

BAB III

ANALISIS DAN PERANCANGAN

3.1 Analisis Sistem

Analisis sistem adalah langkah awal yang sangat penting dalam merancang solusi untuk permasalahan yang dihadapi oleh suatu organisasi. Pada penelitian ini, analisis sistem dilakukan untuk mengevaluasi sistem yang saat ini digunakan oleh Badan Kepegawaian Pelatihan Dan Pendidikan (BKPP) dalam mengelola layanan kepegawaian. Sistem yang digunakan saat ini masih berbasis manual, yang mengandalkan proses pencatatan fisik dan komunikasi tatap muka antar petugas dan pemohon (Hariona et al., 2020). Oleh karena itu, penting untuk memahami dengan baik bagaimana sistem yang ada berfungsi, apa saja kekurangannya, serta bagaimana sistem tersebut mempengaruhi efisiensi dan efektivitas pelayanan.

Dengan melakukan analisis terhadap sistem yang ada, kita dapat mengidentifikasi berbagai masalah operasional yang perlu diatasi, serta memahami kebutuhan pengguna dan *stakeholder* terkait. Analisis sistem ini juga akan memberikan gambaran tentang kekuatan dan kelemahan yang ada pada sistem saat ini, sehingga dapat merancang sistem baru yang lebih baik dan lebih sesuai dengan kebutuhan BKPP. Selain itu, proses ini memungkinkan untuk menentukan area mana saja yang membutuhkan perbaikan melalui otomatisasi, digitalisasi, dan pengintegrasian berbagai elemen yang ada.

Setelah melakukan analisis sistem secara menyeluruh, akan lebih mudah untuk merumuskan solusi yang tepat dan merancang sistem yang dapat mengatasi

masalah yang ada. Oleh karena itu, langkah pertama yang dilakukan dalam penelitian ini adalah menganalisis sistem lama yang masih digunakan di BKPP, untuk kemudian merancang sistem yang lebih efisien, transparan, dan akuntabel. Dalam analisis ini, akan dipetakan berbagai masalah yang ada, mulai dari proses pencatatan manual hingga ketidakmampuan sistem lama dalam memberikan akses informasi secara *real-time* (Irawan & Fauzi, 2025).

Dengan demikian, analisis sistem ini bertujuan untuk mengidentifikasi akar masalah yang ada dan menyusun dasar bagi perancangan sistem baru yang lebih efektif dan efisien. Selanjutnya, analisis ini akan mengarah pada pemahaman yang lebih baik tentang bagaimana alur kerja dapat diatur ulang dan didigitalisasi untuk memberikan layanan yang lebih cepat, lebih transparan, dan lebih akuntabel kepada masyarakat dan pemohon layanan.

3.1.1 Analisa Sistem Lama

Pada tahap awal penelitian ini, penting untuk memahami bagaimana sistem yang digunakan saat ini berfungsi dan mengidentifikasi permasalahan-permasalahan yang timbul dari penerapan sistem tersebut. Sistem yang diterapkan oleh Badan Kepegawaian Pelatihan Dan pendidikan (BKPP) masih menggunakan metode manual dalam pengelolaan permohonan layanan kepegawaian. Meskipun sudah ada proses yang terstruktur, sistem ini sangat bergantung pada pencatatan fisik dan komunikasi tatap muka, yang memunculkan berbagai masalah operasional. Permohonan layanan, seperti pengajuan informasi, legalisasi dokumen, serta klarifikasi data, masih mengharuskan pemohon untuk datang langsung ke kantor atau menghubungi petugas secara langsung. Selain itu,

penggunaan dokumen fisik menyebabkan proses pencatatan dan pengolahan data menjadi kurang efisien, rentan terhadap kesalahan, dan memakan waktu yang cukup lama.

Permasalahan utama lainnya yang ditemukan adalah tidak adanya sistem yang dapat mencatat dan memantau status setiap permohonan secara *real-time*. Pemohon kesulitan untuk mengetahui perkembangan permohonan mereka tanpa harus menunggu kabar langsung dari petugas atau bertanya kepada pihak terkait. Hal ini menyebabkan ketidakpastian yang mengganggu pengalaman pengguna, sekaligus memperpanjang waktu penyelesaian permohonan (Zahra & Nasution, 2025). Proses verifikasi dan persetujuan dokumen juga belum terintegrasi secara efektif, yang berarti setiap langkah memerlukan interaksi manual antar petugas. Keamanan data dan *audit trail* juga menjadi masalah besar, karena tidak adanya jejak digital yang dapat melacak setiap keputusan yang diambil dalam sistem.

Sistem yang ada saat ini memerlukan perbaikan yang mendalam untuk meningkatkan efisiensi, akurasi, dan transparansi layanan yang diberikan. Untuk itu, perlu ada sebuah solusi berbasis teknologi yang dapat mengotomatisasi alur kerja, meningkatkan aksesibilitas informasi, dan memastikan setiap keputusan dapat dipertanggungjawabkan dengan jelas.

3.1.2 Perancangan Sistem Baru

Dengan pemahaman terhadap permasalahan yang ada, pengembangan sistem tiket layanan satu pintu berbasis PHP–MySQL menjadi pilihan yang tepat untuk menggantikan sistem manual yang sudah ada. Sistem baru ini diharapkan mampu mengatasi berbagai masalah yang timbul dengan menyediakan platform digital

yang lebih terstruktur dan efisien. Setiap permohonan akan dicatat sebagai tiket dengan identitas unik dan status yang dapat diperbarui secara real-time, memungkinkan pemohon dan petugas untuk melacak setiap tahap proses.

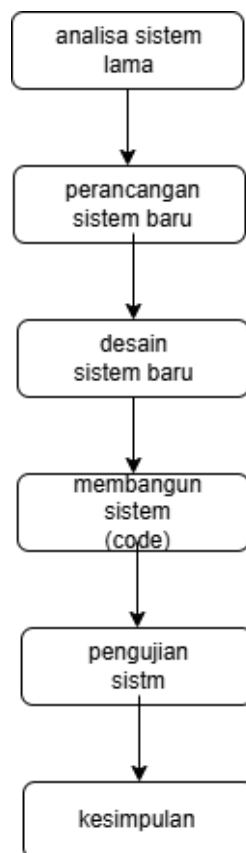
Salah satu keunggulan utama dari sistem ini adalah kemampuannya untuk mengintegrasikan berbagai peran dalam satu platform, mulai dari petugas *front office* yang melakukan verifikasi berkas hingga pejabat penyetuju yang memberikan persetujuan akhir. Selain itu, sistem ini juga dilengkapi dengan fitur notifikasi otomatis yang akan memberi tahu pemohon mengenai perubahan status tiket mereka, sehingga mereka tidak perlu lagi menghubungi petugas untuk mendapatkan informasi terkait.

