

BAB II

TINJAUAN PUSTAKA

2.1 Landasan Teori

2.1.1 Sistem Informasi Manajemen

Sistem Informasi Manajemen menjadi komponen yang krusial dalam rangka pengendalian internal perusahaan, yang mencakup efisiensi penggunaan sumber daya manusia, dokumen, teknologi, dan prosedur oleh pihak manajemen untuk mengatasi berbagai permasalahan bisnis. Dikenal umum sebagai suatu entitas manusia-mesin yang terpadu, sistem informasi manajemen berfungsi menyajikan informasi yang mendukung operasi, manajemen, dan pengambilan keputusan di dalam sebuah organisasi. Komponen-komponen utamanya melibatkan perangkat keras dan perangkat lunak komputer, panduan prosedur, model manajemen dan keputusan, serta sebuah basis data (Abdul Kadir, 2018).

2.1.2 Manajemen Tiket

Manajemen tiket adalah suatu proses untuk mengumpulkan, memproses, dan menyelesaikan keluhan pelanggan. Proses ini biasanya menggunakan sistem tiket untuk memudahkan pengelolaan keluhan. Pentingnya membangun manajemen tiket keluhan didasarkan pada beberapa prinsip kunci. Pertama, transparansi menjadi landasan utama, di mana pelanggan harus memiliki akses untuk mengetahui status keluhannya secara jelas. Kecepatan juga menjadi faktor penting, di mana penyelesaian keluhan harus dilakukan dengan cepat untuk memenuhi kebutuhan dan harapan pelanggan. Selain itu, akurasi menjadi aspek krusial, di mana keluhan harus diselesaikan dengan benar dan sesuai dengan harapan pelanggan untuk memastikan kepuasan mereka. Terakhir, kelengkapan dalam penanganan keluhan melibatkan solusi menyeluruh dan tindak lanjut yang memastikan keluhan diselesaikan dengan tuntas. Dengan adanya proses manajemen pada sistem akan memudahkan petugas sistem

informasi untuk mengetahui masalah dan pengguna dalam pengalamannya menggunakan sistem informasi (Muharam & Sulianta, 2023).

Proses manajemen tiket keluhan dapat dibagi menjadi beberapa tahap, yaitu:

- a. Penerimaan tiket, yaitu tahap untuk menerima keluhan dari pelanggan.
- b. Penilaian tiket, yaitu tahap untuk menilai keluhan dan menentukan tingkat prioritasnya.
- c. Penyelesaian tiket, yaitu tahap untuk menyelesaikan keluhan sesuai dengan prosedur.
- d. Pelaporan tiket, yaitu tahap untuk melaporkan hasil penyelesaian keluhan kepada pelanggan.

2.1.3 Sistem Berbasis Website

Website atau situs web adalah koleksi informasi yang terdiri dari halaman-halaman web yang terhubung satu sama lain dan disajikan oleh individu, kelompok, atau organisasi. Sebuah situs web yang efektif menampilkan tata letak visual yang menarik dan berfungsi sesuai dengan kebutuhan pengguna (Rochmawati, 2019). Sistem berbasis website adalah sistem yang memanfaatkan teknologi *website* untuk menyimpan, mengolah, dan menampilkan data. Sistem berbasis *website* dapat diakses oleh pengguna melalui internet atau intranet.

Tujuan utama dari sistem berbasis *website* adalah menyediakan akses yang mudah terhadap informasi dan layanan bagi pengguna. Dengan memanfaatkan platform web, perusahaan dapat meningkatkan efisiensi dan efektivitas dalam menjalankan proses bisnisnya. Sistem ini juga berperan penting dalam meningkatkan komunikasi dan kolaborasi antara berbagai pihak terkait, baik internal maupun eksternal organisasi. Selain itu, fokus pada penggunaan sistem berbasis *website* bertujuan untuk

meningkatkan kepuasan pengguna dengan memberikan akses yang cepat dan responsif, menciptakan pengalaman pengguna yang memuaskan, serta mendukung kebutuhan dan harapan mereka secara optimal.

