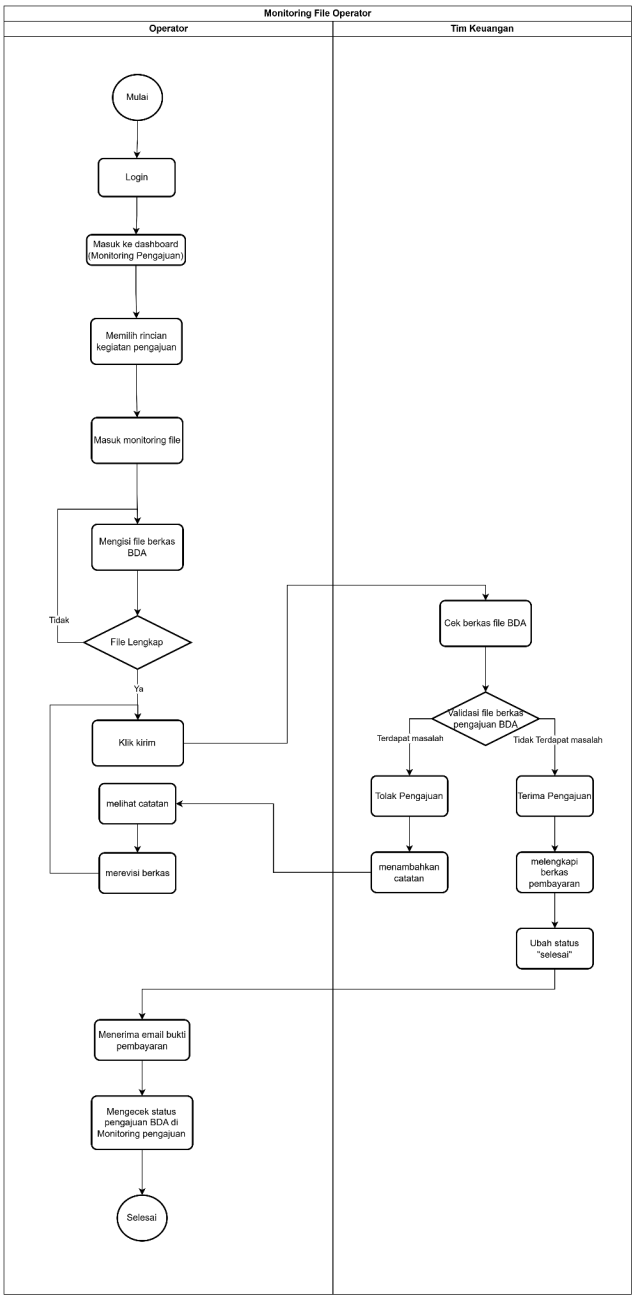


4. Monitoring File
    a. Monitoring File Operator (Pegawai BPS Provinsi DKI Jakarta)

Use case name: Monitoring File Operator (Pegawai BPS Provinsi DKI Jakarta)	Importance level: Kritis
Primary Actor	Operator (Pegawai BPS Provinsi DKI Jakarta) - yaitu Pegawai BPS Provinsi DKI Jakarta melakukan upload berkas pada monitoring file.
Stakeholders and Interest	Operator (Pegawai BPS Provinsi DKI Jakarta) - melengkapi berkas-berkas yang diminta pada akun belanja yang diajukan ke tim keuangan.
Brief Description	Use case ini menggambarkan bagaimana operator melengkapi berkas yang wajib diupload untuk ditindaklanjuti oleh tim keuangan.
Relationship	● Include : Login → Monitoring Pengajuan
Normal Flow of Events	1. Mulai: Operator (Pegawai BPS Provinsi DKI Jakarta) memulai sistem dengan login ke akun. 2. Operator (Pegawai BPS Provinsi DKI Jakarta) memilih menu "Monitoring" di dashboard. 3. Operator (Pegawai BPS Provinsi DKI Jakarta) memilih menu “Monitoring File” di aksi “monitoring pengajuan”. 4. Operator (Pegawai BPS Provinsi DKI Jakarta) melakukan upload berkas di setiap kegiatan pada akun belanja yang diajukan. 5. Operator (Pegawai BPS Provinsi DKI Jakarta) menekan ikon/tombol kirim. 6. Selesai: Sistem menampilkan status “Pengecekan Dokumen” pada dashboard monitoring.
Exceptional Flows	1. Upload berkas tidak lengkap: Sistem akan

	menampilkan pesan error bahwa harus melengkapi upload berkas agar bisa klik kirim. 2. Operator (Pegawai BPS Provinsi DKI Jakarta) mengupload ulang berkas jika tim keuangan menolak karena alasan berkas yang tidak sesuai/kurang lengkap.

Activity Diagram:

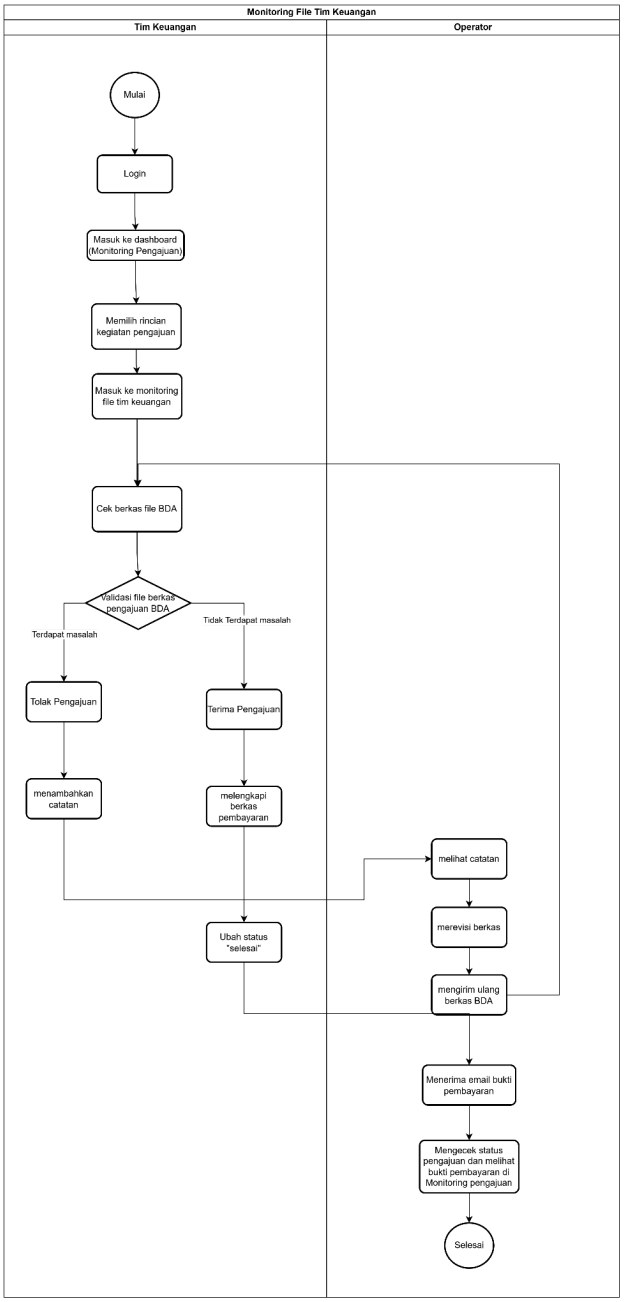


b. Monitoring File Tim Keuangan

Use case name: Monitoring File Tim Keuangan	Importance level: Kritis
Primary Actor	Tim Keuangan - yaitu Pegawai tim keuangan BPS Provinsi DKI Jakarta yang ingin memonitoring upload berkas kegiatan yang diajukan oleh operator dan juga mengupload berkas (seperti bukti transfer dll) pada kegiatan tersebut.
Stakeholders and Interest	Tim keuangan - memeriksa berkas yang telah diupload operator juga mengupload berkas yang dibutuhkan dalam kegiatan yang diajukan.
Brief Description	Use case ini menggambarkan bagaimana tim keuangan memeriksa berkas yang telah diupload operator. Jika sudah sesuai dan lengkap, tim keuangan mengupload berkas yang diperlukan pada akun belanja yang diajukan operator dan status pada “Monitoring” akan berubah menjadi “Proses pembayaran” ataupun “Selesai”. Jika belum sesuai, tim keuangan akan me-*reject* pengajuan lalu akan mengirim catatan kepada operator berkas mana yang harus direvisi/salah lalu operator akan memperbaiki kesalahannya.
Relationship	● Include : Login → Monitoring Pengajuan
Normal Flow of Events	1. Mulai: Tim keuangan memulai sistem dengan login ke akun. 2. Tim keuangan memilih menu "Monitoring" di dashboard. 3. Tim keuangan memilih menu “Monitoring File” di aksi “monitoring pengajuan”. 4. Tim keuangan mengecek berkas yang telah diupload oleh operator dan melakukan upload berkas yang dibutuhkan kegiatan pada akun belanja yang diajukan. 5. Tim keuangan menekan ikon/tombol kirim.

	6. Selesai: Sistem menampilkan status “Proses pembayaran” ataupun “Selesai” pada dashboard monitoring.
Exceptional Flows	Upload berkas tidak lengkap: Sistem akan menampilkan pesan error bahwa harus melengkapi upload berkas agar bisa klik kirim.

Activity Diagram:

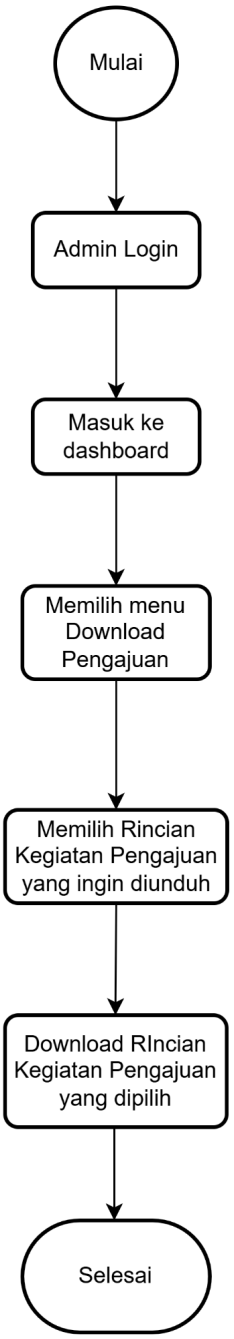


5. Download Data Pengajuan

Use case name: Download Data Pengajuan	Importance level: Kritis
Primary Actor	Tim Operator (Pegawai BPS Provinsi DKI Jakarta)- yaitu Pegawai BPS Provinsi DKI Jakarta yang ingin mendownload pengajuan kegiatan yang diajukan. Tim Keuangan- yaitu Pegawai tim keuangan BPS Provinsi DKI Jakarta yang ingin mendownload pengajuan dari operator. Tim Admin- yaitu Pegawai tim keuangan BPS Provinsi DKI Jakarta yang juga ingin mendownload pengajuan.
Stakeholders and Interest	Seluruh role user - mendownload data pengajuan kegiatan yang telah diisi di form pengajuan.
Brief Description	Use case ini menggambarkan bagaimana Tim keuangan, Tim Operator (Pegawai BPS Provinsi DKI Jakarta), dan admin mendownload data pengajuan kegiatan yang telah diisi di form pengajuan.
Relationship	● Include : Login → Download data pengajuan
Normal Flow of Events	1. Mulai: Tim keuangan, Tim Operator (Pegawai BPS Provinsi DKI Jakarta), dan admin memulai sistem dengan login ke akun. 2. Mulai: Tim keuangan, Tim Operator (Pegawai BPS Provinsi DKI Jakarta), dan admin memilih menu "Download Pengajuan" di dashboard. 3. Tim keuangan, Tim Operator (Pegawai BPS Provinsi DKI Jakarta), dan admin mengisi Rincian Kegiatan Pengajuan yang ingin diunduh. 4. Tim keuangan, Tim Operator (Pegawai BPS Provinsi DKI Jakarta), dan admin mendownload rincian kegiatan pengajuan yang dipilih.

Exceptional Flows	Tanggal tidak terisi dengan benar: Sistem akan menampilkan pesan error bahwa harus mengisi tanggal mulai dan akhir.

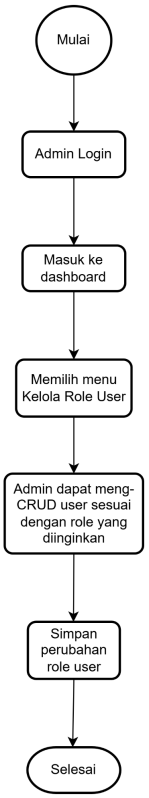
Activity Diagram:



6. Mengatur Role User

Use case name: Manage User	Importance level: Kritis
Primary Actor	Tim Admin - yaitu Pegawai PPK (Pejabat Pengelola Keuangan) BPS Provinsi DKI Jakarta yang memiliki wewenang mengubah role pengguna. Admin juga dapat menambahkan pengguna baru jika terdapat pegawai baru yang terlibat dalam sistem ini.
Stakeholders and Interest	Admin role - memiliki wewenang menambahkan dan mengubah role pengguna
Brief Description	Use case ini menggambarkan bagaimana admin dapat menambahkan pengguna baru dan juga mengubah role pengguna sesuai dengan peran masing-masing pegawai BPS DKI Jakarta yang terlibat dalam proses bisnis ini.

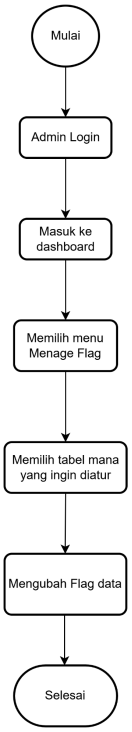
Activity Diagram:



7. Mengelola MAK

Use case name: Mengelola MAK		Importance level: Kritis
Primary Actor	Tim Admin - yaitu Pegawai PPK (Pejabat Pengelola Keuangan) BPS Provinsi DKI Jakarta dapat menambah, mengaktifkan, dan menonaktifkan data pada kolom akun, komponen, subkomponen, dan output.	
Stakeholders and Interest	Tim admin - menambah, mengaktifkan, dan menonaktifkan data pada kolom tabel akun, komponen, subkomponen, dan output.	
Brief Description	Use case ini menggambarkan bagaimana tim admin dapat menambah, mengaktifkan, dan menonaktifkan data pada kolom akun, komponen, subkomponen, dan output. Admin juga dapat menetapkan data-data di setiap kolomnya bisa di "tampilkan" atau "jangan ditampilkan". Tiap kolomnya juga dapat diubah/dikurangi/ditambahkan melalui fitur tambah yang tersedia.	

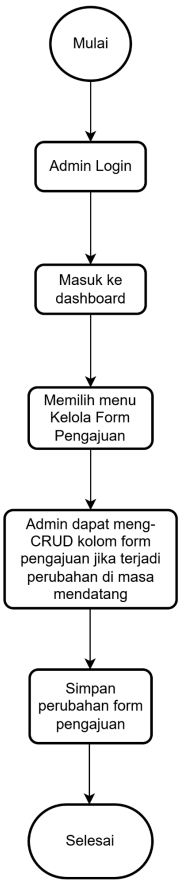
Activity Diagram:



8. Mengelola Form Pengajuan

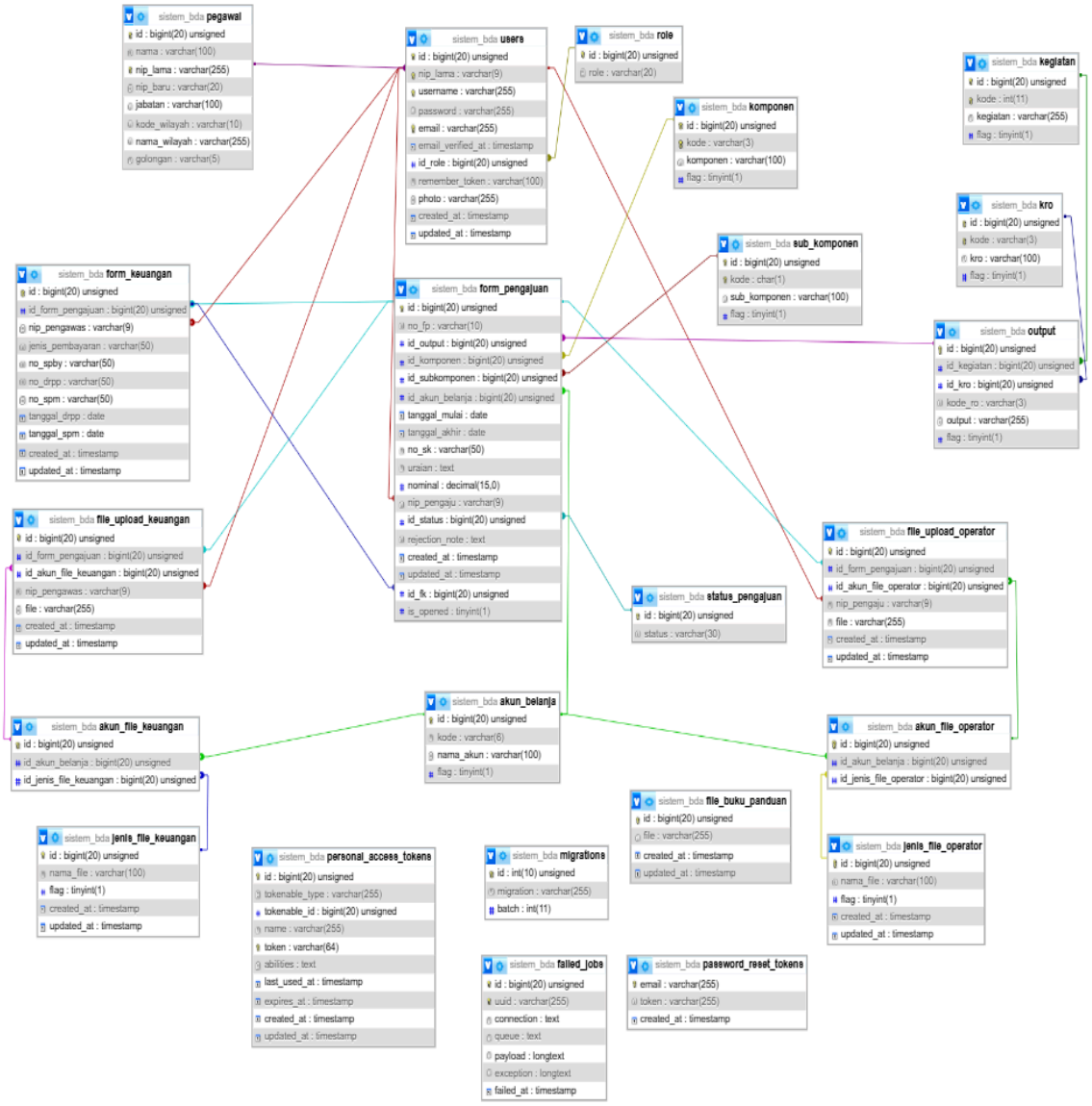
Use case name: Mengelola Form Pengajuan		Importance level: Kritis
Primary Actor	Tim Admin - yaitu Pegawai PPK (Pejabat Pengelola Keuangan) BPS Provinsi DKI Jakarta dapat menambah/menonaktifkan pertanyaan pada form pengajuan.	
Stakeholders and Interest	Admin role - memiliki wewenang menambah, mengaktifkan, dan menonaktifkan pertanyaan pada form pengajuan.	
Brief Description	Use case ini menggambarkan bagaimana tim admin dapat menambah, mengaktifkan, dan menonaktifkan pertanyaan pada form pengajuan. Admin juga dapat menetapkan pertanyaan-pertanyaan di setiap kolomnya bisa di "tampilkan" atau "jangan ditampilkan".	

Activity Diagram:



3.5 Rancangan Basis Data

Entity Relationship Diagram (ERD)



3.5.1 Database bda

	Table	Action						Rows	Type	Collation	Size	Overhead
☐	akun_belanja	Browse	Structure	Search	Insert	Empty	Drop	44	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
☐	akun_file_keuangan	Browse	Structure	Search	Insert	Empty	Drop	47	InnoDB	utf8mb4_unicode_ci	48.0 KiB	-
☐	akun_file_operator	Browse	Structure	Search	Insert	Empty	Drop	104	InnoDB	utf8mb4_unicode_ci	48.0 KiB	-
☐	failed_jobs	Browse	Structure	Search	Insert	Empty	Drop	0	InnoDB	utf8mb4_unicode_ci	32.0 KiB	-
☐	file_buku_panduan	Browse	Structure	Search	Insert	Empty	Drop	0	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
☐	file_upload_keuangan	Browse	Structure	Search	Insert	Empty	Drop	8	InnoDB	utf8mb4_unicode_ci	64.0 KiB	-
☐	file_upload_operator	Browse	Structure	Search	Insert	Empty	Drop	26	InnoDB	utf8mb4_unicode_ci	64.0 KiB	-
☐	form_keuangan	Browse	Structure	Search	Insert	Empty	Drop	13	InnoDB	utf8mb4_unicode_ci	48.0 KiB	-
☐	form_pengajuan	Browse	Structure	Search	Insert	Empty	Drop	13	InnoDB	utf8mb4_unicode_ci	128.0 KiB	-
☐	jenis_file_keuangan	Browse	Structure	Search	Insert	Empty	Drop	6	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
☐	jenis_file_operator	Browse	Structure	Search	Insert	Empty	Drop	86	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
☐	kegiatan	Browse	Structure	Search	Insert	Empty	Drop	16	InnoDB	utf8mb4_unicode_ci	32.0 KiB	-
☐	komponen	Browse	Structure	Search	Insert	Empty	Drop	7	InnoDB	utf8mb4_unicode_ci	32.0 KiB	-
☐	kro	Browse	Structure	Search	Insert	Empty	Drop	8	InnoDB	utf8mb4_unicode_ci	32.0 KiB	-
☐	migrations	Browse	Structure	Search	Insert	Empty	Drop	27	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
☐	output	Browse	Structure	Search	Insert	Empty	Drop	31	InnoDB	utf8mb4_unicode_ci	48.0 KiB	-
☐	password_reset_tokens	Browse	Structure	Search	Insert	Empty	Drop	0	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
☐	pegawai	Browse	Structure	Search	Insert	Empty	Drop	215	InnoDB	utf8mb4_unicode_ci	80.0 KiB	-
☐	personal_access_tokens	Browse	Structure	Search	Insert	Empty	Drop	0	InnoDB	utf8mb4_unicode_ci	48.0 KiB	-
☐	role	Browse	Structure	Search	Insert	Empty	Drop	3	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
☐	status_pengajuan	Browse	Structure	Search	Insert	Empty	Drop	5	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
☐	sub_komponen	Browse	Structure	Search	Insert	Empty	Drop	2	InnoDB	utf8mb4_unicode_ci	32.0 KiB	-
☐	users	Browse	Structure	Search	Insert	Empty	Drop	83	InnoDB	utf8mb4_unicode_ci	80.0 KiB	-
	23 tables	Sum						744	InnoDB	latin1_swedish_ci	944.0 KiB	0 B

3.5.2 Tabel akun belanja

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action		
☐	1	id	bigint(20)		UNSIGNED	No	None		AUTO_INCREMENT	Change	Drop	More
☐	2	kode	varchar(6)	utf8mb4_unicode_ci		No	None			Change	Drop	More
☐	3	nama_akun	varchar(100)	utf8mb4_unicode_ci		No	None			Change	Drop	More
☐	4	flag	tinyint(1)			No	1			Change	Drop	More

3.5.3 Tabel akun file keuangan

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action		
☐	1	id	bigint(20)		UNSIGNED	No	None		AUTO_INCREMENT	Change	Drop	More
☐	2	id_akun_belanja	bigint(20)		UNSIGNED	No	None			Change	Drop	More
☐	3	id_jenis_file_keuangan	bigint(20)		UNSIGNED	No	None			Change	Drop	More

3.5.4 Tabel akun file operator

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action		
☐	1	id	bigint(20)		UNSIGNED	No	None		AUTO_INCREMENT	Change	Drop	More
☐	2	id_akun_belanja	bigint(20)		UNSIGNED	No	None			Change	Drop	More
☐	3	id_jenis_file_operator	bigint(20)		UNSIGNED	No	None			Change	Drop	More

3.5.5 Tabel form pengajuan

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	**id**	bigint(20)		UNSIGNED	No	*None*		AUTO_INCREMENT	Change Drop More
2	**no_fp**	varchar(10)	utf8mb4_unicode_ci		No	*None*			Change Drop More
3	**id_output**	bigint(20)		UNSIGNED	No	*None*			Change Drop More
4	**id_komponen**	bigint(20)		UNSIGNED	No	*None*			Change Drop More
5	**id_subkomponen**	bigint(20)		UNSIGNED	No	*None*			Change Drop More
6	**id_akun_belanja**	bigint(20)		UNSIGNED	No	*None*			Change Drop More
7	**tanggal_mulai**	date			No	*None*			Change Drop More
8	**tanggal_akhir**	date			No	*None*			Change Drop More
9	**no_sk**	varchar(50)	utf8mb4_unicode_ci		No	*None*			Change Drop More
10	**uraian**	text	utf8mb4_unicode_ci		No	*None*			Change Drop More
11	**nominal**	decimal(15,0)			No	*None*			Change Drop More
12	**nip_pengaju**	varchar(9)	utf8mb4_unicode_ci		No	*None*			Change Drop More
13	**id_status**	bigint(20)		UNSIGNED	No	1			Change Drop More
14	**rejection_note**	text	utf8mb4_unicode_ci		Yes	*NULL*			Change Drop More
15	**created_at**	timestamp			Yes	*NULL*			Change Drop More
16	**updated_at**	timestamp			Yes	*NULL*			Change Drop More
17	**id_fk**	bigint(20)		UNSIGNED	Yes	*NULL*			Change Drop More
18	**is_opened**	tinyint(1)			No	0			Change Drop More

3.5.6 Tabel kegiatan

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐	1	**id**	int(11)			No	*None*		AUTO_INCREMENT	Change Drop Mo
☐	2	**kode**	int(4)			No	*None*			Change Drop Mo
☐	3	**kegiatan**	varchar(100)	latin1_swedish_ci		No	*None*			Change Drop Mo
☐	4	**flag**	tinyint(1)			No	*None*			Change Drop Mo

3.5.7 Tabel komponen

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐	1	**id**	int(11)			No	*None*		AUTO_INCREMENT	Change Drop Mor
☐	2	**kode**	int(3)			No	*None*			Change Drop Mor
☐	3	**komponen**	text	latin1_swedish_ci		No	*None*			Change Drop Mor
☐	4	**flag**	tinyint(1)			No	*None*			Change Drop Mor