Integrasi tanda tangan elektronik (TTE) dalam sistem ini juga memungkinkan dokumen resmi diterbitkan tanpa perlu proses manual seperti tanda tangan basah atau penggandaan salinan dokumen. Dengan adanya *audit trail* yang mendetail, setiap keputusan yang diambil dalam proses layanan akan tercatat dengan jelas, meningkatkan akuntabilitas dan transparansi (Hidayah & Ma'ruf, 2018).

Dengan solusi berbasis teknologi ini, diharapkan seluruh alur layanan kepegawaian di BKPP dapat berjalan lebih efisien, lebih transparan, dan lebih mudah dipantau, serta dapat meningkatkan kepuasan pemohon layanan secara keseluruhan.

3.2 Kerangka Penelitian

Kerangka penelitian ini disusun untuk memberikan gambaran jelas mengenai alur dan tahapan yang akan dilalui dalam proses penelitian ini. Dimulai dari tahap pendahuluan, yang mencakup perencanaan penelitian dan studi literatur yang menjadi dasar teori, dilanjutkan dengan landasan teori yang menjelaskan konsep-konsep utama seperti sistem informasi, data, dan sistem informasi manajemen kepegawaian. Selanjutnya, pada metodologi penelitian, dilakukan analisis terhadap sistem yang ada, desain sistem yang mencakup pengumpulan data dan kebutuhan, serta implementasi dan pengujian sistem menggunakan metode Black Box Testing (Hariona et al., 2020).



Gambar 3. 1 Alur kerangka penelitian

Berdasarkan kerangka tersebut, penelitian diawali dengan analisa sistem lama untuk memperoleh gambaran nyata tentang alur layanan yang selama ini berjalan, pihak-pihak yang terlibat, dokumen yang diproses, serta titik-titik masalah yang muncul (misalnya keterlambatan disposisi, informasi status yang tidak transparan, atau kesulitan pelacakan berkas). Pada tahap ini, data dikumpulkan melalui pengamatan proses, telaah dokumen, dan identifikasi kebutuhan pengguna sehingga menghasilkan daftar kebutuhan sistem serta batasan yang harus diperhatikan.

Tahap berikutnya adalah perancangan sistem baru, yaitu menyusun solusi yang dapat menjawab masalah pada sistem lama. Perancangan mencakup penentuan tujuan perbaikan, penyusunan kebutuhan fungsional dan nonfungsional, serta perumusan alur layanan yang lebih terstruktur. Pada tahap ini juga ditentukan pembagian peran pengguna (misalnya pemohon, *front office*, koordinator, petugas, dan pimpinan) beserta hak aksesnya agar setiap proses berjalan sesuai kewenangan dan tidak menimbulkan konflik akses.

Setelah rancangan umum terbentuk, dilakukan desain sistem sebagai pedoman teknis implementasi. Desain sistem memuat pemodelan proses dan interaksi menggunakan diagram (use case untuk cakupan fungsi, activity untuk alur kerja, dan sequence untuk urutan komunikasi antar aktor/komponen). Selain itu, disusun desain basis data untuk memastikan data pemohon, tiket layanan, status proses, dokumen, dan jejak aktivitas dapat tersimpan secara konsisten dan mudah ditelusuri (Atakari, 2025).

Tahap selanjutnya adalah membangun sistem (*coding*), yaitu menerjemahkan hasil desain ke dalam program berbasis web. Pada tahap ini dilakukan pembuatan

modul-modul utama sesuai kebutuhan, seperti autentikasi dan manajemen sesi, pengajuan permohonan, pengelolaan tiket dan perubahan status, unggah/unduh dokumen, notifikasi, serta pencatatan *audit trail*. Implementasi dilakukan dengan mengikuti struktur aplikasi dan standar penulisan kode yang rapi agar sistem mudah dipelihara dan dikembangkan di kemudian hari.

Setelah sistem selesai dibangun, dilakukan pengujian sistem untuk memastikan seluruh fungsi berjalan sesuai rancangan dan kebutuhan. Pengujian meliputi pemeriksaan setiap fitur utama (misalnya proses pengajuan sampai ACC/TTE), validasi input, akses sesuai *role*, kebenaran penyimpanan data ke basis data, serta memastikan dokumen dapat diunggah dan diakses dengan benar (Jhon, 2023). Hasil pengujian digunakan untuk menemukan kekurangan, melakukan perbaikan, dan memastikan sistem stabil sebelum dinyatakan siap digunakan.

Tahap terakhir adalah kesimpulan, yaitu merangkum hasil penelitian berdasarkan implementasi dan pengujian yang telah dilakukan. Pada bagian ini dijelaskan sejauh mana sistem baru mampu memperbaiki proses layanan dibanding sistem lama, termasuk peningkatan efisiensi alur kerja, keterlacakan status tiket, dan kemudahan pengelolaan dokumen.

3.3 Desain Sistem

Desain sistem merupakan tahapan krusial dalam proses pengembangan sistem informasi, di mana hasil dari analisis sistem yang telah dilakukan sebelumnya diterjemahkan ke dalam suatu blueprint atau rancangan yang akan menjadi acuan dalam pengembangan sistem yang lebih baik. Pada tahap desain sistem, tujuan utamanya adalah merancang solusi yang efektif untuk mengatasi

masalah yang telah diidentifikasi dan memastikan bahwa sistem yang dihasilkan dapat berjalan secara efisien, sesuai dengan kebutuhan organisasi, serta mudah digunakan oleh para penggunanya.

Desain sistem ini tidak hanya mencakup struktur teknis atau arsitektur perangkat lunak, tetapi juga mencakup perencanaan tentang bagaimana alur kerja (*workflow*), interaksi antar pengguna (*user interaction*), serta pengelolaan data dalam sistem dapat diatur dengan baik. Sistem yang dirancang haruslah mampu mendukung proses bisnis yang ada, memperbaiki kekurangan dari sistem lama, dan memberikan kemudahan akses serta transparansi bagi pengguna, baik itu pemohon layanan, petugas, maupun pimpinan (Zahra & Nasution, 2025).