2.1.4 Konsep *Unified Modelling Language (UML)*

Unified Modeling Language (UML) merupakan salah satu bahasa standar yang secara luas digunakan dalam industri untuk menguraikan kebutuhan, melakukan analisis dan desain, serta mengilustrasikan arsitektur dalam konteks pemrograman berorientasi objek (Putra & Andriani, 2019). Landasan teori UML didasarkan pada konsep-konsep dasar yang mencakup objek, kelas, relasi, dan diagram. Objek, sebagai unit dasar dalam sistem perangkat lunak, memiliki atribut dan perilaku yang mendefinisikannya secara unik. Kelas merupakan kumpulan objek yang memiliki atribut dan perilaku yang serupa. Relasi mewakili koneksi antara dua atau lebih objek atau kelas yang mencerminkan interaksi di dalam sistem. Diagram, sebagai representasi grafis, digunakan untuk menggambarkan secara visual elemen-elemen dalam sistem perangkat lunak, memberikan gambaran yang jelas dan terstruktur.

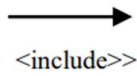
UML terdiri dari beberapa diagram yang dapat digunakan untuk menggambarkan berbagai aspek sistem perangkat lunak. Diagram-digram UML tersebut adalah:

1. *Diagram Usecase*

Use case diagram merupakan pemodelan perilaku (behavior) dari suatu sistem yang dibutuhkan serta diharapkan pengguna (Suharni et al., 2023). *Diagram Usecase* memberikan gambaran interaksi antara pengguna dan perangkat lunak suatu sistem. *Usecase* berfungsi dengan menggambarkan secara rinci interaksi tipikal antara pengguna suatu sistem dan bagaimana sistem tersebut digunakan melalui narasi atau cerita

penggunaannya. Berikut adalah simbol-simbol yang digunakan dalam *diagram usecase*.

Tabel 2. 1 Simbol *Diagram Usecase*

Simbol	Nama	Deskripsi
	Actor	Digunakan untuk menjelaskan sesuatu atau seseorang yang sedang berinteraksi dengan sistem.
	Use Case	Menggambarkan suatu perilaku dari sistem tanpa mengungkapkan struktur internal dari sistem tersebut.
	Assosiation	Jalur komunikasi antar actor dengan use case yang saling berpartisipasi.
	Extend	Penambahan perilaku ke dalam use case dasar yang tidak tahu tentang hal tersebut.
	Use case generalization	Hubungan antara use case umum dengan use case yang lebih spesifik, yang mewarisi dan menambah fitur terhadapnya.
	Include	Penambahan perilaku ke dalam use case dasar yang secara eksplisit menjelaskan penambahannya.

Sumber: (Suharni et al., 2023)

2. *Diagram Activity*

Diagram Activity adalah representasi grafis yang mengilustrasikan alur kerja atau aktivitas dalam suatu sistem perangkat lunak (Putra & Andriani, 2019). *Diagram Activity* menggambarkan aliran kerja dari sistem perangkat lunak.

Berikut adalah simbol-simbol yang digunakan dalam *diagram activity*.

Tabel 2. 2 Simbol *Diagram Activity*

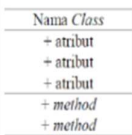
Simbol	Nama	Deskripsi
	Initial	Menunjukkan di mana aliran kerja dimulai.
	Final	Menunjukkan di mana aliran kerja berakhir.
	Action	Langkah-langkah dalam sebuah activity.
	Decision	Menunjukkan di mana keputusan akan dibuat.
	Swimlane	Mengelompokkan activity berdasarkan actor.