3.5.8 Tabel kro

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐	1	**id**	int(11)			No	*None*		AUTO_INCREMENT	Change Drop More
☐	2	**kode**	varchar(3)	latin1_swedish_ci		No	*None*			Change Drop More
☐	3	**kro**	varchar(100)	latin1_swedish_ci		No	*None*			Change Drop More
☐	4	**flag**	tinyint(1)			No	*None*			Change Drop More

3.5.9 Tabel output

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	**id**	int(11)			No	*None*		AUTO_INCREMENT	Change Drop
2	**kegiatan**	varchar(4)	latin1_swedish_ci		No	*None*			Change Drop
3	**kro**	varchar(3)	latin1_swedish_ci		No	*None*			Change Drop
4	**ro**	varchar(3)	latin1_swedish_ci		No	*None*			Change Drop
5	**output**	text	latin1_swedish_ci		No	*None*			Change Drop
6	**flag**	tinyint(1)			No	*None*			Change Drop

3.5.10 Tabel pegawai

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	**id**	int(11)			No	*None*		AUTO_INCREMENT	Change D
2	**nama**	varchar(100)	utf8mb4_general_ci		No	*None*			Change D
3	**niplama**	int(11)			No	*None*			Change D
4	**nipbaru**	varchar(20)	utf8mb4_general_ci		Yes	*NULL*			Change D
5	**jabatan**	varchar(100)	utf8mb4_general_ci		Yes	*NULL*			Change D
6	**wilayah**	varchar(255)	utf8mb4_general_ci		No	*None*			Change D
7	**namawilayah**	varchar(255)	utf8mb4_general_ci		No	*None*			Change D
8	**golongan**	varchar(255)	utf8mb4_general_ci		No	*None*			Change D

3.5.11 Tabel sub komponen

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	**id**	int(11)			No	*None*		AUTO_INCREMENT	Change Dr
2	**kode**	varchar(1)	latin1_swedish_ci		No	*None*			Change Dr
3	**subkomponen**	varchar(100)	latin1_swedish_ci		No	*None*			Change Dr
4	**flag**	tinyint(1)			No	*None*			Change Dr

3.5.12 Tabel users

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	**id**	bigint(20)		UNSIGNED	No	*None*		AUTO_INCREMENT	Change Drop More
2	**nip_lama**	varchar(9)	utf8mb4_unicode_ci		No	*None*			Change Drop More
3	**username**	varchar(255)	utf8mb4_unicode_ci		No	*None*			Change Drop More
4	**password**	varchar(255)	utf8mb4_unicode_ci		No	*None*			Change Drop More
5	**email**	varchar(255)	utf8mb4_unicode_ci		No	*None*			Change Drop More
6	**email_verified_at**	timestamp			Yes	*NULL*			Change Drop More
7	**id_role**	bigint(20)		UNSIGNED	No	*None*			Change Drop More
8	**remember_token**	varchar(100)	utf8mb4_unicode_ci		Yes	*NULL*			Change Drop More
9	**photo**	varchar(255)	utf8mb4_unicode_ci		Yes	*NULL*			Change Drop More
10	**created_at**	timestamp			Yes	*NULL*			Change Drop More
11	**updated_at**	timestamp			Yes	*NULL*			Change Drop More

3.5.13 Tabel file buku panduan

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	bigint(20)		UNSIGNED	No	None		AUTO_INCREMENT	Change Drop More
2	file	varchar(255)	utf8mb4_unicode_ci		No	None			Change Drop More
3	created_at	timestamp			Yes	NULL			Change Drop More
4	updated_at	timestamp			Yes	NULL			Change Drop More

3.5.14 Tabel file upload keuangan

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	bigint(20)		UNSIGNED	No	None		AUTO_INCREMENT	Change Drop
2	id_form_pengajuan	bigint(20)		UNSIGNED	No	None			Change Drop
3	id_akun_file_keuangan	bigint(20)		UNSIGNED	No	None			Change Drop
4	nip_pengawas	varchar(9)	utf8mb4_unicode_ci		No	None			Change Drop
5	file	varchar(255)	utf8mb4_unicode_ci		No	None			Change Drop
6	created_at	timestamp			Yes	NULL			Change Drop
7	updated_at	timestamp			Yes	NULL			Change Drop

3.5.15 Tabel file upload operator

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	bigint(20)		UNSIGNED	No	None		AUTO_INCREMENT	Change Drop M
2	id_form_pengajuan	bigint(20)		UNSIGNED	No	None			Change Drop M
3	id_akun_file_operator	bigint(20)		UNSIGNED	No	None			Change Drop M
4	nip_pengaju	varchar(9)	utf8mb4_unicode_ci		No	None			Change Drop M
5	file	varchar(255)	utf8mb4_unicode_ci		No	None			Change Drop M
6	created_at	timestamp			Yes	NULL			Change Drop M
7	updated_at	timestamp			Yes	NULL			Change Drop M

3.5.16 Tabel form keuangan

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	bigint(20)		UNSIGNED	No	None		AUTO_INCREMENT	Change Drop More
2	id_form_pengajuan	bigint(20)		UNSIGNED	No	None			Change Drop More
3	nip_pengawas	varchar(9)	utf8mb4_unicode_ci		No	None			Change Drop More
4	jenis_pembayaran	varchar(50)	utf8mb4_unicode_ci		Yes	NULL			Change Drop More
5	no_spby	varchar(50)	utf8mb4_unicode_ci		No	None			Change Drop More
6	no_drpp	varchar(50)	utf8mb4_unicode_ci		No	None			Change Drop More
7	no_spm	varchar(50)	utf8mb4_unicode_ci		No	None			Change Drop More
8	tanggal_drpp	date			No	None			Change Drop More
9	tanggal_spm	date			No	None			Change Drop More
10	created_at	timestamp			Yes	NULL			Change Drop More
11	updated_at	timestamp			Yes	NULL			Change Drop More

3.5.17 Tabel jenis file keuangan

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	**id**	bigint(20)		UNSIGNED	No	*None*		AUTO_INCREMENT	Change Drop More
2	**nama_file**	varchar(100)	utf8mb4_unicode_ci		No	*None*			Change Drop More
3	**flag**	tinyint(1)			No	1			Change Drop More
4	**created_at**	timestamp			Yes	*NULL*			Change Drop More
5	**updated_at**	timestamp			Yes	*NULL*			Change Drop More

3.5.18 Tabel jenis file operator

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	**id**	bigint(20)		UNSIGNED	No	*None*		AUTO_INCREMENT	Change Drop More
2	**nama_file**	varchar(100)	utf8mb4_unicode_ci		No	*None*			Change Drop More
3	**flag**	tinyint(1)			No	1			Change Drop More
4	**created_at**	timestamp			Yes	*NULL*			Change Drop More
5	**updated_at**	timestamp			Yes	*NULL*			Change Drop More

3.5.19 Tabel role

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	**id**	bigint(20)		UNSIGNED	No	*None*		AUTO_INCREMENT	Change Drop More
2	**role**	varchar(20)	utf8mb4_unicode_ci		No	*None*			Change Drop More

3.5.20 Tabel status pengajuan

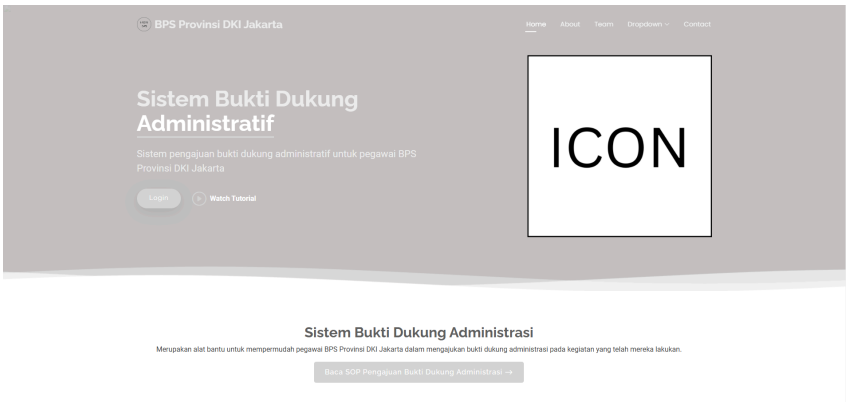
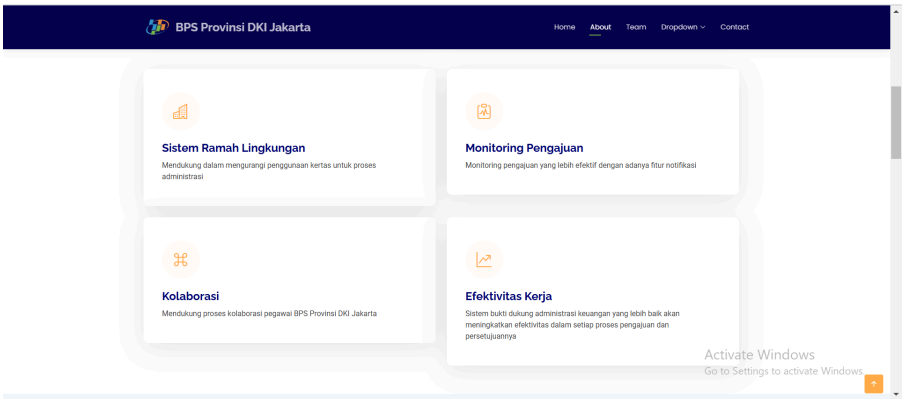
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	**id**	bigint(20)		UNSIGNED	No	*None*		AUTO_INCREMENT	Change Drop More
2	**status**	varchar(30)	utf8mb4_unicode_ci		No	*None*			Change Drop More

3.6 UI/UX Wireframe dan Mock-up:

Link figma wireframe:
https://www.figma.com/design/HxDEBvC4fSxzkqR4vsvQIM/Operator_Monitoring-Pengajuan?node-id=0-1&t=TxLfbXyyVxTjbw2R-1

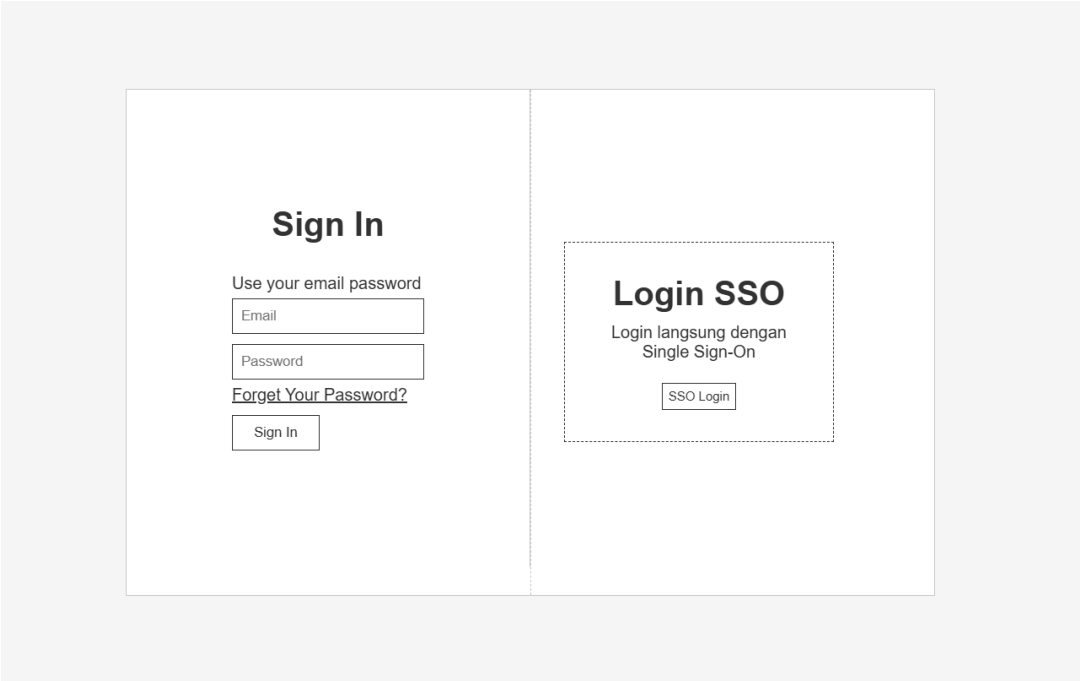
Link git-stis frontend html: *https://github.com/imaliarosyidaa/sistem-bda.git*

3.6.1 Landing page

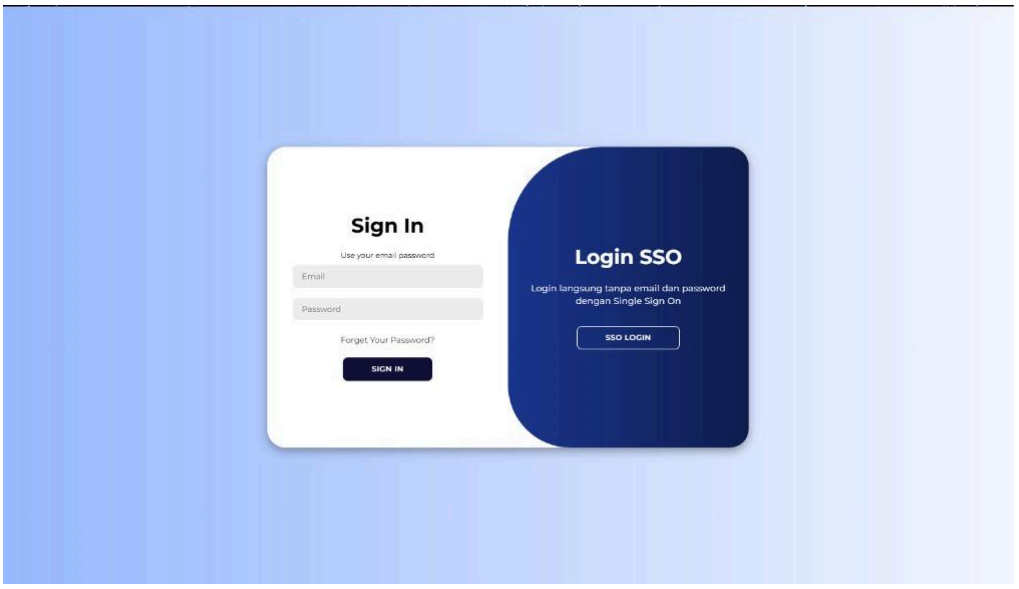
WIREFRAME

MOCKUP
  

3.6.2 Login

WIREFRAME



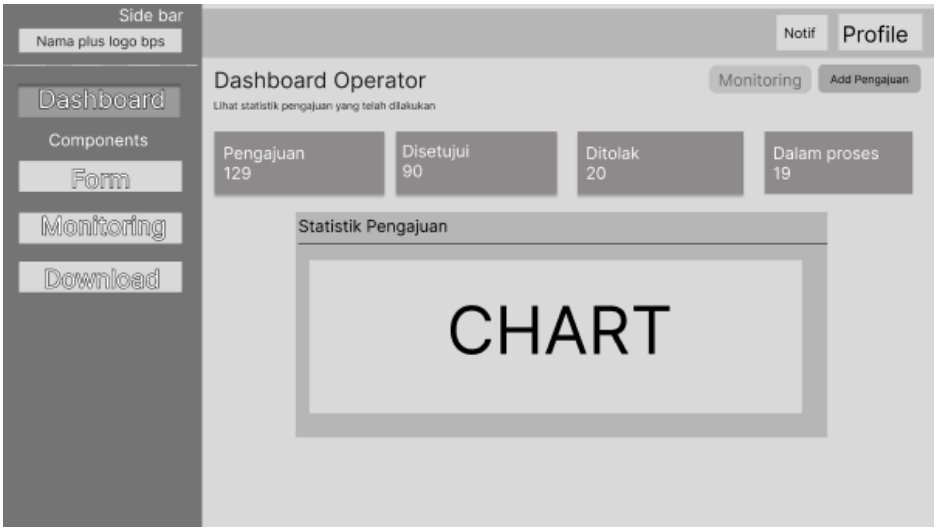
MOCKUP



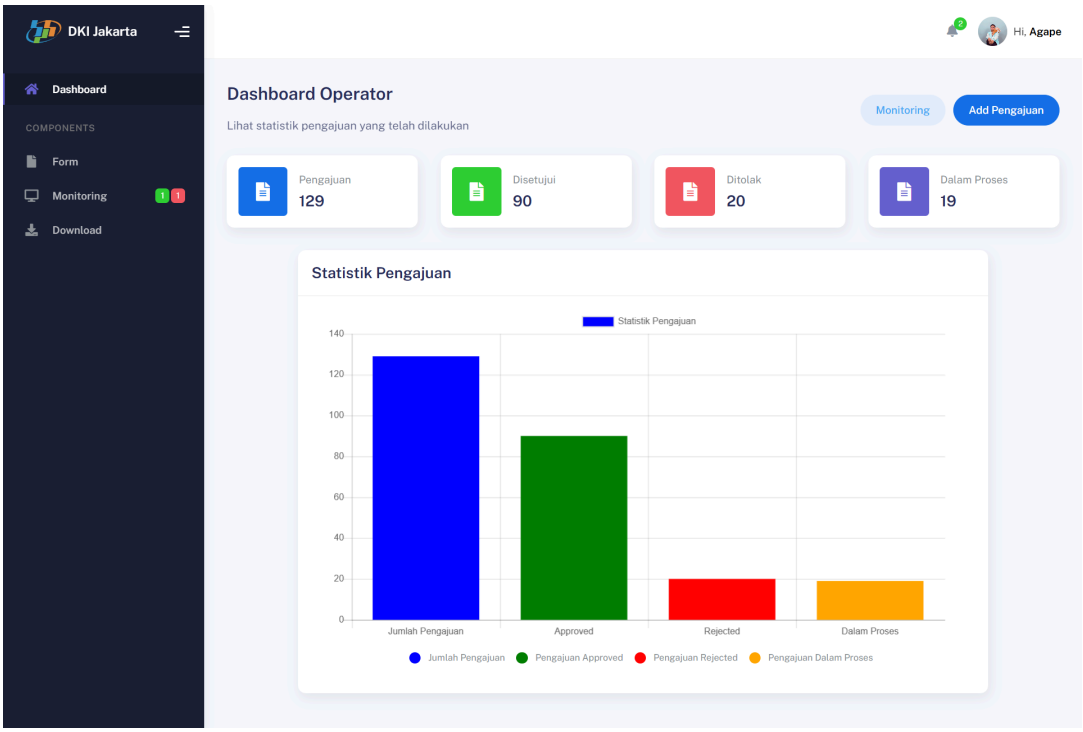
3.6.3 Operator

1) Dashboard Operator (Pegawai BPS Provinsi DKI Jakarta)

WIREFRAME

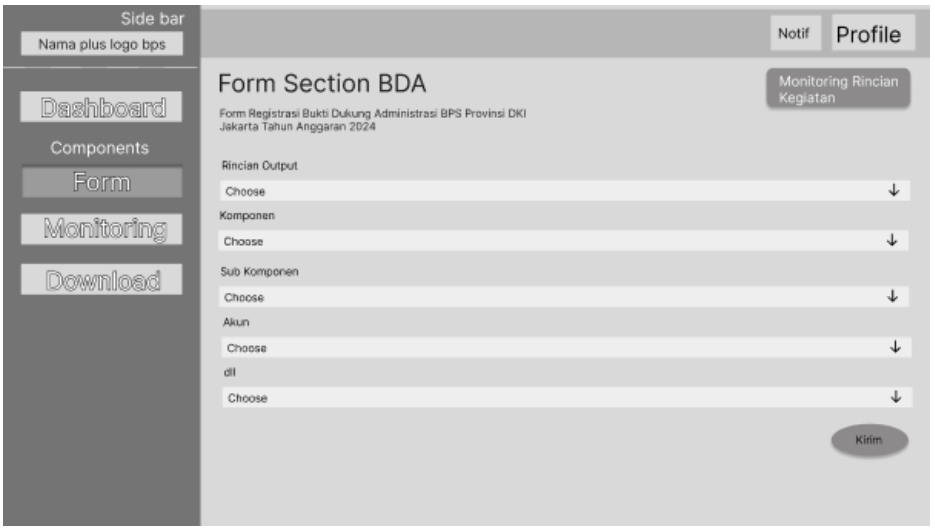


MOCKUP

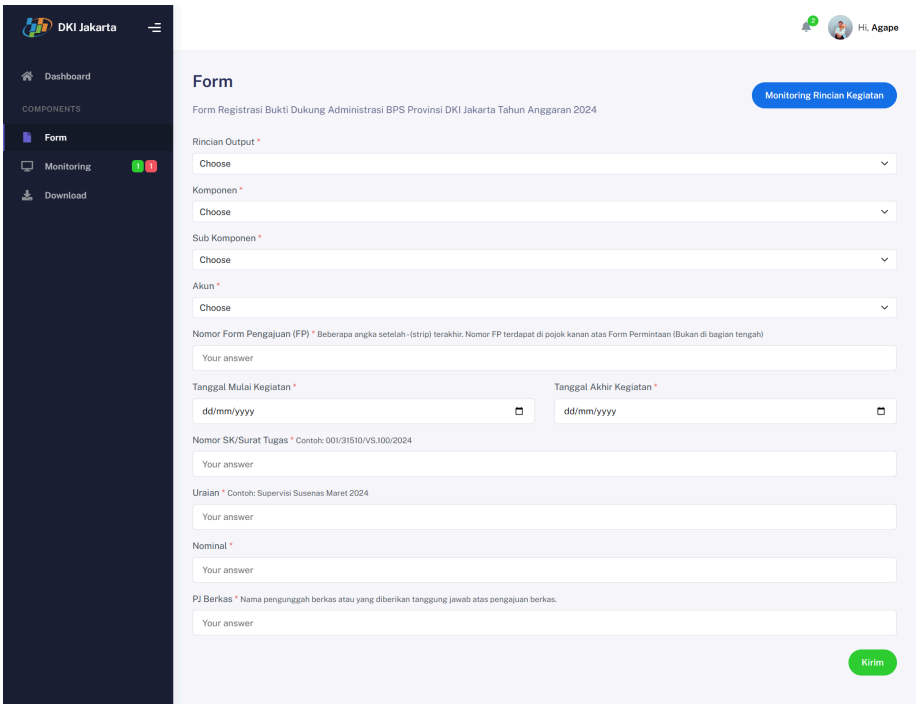


2) Form Operator (Pegawai BPS Provinsi DKI Jakarta)

WIREFRAME

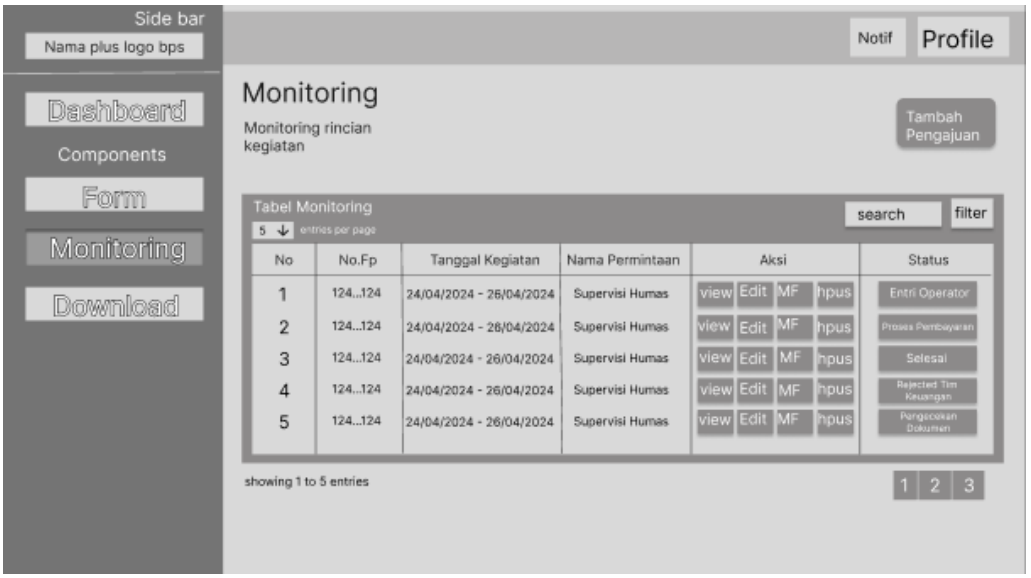


MOCKUP

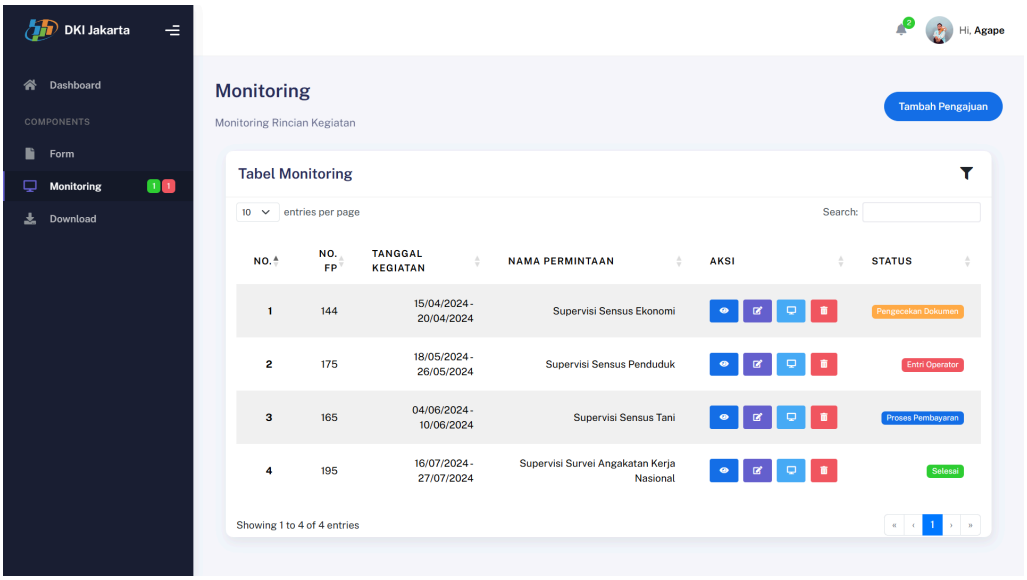


3) Monitoring Operator (Pegawai BPS Provinsi DKI Jakarta)