Pada penelitian ini, desain sistem difokuskan pada pembuatan sistem tiket layanan satu pintu yang berbasis PHP–MySQL dengan integrasi tanda tangan elektronik (TTE) [15]. Rancangan sistem ini berfungsi untuk mengoptimalkan pengelolaan permohonan layanan kepegawaian yang selama ini masih dilakukan secara manual. Dengan merancang sistem yang memanfaatkan teknologi terkini, diharapkan proses pengajuan dan penyelesaian permohonan dapat dilakukan secara digital, lebih cepat, lebih akurat, dan lebih efisien. Selain itu, sistem baru ini juga diharapkan mampu mengurangi risiko kesalahan manusia dan meningkatkan akuntabilitas melalui pencatatan yang dapat diaudit secara otomatis.

Desain sistem ini tidak hanya memperhatikan aspek teknis seperti pengelolaan database dan antarmuka pengguna (*user interface*), tetapi juga mempertimbangkan peran dan hak akses pengguna yang berbeda-beda, alur kerja yang efisien, serta fitur-fitur yang akan mendukung proses verifikasi, persetujuan,

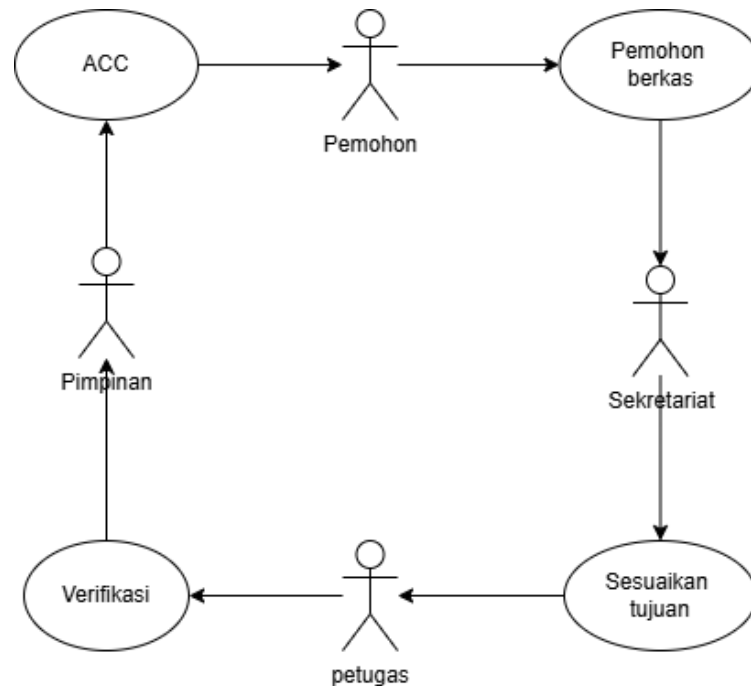
dan penerbitan dokumen secara elektronik (Nur Fikri Zaki et al., 2025). Oleh karena itu, pada bagian ini akan dijelaskan berbagai elemen desain sistem mulai dari use case diagram, activity diagram, hingga class diagram yang menggambarkan secara rinci komponen-komponen sistem yang akan dibangun.

Dengan demikian, desain sistem ini bertujuan untuk memastikan bahwa sistem yang dikembangkan tidak hanya memenuhi kebutuhan teknis, tetapi juga memberikan solusi praktis yang sesuai dengan konteks operasional yang ada di Badan Kepegawaian Pelatihan Dan Pendidikan (BKPP). Selanjutnya, setiap komponen dalam sistem ini akan dijelaskan lebih lanjut untuk memastikan implementasi yang tepat dan sesuai dengan tujuan penelitian ini.

3.3.1 Use Case Diagram

Use Case Diagram menggambarkan interaksi antara aktor-aktor yang terlibat dalam sistem tiket layanan satu pintu dan sistem itu sendiri. Dalam konteks BKPP, aktor utama yang terlibat antara lain Pemohon, Petugas Sekretariat, Petugas Bidang, dan Pimpinan. Setiap aktor memiliki fungsi tertentu yang dapat dijalankan, seperti berikut.

1. Pemohon: Mengajukan tiket layanan, melengkapi persyaratan, melacak status tiket.
2. Petugas Sekretariat: Memverifikasi kelengkapan berkas dan menyesuaikan berkas yang akan dilanjutkan ke bidang yang dipilih.
3. Petugas Bidang: Melakukan verifikasi berkas.
4. Pimpinan: Menandatangani dan menolak permohonan berkas yang akan dikirim lagi kepada pemohon