Sumber: (Suharni et al., 2023)

3. *Diagram Class*

Diagram Class adalah gambaran visual dari struktur sistem yang menitikberatkan pada penentuan kelas-kelas yang akan digunakan dalam proses pengembangan sistem. Diagram ini mencakup atribut dan operasi, memungkinkan para pengembang perangkat lunak untuk menetapkan hubungan antara dokumentasi perancangan dan implementasi perangkat lunak (Putra & Andriani, 2019). *Diagram Class* menggambarkan struktur kelas dari sistem perangkat lunak, memberikan pandangan yang jelas terhadap bagaimana kelas-kelas tersebut terkait dan berinteraksi dalam rangkaian pembangunan perangkat lunak. Berikut adalah simbol-simbol yang digunakan dalam *diagram class*.

Tabel 2. 3 Simbol *Diagram Class*

Simbol	Nama	Deskripsi
	Class	Himpunan objek-objek dari berbagai atribut yang me-miliki operasi yang sama.
	Association	Relasi antar kelas dengan makna umum dan biasanya disertai multiplicity.
	Directed Association	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas lain.
	Aggregation	Mengindikasikan keseluruhan bagian relationship disebut sebagai relasi.
	Composition	Relasi Composition terhadap class tempat dia bergantung.
	Dependency	Menunjukkan operasi pada suatu class yang menggunakan class yang lain.

Sumber: (Suharni et al., 2023)

4. Entity Relationship Diagram

Entity Relationship Diagram (ERD) merupakan representasi grafis yang digunakan dalam perancangan basis data untuk menggambarkan hubungan antar data (’Afiifah et al., 2022). Fungsi utama ERD adalah sebagai alat bantu dalam proses pembuatan basis data, serta memberikan pandangan menyeluruh mengenai struktur dan interaksi antara berbagai entitas di dalam sistem yang akan dibuat. Komponen utama pada ERD adalah sebagai berikut:

1. Entitas

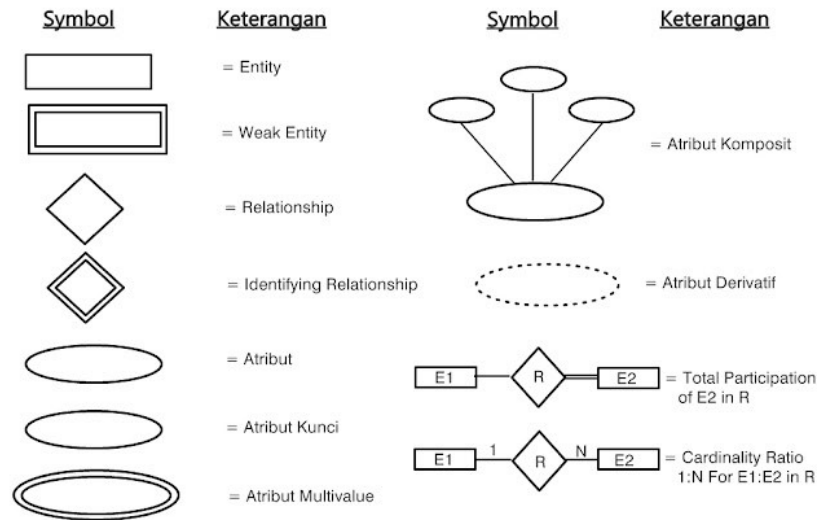
Entitas adalah elemen dasar dalam ERD yang mewakili objek nyata atau konsep yang memiliki data relevan yang akan disimpan dalam basis data. Entitas dapat berupa orang, benda, peristiwa, atau konsep yang penting dalam konteks sistem yang sedang dikembangkan. Dalam diagram, entitas digambarkan sebagai kotak dengan nama entitas di dalamnya.

2. Atribut

Atribut adalah detail atau informasi spesifik yang dimiliki oleh sebuah entitas. Setiap entitas memiliki sejumlah atribut yang mendeskripsikan properti atau karakteristik entitas tersebut. Dalam ERD, atribut digambarkan sebagai oval yang terhubung dengan entitas.