WIREFRAME

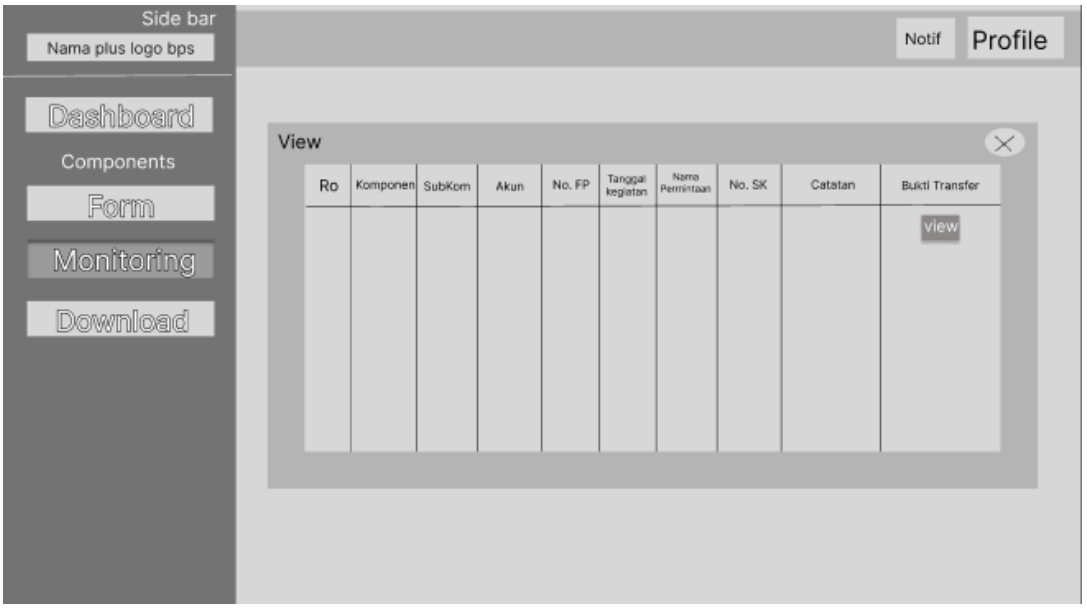


MOCKUP

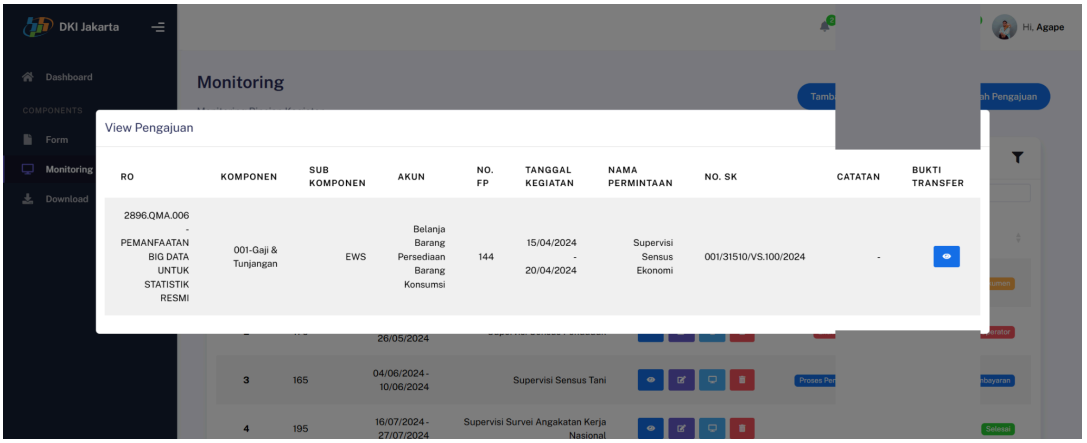


4) View Pengajuan

WIREFRAME

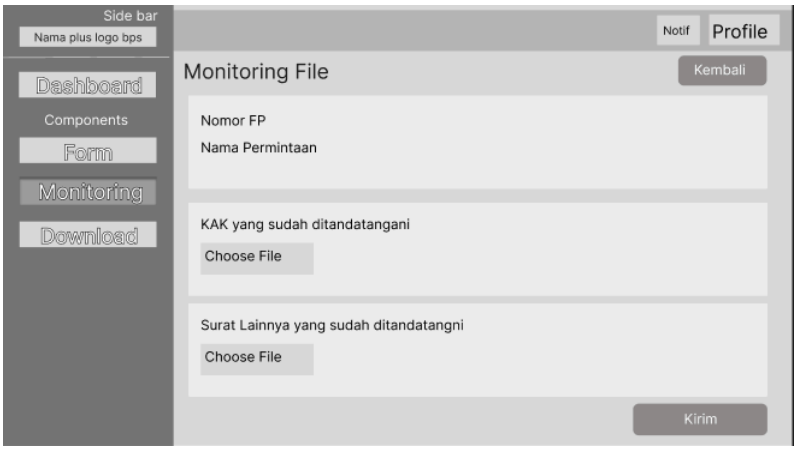


MOCKUP

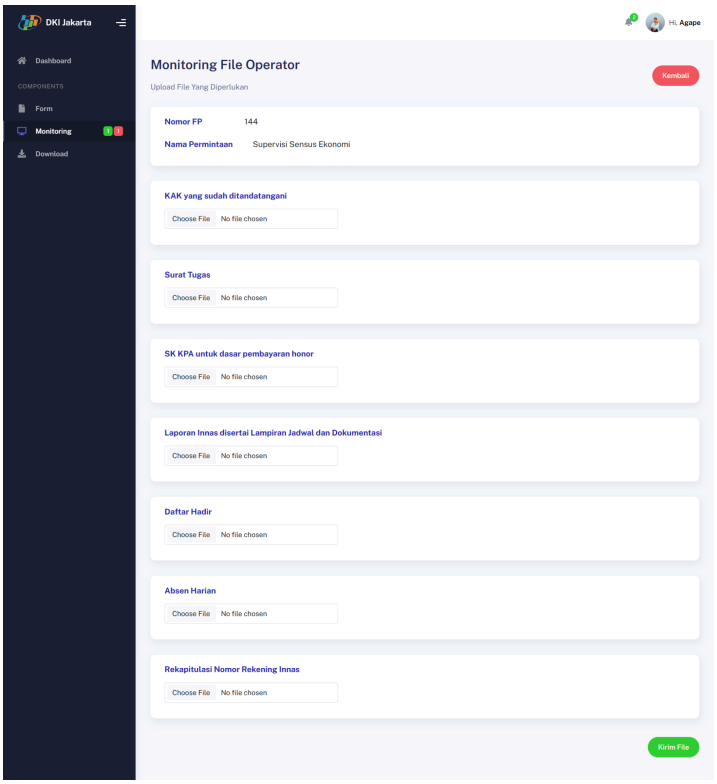


5) Monitoring File

WIREFRAME

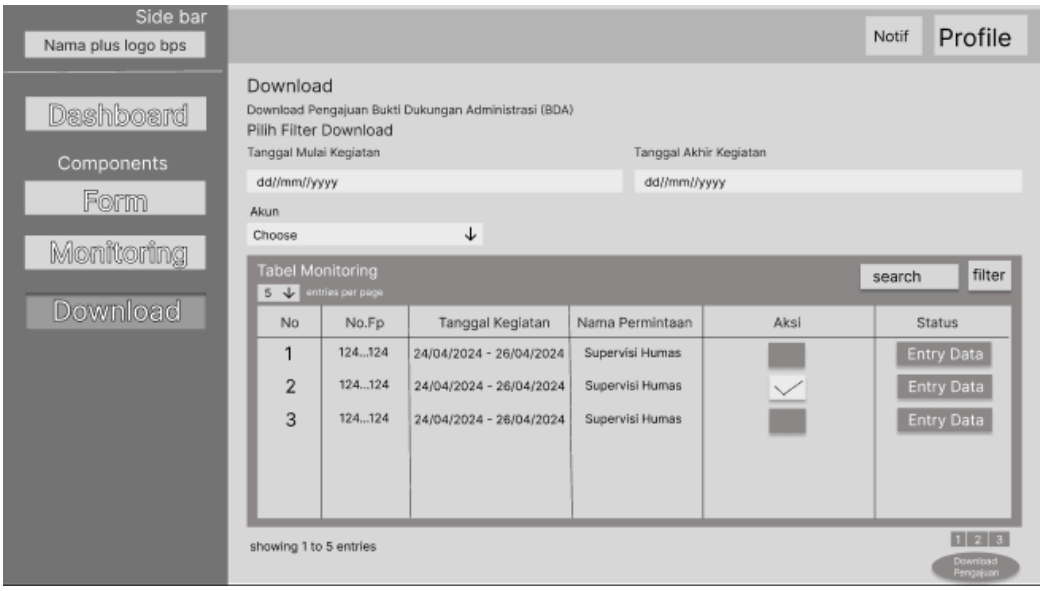


MOCKUP

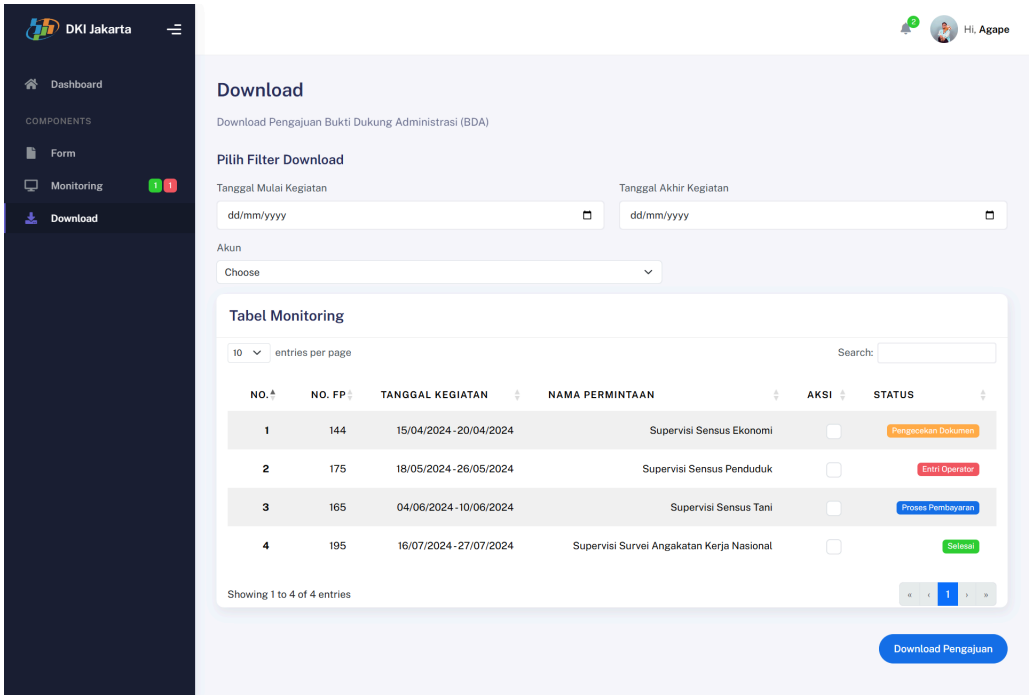


6) Download

WIREFRAME



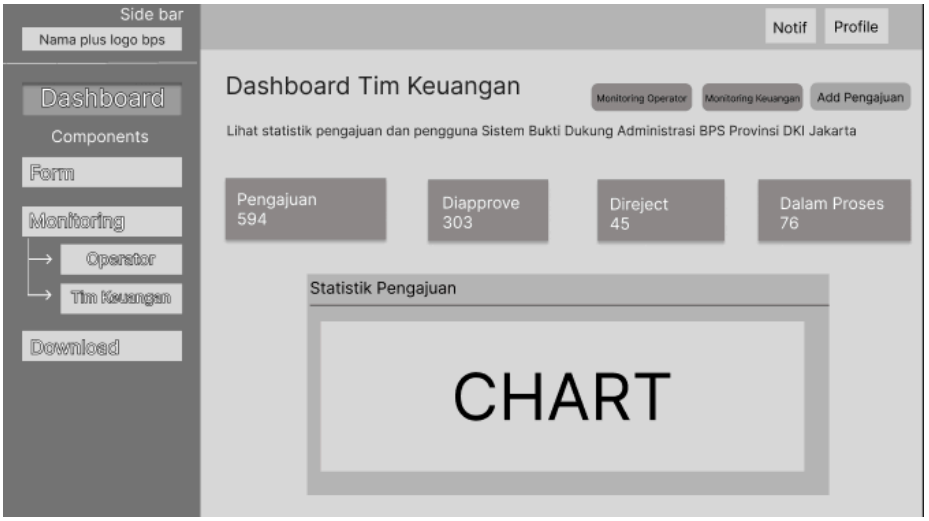
MOCKUP



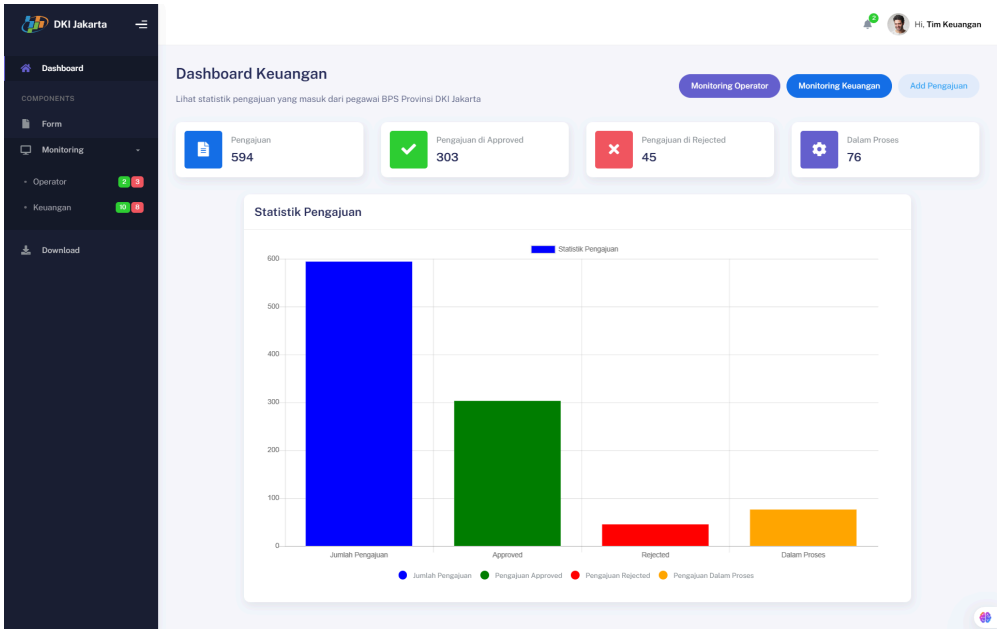
3.6.4 Tim Keuangan

1) Dashboard Tim Keuangan

WIREFRAME

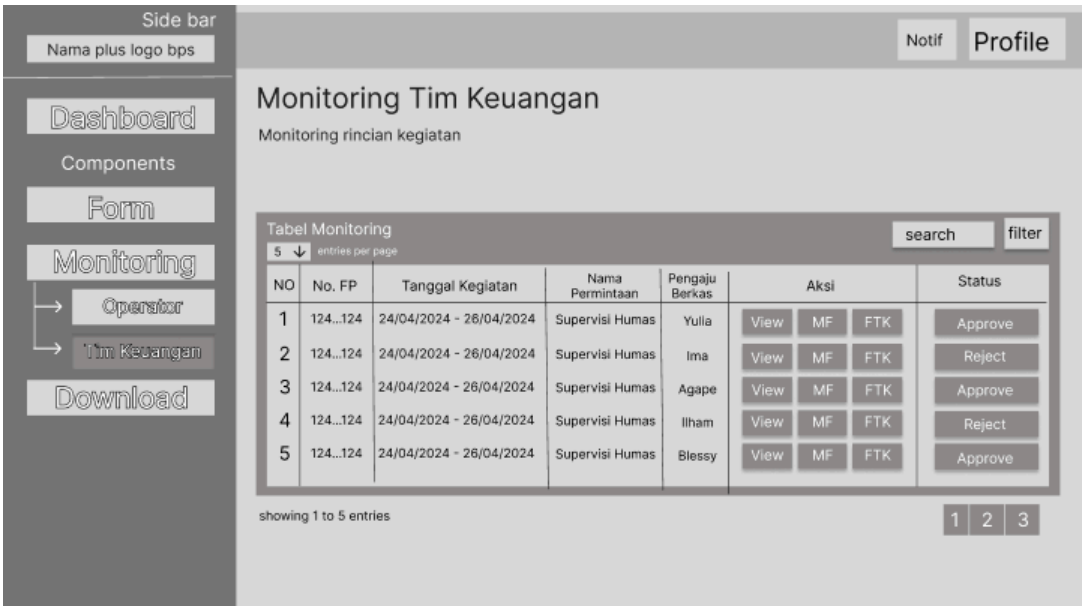


MOCKUP

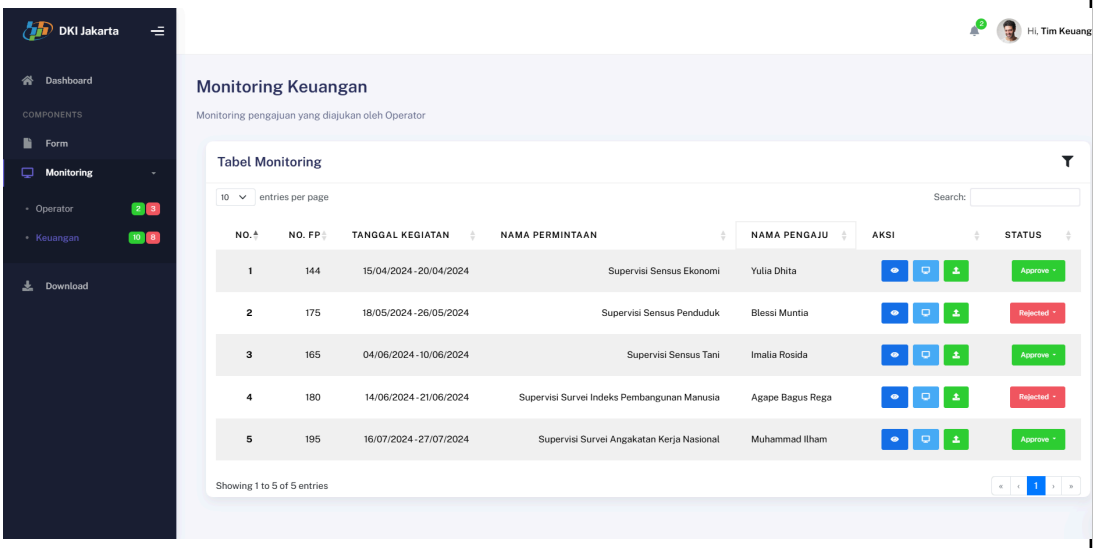


2) Monitoring Keuangan

WIREFRAME

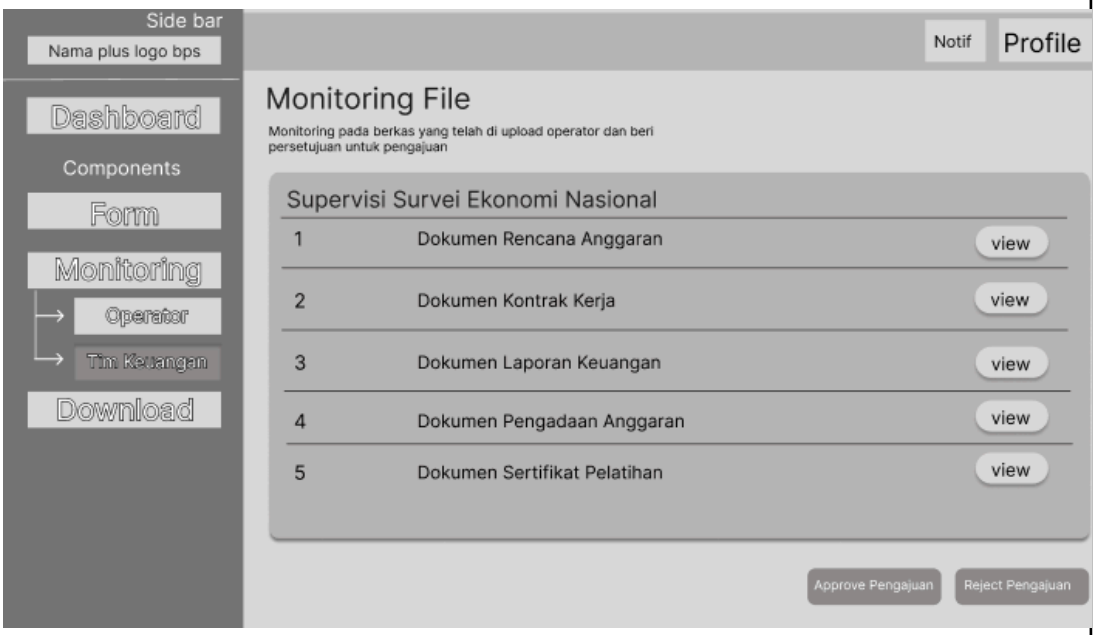


MOCKUP

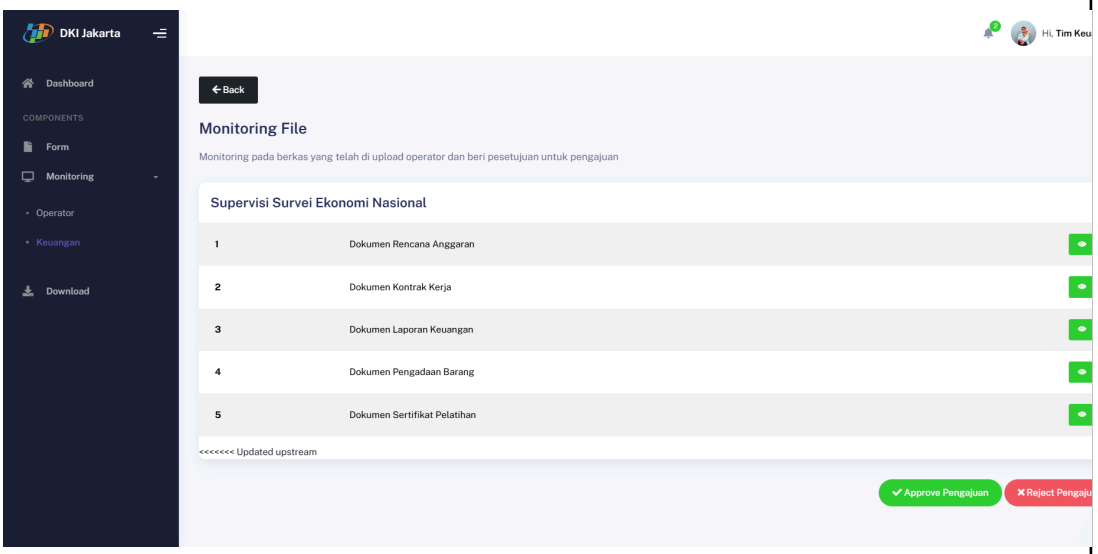


3) Monitoring File Keuangan

WIREFRAME

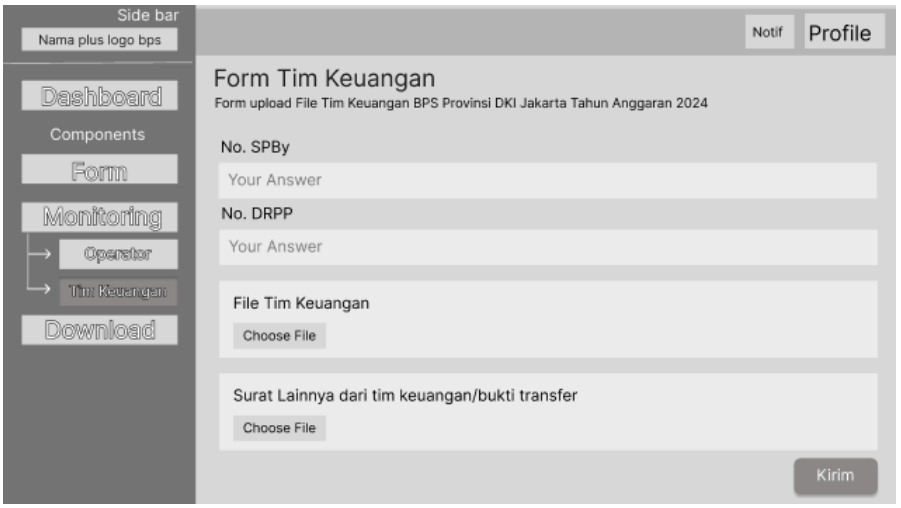


MOCKUP

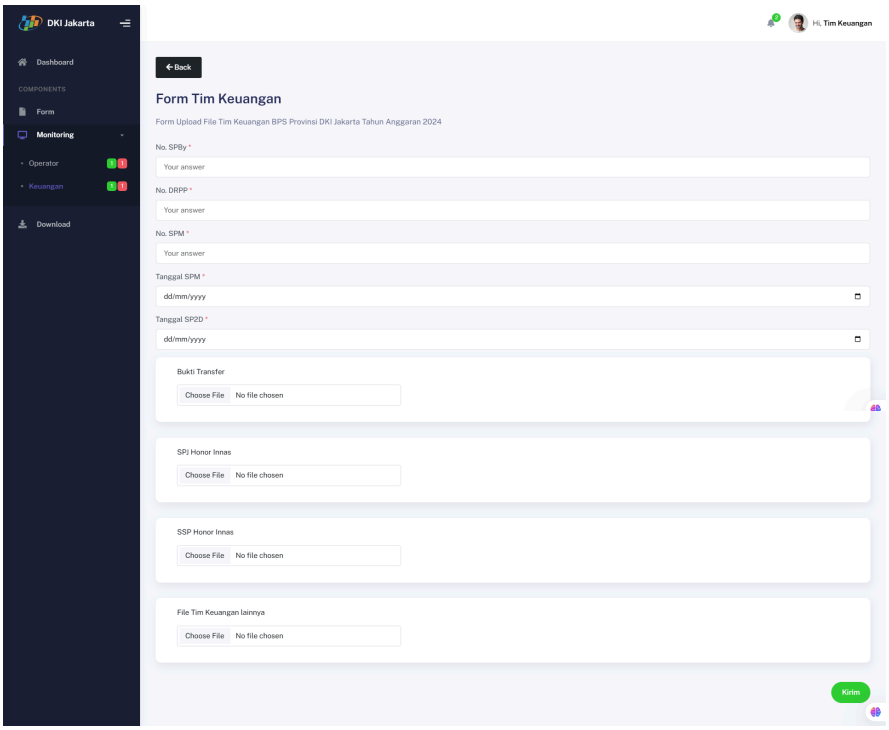


4) Form Tim Keuangan

WIREFRAME



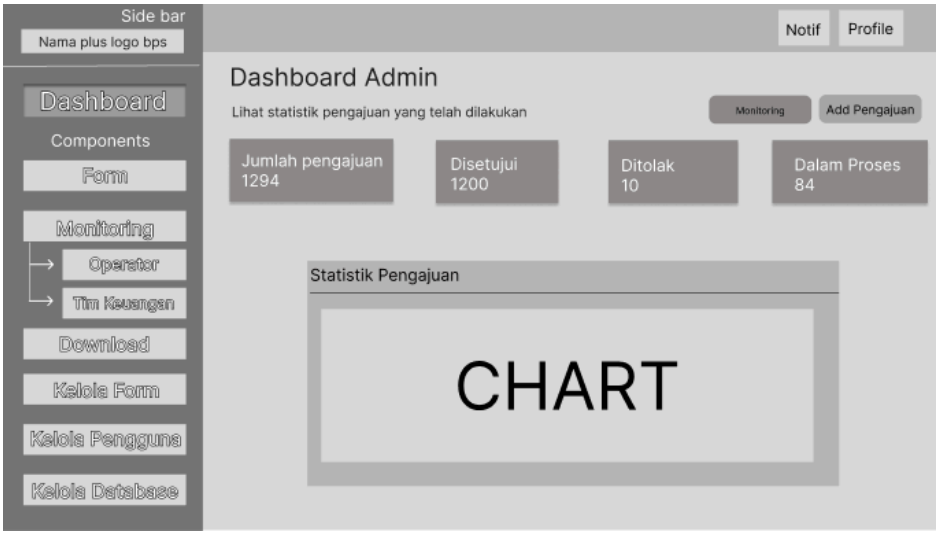
MOCKUP



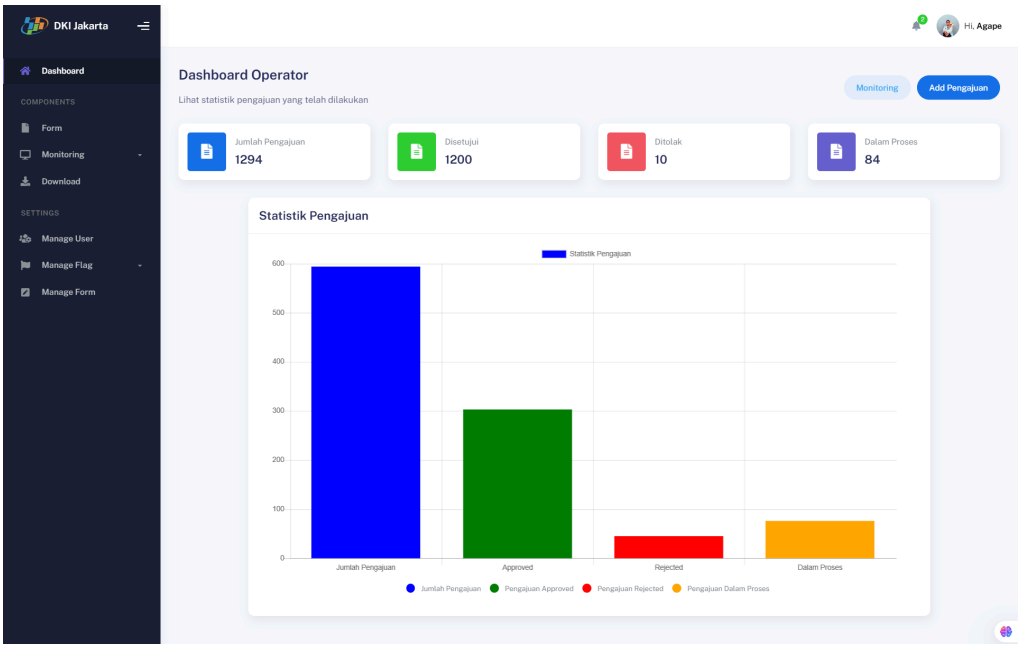
3.6.5 Admin

1) Dashboard Admin

WIREFRAME

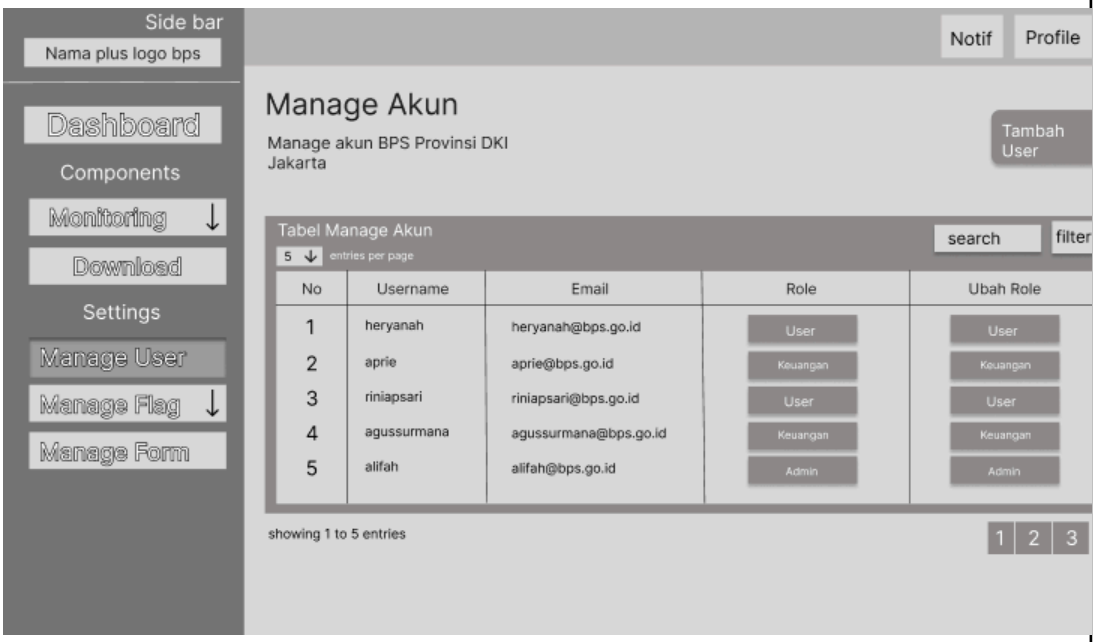


MOCKUP

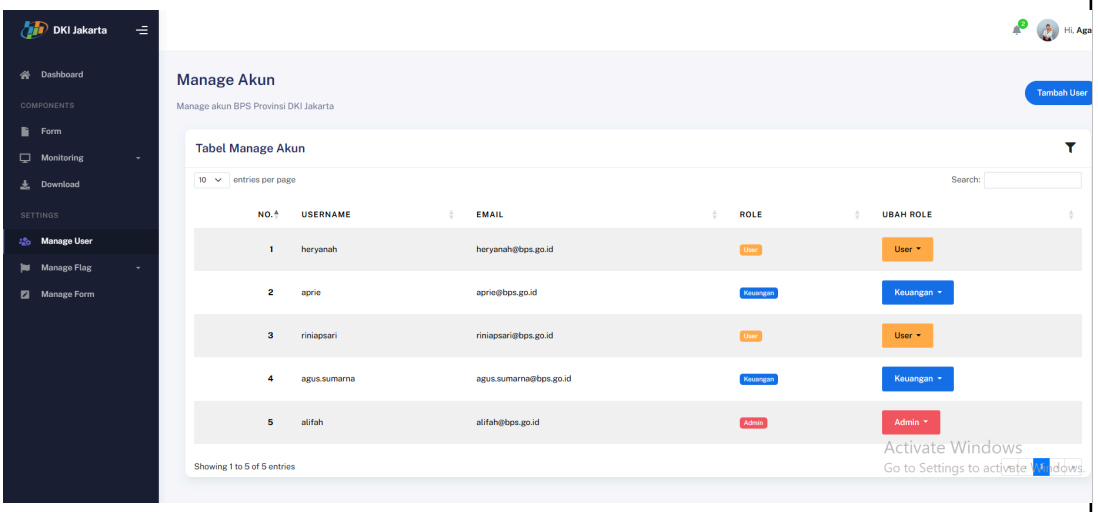


2) Mengatur Role User

WIREFRAME



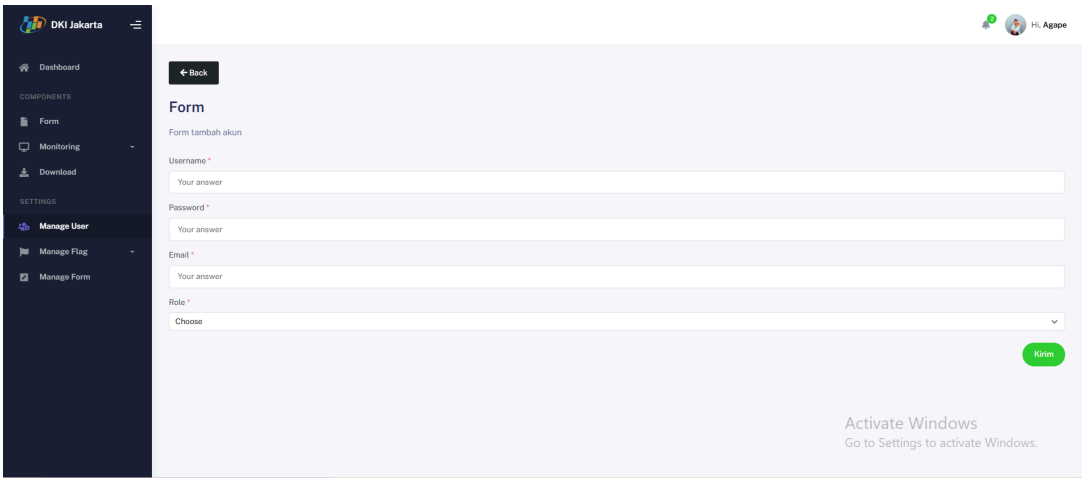
MOCKUP



3) Menambah User Baru

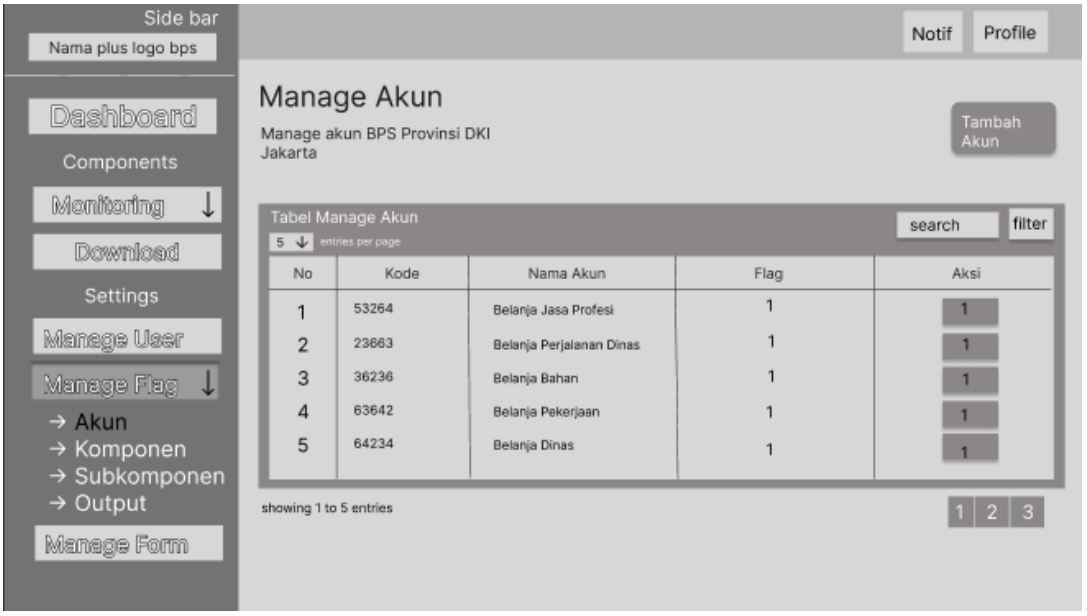
WIREFRAME

MOCKUP



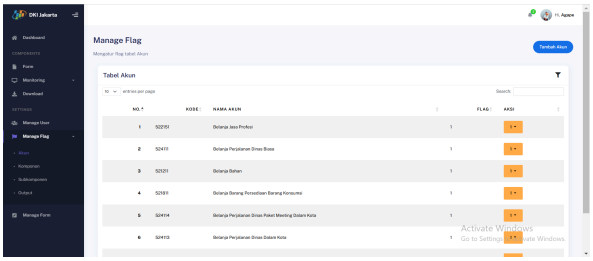
4) Manage MAK

WIREFRAME

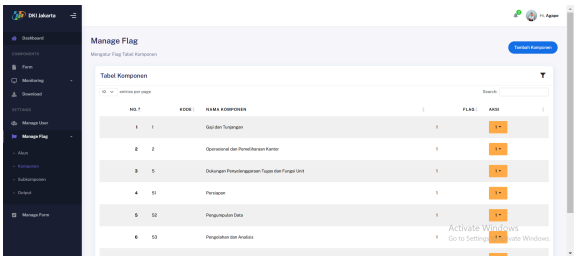


MOCKUP

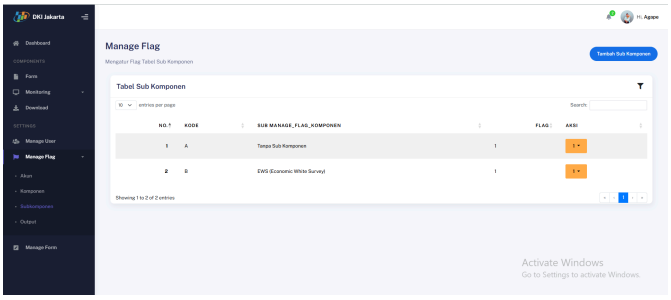
- ***Akun***



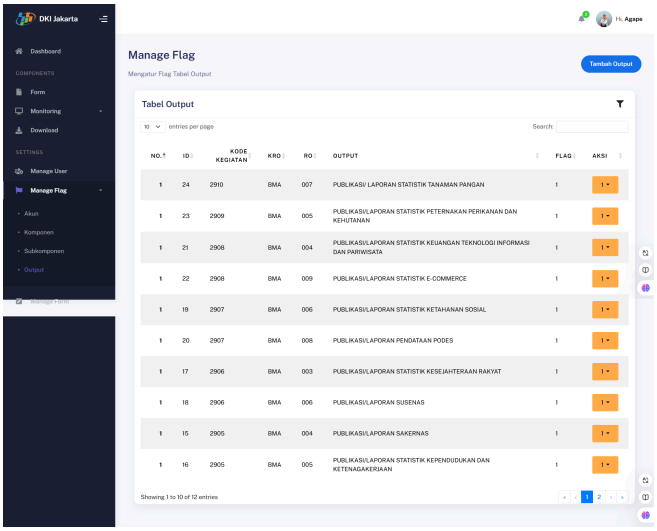
- ***Komponen***



- ***Sub Komponen***

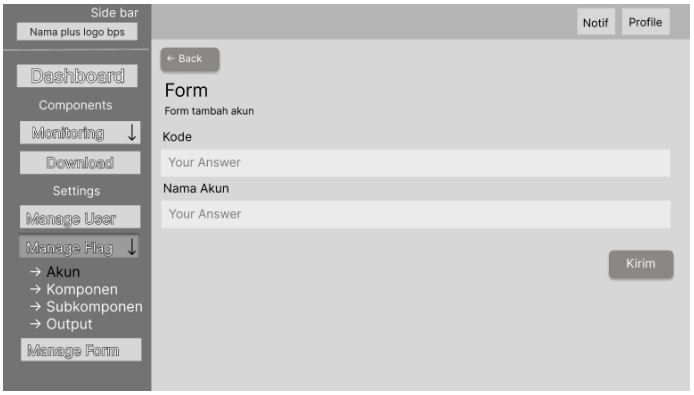


- ***Output***



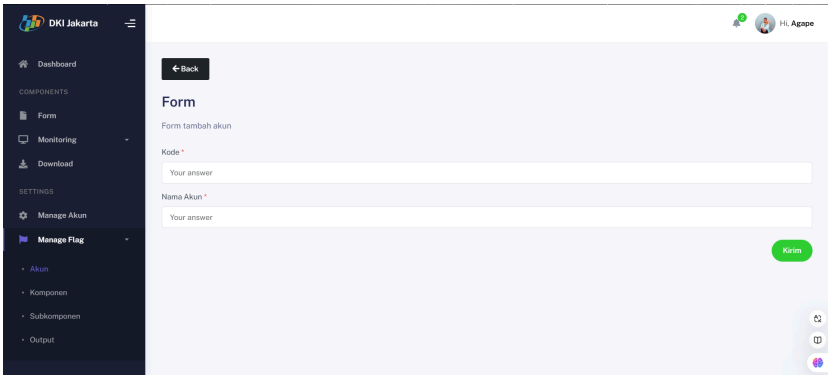
5) Tambah Pilihan pada Manage MAK

WIREFRAME

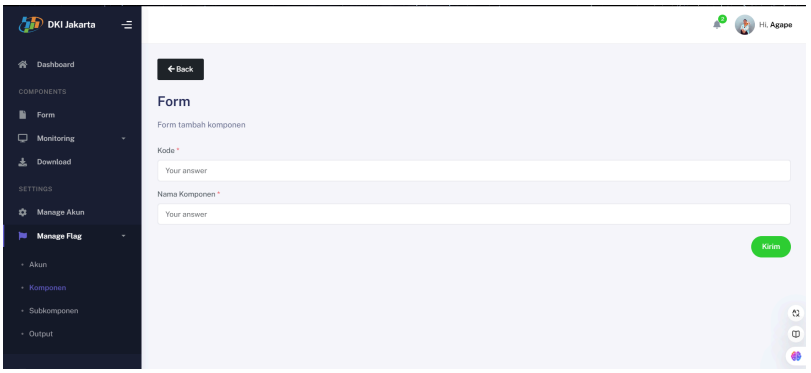


MOCKUP

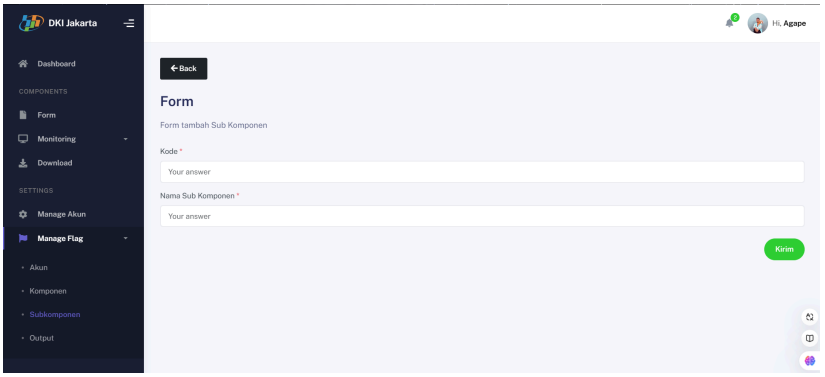
- ***Tambah Akun***



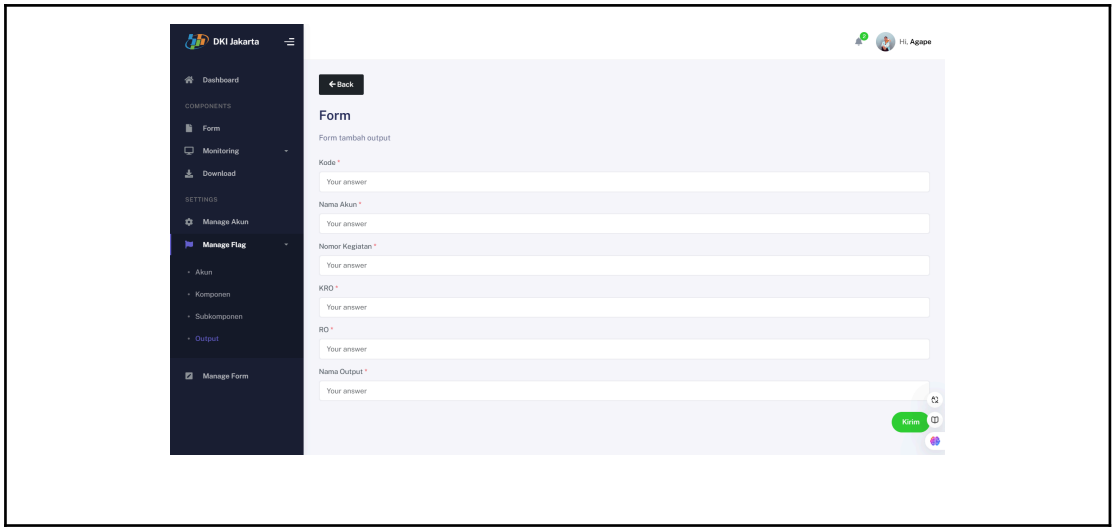
- ***Tambah Komponen***



- ***Tambah Sub Komponen***

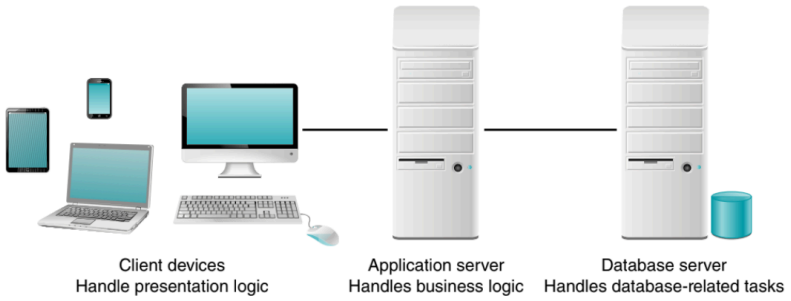


- ***Tambah Output***



3.7 Arsitektur System

3.7.1 Rancangan Arsitektur Sistem



3.7.2 Penjelasan Rancangan Arsitektur Sistem

Diagram arsitektur ini menggambarkan model Three-Tier yang terdiri dari tiga lapisan utama, yaitu Client Devices, Application Server, dan Database Server. Model ini dirancang untuk memisahkan setiap lapisan agar dapat dikembangkan, dioperasikan, dan ditingkatkan secara independen sesuai kebutuhan aplikasi.

1. Client Devices (Presentation Layer)

   Lapisan ini bertanggung jawab untuk menangani presentation logic atau logika presentasi yang berfokus pada antarmuka pengguna. Komponen ini mencakup perangkat-perangkat klien seperti komputer, tablet, dan ponsel, yang digunakan pengguna untuk berinteraksi dengan aplikasi. Di sini, sistem menerima dan menampilkan informasi kepada pengguna melalui antarmuka,

dan melakukan validasi awal input yang diberikan oleh pengguna sebelum diteruskan ke lapisan berikutnya. Lapisan ini dirancang agar ramah pengguna dan berfungsi sebagai pintu masuk utama bagi pengguna untuk mengakses layanan aplikasi.