Gambar 3. 2 Use Case Diagram

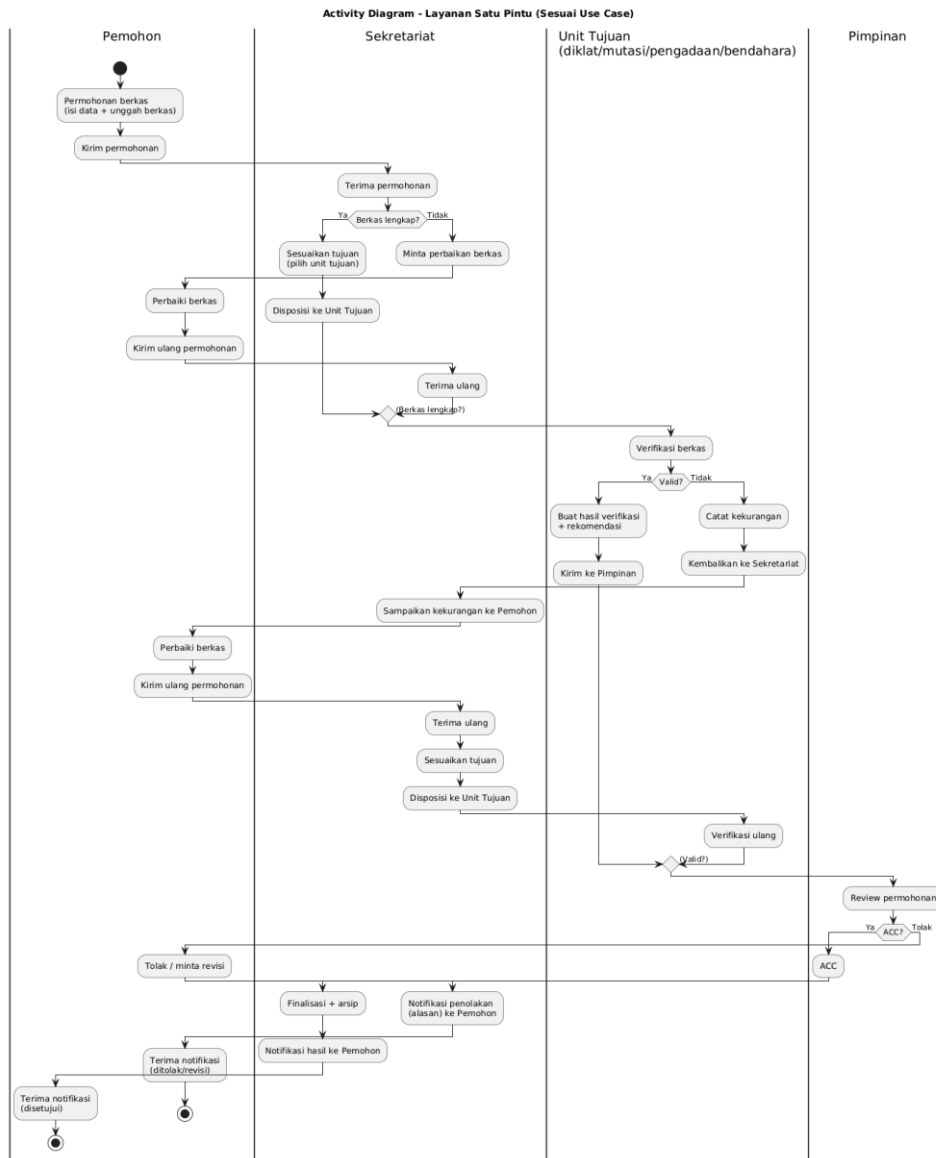
3.3.2 Activity Diagram

Activity Diagram menggambarkan alur aktivitas yang lebih rinci dalam sistem tiket. Diagram ini menjelaskan langkah-langkah yang diambil oleh setiap aktor dalam memproses tiket permohonan layanan. Misalnya, langkah pertama dimulai dengan pemohon yang mengajukan tiket layanan, kemudian proses berlanjut ke tahap verifikasi, pemeriksaan substansi, hingga diterbitkannya dokumen yang ditandatangani secara elektronik.

Beberapa alur yang digambarkan dalam diagram ini termasuk:

1. Pemohon mengisi form pengajuan dan mengunggah berkas.
2. Petugas sekretarian memverifikasi kelengkapan dokumen.
3. Tiket diproses dan diverifikasi oleh petugas bidang.
4. Pejabat Penyetuju memberikan persetujuan akhir.

5. Penandatanganan elektronik dilakukan, dan tiket ditutup.



Gambar 3.3 Activity Diagram

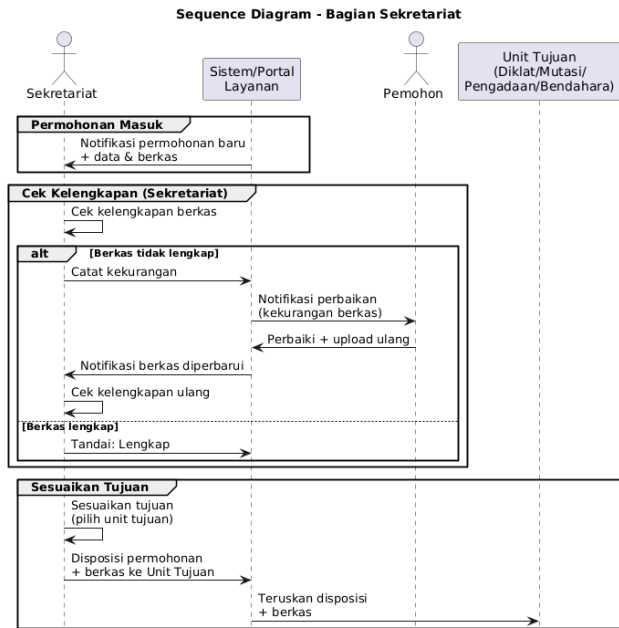
3.3.3 Sequence Diagram

Sequence Diagram menggambarkan urutan interaksi antar objek atau entitas dalam sistem selama proses pengajuan tiket berlangsung. Diagram ini menunjukkan

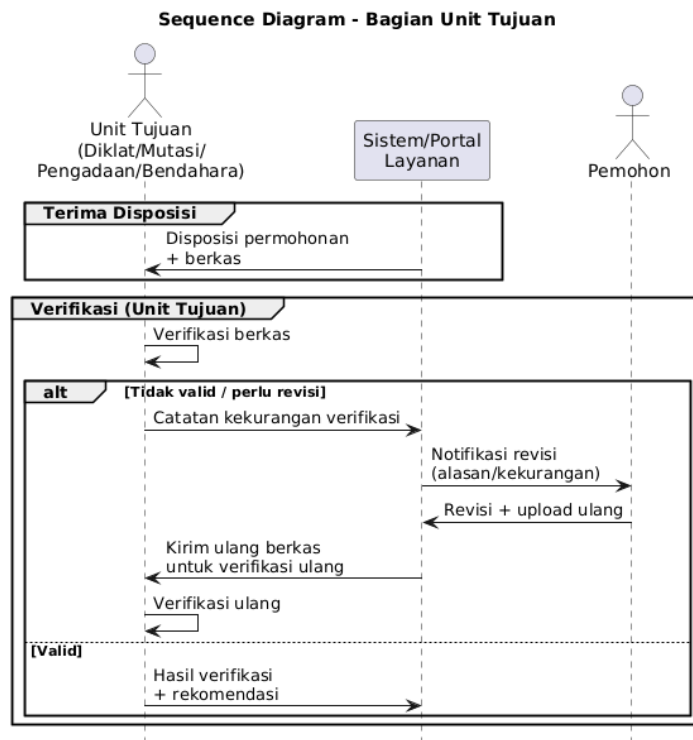
bagaimana setiap objek (misalnya, Pemohon, Petugas Sekretariat, Petugas bidang, Pimpinan) berinteraksi dan bertukar pesan pada urutan waktu tertentu.