3. Hubungan

Hubungan menggambarkan koneksi atau interaksi antara dua atau lebih entitas. Hubungan ini menunjukkan bagaimana entitas-entitas tersebut berinteraksi satu sama lain dalam konteks sistem. Dalam ERD, hubungan digambarkan dengan garis yang menghubungkan entitas-entitas terkait, sering kali disertai dengan rombus yang mendeskripsikan jenis hubungan.



Gambar 2. 1 Simbol Entity Relationship Diagram

Sumber: (Afiifah et al., 2022)

2.1.5 Network Operation Center

Network Operation Center (NOC) berfungsi sebagai pusat operasional yang mengelola, mengendalikan, dan memantau satu atau lebih jaringan dengan partisipasi administrator jaringan. Misi utama NOC adalah menjamin bahwa operasi jaringan berjalan secara optimal melibatkan beragam platform, media, dan jalur komunikasi. NOC menyajikan representasi visual dari jaringan yang sedang dimonitor, termasuk *workstation* yang memberikan informasi rinci mengenai status jaringan, dan menggunakan perangkat lunak sebagai alat bantu dalam pengelolaan jaringan. (Fauzia et al., 2018).

Fungsi utama dari *Network Operation Center (NOC)* mencakup pemantauan, pengelolaan, dan pemulihan jaringan. Dalam konteks pemantauan, NOC secara aktif memantau kinerja jaringan dengan tujuan mendeteksi masalah potensial atau anomali yang dapat mempengaruhi operasionalnya. Pada aspek pengelolaan, NOC bertanggung jawab untuk mengelola jaringan secara menyeluruh, termasuk konfigurasi, pemeliharaan, dan upaya keamanan. Selain itu, peran NOC melibatkan

kemampuan untuk melakukan pemulihan jaringan dengan cepat dan efisien apabila terjadi kegagalan atau gangguan. Dengan demikian, NOC memiliki peran krusial dalam memastikan kinerja optimal, keamanan, dan ketersediaan jaringan, menjadikannya elemen penting dalam mendukung operasional suatu organisasi atau perusahaan.

2.1.6 Service Operation Processes pada Information Technology Infrastructure Library (ITIL)

Information Technology Infrastructure Library (ITIL) adalah suatu *framework* praktik manajemen layanan IT yang menyediakan panduan terstruktur untuk merancang, mengembangkan, mengimplementasikan, dan mengelola layanan IT. ITIL memberikan serangkaian praktik terbaik yang dapat diadaptasi dan diimplementasikan oleh organisasi untuk meningkatkan kualitas dan efisiensi layanan IT mereka. Salah satu unsur pada ITIL adalah *Service Operation* yang dapat membantu organisasi dalam mengoperasikan layanan IT sehari-hari. Tujuan dari *Service Operation* adalah untuk mengatur dan menjalankan aktivitas serta proses yang diperlukan dalam memberikan dan mengelola layanan sesuai dengan tingkat yang telah disepakati untuk kebutuhan bisnis pengguna dan pelanggan (Office of Government Commerce, 2007). Bagian *Service Operation* juga memiliki tanggung jawab terhadap keberlanjutan pengelolaan teknologi yang digunakan untuk menyampaikan dan mendukung layanan.

Service Operation yang efektif bergantung pada mengetahui status infrastruktur dan mendeteksi penyimpangan dari operasi normal atau yang diharapkan. *Event Management* merupakan kemampuan untuk mengidentifikasi peristiwa, memahaminya, dan menetapkan tindakan kontrol yang sesuai. Oleh karena itu, *Event Management* merupakan landasan untuk Pemantauan dan Pengendalian Operasional (Office of Government Commerce, 2007). Terutama ketika *event* tersebut diatur untuk tujuan komunikasi operasional, memberikan informasi, peringatan,

dan pengecualian, hal ini dapat menjadi dasar untuk otomatisasi berbagai kegiatan rutin dalam Manajemen Operasional. Misalnya, melalui otomatisasi, dapat dilakukan eksekusi skrip pada perangkat jarak jauh, pengiriman pekerjaan untuk diproses, atau bahkan dinamis menyeimbangkan permintaan layanan di berbagai perangkat untuk meningkatkan kinerja.