2. Application Server (Business Logic Layer)

   Lapisan ini bertindak sebagai business logic tier yang bertanggung jawab atas pengolahan logika bisnis aplikasi. Komponen pada lapisan ini berperan sebagai perantara antara Client Devices dan Database Server. Di sinilah semua aturan dan proses bisnis diterapkan, termasuk pemrosesan instruksi dari pengguna, pengelolaan sesi, serta pemisahan antara data dan interaksi pengguna. Server aplikasi menerima permintaan dari perangkat klien, memprosesnya sesuai dengan aturan bisnis, dan meneruskan hasilnya ke lapisan presentasi atau mengambil data dari lapisan data jika diperlukan. Model ini memungkinkan keamanan dan kontrol yang lebih baik karena pengguna tidak memiliki akses langsung ke database.

3. Database Server (Data Layer)

   Lapisan ini adalah data layer yang menangani semua tugas yang berhubungan dengan pengelolaan data. Database server menyimpan, mengatur, dan menyediakan akses ke data yang dibutuhkan oleh aplikasi. Komponen ini menyimpan data dalam sistem manajemen basis data (DBMS) dan memastikan data dapat diakses dengan cepat dan aman. Selain menyimpan dan mengelola data, lapisan ini juga menyediakan fungsi seperti pengambilan data, pembaruan, dan penghapusan data sesuai dengan permintaan dari Application Server. Arsitektur ini memungkinkan data tetap terlindungi dan terisolasi dari akses langsung pengguna, meningkatkan keamanan dan integritas data.

Arsitektur Three-Tier ini membuat aplikasi menjadi lebih terstruktur dan mudah dipelihara, karena setiap lapisan memiliki tanggung jawab yang spesifik dan dapat dikembangkan secara independen. Struktur ini memungkinkan aplikasi untuk menangani jumlah pengguna yang lebih besar karena beban kerja dapat dibagi antara berbagai server.

BAB IV
DESAIN PROGRAM

4.1 User Access Matrix

Role Use	Modul					
	Form Registrasi BDA	Mengunggah Berkas BDA	Memantau Pengajuan & Berkas BDA	Pengecekan Berkas BDA	Download Pengajuan BDA	Manage Users
Operator	C	C	RUD	R	R	
Tim Keuangan	C	C	RUD	RU	R	
Admin	CRUD	CRUD	RUD	RU	R	CRUD

Keterangan: C = Create

R = Read

U = Update/Approval

D = Delete

Penjelasan:

- Form Registrasi BDA

  Form ini digunakan operator untuk mengajukan membuat (create) pengajuan kepada tim keuangan, tim keuangan juga dapat mengakses untuk pengajuan form juga. Sedangkan admin dapat CRUD, karena role admin bisa melihat, mengupdate, dan mendelete isi pengajuan form.
- Mengunggah Berkas BDA

  Form ini digunakan operator untuk mengunggah (create) berkas kepada tim keuangan, tim keuangan juga dapat mengakses untuk mengunggah berkas juga. Sedangkan admin dapat CRUD, karena role admin bisa melihat, mengupdate, dan mendelete berkas yang diunggah.
- Memantau Pengajuan & Berkas BDA

  Pada bagian ini, Operator, tim keuangan, dan admin dapat memantau isian form dan berkas-berkas yang diunggah sebelumnya, lalu bisa juga mengupdate file-file yang diperlukan untuk kelengkapan pengajuan. Dan bisa juga mendelete pengajuan yang diajukan.