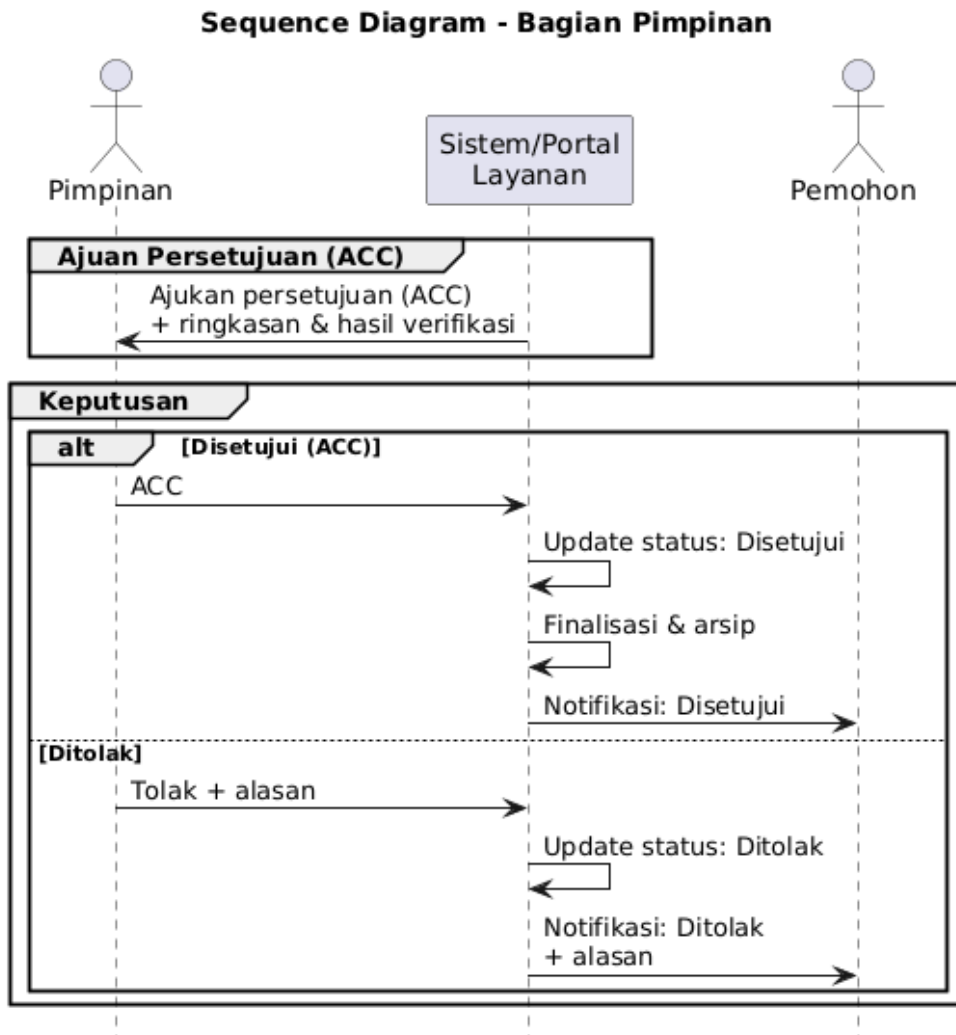
Gambar 3. 4 Sequence Diagram Pemohon



Gambar 3. 5 Diagram Sequence Sekretariat



Gambar 3. 6 Diagram Sequence Unit



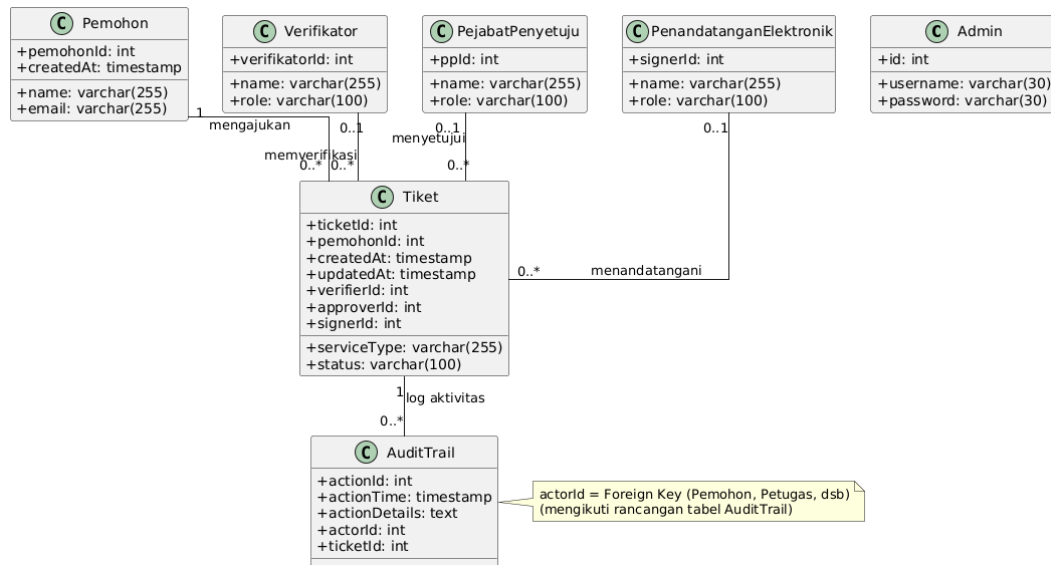
Gambar 3. 7 Diagram Sequence Pimpinan

3.3.4 Class Diagram

Class Diagram menggambarkan struktur objek dalam sistem dan hubungan antar kelas yang ada. Dalam sistem ini, kelas utama meliputi Pemohon, yang menyimpan informasi pemohon tiket; Tiket, yang berisi detail permohonan layanan; Verifikator, yang bertugas memverifikasi tiket; Penyetuju, yang menyetujui tiket yang telah diverifikasi; dan Penandatanganan Elektronik, yang menandatangani dokumen secara digital. Kelas-kelas ini saling terhubung untuk

menggambarkan alur proses tiket, mulai dari pengajuan hingga penandatanganan, dengan atribut dan relasi yang mendefinisikan interaksi antar entitas dalam sistem layanan kepegawaian. Kelas utama yang ada dalam sistem ini antara lain.

1. Pemohon: Kelas yang menyimpan informasi terkait pemohon tiket.
2. Tiket: Kelas yang menyimpan detail tiket layanan yang diajukan.
3. Dokumen: Kelas yang menyimpan informasi dokumen yang diterbitkan.
4. Tanda Tangan Elektronik: Kelas yang menangani proses tanda tangan digital pada dokumen.



Gambar 3. 8 Class Diagram

3.4 Rancangan Interface

Rancangan *interface* atau antarmuka pengguna (UI) merupakan salah satu elemen penting dalam pengembangan sistem informasi, karena antarmuka ini menjadi titik interaksi utama antara pengguna dan sistem. Desain antarmuka yang baik harus mampu memenuhi dua tujuan utama: pertama, memberikan pengalaman

pengguna yang intuitif dan mudah dipahami, serta kedua, mendukung efisiensi dan efektivitas dalam menjalankan tugas-tugas yang diinginkan pengguna (Sulaiman et al., 2022). Dalam konteks sistem tiket layanan satu pintu ini, antarmuka yang dirancang harus mendukung alur kerja yang terstruktur, memungkinkan pengguna untuk mengakses, mengajukan, memproses, dan memverifikasi tiket dengan mudah.

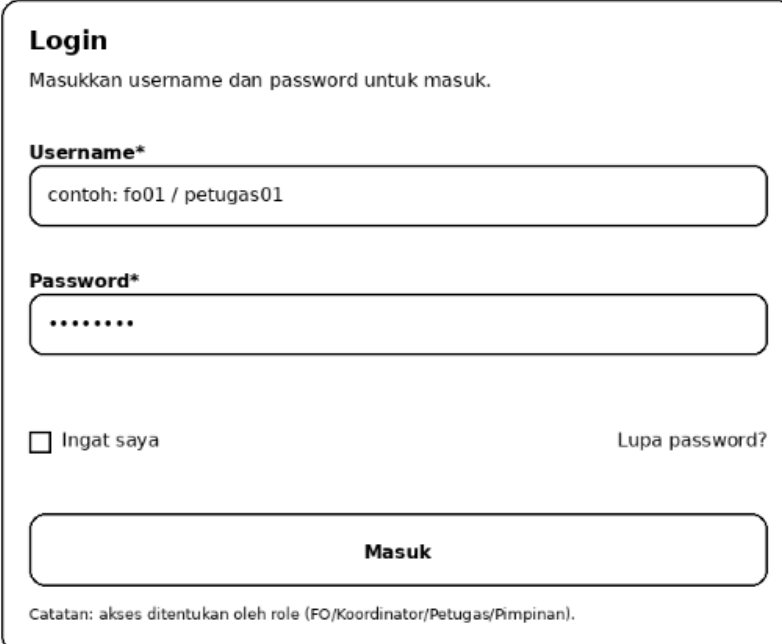
Salah satu tantangan dalam merancang antarmuka untuk sistem layanan publik adalah memastikan bahwa pengguna dengan berbagai tingkat keterampilan teknologi dapat menggunakan sistem dengan lancar. Oleh karena itu, rancangan interface yang efektif harus mempertimbangkan aspek kemudahan akses, kesederhanaan, dan kejelasan informasi yang disajikan. Setiap elemen desain harus mendukung tujuan fungsionalitas sistem, seperti kemudahan dalam mengajukan permohonan, memverifikasi berkas, hingga memantau status tiket secara *real-time*.

Pada penelitian ini, rancangan interface difokuskan pada pembuatan antarmuka yang dapat mengakomodasi kebutuhan berbagai aktor yang terlibat dalam proses layanan, yaitu pemohon, petugas *front office*, petugas *back office*, verifikator, pejabat penyetuju, dan penandatangan (Dzulfian Syafrian, 2025). Setiap aktor memiliki kebutuhan yang berbeda dalam berinteraksi dengan sistem, sehingga desain antarmuka harus disesuaikan dengan peran dan tugas masing-masing.

Dengan desain *interface* yang baik, proses perencanaan tampilan masukan (input) dan keluaran (output) pada suatu program yang bertujuan untuk mempermudah pengguna dalam berinteraksi dengan sistem.

3.4.1 Rancangan Halaman *Login*

Halaman *Login* merupakan antarmuka awal yang digunakan untuk melakukan autentikasi pengguna sebelum mengakses fitur Sistem Layanan Satu Pintu. Pada form ini pengguna memasukkan username dan password yang telah terdaftar. Setelah proses verifikasi berhasil, sistem membentuk sesi (session) dan mengarahkan pengguna ke halaman sesuai hak akses/*role* (misalnya *Front office*, Koordinator, Petugas, atau Pimpinan). Dengan adanya mekanisme *login*, akses terhadap data tiket, dokumen, dan aktivitas pelayanan dapat dibatasi hanya untuk pengguna yang berwenang sehingga keamanan dan keterlacakan proses layanan tetap terjaga.



Login
Masukkan usemame dan password untuk masuk.

Username*
contoh: fo01 / petugas01

Password*
.....

Ingat saya Lupa password?

Masuk

Catatan: akses ditentukan oleh role (FO/Koordinator/Petugas/Pimpinan).

Gambar 3. 9 Interface form login

3.4.2 Rancangan Halaman Pengajuan

Halaman pengajuan dirancang dengan antarmuka yang *user-friendly*, memudahkan pemohon untuk mengisi data, mengunggah dokumen, dan memantau status tiket. Setiap form pengajuan dilengkapi dengan panduan untuk memastikan kelengkapan data.

Form Pengajuan Layanan

Isi data berikut untuk mengajukan layanan. (*) wajib diisi.

Nama*	Email
<input type="text" value="Nama lengkap"/>	<input type="text" value="email@contoh.com"/>
Telepon	Alamat
<input type="text" value="08xxxxxxxxxx"/>	<input type="text" value="Alamat pemohon"/>
Layanan*	
<input type="text" value="Pilih jenis layanan (SLA: xx jam)"/>	
Berkas pendukung	
<input type="text" value="[Pilih file... (multiple)]"/>	

Catatan: unggah berkas pendukung (pdf/jpg) bila diperlukan.

Setelah dikirim, pemohon dapat memantau status melalui notifikasi.

Kirim Pengajuan

Gambar 3. 10 Interface Halaman Pengajuan

3.4.3 Rancangan Halaman *Front office*

Halaman ini memberikan akses bagi petugas *Front office* untuk memverifikasi tiket yang diajukan oleh pemohon. Di halaman ini, petugas dapat melihat detail tiket, memeriksa kelengkapan dokumen, dan melakukan pembaruan status tiket.

Perlu Screening (Front Office)

Pencarian:

Kode	Pemohon	Layanan	Arahkan ke Bidang	Aksi
			<input type="button" value="Pilih bidang..."/>	<input type="button" value="Terima"/> <input type="button" value="Tolak"/>

Petugas FO memilih bidang tujuan lalu menekan Terima/Tolak.