- Pengecekan Berkas BDA

  Pada bagian ini, operator hanya dapat membaca/memantau saja apakah berkasnya sudah di approve atau belum. Sedangkan tim keuangan dan admin dapat melakukan tidak hanya membaca/memantau, tapi juga melakukan update terkait berkas yang telah diajukan (melakukan approve atau reject, bahkan mengunggah file yang diperlukan).

- Download Pengajuan BDA

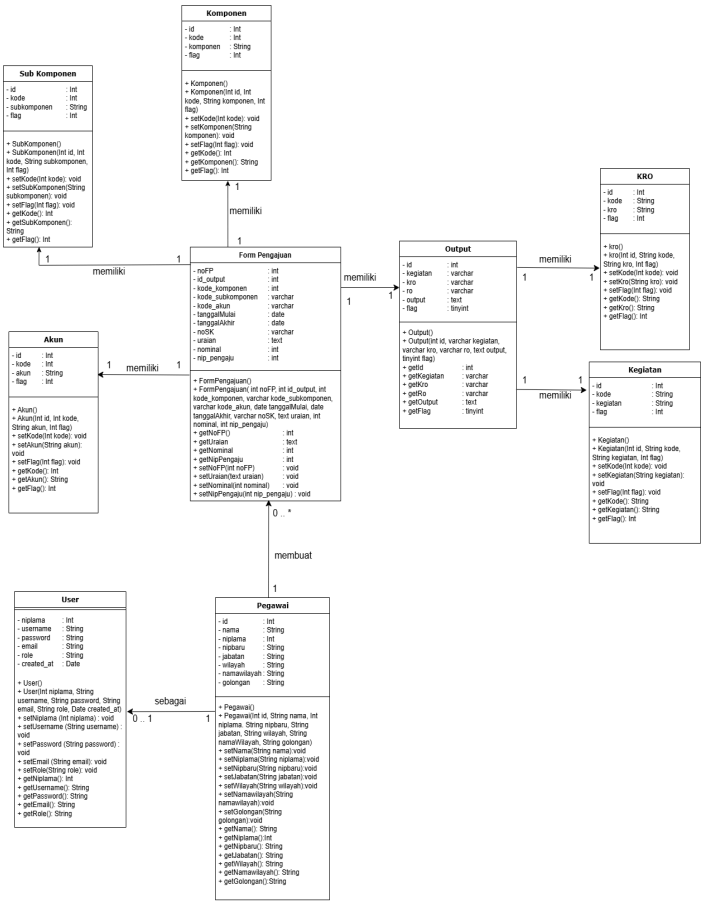
  Pada bagian ini, operator, tim keuangan, dan admin dapat melakukan unduhan terhadap pengajuan BDA yang sudah di selesai dilakukan pengecekan menyeluruh oleh tim keuangan.

- Manage Users

  Pada bagian ini, hanya ada pada role admin, selain role admin tidak bisa mengakses bagian ini. Admin dapat mengubah role akun orang, dan menambah dan delete akun.

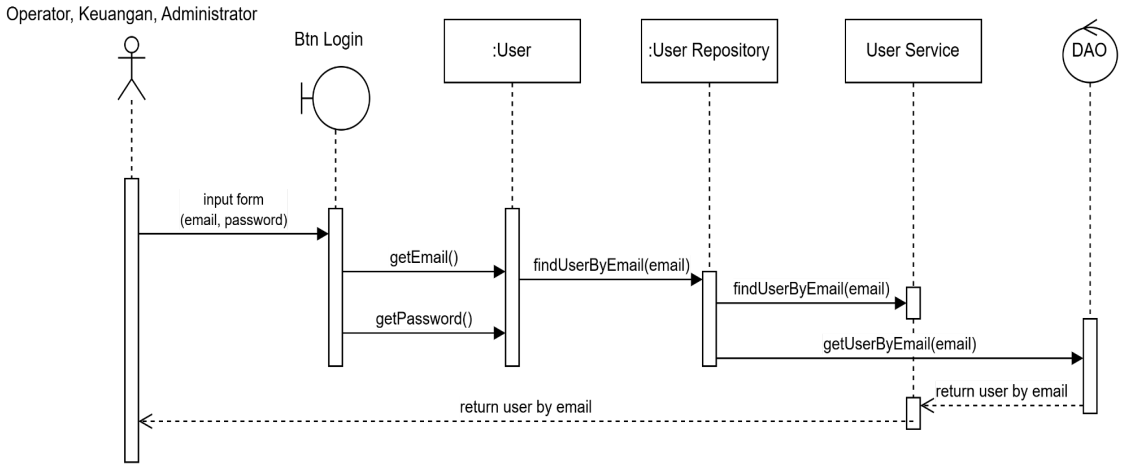
4.2 Desain Program

A. Class Diagram Sistem BDA



B. Sequence Diagram Sistem BDA

1. Login



Sequence diagram tersebut menggambarkan alur autentikasi pengguna dalam sistem bukti dukung administrasi. Proses dimulai dari antarmuka *Login*, di mana pengguna memasukkan data berupa email dan password melalui form. Setelah data diterima, *Login* memanggil *User Service* untuk memproses autentikasi. *User Service* kemudian mengambil data input dengan memanggil metode getEmail() dan getPassword() sebelum meneruskan permintaan ke *User Repository*. Selanjutnya, *User Repository* menggunakan DAO (Data Access Object) untuk mengakses database dan menjalankan query pencarian data pengguna berdasarkan email dengan metode findUserByEmail(email). Setelah data ditemukan, DAO mengembalikan hasilnya ke *User Repository*, yang kemudian meneruskan informasi tersebut ke *User Service*. Di tahap ini, *User Service* memvalidasi data pengguna dengan mencocokkan email dan password. Jika validasi berhasil, *User Service* mengirimkan respon ke *Login* berupa informasi pengguna yang valid. Proses ini mencakup kerja sama antara komponen antarmuka, layanan, repositori, dan akses data untuk memastikan autentikasi pengguna berjalan dengan aman dan efisien.

2. Mengisi Form Pengajuan

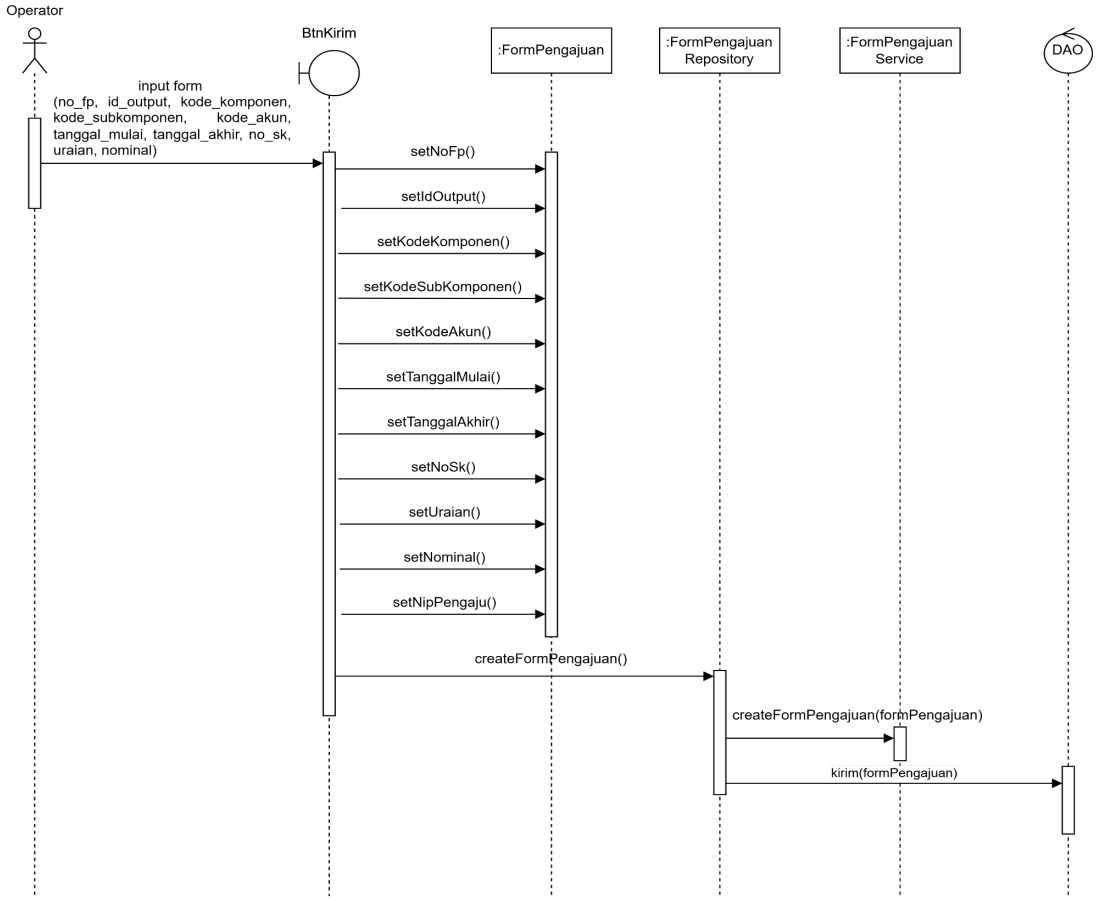
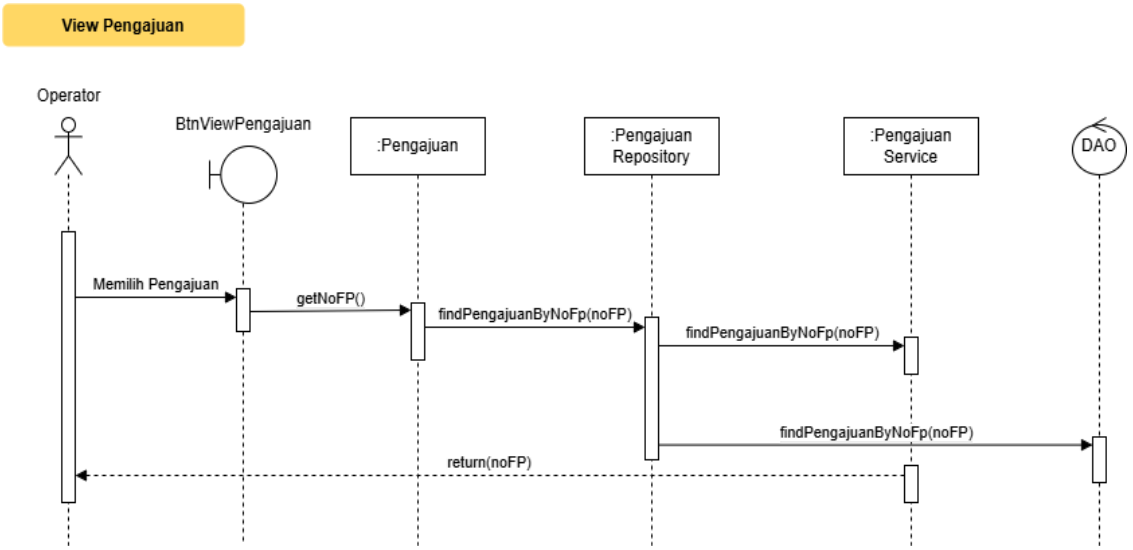


Diagram sequence di atas menggambarkan proses pengisian form pengajuan yang dimulai dari pengguna (aktor) yang menginput data form, termasuk informasi seperti nomor form (no_fp), ID output (id_output), kode komponen (kode_komponen), kode subkomponen (kode_subkomponen), kode akun (kode_akun), tanggal mulai, tanggal akhir, nomor surat keputusan (no_sk), uraian, nominal, dan NIP pengaju. Setelah data dimasukkan, setiap atribut pada form diproses melalui serangkaian metode set yang ada di objek FormPengajuan, seperti setNoFp(), setIdOutput(), dan seterusnya, hingga semua data tersimpan di dalam objek. Selanjutnya, form tersebut diproses melalui metode createFormPengajuan() pada FormPengajuanService, yang kemudian memanggil createFormPengajuan(formPengajuan) di FormPengajuanRepository untuk menyimpan data ke database menggunakan DAO. Terakhir, form dikirim melalui metode kirim(formPengajuan) untuk melanjutkan ke tahap berikutnya dalam proses pengajuan.

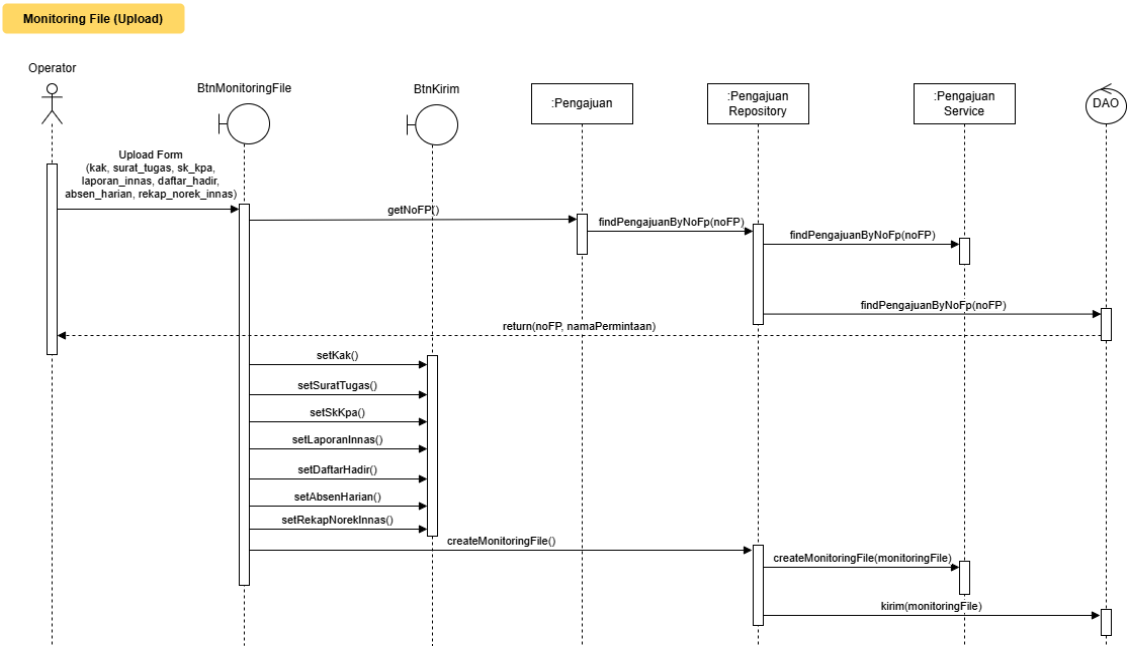
3. **Monitoring Pengajuan Operator**

- **View Pengajuan**



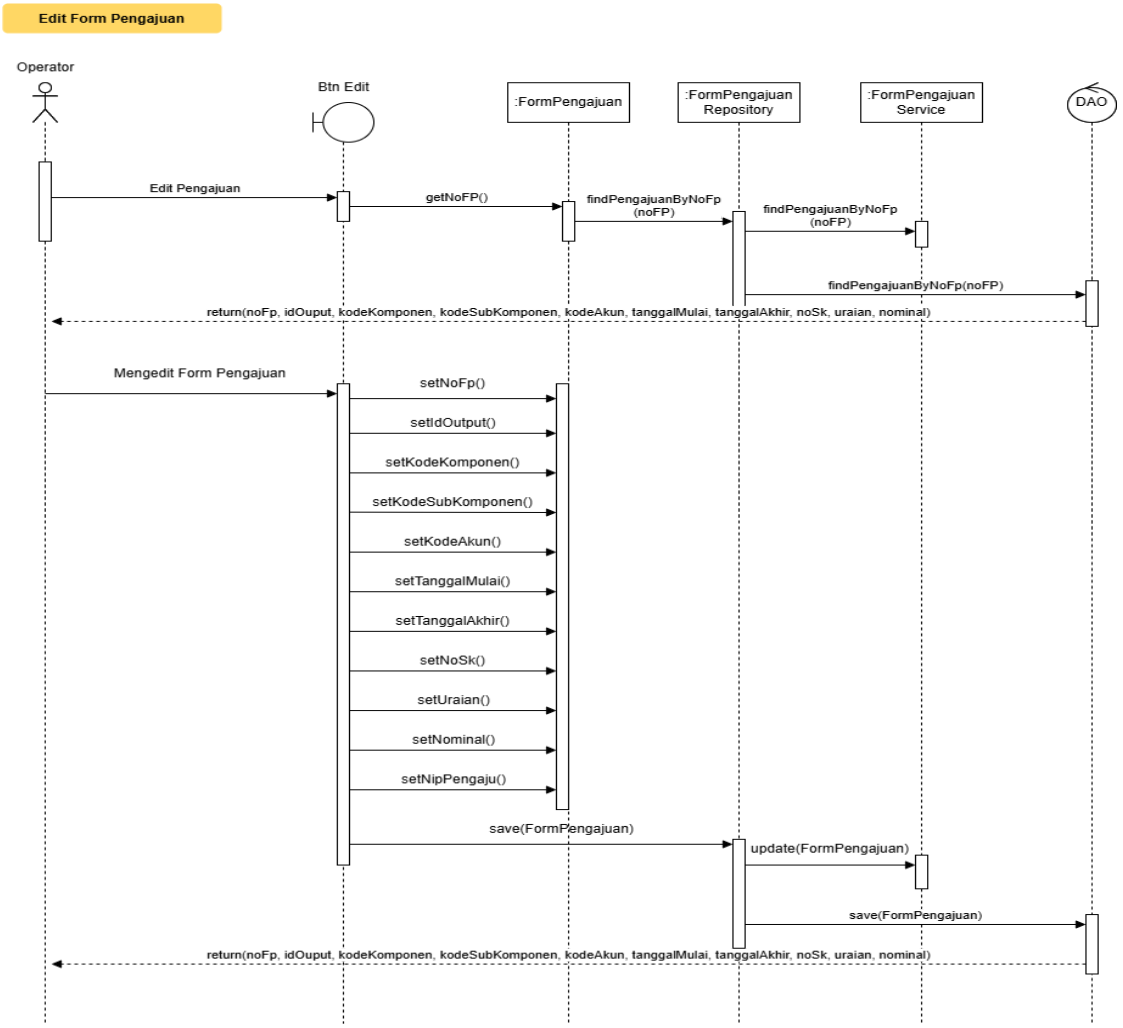
Proses View Pengajuan dimulai ketika Operator memilih pengajuan yang ingin dilihat, memicu tombol *View Pengajuan* untuk menjalankan metode getNoFP(). Tombol ini kemudian memanggil entitas Pengajuan untuk mengambil nomor faktur pengajuan (noFP). Selanjutnya, entitas Pengajuan mengirim permintaan ke Pengajuan Repository melalui metode findPengajuanByNoFp(noFP) untuk mencari data pengajuan berdasarkan noFP di dalam basis data. Repository tersebut kemudian berinteraksi dengan Pengajuan Service menggunakan metode yang sama untuk memastikan data yang ditemukan sesuai dengan logika bisnis. Setelah data berhasil ditemukan, Pengajuan Repository mengembalikan data ke entitas Pengajuan, yang kemudian diteruskan ke antarmuka tombol *View Pengajuan* dan akhirnya ditampilkan kepada Operator. Proses ini memastikan alur data berjalan dari pengguna hingga akses data dengan struktur yang terorganisasi.

- **Monitoring File (Upload)**



Monitoring File (Upload File) merupakan proses upload file untuk monitoring pada sebuah sistem. Proses dimulai dengan pengguna mengunggah form melalui fitur "Upload Form," yang memanggil metode getNoFP() untuk mendapatkan nomor form pengajuan (noFP). Nomor ini kemudian digunakan untuk mencari detail pengajuan melalui metode findPengajuanByNoFp(noFP) di repository, yang mengembalikan data terkait, termasuk noFP dan nama permintaan. Setelah itu, berbagai atribut file, seperti setKak(), setSuratTugas(), setSkKpa(), dan lainnya, diatur sesuai kebutuhan. Data monitoring file kemudian dibuat menggunakan metode createMonitoringFile() dan dikirim ke repository untuk disimpan. Akhirnya, file yang telah diunggah diteruskan melalui metode kirim(monitoringFile). Diagram ini menampilkan interaksi yang terstruktur antara pengguna, service, dan repository untuk mendukung proses upload dan manajemen file.

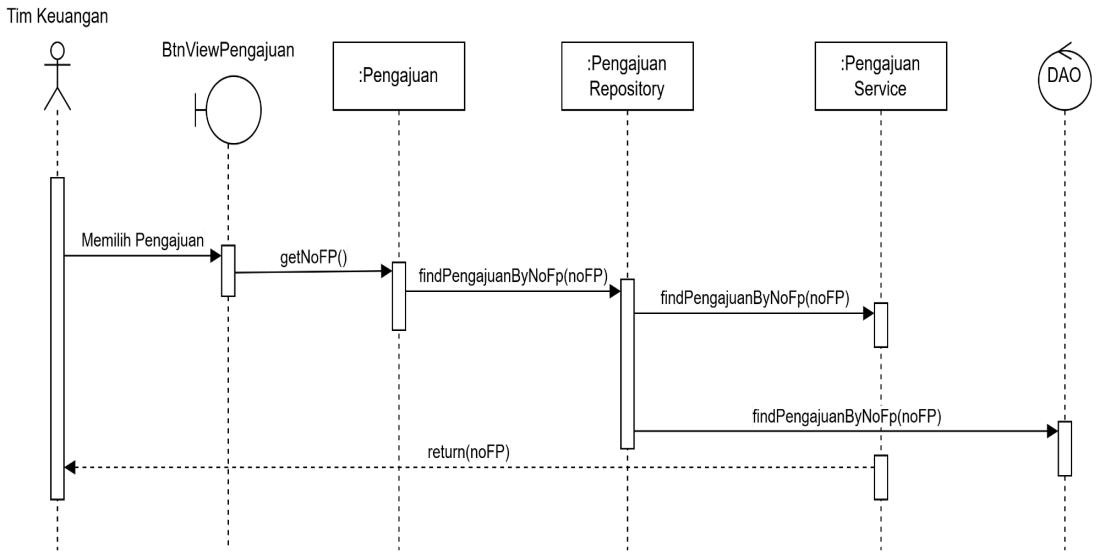
- **Edit Pengajuan**



Edit Form Pengajuan merupakan proses pengeditan form pengajuan dalam sebuah sistem. Proses dimulai dengan pengguna melakukan aksi "Edit Pengajuan," yang memanggil metode getNoFP() untuk mendapatkan nomor form pengajuan (noFP). Nomor ini kemudian digunakan untuk mencari data terkait di database melalui metode findPengajuanByNoFp(noFP) pada repository, yang mengembalikan informasi detail dari form pengajuan. Setelah data diperoleh, pengguna dapat mengubah informasi yang diinginkan. Perubahan tersebut diteruskan kembali ke service dan repository untuk diperbarui dalam database menggunakan metode save(FormPengajuan). Akhirnya, form pengajuan yang telah diperbarui disimpan dan ditampilkan kembali kepada pengguna. Diagram ini mencerminkan arsitektur berorientasi layanan dengan komponen DAO (Data Access Object) untuk manipulasi data.

4. **Monitoring Pengajuan Tim Keuangan**

- **View Pengajuan**



Proses View Pengajuan dimulai ketika Tim Keuangan memilih pengajuan yang ingin dilihat, memicu tombol *View Pengajuan* untuk menjalankan metode getNoFP(). Tombol ini kemudian memanggil entitas Pengajuan untuk mengambil nomor faktur pengajuan (noFP). Selanjutnya, entitas Pengajuan mengirim permintaan ke Pengajuan Repository melalui metode findPengajuanByNoFp(noFP) untuk mencari data pengajuan berdasarkan noFP di dalam basis data. Repository tersebut kemudian berinteraksi dengan Pengajuan Service menggunakan metode yang sama untuk memastikan data yang ditemukan sesuai dengan logika bisnis. Setelah data berhasil ditemukan, Pengajuan Repository mengembalikan data ke entitas Pengajuan, yang kemudian diteruskan ke antarmuka tombol *View Pengajuan* dan akhirnya ditampilkan kepada Tim Keuangan. Proses ini memastikan alur data berjalan dari pengguna hingga akses data dengan struktur yang terorganisasi.

- **Monitoring File (Approve Pengajuan)**

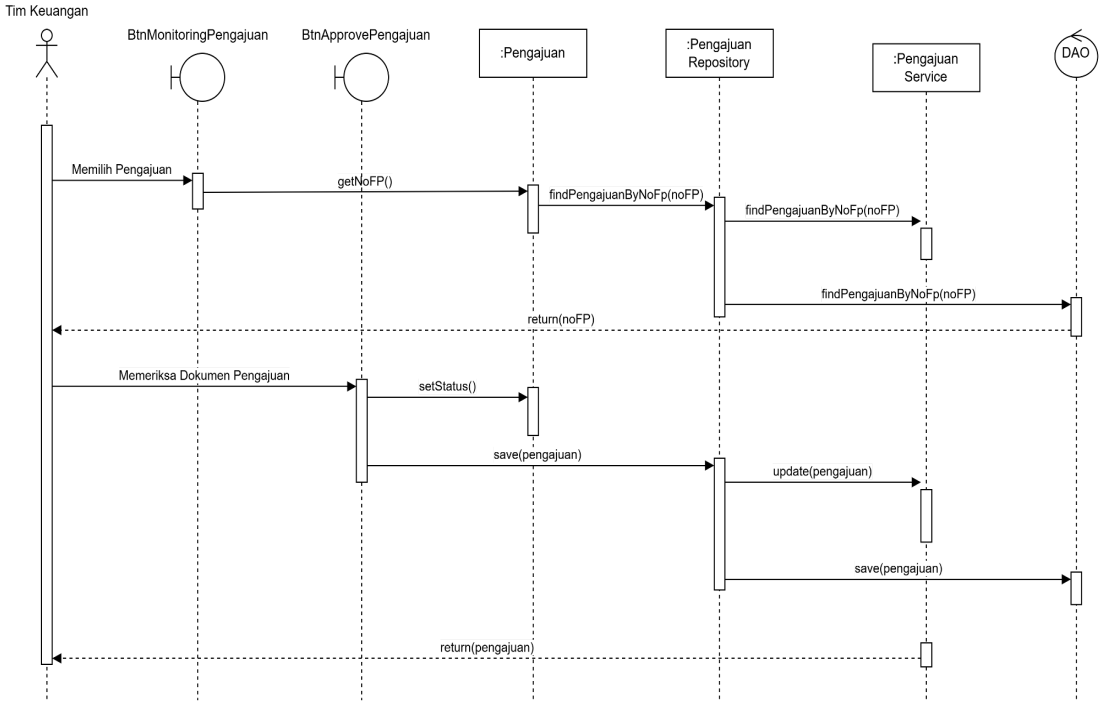


Diagram sequence di atas menggambarkan proses persetujuan pengajuan oleh tim keuangan. Proses dimulai dengan tim keuangan memilih pengajuan melalui tombol monitoring pengajuan (BtnMonitoringPengajuan). Sistem kemudian memanggil metode get(noFP) untuk mengambil data pengajuan berdasarkan nomor form pengajuan (noFP). Data pengajuan diambil melalui metode findPengajuanByNoFp(noFP) yang diproses oleh PengajuanRepository dan diteruskan oleh PengajuanService untuk mengakses data di DAO. Setelah data berhasil diambil, tim keuangan memeriksa dokumen pengajuan. Jika dokumen sesuai, status pengajuan diubah melalui metode setStatus() dan disimpan kembali ke database menggunakan metode save(pengajuan). Proses penyimpanan ini melalui PengajuanService yang memanggil update(pengajuan) dan diteruskan ke PengajuanRepository. Setelah pengajuan berhasil disimpan, sistem mengembalikan data pengajuan ke tim keuangan untuk memastikan status pengajuan telah diperbarui.