Gambar 3. 11 Interface *Front office*

3.4.4 Rancangan Halaman *Back office*

Petugas *Back office* dapat menggunakan halaman ini untuk memproses tiket yang lebih kompleks. Halaman ini dilengkapi dengan fitur untuk memeriksa substansi permohonan dan mengelola data terkait.

KoordinatorPetugasRiwayat

Antrian Koordinator

Kode	Layanan	Prioritas	Assign ke Petugas
			Pilih petugas... Assign

Koordinator melakukan penugasan (assign) tiket ke petugas.

Tugas Saya (Petugas Back Office)

Kode	Layanan	Status	Aksi
			Minta Data
			Siap TTE
			Tolak

Petugas memproses tiket hingga status SIAP_TTD (Siap TTE).

Gambar 3.12 *Interface Back office*

3.4.5 Rancangan Halaman Pimpinan

Halaman ini dirancang untuk memudahkan pimpinan memantau kinerja sistem secara keseluruhan. *Dashboard* yang tersedia menampilkan data statistik, seperti jumlah tiket yang diajukan, tingkat penyelesaian, dan SLA yang terpenuhi.

Menunggu TTE (Unggah Hasil TTE)

Kode	Pemohon	Layanan	Unggah Hasil TTE
			[Pilih file...] <input type="button" value="Unggah"/>

Pimpinan mengunggah dokumen hasil TTE untuk menyelesaikan tiket.

Gambar 3. 13 Interface Pimpinan

3.4.6 Rancangan Halaman Cek Status

Halaman cek status dirancang untuk menampilkan informasi perkembangan permohonan yang telah diajukan oleh pengguna. Pada halaman ini ditampilkan data seperti kode tiket, nama pemohon, jenis layanan, dan status permohonan. Selain itu, sistem juga menampilkan riwayat status proses permohonan beserta waktu dan catatan pada setiap tahapannya. Jika proses telah selesai, pengguna dapat mengunduh dokumen yang telah ditandatangani secara elektronik (TTE).

Cek Status [Login](#)

Hasil Cek Status

Kode Tiket: TCK-20260308-7195

Pemohon: kymul

Layanan: Informasi Umum

Status: SELESAI

Riwayat Status [Download Berkas TTE](#)

STATUS	CATATAN	WAKTU
DIAJUKAN	Pengajuan via web	2026-03-08 22:11:47
DITERIMA_FO	Screening FO: diterima	2026-03-08 22:12:30
DALAM_PROSES	Penugasan ke petugas	2026-03-08 22:12:38
SIAP_TTD		2026-03-08 22:13:06
SELESAI	Dokumen keluar TTE tersimpan	2026-03-08 22:13:20

Gambar 3. 14 Interface pimpinan

3.5 Rancangan Database

Rancangan *database* ini menggunakan MySQL untuk menyimpan data yang relevan, seperti data pemohon, tiket, dokumen, status tiket, serta jejak audit. Setiap tabel akan memiliki relasi yang jelas, dengan kunci utama dan kunci asing untuk memastikan integritas data.

3.5.1 Tabel Admin

Disini akan dijelaskan mengenai struktur tabel *Admin* yang digunakan untuk menyimpan informasi pengguna yang memiliki akses untuk mengelola sistem. Tabel ini menyimpan data terkait identitas admin, seperti nama pengguna dan password untuk *login* ke dalam sistem.

Tabel 3. 1 database admin

Field	Type	Length	Keterangan
id	int	10	Primary Key
username	varchar	30	Nama Pengguna
password	varchar	30	Password untuk <i>login</i>

3.5.2 Tabel Pemohon

Selanjutnya, akan dibahas mengenai tabel Pemohon yang menyimpan data mengenai pemohon layanan. Tabel ini menyertakan informasi penting seperti nama pemohon, alamat *email*, dan waktu pendaftaran mereka.

Tabel 3. 2 database pemohon

Field	Type	Length	Keterangan
pemohonId	int	10	Primary Key
name	varchar	20	Nama Pemohon
email	varchar	20	Email Pemohon

createdAt	timestamp	-	Waktu Pendaftaran
-----------	-----------	---	-------------------

3.5.3 Tabel Tiket

Pada bagian ini akan dijelaskan tentang tabel *Tiket*, yang berfungsi untuk menyimpan informasi terkait tiket yang dibuat oleh pemohon. Tabel ini mencakup berbagai informasi seperti jenis layanan yang diminta, status tiket, serta relasi dengan pemohon, verifikator, pejabat penyetuju, dan penandatanganan elektronik.

Tabel 3. 3 Database Tiket

Field	Type	Length	Keterangan
ticketId	int	11	Primary Key
pemohonId	int	11	Foreign Key (Pemohon)
serviceType	varchar	25	Jenis Layanan
status	varchar	100	Status Tiket
createdAt	timestamp	-	Waktu Pembuatan Tiket
updatedAt	timestamp	-	Waktu Pembaruan Tiket
verifierId	int	11	Foreign Key (Verifikator)

approverId	int	11	Foreign Key (Pejabat Penyetuju)
signerId	int	11	Foreign Key (Penandatanganan Elektronik)

3.5.4 Tabel Verifikator

Bagian ini akan mengulas tabel *Verifikator*, yang menyimpan data tentang individu yang bertugas untuk memverifikasi tiket. Informasi yang disimpan dalam tabel ini meliputi nama dan peran verifikator dalam proses pengelolaan tiket.

Tabel 3. 4 Database Verifikator

Field	Type	Length	Keterangan
verifikatorId	int	11	Primary Key
name	varchar	50	Nama Verifikator
<i>role</i>	varchar	100	Peran Verifikator

3.5.5 Tabel Pejabat Penyetuju

Pada bagian ini, dibahas mengenai tabel *Pejabat Penyetuju*, yang menyimpan data pejabat yang berwenang untuk menyetujui tiket yang telah diverifikasi. Tabel ini menyertakan informasi mengenai nama dan peran pejabat penyetuju.

Tabel 3. 5 Database Pejabat Penyetuju

Field	Type	Length	Keterangan
ppId	int	11	Primary Key
name	varchar	50	Nama Pejabat Penyetuju
role	varchar	100	Peran Pejabat Penyetuju

3.5.6 Tabel Penandatanganan Elektronik

Menjelaskan tabel *Penandatanganan Elektronik*, yang menyimpan data mengenai individu yang memiliki otoritas untuk menandatangani tiket secara elektronik. Data yang disimpan termasuk nama dan peran penandatanganan elektronik.