- **Monitoring File (Reject Pengajuan)**

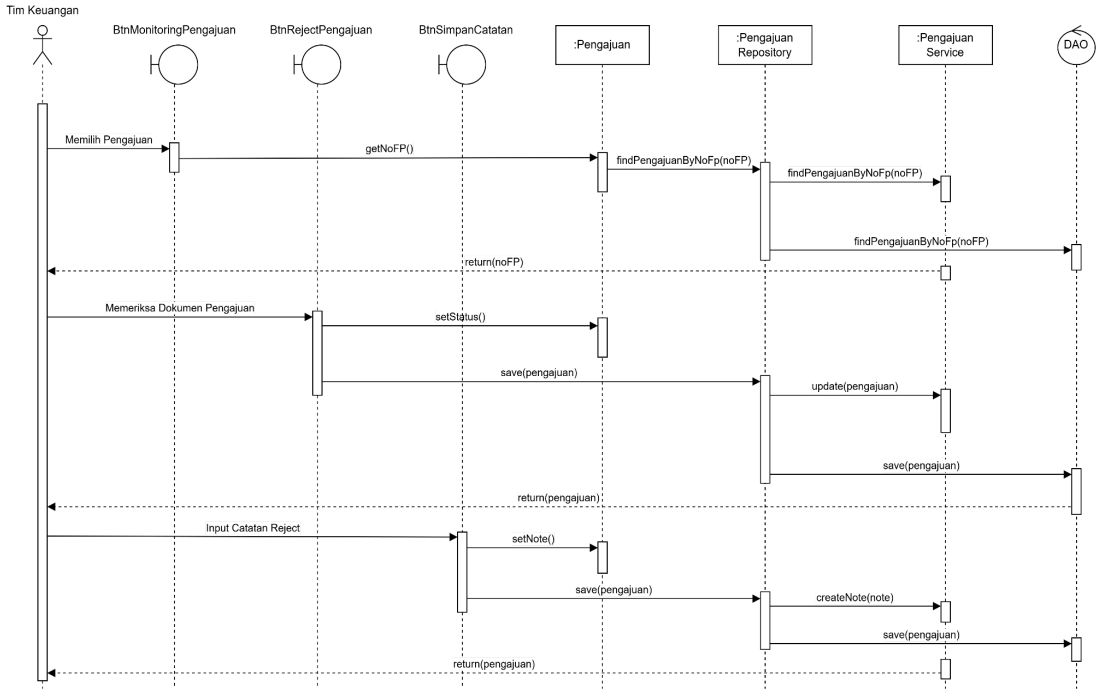
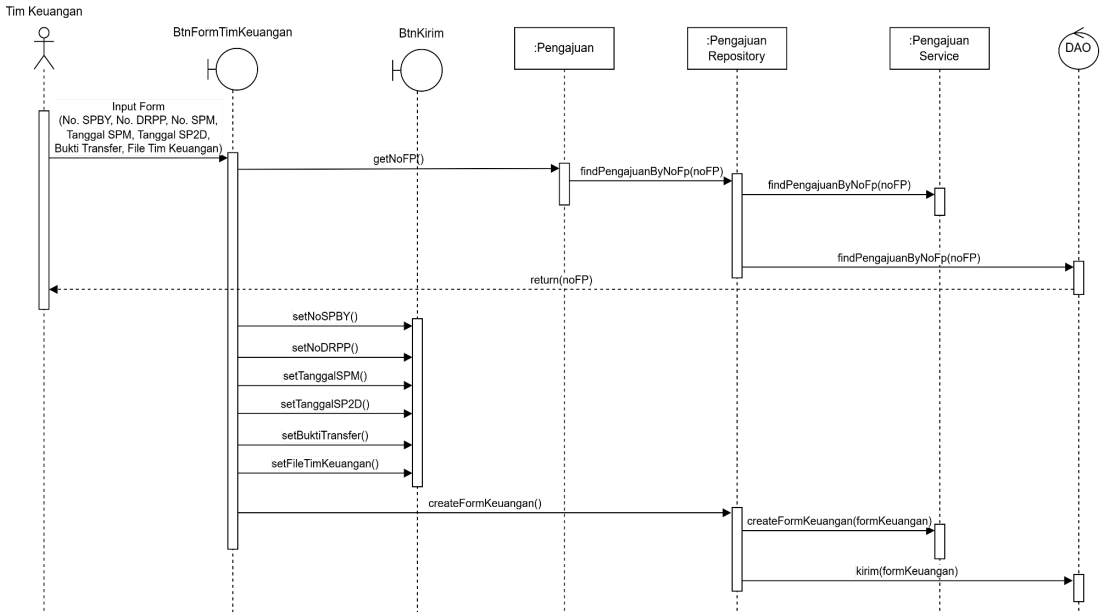


Diagram sequence di atas menggambarkan proses penolakan pengajuan oleh tim keuangan. Proses dimulai dengan tim keuangan memilih pengajuan melalui tombol monitoring pengajuan (BtnMonitoringPengajuan). Sistem memanggil metode get(noFP) untuk mengambil data pengajuan berdasarkan nomor form pengajuan (noFP). Data pengajuan diambil melalui metode findPengajuanByNoFp(noFP) yang diproses oleh PengajuanRepository dan diteruskan oleh PengajuanService untuk mengakses data di DAO. Setelah data berhasil diambil, tim keuangan memeriksa dokumen pengajuan. Jika dokumen tidak sesuai, status pengajuan diubah melalui metode setStatus() dan data pengajuan diperbarui serta disimpan kembali ke database menggunakan metode save(pengajuan). Selanjutnya, tim keuangan memasukkan catatan penolakan dengan menggunakan metode setNote() dan menyimpan catatan tersebut melalui metode save(pengajuan) yang memanggil createNote(note) untuk menambahkan catatan ke dalam sistem. Setelah proses selesai, data pengajuan dikembalikan ke tim keuangan untuk memastikan bahwa status dan catatan penolakan telah tersimpan.

- **Form Tim Keuangan**



Proses pengisian form oleh Tim Keuangan dimulai dengan membuka BtnFormTimKeuangan, di mana Tim Keuangan memasukkan data yang diperlukan, seperti nomor SPBY, nomor DRPP, nomor SPM, tanggal SPM, tanggal SP2D, bukti transfer, dan file pendukung. Data ini diatur melalui metode setNoSPBY(), setNoDRPP(), setTanggalSPM(), setTanggalSP2D(), setBuktiTransfer(), dan setFileTimKeuangan() pada entitas Form Tim Keuangan. Setelah semua data diisi, tombol BtnKirim diaktifkan untuk memproses data lebih lanjut. Tombol ini memanggil metode getNoFP() pada entitas Pengajuan untuk mendapatkan nomor faktur pengajuan (noFP). Entitas Pengajuan kemudian memanggil Pengajuan Repository menggunakan metode findPengajuanByNoFp(noFP) untuk mencari data pengajuan terkait. Repository ini melanjutkan proses ke Pengajuan Service untuk validasi dan pemrosesan data. Setelah data divalidasi, repository mengembalikan nomor faktur ke entitas Pengajuan. Selanjutnya, metode createFormKeuangan(formKeuangan) dijalankan untuk membuat form berdasarkan data yang telah diisi. Data yang lengkap dan valid kemudian dikirimkan melalui metode kirim(formKeuangan) untuk diproses lebih lanjut, sehingga proses pengisian form oleh Tim Keuangan dapat diselesaikan.

5. Download Data Pengajuan

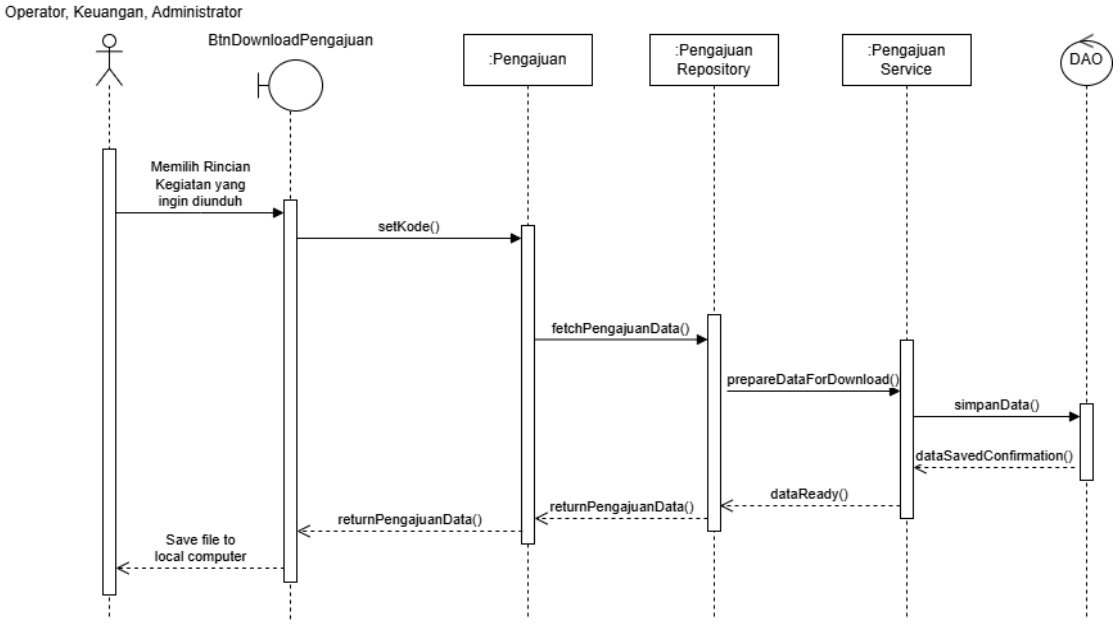
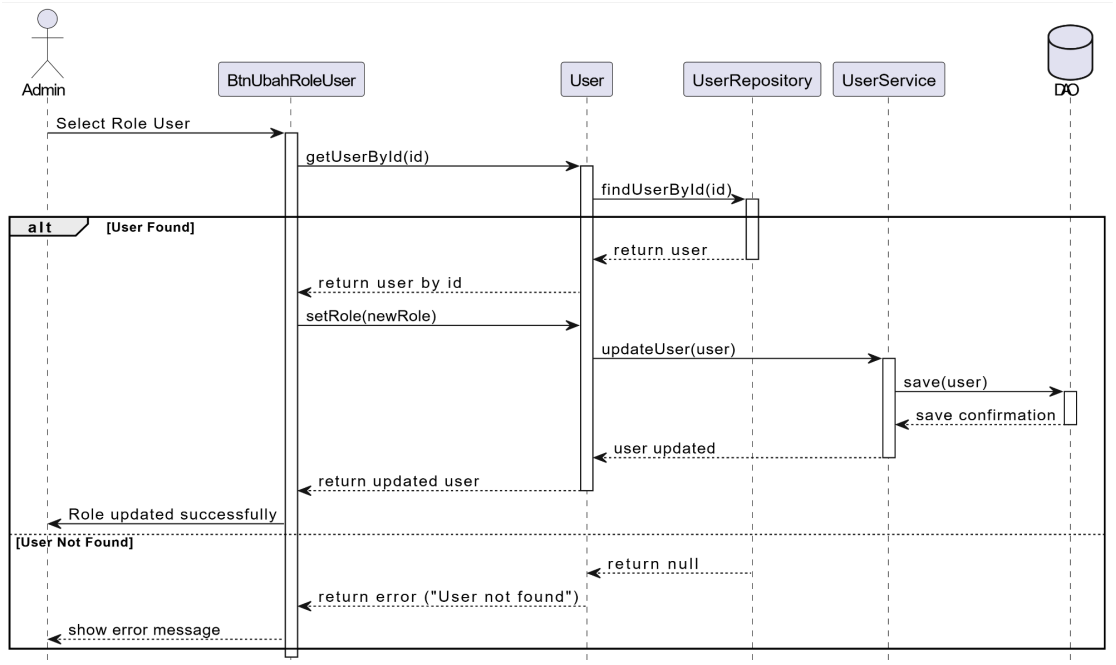


Diagram ini menggambarkan proses pengguna (Operator, Keuangan, atau Administrator) untuk mengunduh data pengajuan kegiatan. Proses dimulai ketika pengguna memilih rincian kegiatan yang ingin diunduh, lalu menekan tombol BtnDownloadPengajuan. Tombol ini berkomunikasi dengan objek :Pengajuan untuk menetapkan kode pengajuan melalui metode setKode(). Selanjutnya, :Pengajuan memanggil fetchPengajuanData() ke :Pengajuan Repository untuk mengambil data yang diperlukan. Data yang diambil dikembalikan ke :Pengajuan melalui metode returnPengajuanData(). Setelah itu, :Pengajuan Service mempersiapkan data untuk diunduh menggunakan prepareDataForDownload(). Data kemudian disimpan melalui metode simpanData() menggunakan DAO.

Setelah data siap, sistem memberikan konfirmasi dataSavedConfirmation() kepada pengguna. Akhirnya, data yang telah diproses diunduh dan disimpan ke komputer lokal pengguna. Proses ini memastikan bahwa semua pengguna dengan peran relevan dapat mengakses data pengajuan yang valid. Proses ini juga mencakup pengelolaan kesalahan, seperti memastikan rincian pengajuan (termasuk tanggal mulai dan akhir) diisi dengan benar. Jika ada kesalahan, sistem akan memberikan pesan peringatan kepada pengguna, sehingga proses tetap aman dan terstruktur.

6. Mengatur Role User



Dalam Diagram sequence "Manage Role User" ini, Tim Admin yang merupakan Pegawai PPK (Pejabat Pengelola Keuangan) BPS Provinsi DKI Jakarta, memiliki tanggung jawab utama untuk mengelola akun pengguna. Mereka memiliki wewenang untuk mengubah role pengguna berdasarkan peran mereka dalam proses bisnis serta menambahkan pengguna baru saat pegawai baru bergabung.

Alur Utama (Normal Flow)

Proses dimulai ketika Admin memilih pengguna melalui fitur BtnUbahRoleUser untuk mengubah role-nya. Sistem memanggil metode getUserById(id) pada entitas User untuk mencari data pengguna berdasarkan ID. Data ini kemudian dicari di database melalui UserRepository menggunakan metode findUserById(id). Jika data ditemukan, pengguna dikembalikan ke sistem. Admin kemudian menetapkan role baru melalui metode setRole(newRole) pada objek User.

Perubahan ini diteruskan ke UserService melalui metode updateUser(user). Layanan ini bertugas memproses pembaruan data pengguna, termasuk menyimpannya ke basis data melalui DAO (save(user)). Setelah data berhasil disimpan, sistem memberikan konfirmasi bahwa pembaruan role berhasil dilakukan, yang dikembalikan kepada Admin. Hal ini memungkinkan

Admin untuk memastikan bahwa perubahan sudah diterapkan sesuai kebutuhan.

Alur Alternatif (Kesalahan: User Not Found)

Jika pengguna yang dicari tidak ditemukan di database, UserRepository akan mengembalikan nilai null atau pesan error. Hal ini terjadi misalnya, jika ID pengguna tidak valid atau pengguna belum terdaftar dalam sistem. Pesan error ini diteruskan ke tombol BtnUbahRoleUser, yang kemudian menampilkan notifikasi "User not found" kepada Admin. Dengan demikian, Admin diberi informasi untuk memeriksa kembali ID pengguna atau melakukan langkah lain, seperti menambahkan pengguna baru.

7. Mengelola MAK (Mata Anggaran Keuangan)

a. Tambah Akun Belanja

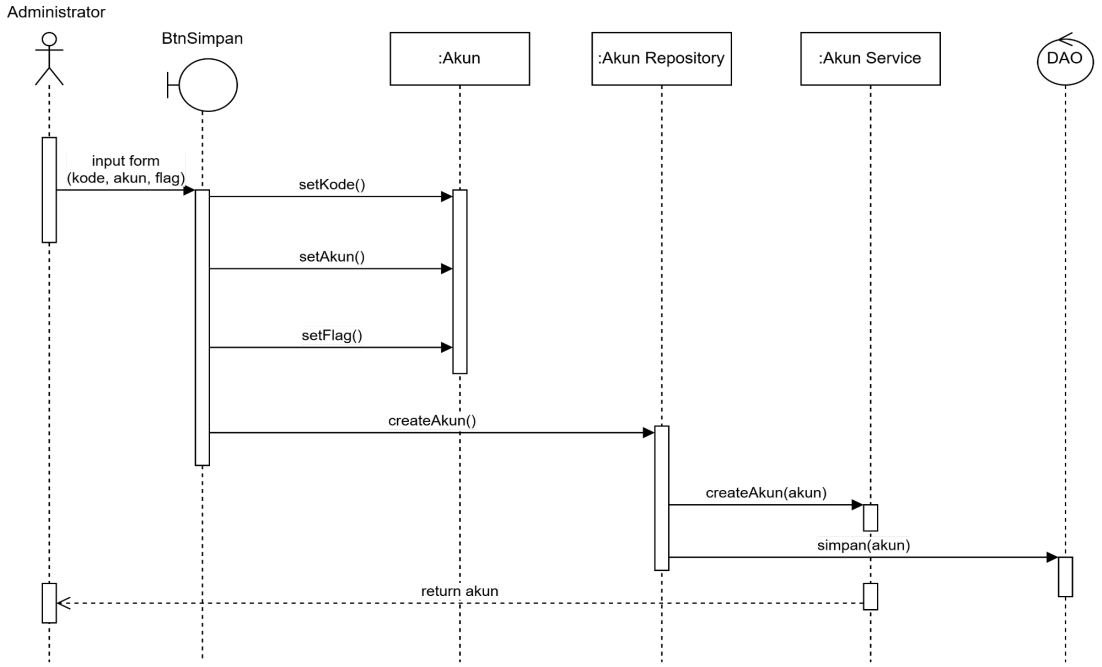


Diagram sequence ini menggambarkan proses penambahan akun belanja oleh administrator menggunakan tombol simpan (BtnSimpan). Awalnya, administrator mengisi formulir input dengan parameter berupa kode, akun, dan MAK. Formulir ini kemudian memanggil metode setKode(), setAkun(), dan setMAK() pada objek :Akun untuk menetapkan nilai-nilai tersebut. Setelah nilai-nilai atribut akun diatur, metode createAkun() dipanggil untuk membuat objek akun baru. Objek :Akun Repository kemudian

menangani permintaan createAkun(akun) yang selanjutnya memanggil simpan(akun) pada :Akun Service untuk menyimpan akun ke dalam database melalui lapisan DAO. Setelah akun berhasil disimpan, objek akun yang telah dibuat dikembalikan ke administrator sebagai hasil akhir.

b. Tambah Komponen

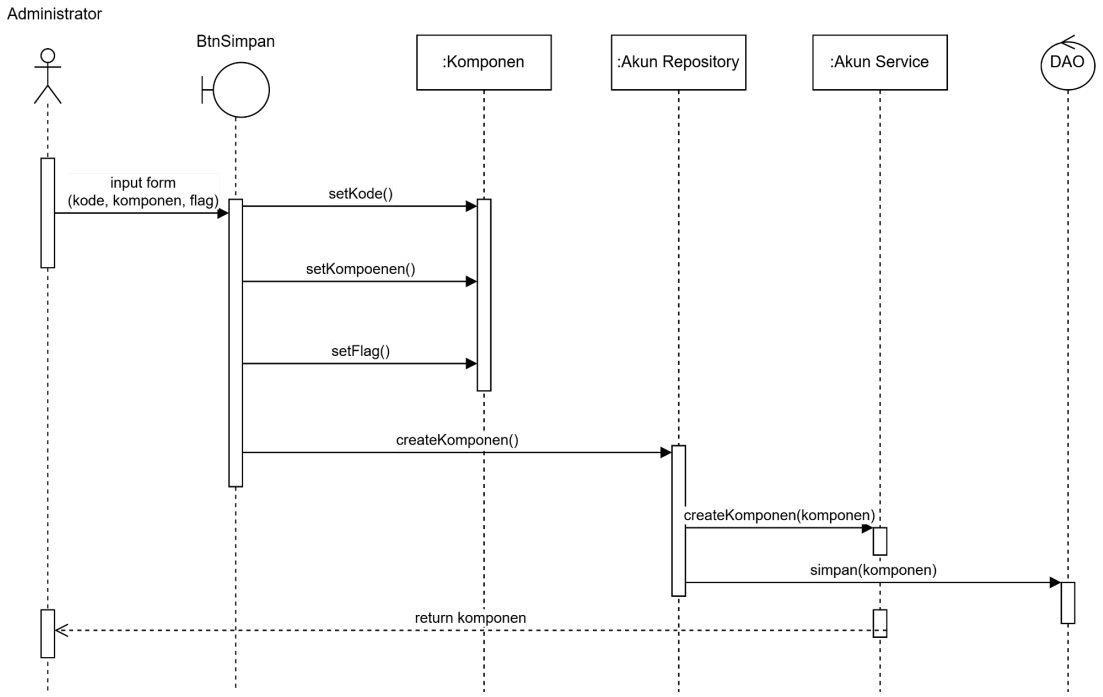


Diagram sequence ini menggambarkan proses penambahan komponen oleh administrator melalui tombol simpan (BtnSimpan). Administrator mengisi formulir input dengan parameter berupa kode, komponen, dan MAK. Data dari formulir ini diteruskan untuk memanggil metode setKode(), setKomponen(), dan setMAK() pada objek :Komponen, yang bertugas menetapkan nilai-nilai tersebut pada objek komponen. Setelah itu, metode createKomponen() dipanggil untuk membuat objek komponen baru. Objek :Akun Repository menerima permintaan createKomponen(komponen) yang kemudian memanggil simpan(komponen) pada :Akun Service. Proses ini memastikan data komponen disimpan dalam database melalui lapisan DAO. Setelah berhasil, objek komponen yang telah dibuat dikembalikan ke administrator sebagai hasil akhir.

c. Tambah SubKomponen

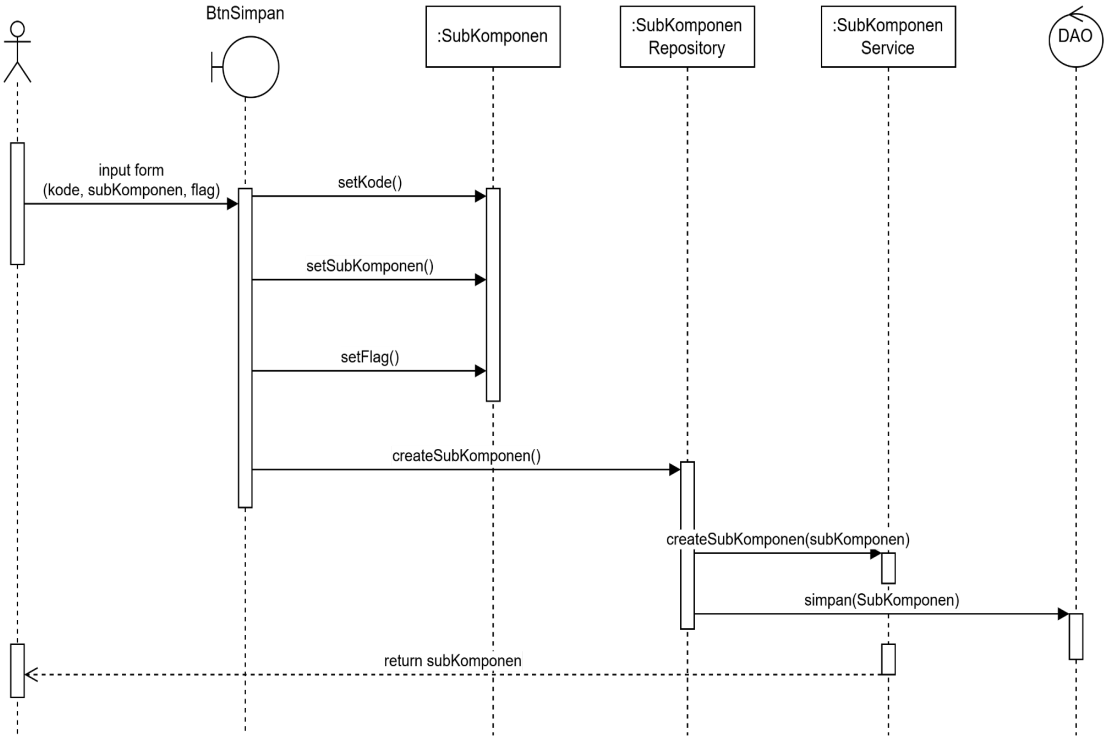


Diagram sequence ini menggambarkan proses penambahan subkomponen oleh administrator menggunakan tombol simpan (BtnSimpan). Administrator mengisi formulir input dengan parameter berupa kode, subkomponen, dan MAK. Data dari formulir ini diteruskan untuk memanggil metode setKode(), setSubKomponen(), dan setMAK() pada objek :SubKomponen, yang bertugas menetapkan nilai-nilai tersebut pada objek subkomponen. Setelah itu, metode createSubKomponen() dipanggil untuk membuat objek subkomponen baru. Objek :SubKomponen Repository menerima permintaan createSubKomponen(subKomponen) yang kemudian memanggil simpan(SubKomponen) pada :SubKomponen Service. Proses ini menyimpan data subkomponen ke dalam database melalui lapisan DAO. Setelah berhasil, objek subkomponen yang telah dibuat dikembalikan ke administrator sebagai hasil akhir.

d. Tambah Output

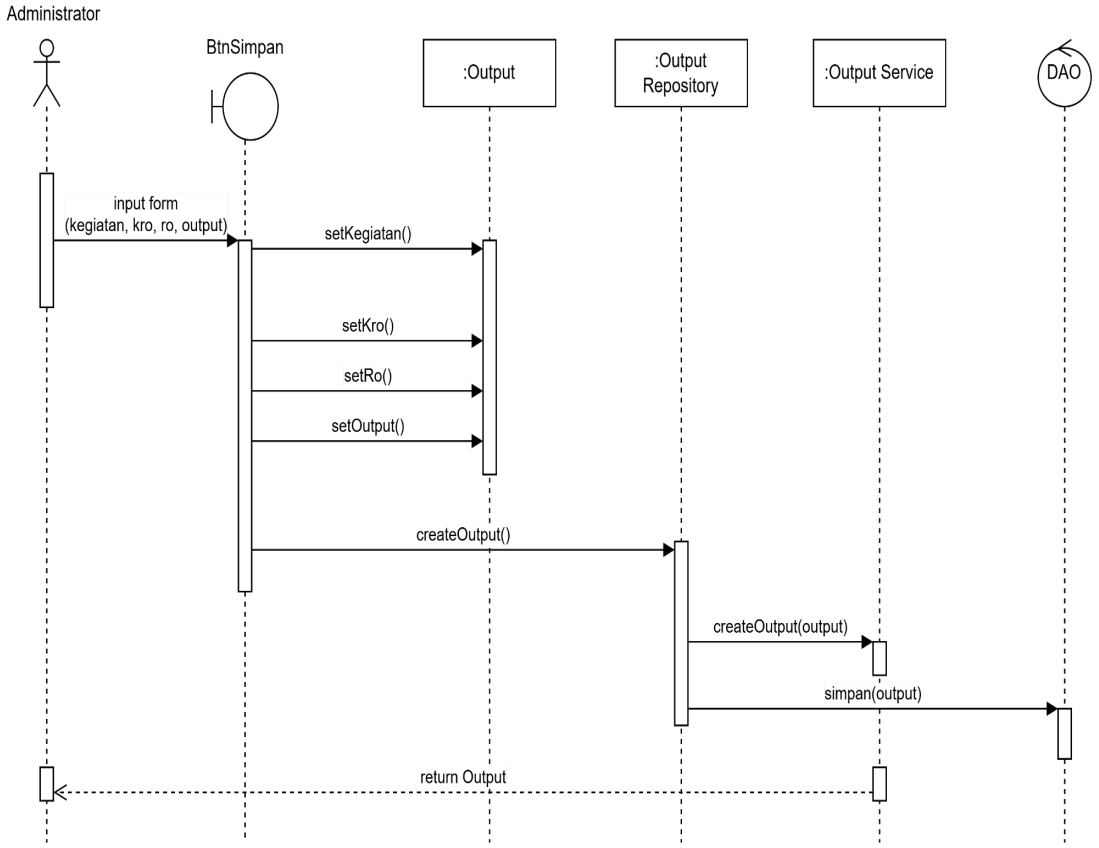


Diagram sequence ini menunjukkan alur proses untuk menambah data output dalam sistem. Administrator memulai proses dengan mengisi formulir input yang berisi data kegiatan, KRO (Komponen Rencana Operasional), RO (Rencana Operasional), dan output, lalu menekan tombol BtnSimpan. Tombol ini memanggil metode seperti setKegiatan(), setKro(), setRo(), dan setOutput() pada objek Output untuk menetapkan nilai-nilai yang dimasukkan. Setelah itu, metode createOutput() dipanggil untuk membuat entitas output baru. Proses dilanjutkan ke Output Repository, di mana data output yang dibuat dikirimkan melalui metode createOutput(output), kemudian diproses oleh Output Service menggunakan metode simpan(output). Akhirnya, data disimpan ke dalam sistem basis data (DAO), dan output yang telah berhasil disimpan dikembalikan sebagai hasil proses.

e. Manage Akun

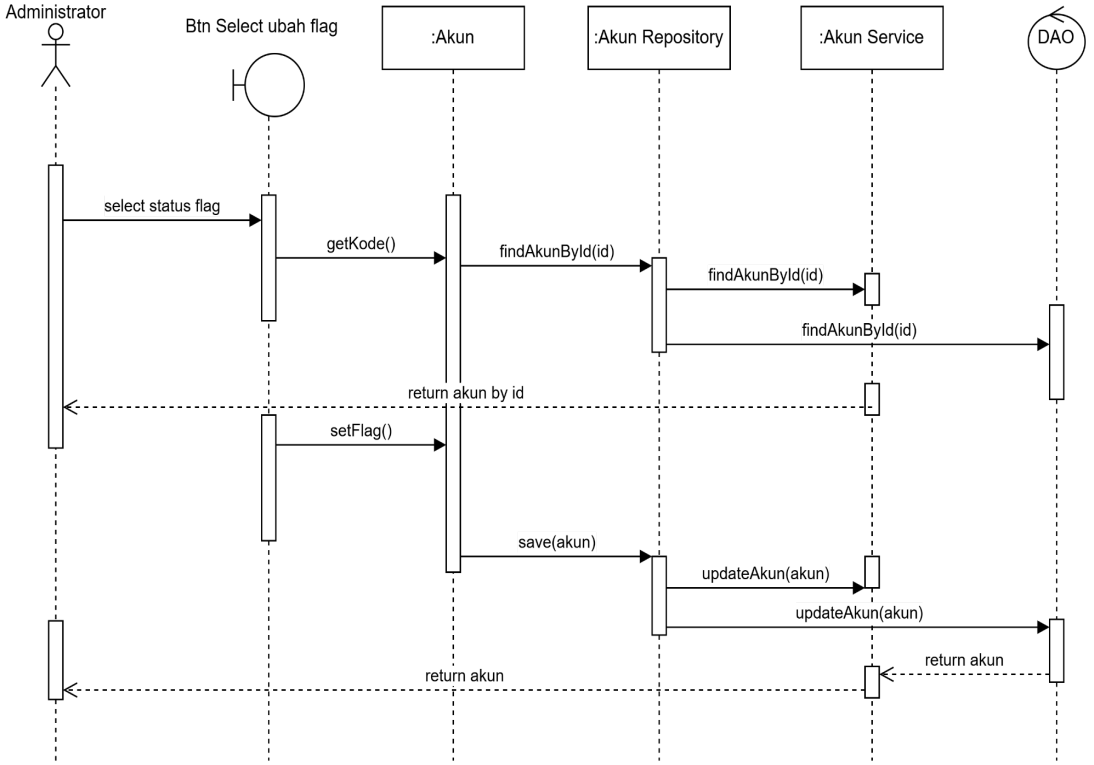


Diagram sequence ini menjelaskan proses pengaturan status MAK pada akun belanja. Administrator memulai dengan memilih status MAK melalui tombol Btn Select ubah MAK, yang memanggil metode getKode() pada objek Akun untuk mendapatkan kode atau identifikasi akun yang dipilih. Sistem kemudian mencari akun berdasarkan ID dengan memanggil metode findAkunById(id) pada Akun Repository, yang diteruskan ke Akun Service hingga mencapai DAO untuk mengambil data akun terkait. Setelah akun ditemukan, data tersebut dikembalikan ke sistem. Administrator kemudian memperbarui status MAK dengan memanggil metode setMAK() pada objek akun. Proses dilanjutkan dengan menyimpan perubahan menggunakan metode save(akun), yang mengarahkan data ke Akun Repository dan Akun Service melalui metode updateAkun(akun). Akhirnya, data akun yang diperbarui disimpan di basis data, dan hasil akhir dikembalikan ke Administrator.

f. Manage Komponen

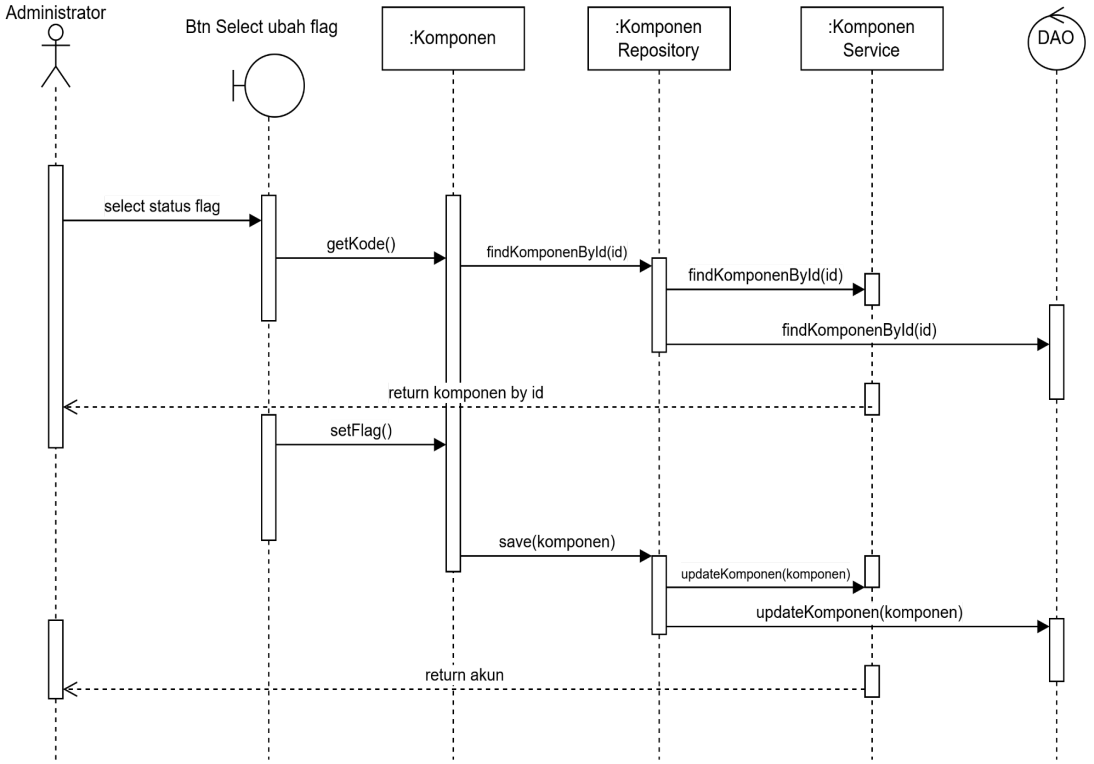


Diagram sequence ini menjelaskan proses pengaturan status MAK pada komponen. Administrator memulai dengan memilih status MAK melalui tombol Btn Select ubah MAK, yang memanggil metode getKode() pada objek Komponen untuk mendapatkan kode atau identifikasi komponen yang dipilih. Sistem kemudian mencari komponen berdasarkan ID dengan memanggil metode findKomponenById(id) pada Komponen Repository, yang diteruskan ke Komponen Service hingga mencapai DAO untuk mengambil data komponen terkait. Setelah komponen ditemukan, data tersebut dikembalikan ke sistem. Administrator kemudian memperbarui status MAK dengan memanggil metode setMAK() pada objek komponen. Proses dilanjutkan dengan menyimpan perubahan menggunakan metode save(komponen), yang mengarahkan data ke Komponen Repository dan Komponen Service melalui metode updateKomponen(komponen). Akhirnya, data komponen yang diperbarui disimpan di basis data, dan hasil akhir dikembalikan ke Administrator.

g. Manage SubKomponen

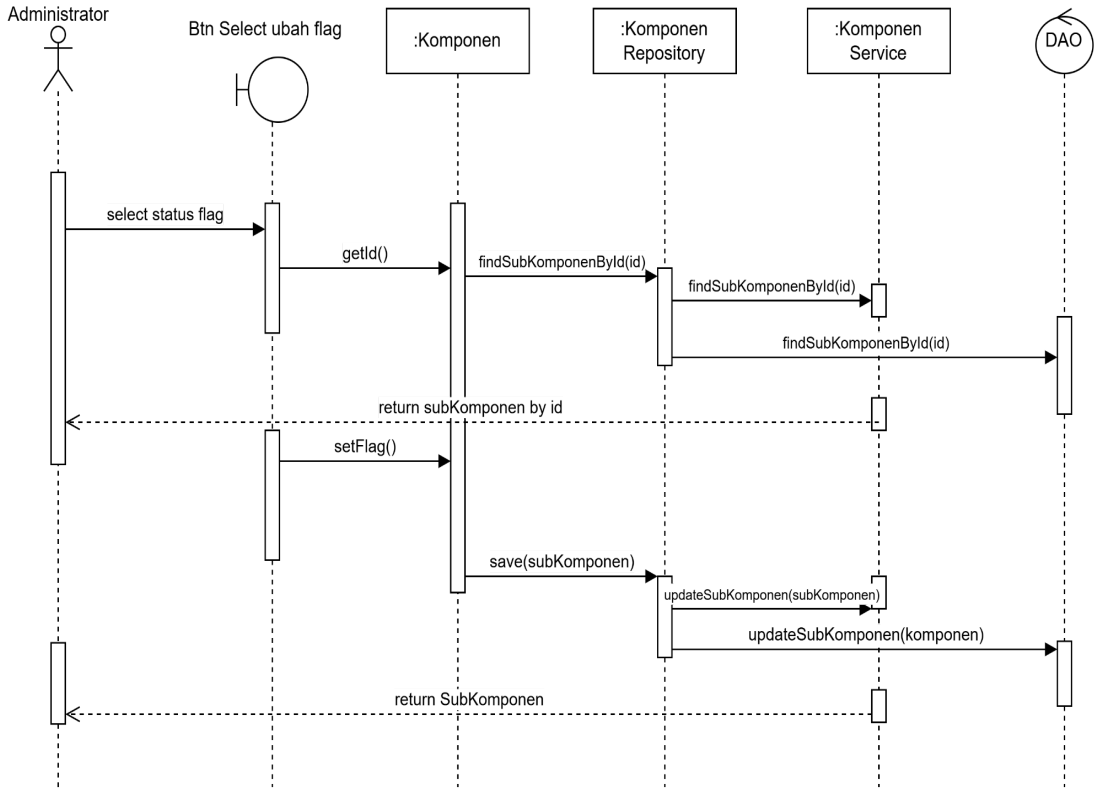


Diagram sequence ini menjelaskan proses pengaturan status MAK pada subkomponen. Administrator memulai dengan memilih status MAK melalui tombol Btn Select ubah MAK, yang memanggil metode getKode() pada objek SubKomponen untuk mendapatkan kode atau identifikasi subkomponen yang dipilih. Sistem kemudian mencari subkomponen berdasarkan ID dengan memanggil metode findSubKomponenById(id) pada SubKomponen Repository, yang diteruskan ke SubKomponen Service hingga mencapai DAO untuk mengambil data komponen terkait. Setelah subkomponen ditemukan, data tersebut dikembalikan ke sistem. Administrator kemudian memperbarui status MAK dengan memanggil metode setMAK() pada objek subkomponen. Proses dilanjutkan dengan menyimpan perubahan menggunakan metode save(subkomponen), yang mengarahkan data ke SubKomponen Repository dan SubKomponen Service melalui metode updateSubKomponen(subkomponen). Akhirnya, data subkomponen yang diperbarui disimpan di basis data, dan hasil akhir dikembalikan ke Administrator.

h. Manage Output

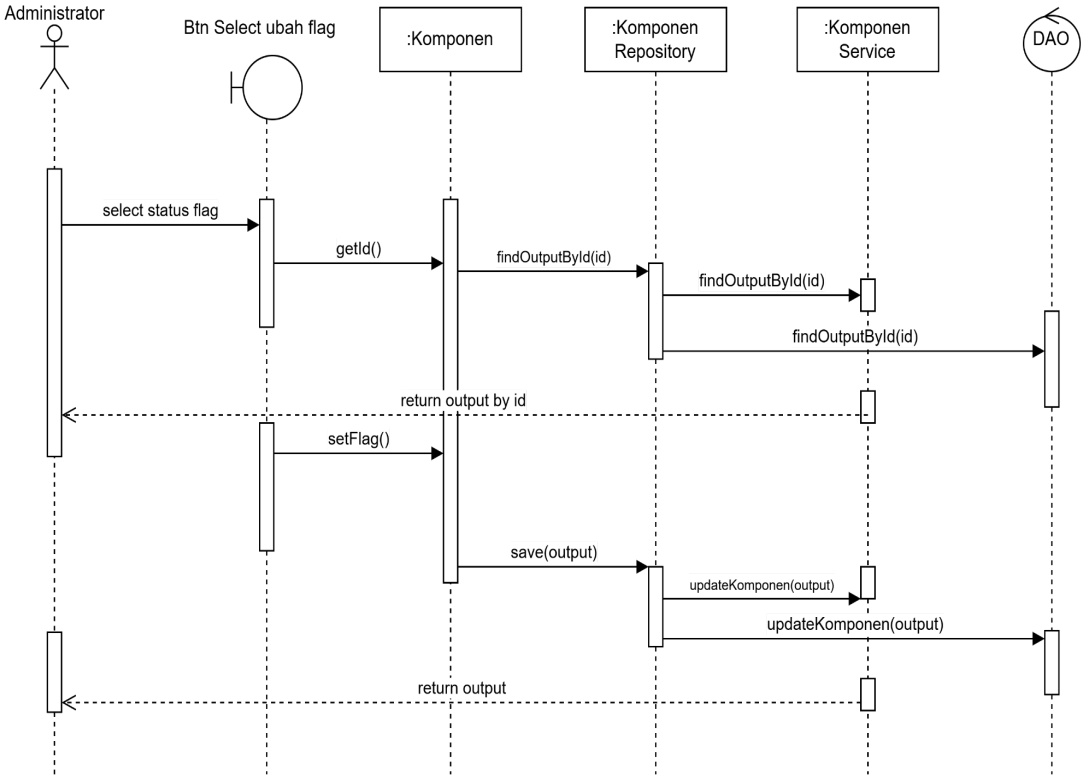


Diagram sequence ini menjelaskan proses pengaturan status MAK pada output. Administrator memulai dengan memilih status MAK melalui tombol Btn Select ubah MAK, yang memanggil metode getKode() pada objek Output untuk mendapatkan kode atau identifikasi output yang dipilih. Sistem kemudian mencari output berdasarkan ID dengan memanggil metode findOutputById(id) pada Ouput Repository, yang diteruskan ke Output Service hingga mencapai DAO untuk mengambil data output terkait. Setelah oupput ditemukan, data tersebut dikembalikan ke sistem. Administrator kemudian memperbarui status MAK dengan memanggil metode setMAK() pada objek output. Proses dilanjutkan dengan menyimpan perubahan menggunakan metode save(ouput), yang mengarahkan data ke Output Repository dan Ouput Service melalui metode updateOutput(output). Akhirnya, data output yang diperbarui disimpan di basis data, dan hasil akhir dikembalikan ke Administrator.

4.3 Identifikasi Resiko Sistem, Sisi Keamanan, Performa, dan Keandalan

Kebutuhan keamanan Sistem BDA Provinsi DKI Jakarta:

- Privasi data

  Data yang diunggah oleh pengguna/operator (pegawai BPS DKI Jakarta) hanya dapat diakses oleh pihak yang berwenang sesuai dengan peran masing-masing (role-based access control). Informasi sensitif, seperti rincian pengajuan dan dokumen pendukung, dijamin kerahasiaannya dari akses luar.
- Integritas data

  Informasi tidak dapat diubah tanpa otorisasi sesuai dengan hak akses pengguna. Sistem mencatat log semua perubahan data untuk memudahkan audit.
- Access Control
  - Pengguna (Operator): Mengisi dan mengelola pengajuan serta dokumen miliknya.
  - Tim Keuangan: Mengelola semua pengajuan dan dokumen yang diajukan operator, tetapi bisa juga berperan menjadi operator.
  - Admin: Memiliki akses penuh ke seluruh data dan pengelolaan sistem.
- Authorization
  - Create Authorization: Operator, Tim Keuangan, dan Admin dapat menambahkan data pengajuan.
  - Read Authorization: Pada monitoring file keuangan, operator hanya bisa melihat saja apa perkembangan ataupun perubahan dari tim keuangan dan admin.
  - Update Authorization: Operator, tim keuangan, dan admin dapat memperbarui dokumen atau data sesuai perannya.
  - Delete Authorization: Hanya Admin dapat menghapus data yang tidak relevan.
- Authentication
  - Sistem memvalidasi pengguna melalui mekanisme login menggunakan email BPS dan password, juga bisa menggunakan SSO.
  - Sistem mendukung multi-factor authentication untuk meningkatkan keamanan.

Risiko Bahaya dan Cara Meminimalisir

Risiko Kemananan	Cara Meminimalisir
Serangan SQL Injection	Dengan menggunakan parameterized queries untuk memastikan input data diproses dengan aman dan tidak disalahgunakan sebagai perintah SQL dan dengan melakukan validasi input dengan membatasi karakter tertentu yang tidak diharapkan.
Kehilangan Data	Dengan mengimplementasikan backup otomatis secara berkala dan mengenkripsi data sebelum disimpan untuk memastikan kerahasiaan data meskipun terjadi pelanggaran keamanan.
Eksposure data	Dengan menggunakan enkripsi seperti AES-256 untuk melindungi data ketika transit dan ketika disimpan dan memastikan kontrol akses diterapkan dengan ketat sehingga hanya pengguna dengan izin tertentu yang memiliki akses terhadap data tertentu.
Serangan DDoS	Dengan menggunakan mekanisme load balancing untuk mendistribusikan lalu lintas jaringan dan mengurangi dampak lonjakan beban, selain itu juga dengan menerapkan pemantauan jaringan secara aktif yang memitigasi serangan secara real-time
Kesalahan Konfigurasi	Dengan melakukan audit konfigurasi secara terjadwal dan menonaktifkan layanan dan fitur yang tidak diperlukan untuk mengurangi potensi ekspoitasi

Identifikasi kebutuhan performa dalam kondisi normal dan kondisi puncak dan cara mendapatkan performa terbaik.

1) Kebutuhan Performa:

Performa	Keterangan	Penerapan
Speed (kecepatan)	Waktu yang dibutuhkan sistem untuk menjalankan fungsinya.	- Waktu respon untuk setiap transaksi (pengajuan dan upload

		dokumen BDA) harus kurang dari 7 detik untuk setiap kondisi jaringan
Capacity (Kapasitas)	Total dan jumlah puncak pengguna dan volume data yang diharapkan.	- Pada kondisi normal mencapai 2 - 15 orang.
Availability and Reliability (ketersediaan dan keandalan)	Sejauh mana sistem tersedia untuk pengguna dan seberapa besar tingkat kegagalan yang diizinkan karena kesalahan.	- Sistem akan menyimpan pengguna dengan jumlah 85 pengguna (user) - Waktu yang digunakan untuk maintenance (pemeliharaan) tidak melebihi 3 jam selama seminggu - Sistem harus memiliki kinerja uptime 99%.

2) Kebutuhan Performa dalam kondisi puncak

   Kondisi puncak diperkirakan terjadi pada saat akhir tahun, sebelum batas pembuatan laporan keuangan akhir tahun oleh tim keuangan BPS Provinsi DKI Jakarta.

3) Cara mendapatkan performa terbaik:
   - Menggunakan server dengan kapasitas sesuai kebutuhan dan melakukan penambahan jika diperlukan.
   - Pembuatan tampilan pengguna atau *interface* yang responsif dan intuitif sehingga dapat menyesuaikan kebutuhan pengguna, memberikan kemudahan navigasi dan pemahaman, serta mendukung kustomisasi dan personalisasi.
   - Membatasi ukuran file untuk menjaga ketersediaan *storage database*.
   - Memperbesar ukuran bandwith server

Metrik Keandalan

No	Failure Class	Contoh	Avoidance	Detection	Tolerance	Metric (Requirement)
1.	Transient non-corrupting	Gangguan sementara pada koneksi jaringan saat mengunggah dokumen.	Implementasi metode buffering untuk dokumen yang sedang diunggah agar bisa dilanjutkan setelah jaringan pulih.	Pemantauan status jaringan dan notifikasi otomatis saat terjadi gangguan koneksi saat proses unggah berlangsung.	Sistem secara otomatis melanjutkan unggahan setelah jaringan pulih tanpa memulai ulang proses	1 per 1.000 transaksi
2.	Transient non-corrupting	Sistem lambat saat memvalidasi data input pengguna.	Optimasi logika validasi dan pengurangan proses yang redundan untuk mempercepat eksekusi sistem.	Pemantauan kinerja aplikasi menggunakan alat seperti New Relic untuk mendeteksi latensi tinggi pada waktu respons server.	Menampilkan notifikasi kepada pengguna bahwa proses sedang berjalan dan akan selesai dalam beberapa saat.	1 per 1.000 transaksi
3.	Transient non-corrupting	Gagal menyimpan dokumen bukti administrasi pada server sementara sistem sedang sibuk.	Implementasi retry mechanism untuk menyimpan data jika terjadi kegagalan pada percobaan awal.	Pemantauan log aplikasi untuk mendeteksi kegagalan saat proses penyimpanan dokumen.	Sistem menyimpan status unggahan sementara untuk menghindari kehilangan data, dan mencoba kembali setelah server lebih stabil.	1 per 1.000 transaksi
4.	Permanent non-corrupting	Kegagalan perangkat keras seperti server yang menyimpan dokumen.	Menjadwalkan pemeliharaan perangkat keras secara rutin dan memastikan semua perangkat keras	Monitor kesehatan perangkat keras menggunakan alat seperti SMART monitoring untuk mendeteksi	Implementasi solusi cadangan seperti server failover dan replikasi data untuk memastikan	1 per 1.000 hari

			berada di lingkungan yang sesuai.	kegagalan lebih awal.	ketersediaan layanan.	
5.	Permanent non-corrupting	Beberapa fitur sistem tidak berfungsi setelah pembaruan perangkat lunak administrasi.	Melakukan uji coba dan simulasi pembaruan perangkat lunak pada lingkungan staging sebelum diimplementasikan ke lingkungan produksi.	Pemantauan log dan sistem secara real-time untuk mendeteksi fitur yang gagal bekerja setelah pembaruan.	Menyiapkan mekanisme rollback untuk kembali ke versi perangkat lunak sebelumnya jika pembaruan menyebabkan kerusakan signifikan.	1 per 1.000 hari
6.	Corrupting	Kegagalan pencadangan data bukti administrasi keuangan.	Menggunakan metode pencadangan yang teruji seperti pencadangan inkremental dan redundansi lintas lokasi.	Pemantauan otomatis terhadap proses pencadangan untuk mendeteksi kesalahan, seperti file yang tidak lengkap atau rusak.	Menyediakan pencadangan multi-situs atau replikasi waktu nyata untuk memastikan data selalu tersedia meskipun terjadi kegagalan pada satu lokasi penyimpanan.	Never
7.	Corrupting	File dokume rusak selam proses unggahan akibat gangguan sistem.	Menggunakan algoritma checksum untuk mendeteksi korupsi data sejak awal, dan melakukan pengecekan di setiap tahapan unggahan.	Verifikasi integritas file yang diunggah dengan checksum atau hash MD5 untuk memastikan file tidak rusak selama unggahan.	Mengimplementasi kan unggahan ulang otomatis untuk file yang terdeteksi rusak tanpa meminta pengguna untuk memulai ulang secara manual.	Never

BAB V
IMPLEMENTASI SISTEM

5.1 Hasil Implementasi Sistem

Hasil implementasi sistem berupa screenshot antarmuka beserta penjelasannya silahkan dilihat di buku panduan yang ada di dalam link gdocs di bawah ini:

- **https://s.stis.ac.id/BukuPanduan_SistemBDA**

Dan berikut link aplikasi dari Sistem BDA yang telah di-*deploy*:

- **https://bda.web.bps.go.id**

5.2 Akun untuk Masuk ke Sistem

Akun-akun dibawah ini adalah akun perwakilan yang sementara dipakai sebelum login menggunakan SSO:

1. Akun sebagai *role operator*:
   email: heryanah@bps.go.id
   password: password123

2. Akun sebagai *role tim keuangan*:
   email: erikha.anindita@bps.go.id
   password: password123

3. Akun sebagai *role admin*:
   email: anggitn@bps.go.id
   password: password123

BAB VI

PENGUJIAN SISTEM

6.1 Penjelasan terkait Pengujian Sistem

Kelompok kami melakukan pengujian sistem terhadap Ssitem BDA yang sudah dirancang dengan metode Teknik *Black Box* dan metode *System Usability Scale* (SUS). Pengujian Black Box merupakan metode pengujian yang berfungsi untuk memastikan bahwa semua fungsi perangkat lunak berjalan sesuai dengan kebutuhan fungsional yang telah ditentukan, tanpa memperhatikan struktur internal atau kode programnya. Sementara metode *System Usability Scale* (SUS) berfungsi untuk mengevaluasi tingkat kenyamanan dan kemudahan penggunaan aplikasi oleh pengguna.