Tabel 3. 6 Database TTE

Field	Type	Length	Keterangan
signerId	int	11	Primary Key
name	varchar	50	Nama Penandatanganan Elektronik
role	varchar	100	Peran Penandatanganan Elektronik

3.5.7 Tabel *Audit trail*

Disini akan dibahas mengenai tabel *AuditTrail*, yang digunakan untuk merekam jejak audit terkait tindakan yang dilakukan dalam sistem. Tabel ini mencatat informasi seperti waktu tindakan, rincian tindakan yang dilakukan, serta identitas aktor yang terlibat dan tiket yang terkait.

Tabel 3. 7 Database Audit trail

Field	Type	Length	Keterangan
actionId	int	11	Primary Key
actionTime	timestamp	-	Waktu Tindakan
actionDetails	text	-	Rincian Tindakan
actorId	int	11	Foreign Key (Pemohon, Petugas, dsb)
ticketId	int	11	Foreign Key (Tiket)

3.5.8 Component Diagram

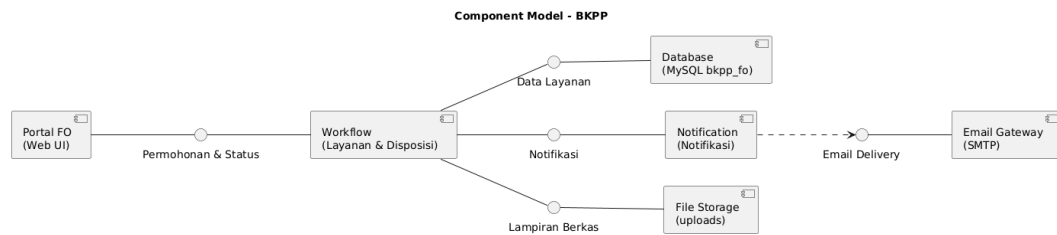
Component Diagram adalah diagram UML untuk menggambarkan struktur modul/komponen perangkat lunak dan hubungan ketergantungan antar komponen tersebut. Fokusnya bukan alur proses (itu tugas *activity/sequence*), tetapi “bagian-bagian sistem ini tersusun dari apa saja” dan bagaimana mereka berkomunikasi lewat antarmuka (*interface*).

Dalam konteks use case (Permohonan berkas → Sesuaikan tujuan → Verifikasi → ACC), *Component Diagram* dipakai untuk:

1. Memetakan modul utama sistem (misalnya: Portal Web, *Backend API*, Permohonan, Disposisi/tujuan, Verifikasi, Persetujuan/ACC, Notifikasi, Dokumen).
2. Menjelaskan pembagian tanggung jawab tiap komponen (sehingga desain lebih rapi, mudah dikembangkan, dan mudah diuji).
3. Menunjukkan interface yang disediakan (*provided*) dan dibutuhkan (*required*), misalnya:
 - a) Portal Web “memanggil” layanan Permohonan/Verifikasi/ACC melalui API.
 - b) Komponen Permohonan menyimpan data ke Database dan berkas ke Storage.
 - c) Komponen Notifikasi mengirim pesan ke Email/WA *Gateway*.

Elemen yang biasa muncul:

1. Component: modul perangkat lunak (Portal Web, Backend API, dll).
2. Interface (*provided/required*): kontrak layanan (mis. IPermohonan, IVerifikasi, IApproval).
3. Dependency: ketergantungan antar komponen (mis. Modul Permohonan bergantung pada *Database dan Storage*)



Gambar 3. 15 Component Diagram

3.5.9 Deployment Diagram

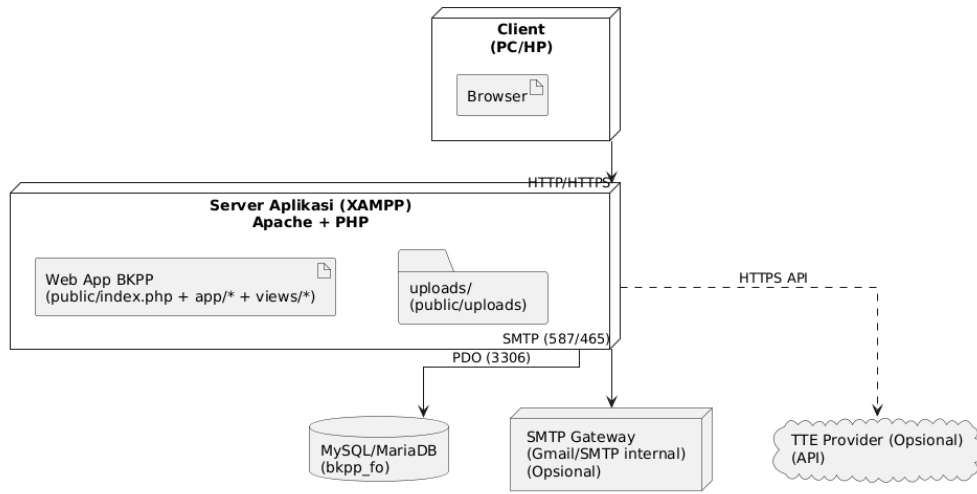
Deployment Diagram adalah diagram UML untuk menggambarkan penempatan (deployment) komponen perangkat lunak ke infrastruktur fisik/virtual. Kalau Component Diagram menjawab “sistem tersusun dari apa”, Deployment Diagram menjawab “sistem ini dijalankan di mana”. Dalam konteks sistem Anda, *Deployment Diagram* penting untuk menjelaskan.

1. Node: perangkat/lingkungan eksekusi (*Client, Reverse Proxy, App Server, DB Server, Storage*).
2. Artifact: paket yang dideploy (mis. *portal-web, backend-api, worker-notifikasi*).
3. Communication path: jalur komunikasi antar node (HTTPS dari client ke server, koneksi app ke DB, akses ke storage, panggilan ke *gateway* notifikasi).

Deployment Diagram juga membantu mengaitkan kebutuhan non-fungsional seperti:

- a) Keamanan (HTTPS/TLS, pemisahan jaringan DMZ vs internal).
- b) Kinerja (worker untuk notifikasi, scaling app server).
- c) Keandalan (*backup DB, storage terpisah, logging/audit*).

Deployment Diagram - Sistem Layanan Satu Pintu



Gambar 3. 16 Deployment Diagram