Dalam pengujian *Black Box*, kami akan memasukkan data secara acak untuk memastikan apakah sistem pada aplikasi BDA dapat menghasilkan output yang sesuai dengan desain pengujian yang ditetapkan. Ini bertujuan untuk memastikan bahwa sistem berjalan sesuai dengan tujuan awal pembuatannya dan mampu memberikan *output* atau respons yang tepat, baik ketika menerima data yang valid maupun tidak valid. Sedangkan metode SUS menggunakan kuesioner dengan sepuluh pertanyaan yang mengukur berbagai aspek kegunaan sistem, seperti kesederhanaan antarmuka, efektivitas, dan efisiensi aplikasi dalam memenuhi kebutuhan pengguna. Hasil pengukuran SUS berupa skor yang diinterpretasikan untuk menentukan tingkat kepuasan pengguna, yang selanjutnya menjadi acuan dalam perbaikan atau pengembangan lebih lanjut dari sistem aplikasi BDA.

Berikut adalah tabel hasil pengujian *Black Box* yang telah dilakukan:

	Metode Pengujian	Hasil Pengujian
Fungsional Testing	Memastikan semua fitur berfungsi dengan baik	Semua fitur utama berfungsi dengan baik, termasuk pengajuan dokumen dan monitoring status
Usability Testing	Menguji kemudahan penggunaan antarmuka sistem	Pengguna merasa antarmuka sistem intuitif dan mudah digunakan
Performance Testing	Mengukur kecepatan dan respons sistem di bawah beban pengguna yang tinggi	Sistem mampu menangani hingga 50 pengguna secara bersamaan dengan waktu respons kurang dari 3 detik
Security Testing	Memastikan data pengguna aman dan tidak dapat diakses oleh pihak yang tidak berwenang	Tidak ada celah keamanan yang ditemukan, dan data pengguna terlindungi dengan baik

6.1.1 Scenario Testing Sistem

No.	Action	Input	Hasil yang diharapkan	Hasil sebenarnya	Test browser	Hasil test browser	Komentar
Login							
1.	Memasukkan email dan password yang benar kemudian mengklik tombol login	Email dan password user yang terdaftar	Login berhasil, pengguna/user masuk ke halaman dashboard	Login berhasil	Ms edge, Google chrome, Mozilla Firefox	Pass (berhasil)	
2.	Memasukkan email dan password yang salah kemudian mengklik tombol login	Email dan password user yang belum terdaftar	Login tidak berhasil dan muncul pesan error "password/username salah"	Login tidak berhasil	Ms edge, Google chrome, Mozilla Firefox	Pass	
Form Pengajuan							
1.	Mengisi detail form pengajuan sesuai validasi dan lengkap, lalu klik tombol kirim	Detail form yang ingin diajukan sudah lengkap	Berhasil terkirim, user masuk ke halaman monitoring operator dan status berubah menjadi entri operator	Berhasil terkirim	Ms edge, Google chrome, Mozilla Firefox	Pass	
2.	Mengisi detail form pengajuan tidak sesuai validasi dan tidak lengkap, lalu klik tombol kirim	Detail form yang ingin diajukan tidak lengkap	Tidak berhasil terkirim, user tetap di halaman form pengajuan dan mendapatkan warning error	Gagal terkirim	Ms edge, Google chrome, Mozilla Firefox	Pass	
Monitoring Operator							
1.	Mengklik tombol view pada monitoring operator	-	Detail form pengajuan ditampilkan	Berhasil ditampilkan	Ms edge, Google chrome, Mozilla Firefox	Pass	
2.	Mengklik tombol edit pada monitoring operator	Detail form yang ingin diubah	Detail form pengajuan berhasil diubah	Berhasil diubah	Ms edge, Google chrome, Mozilla Firefox	Pass	
3.	Mengklik tombol delete pada	Konfirmasi Ya atau tidak	Data form pengajuan	Berhasil dihapus	Ms edge, Google	Pass	

	monitoring operator	untuk menghapus form pengajuan	berhasil dihapus dari pengajuan		chrome, Mozilla Firefox		
Monitoring Keuangan							
1.	Mengklik tombol view pada monitoring keuangan	-	Detail form pengajuan ditampilkan	Berhasil ditampilkan	Ms edge, Google chrome, Mozilla Firefox	Pass	
2.	Jika pengajuan dari user sesuai dan melengkapi syarat, tim keuangan mengklik tombol approve pengajuan	-	Status di monitoring operator berubah menjadi proses pembayaran	Status pengajuan di monitoring berhasil berubah	Ms edge, Google chrome, Mozilla Firefox	Pass	
3.	Jika pengajuan dari user ada yang tidak sesuai dan belum melengkapi syarat, tim keuangan mengklik tombol reject pengajuan	-	Status di monitoring operator berubah menjadi entri operator.	Status pengajuan di monitoring berhasil berubah	Ms edge, Google chrome, Mozilla Firefox	Pass	
Upload File Operator							
1.	Mengupload file yang diperlukan untuk memenuhi syarat-syarat pengajuan secara lengkap	File yang sesuai	Berhasil mengirim file-file yang memenuhi syarat dan status di monitoring berubah menjadi Pengecekan Dokumen	File berhasil terkirim dan status pengajuan di monitoring berhasil berubah	Ms edge, Google chrome, Mozilla Firefox	Pass	
2.	Tidak lengkap dalam mengupload file yang diperlukan untuk memenuhi syarat-syarat pengajuan	File yang tidak sesuai	Tidak berhasil mengirim file dan kembali ke halaman upload operator	File tidak berhasil terkirim dan status pengajuan di monitoring masih tetap sama	Ms edge, Google chrome, Mozilla Firefox	Pass	
Upload File Tim Keuangan							
1.	Mengupload file yang diperlukan untuk menyetujui dan mendukung	File yang sesuai	Berhasil mengirim file-file yang memenuhi syarat dan status	File berhasil terkirim dan status pengajuan di	Ms edge, Google chrome, Mozilla	Pass	

	pengajuan operator seperti bukti pembayaran dll.		di monitoring berubah menjadi Pengecekan Dokumen	monitoring berhasil berubah	Firefox		
2.	Tidak lengkap dalam mengupload file yang diperlukan untuk mendukung pengajuan operator	File yang tidak sesuai	Tidak berhasil mengirim file dan kembali ke halaman upload file	File tidak berhasil terkirim dan status pengajuan di monitoring masih tetap sama	Ms edge, Google chrome, Mozilla Firefox	Pass	
Download Data Pengajuan							
1.	Memilih pengajuan yang ingin di download dan klik tombol download	-	Download file dalam bentuk csv atau excel berhasil	Download file berhasil	Ms edge, Google chrome, Mozilla Firefox	Pass	
2.	Memilih pengajuan yang ingin di download dan klik tombol download	-	Gagal mendownload file dalam bentuk file csv atau excel	Download file gagal	Ms edge, Google chrome, Mozilla Firefox	Pass	
Manage User							
1.	Admin mengubah role user dengan klik tombol dropdown role	-	Role user berubah sesuai dengan yang diganti admin	Role berhasil diubah	Ms edge, Google chrome, Mozilla Firefox	Pass	
2.	Admin menambah user/pengguna ke dalam sistem di form tambah user	Melengkapi detail form dengan benar	User berhasil ditambahkan dan masuk ke dalam database	Tambah user berhasil	Ms edge, Google chrome, Mozilla Firefox	Pass	
3.	Admin menambah user/pengguna ke dalam sistem di form tambah user tetapi tidak lengkap	Melengkapi detail form dengan tidak lengkap	User gagal ditambahkan dan tidak masuk ke dalam database	Tambah user gagal	Ms edge, Google chrome, Mozilla Firefox	Pass	
Manage MAK							
1.	Admin mengubah MAK MAK (Akun belanja, komponen, sub-komponen, dan output) dengan klik	-	Aksi setiap MAK berubah sesuai dengan yang diganti admin	Aksi MAK yang diubah berhasil	Ms edge, Google chrome, Mozilla Firefox	Pass	

	tombol dropdown aksi						
2.	Admin menambah MAK (Akun belanja, komponen, sub-komponen, dan output) ke dalam sistem di form tambah MAK (Akun belanja, komponen, sub-komponen, dan output)	Melengkapi detail form dengan benar sesuai yang diminta	MAK (Akun belanja, komponen, sub-komponen, dan output) berhasil ditambahkan dan masuk ke dalam database	Tambah MAK (Akun belanja, komponen, sub-kompone n, dan output) berhasil	Ms edge, Google chrome, Mozilla Firefox	Pass	
3.	Admin menambah MAK (Akun belanja, komponen, sub-komponen, dan output) ke dalam sistem di form tambah MAK (Akun belanja, komponen, sub-komponen, dan output) tetapi tidak lengkap	Melengkapi detail form dengan tidak lengkap	MAK (Akun belanja, komponen, sub-komponen, dan output) gagal ditambahkan dan tidak masuk ke dalam database	Tambah MAK (Akun belanja, komponen, sub-kompone n, dan output) gagal	Ms edge, Google chrome, Mozilla Firefox	Pass	

6.1.2 Instrumen atau Kuesioner yang Digunakan

Deliverable	Kesiapan
Embedded System	Tidak diperlukan untuk web
Persyaratan sistem	Sudah disiapkan (dokumentasi spesifikasi)
File source code	Disiapkan dalam repository GitHub dan GitLab STIS
Dokumentasi code	Disusun menggunakan tool Vscode dengan bahasa PHP di framework Laravel
Buku panduan admin	Sudah disiapkan
Buku panduan pengguna	Sudah disiapkan
Lisensi hak cipta	Lisensi open-source atau interval ditentukan
Skrip instalasi	Tidak diperlukan (deploy langsung via web hosting)
Skrip, data, dan laporan pengujian	Sudah disiapkan

6.1.3 Analisis dan Hasil Kuesioner Testing

Tautan kuesioner testing untuk mengetahui hasil dari SUS dan EUCS dapat diakses melalui : **https://s.stis.ac.id/KuesionerSUS_SistemBDA**

1. **Analisis Kuesioner System Usability Scale (SUS)**

System Usability Scale (SUS) adalah metode evaluasi kegunaan yang menggunakan kuesioner untuk mengukur tingkat kepuasan pengguna terhadap suatu sistem, produk, atau layanan. Kuesioner ini terdiri dari 10 pernyataan yang mencakup berbagai dimensi pengalaman pengguna, termasuk kemudahan penggunaan, efisiensi, dan kenyamanan.

No	Kode	Pertanyaan
1.	Q1	Saya berpikir akan menggunakan sistem ini lagi
2.	Q2	Saya merasa sistem ini rumit untuk digunakan
3.	Q3	Saya merasa sistem ini mudah digunakan
4.	Q4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini
5.	Q5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya
6.	Q6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi pada sistem ini)
7.	Q7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat.
8.	Q8	Saya merasa sistem ini membingungkan
9.	Q9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini
10.	Q10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini

Setelah data kuesioner terkumpul, dilakukan perhitungan untuk menentukan nilai kelayakan sistem. Sepuluh item tersebut terbagi menjadi dua jenis: item positif (pertanyaan 1, 3, 5, 7, dan 9), di mana skor dihitung dengan mengurangi nilai skala dari 5; dan item negatif (pertanyaan 2, 4, 6, 8, dan 10), di mana skor dihitung dengan mengurangi nilai skala dari 5. Total skor dari seluruh item dijumlahkan, kemudian dikalikan dengan 2,5. Langkah terakhir adalah menghitung rata-rata keseluruhan untuk memperoleh skor SUS akhir.

$$SKOR\ SUS = ((R1 - 1) + (5 - R2) + (R3 - 1) + (5 - R6)$$

$$+ (R5 - 1) + (5 - R6) + (R7 - 1) + (5 - R8)$$

$$+ (R9 - 1) + (5 - R10)) \, x \, 2.5$$

Skor tersebut didapatkan dengan cara berikut.

1. Setiap pertanyaan bernomor ganjil, skor setiap pertanyaan didapat dari skor pengguna akan dikurangi 1.
2. Setiap pertanyaan bernomor genap, skor akhir didapat dari nilai 5 dikurangi skor pertanyaan yang didapat dari pengguna.
3. Skor SUS didapat dari hasil penjumlahan skor setiap pertanyaan yang kemudian dikali 2,5.

Interpretasi hasil rata-rata keseluruhan skor SUS dapat dilakukan menggunakan beberapa pendekatan, yaitu grade, SUS, percentile range, dan adjective.

Grade	**SUS**	**Percentile**	**Adjective**
(1)	(2)	(3)	(4)
A+	84.1 - 100	96 - 100	Best Imaginable
A	80.8 - 84.0	90 - 95	Excellent
A-	78.9 - 80.7	85 - 89	Good
B+	77.2 - 78.8	80 - 84	
B	74.1 - 77.1	70 - 79	
B-	72.6 - 74	65 - 69	
C+	71.1 - 72.5	60 - 64	
C	65.0 - 71.0	41 - 59	OK
C-	62.5 - 64.9	35 - 40	
D	51.7 - 62.6	15 - 34	

Untuk menentukan tingkat kepuasan pengguna akhir terhadap situs web Sistem BDA, kami mengonversi pernyataan puas dan tidak puas pada kuesioner pengguna sebagai berikut:

Tabel Skala Tingkat Kepuasan Pengguna

Level	Skor (0-5)	Tingkat Kepuasan
1	0 -1.5	Sangat Tidak Puas
2	1.5-2.5	Tidak Puas
3	2.5-3.0	Netral
4	3.0-3.5	Puas
5	3.5-4.0	Sangat Puas

2. Berikut hasil dari kuesioner pengujian Sistem BDA:

Responden dalam kuesioner ini dipilih menggunakan metode snowball sampling, berdasarkan referensi dari individu yang telah diwawancarai sebelumnya, sehingga memungkinkan perluasan jaringan responden yang relevan dengan sistem yang diuji.

Kuesioner black box (SUS) ini dirancang dalam satu media google form yang selanjutnya diisi oleh tujuh responden. Tujuh responden meliputi 1 Admin, 3 Tim Keuangan, dan 6 Operator yaitu pegawai yang ada di BPS DKI Jakarta.

Dari hasil uji black box (SUS) tersebut, dapat disimpulkan bahwa secara keseluruhan sistem dapat berjalan dengan baik. Fitur utama seperti pengajuan form kepada Tim Keuangan oleh operator dan respon Tim Keuangan sampai proses pengajuan selesai berjalan sesuai dengan yang diharapkan.

Kuesioner dengan metode SUS ini dirancang dalam satu media google form yang selanjutnya diisi oleh sepuluh responden. Tiap role meliputi Admin, Tim Keuangan, dan Operator yang mewakili sebagai responden kuesioner.

Hasil Kuesioner SUS

Berikut ini adalah hasil pengujiannya yang mana untuk jawaban hasil test sendiri itu diisi sesuai dengan jawaban klien pada kuesioner melalui tautan google form yang telah dilampirkan sebelumnya dengan 10 pertanyaan.

Participant	Skor Pertanyaan (Q)										Total Skor
	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	
P1	5	2	5	2	5	2	5	2	5	2	87,5
P2	5	1	5	2	4	1	5	1	5	2	92,5
P3	5	1	5	2	4	1	5	1	4	1	92,5
P4	5	2	5	2	5	2	5	2	5	2	87,5
P5	5	2	5	2	5	2	5	2	5	2	87,5
P6	5	2	5	2	5	2	5	2	5	2	87,5
P7	5	1	5	2	4	1	5	1	4	1	92,5
P8	5	2	5	2	5	2	5	2	5	2	87,5

P9	5	2	5	2	5	2	5	2	5	2	87,5
P10	5	2	5	2	5	2	5	2	5	2	87,5
Skor Rata-Rata SUS (Hasil Akhir)											89,0

Dari tabel di atas, rata-rata keseluruhan skor SUS sebesar 89,0. Hal ini menunjukkan bahwa sistem yang dibangun termasuk kategori "Best Imaginable" yang berada pada grade A+. Sistem yang dibangun telah memenuhi dan sudah sesuai dalam memenuhi kebutuhan pengguna.

Dari hasil survei SUS ini dapat disimpulkan bahwa website Sistem BDA sudah berhasil dalam memenuhi kebutuhan di lingkungan pegawai BPS Provinsi DKI Jakarta terkait sistem bukti dukung administrasi keuangan.

6.1.4 Review Client yang Melakukan Testing

Kami mengambil 2 sampel responden dari pegawai BPS Provinsi DKI Jakarta untuk melakukan pengujian sistem. Kedua pegawai BPS yang diambil bersumber dari jabatan fungsional dan salah satu diambil dari unit PPK yang menjadi administrator sekaligus tim keuangan. Tujuan diambil dari jabatan fungsional adalah untuk mengetahui apakah sistem sudah dapat dijalankan secara efektif dan efisien guna membantu melakukan proses bisnis BDA di lingkungan kantor Badan Pusat Statistik DKI Jakarta. sedangkan unit PPK diambil untuk mengetahui kinerja sistem yang diambil dari sudut pandang administrator yang bertanggung jawab untuk melakukan pengecekan BDA pada pengujian yang dilakukan pegawai Badan Pusat Statistik. Hasil testing didapatkan sebagai berikut.

Pegawai fungsional:

No.	Action	Input	Hasil sesuai yang diharapkan
1.	Memasukkan email dan password yang benar kemudian mengklik tombol login	Email dan password user yang terdaftar	✔
2.	Memasukkan email dan password yang salah kemudian mengklik tombol login	Email dan password user yang belum terdaftar	✔

3.	Mengisi detail form pengajuan sesuai validasi dan lengkap, lalu klik tombol kirim	Detail form yang ingin diajukan sudah lengkap	✔
4.	Mengisi detail form pengajuan tidak sesuai validasi dan tidak lengkap, lalu klik tombol kirim	Detail form yang ingin diajukan tidak lengkap	✔
5.	Mengklik tombol view pada monitoring operator	-	✔
6.	Mengklik tombol edit pada monitoring operator	Detail form yang ingin diubah	✔
7.	Mengklik tombol delete pada monitoring operator	Konfirmasi Ya atau tidak untuk menghapus form pengajuan	✔
8.	Mengupload file yang diperlukan untuk memenuhi syarat-syarat pengajuan secara lengkap	File yang sesuai	✔
9.	Tidak lengkap dalam mengupload file yang diperlukan untuk memenuhi syarat-syarat pengajuan	File yang tidak sesuai	✔
10.	Memilih pengajuan yang ingin di download dan klik tombol download	-	✔
11.	Memilih pengajuan yang ingin di download dan klik tombol download	-	✔

Pegawai unit PPK:

No.	Action	Input	Hasil sesuai yang diharapkan
1.	Memasukkan email dan password yang benar kemudian mengklik tombol login	Email dan password user yang terdaftar	✔
2.	Memasukkan email dan password yang salah kemudian mengklik tombol login	Email dan password user yang belum terdaftar	✔
3.	Mengisi detail form pengajuan sesuai validasi dan lengkap, lalu klik tombol kirim	Detail form yang ingin diajukan sudah lengkap	✔
4.	Mengisi detail form pengajuan tidak sesuai validasi dan tidak lengkap, lalu klik tombol kirim	Detail form yang ingin diajukan tidak lengkap	✔
5.	Mengklik tombol view pada monitoring operator	-	✔
6.	Mengklik tombol edit pada monitoring operator	Detail form yang ingin diubah	✔
7.	Mengklik tombol delete pada monitoring operator	Konfirmasi Ya atau tidak untuk menghapus form pengajuan	✔
8.	Mengklik tombol view pada monitoring keuangan	-	✔
9.	Jika pengajuan dari user sesuai dan melengkapi syarat, tim keuangan mengklik tombol approve pengajuan	-	✔
10.	Jika pengajuan dari user ada yang tidak sesuai dan belum melengkapi syarat, tim keuangan	-	✔

	mengklik tombol reject pengajuan		
11.	Mengupload file yang diperlukan untuk memenuhi syarat-syarat pengajuan secara lengkap	File yang sesuai	✔
12.	Tidak lengkap dalam mengupload file yang diperlukan untuk memenuhi syarat-syarat pengajuan	File yang tidak sesuai	✔
13.	Mengupload file yang diperlukan untuk menyetujui dan mendukung pengajuan operator seperti bukti pembayaran dll.	File yang sesuai	✔
14.	Tidak lengkap dalam mengupload file yang diperlukan untuk mendukung pengajuan operator	File yang tidak sesuai	✔
15.	Memilih pengajuan yang ingin di download dan klik tombol download	-	✔
16.	Memilih pengajuan yang ingin di download dan klik tombol download	-	✔
17.	Admin mengubah role user dengan klik tombol dropdown role	-	
18.	Admin menambah user/pengguna ke dalam sistem di form tambah user	Melengkapi detail form dengan benar	✔
19.	Admin menambah user/pengguna ke dalam sistem di form tambah user tetapi tidak lengkap	Melengkapi detail form dengan tidak lengkap	✔

20.	Admin mengubah MAK MAK (Akun belanja, komponen, sub-komponen, dan output) dengan klik tombol dropdown aksi	-	✔
21.	Admin menambah MAK (Akun belanja, komponen, sub-komponen, dan output) ke dalam sistem di form tambah MAK (Akun belanja, komponen, sub-komponen, dan output)	Melengkapi detail form dengan benar sesuai yang diminta	✔
22.	Admin menambah MAK (Akun belanja, komponen, sub-komponen, dan output) ke dalam sistem di form tambah MAK (Akun belanja, komponen, sub-komponen, dan output) tetapi tidak lengkap	Melengkapi detail form dengan tidak lengkap	✔

Dari tabel tersebut pengujian dilakukan pada semua komponen dengan mencocokan fitur-fitur role yang didapatkan dan hasilnya semua berjalan sesuai yang diharapkan berdasarkan tabel 6.1.1 Scenario Testing.

BAB VII
KESIMPULAN

Proyek "Sistem Bukti Dukung Administrasi BPS DKI Jakarta" merupakan solusi untuk mengatasi permasalahan dalam pengelolaan administrasi bukti dukung yang selama ini dilakukan secara manual. Dengan adanya sistem ini, BPS DKI Jakarta dapat memperbaiki proses pengumpulan, verifikasi, dan penyimpanan dokumen menjadi lebih terstruktur dan efisien. Sistem yang diusulkan menawarkan berbagai keuntungan, antara lain peningkatan efisiensi waktu, penghematan biaya, percepatan verifikasi, dan transparansi proses administrasi yang lebih baik.

Implementasi sistem ini berdampak positif terhadap beberapa aspek organisasi. Dari segi sumber daya manusia, sistem ini memungkinkan staf untuk mengakses dan mengelola dokumen dengan lebih mudah, serta meningkatkan koordinasi antar unit kerja. Dari segi proses, digitalisasi alur kerja administrasi menjadikan proses lebih cepat, efisien, dan transparan, yang pada akhirnya mendukung pengambilan keputusan yang lebih efektif. Dari segi teknologi, sistem ini menghadirkan platform yang aman, mudah diakses, dan terpusat untuk memfasilitasi pengelolaan dokumen administrasi secara real-time.

Secara keseluruhan, pengembangan "Sistem Bukti Dukung Administrasi" merupakan langkah strategis bagi BPS DKI Jakarta untuk beradaptasi dengan kebutuhan era digital, meningkatkan kinerja, dan memenuhi harapan pemangku kepentingan dalam penyediaan layanan administrasi yang akurat dan transparan. Dengan adanya sistem ini, BPS DKI Jakarta tidak hanya dapat mendukung kelancaran operasional sehari-hari, tetapi juga berkontribusi pada pencapaian visi dan misi sebagai penyedia data dan informasi yang andal dalam mendukung pembangunan daerah.

LAMPIRAN

Tautan web

https://bda.web.bps.go.id/

Link git-stis

https://github.com/imaliarosyidaa/sistem-bda.git

Link record zoom Milestone 2

https://drive.google.com/drive/folders/1hsZeiSIvR8pWawyuXS4zUOjPF94N3781?usp=sharing

Link record zoom Milestone 3

https://drive.google.com/file/d/1ZPls23VSCSeqwAwSLKeGVzzsdbilahTb/view?usp=sharing

Link record zoom Milestone 4

https://drive.google.com/file/d/1IuCnewTNRMzEMQr9THkje41a5JJb2_tz/view?usp=sharing