Perbedaan antara *monitoring* dan *Event management* terletak pada fokus dan cakupannya. *Event management* lebih berorientasi pada penghasilan dan deteksi notifikasi yang signifikan terkait status Infrastruktur dan layanan TI. Meskipun *monitoring* juga terlibat dalam mendeteksi notifikasi ini, ruang lingkupnya lebih luas daripada *Event management*. Sebagai contoh, alat pemantauan dapat memeriksa status perangkat untuk memastikan operasionalitasnya, bahkan jika perangkat tersebut tidak menghasilkan peristiwa. Dengan kata lain, *Event management* fokus pada kejadian spesifik yang dihasilkan untuk dipantau, sementara *monitoring* juga mencakup pencarian aktif terhadap kondisi tanpa perlu adanya kejadian yang terjadi.

2.1.7 Konsep PERN

Konsep PERN merupakan konsep yang terdiri dari beberapa unsur pembentuk sistem informasi. PERN merupakan singkatan dari PostgreSQL, ExpressJS, ReactJS, dan NodeJS. PostgreSQL merupakan suatu sistem basis data yang bersifat *open source*, menonjol dengan kemampuannya yang sangat baik jika dibandingkan dengan database lainnya. Keunggulan PostgreSQL terletak pada fitur-fitur yang sangat komprehensif, menjadikannya pilihan yang sangat mendukung untuk aplikasi basis data baik dengan skala menengah maupun besar (Dwi Praba, 2018). Express.js merupakan sebuah kerangka kerja atau *framework* untuk aplikasi web pada platform Node.js yang dikodekan dengan bahasa pemrograman JavaScript. Fungsinya terletak pada kemampuannya dalam

efisien dan optimal membangun aplikasi dari sisi *backend*. (Lisgiani & Nurmajid, 2022).

ReactJS merupakan sebuah perpustakaan front-end yang diproduksi oleh Facebook, berfungsi sebagai penopang bagi berbagai web-framework. Kelebihan utama ReactJS mencakup kemampuannya dalam menyajikan kecepatan, kesederhanaan, dan skalabilitas. ReactJS memberikan peluang bagi pengembang untuk membangun komponen antarmuka pengguna yang lebih interaktif, memiliki keadaan (stateful), dan dapat digunakan kembali dengan efektif (Panjaitan & Pakpahan, 2021). NodeJS adalah sebuah perangkat lunak yang diciptakan untuk pembangunan aplikasi web, dan pengembangannya dilakukan dengan menggunakan bahasa pemrograman JavaScript (Hasanuddin et al., 2022). Berbeda dengan konsep awal JavaScript yang umumnya berjalan di sisi klien atau *browser*, NodeJS hadir melengkapi peran JavaScript sehingga dapat berfungsi sebagai bahasa pemrograman yang beroperasi di sisi *server*.

2.1.8 Systems Development Life Cycle

Systems Development Life Cycle, atau yang akrab disebut SDLC, adalah metode umum yang lazim digunakan dalam proses pengembangan sistem informasi. (Wahid Abdul, 2020). SDLC melibatkan sejumlah tahapan yang dimulai dari perencanaan, analisis, perancangan, implementasi, hingga pemeliharaan sistem. Prinsip SDLC menjadi dasar dalam berbagai proyek pengembangan sistem informasi, menyediakan struktur kerja untuk perencanaan dan manajemen sistem informasi.

SDLC terdiri dari beberapa fase yang membentuk langkah-langkah esensial dalam pengembangan sistem informasi. Fase perencanaan, sebagai tahap awal, berfokus pada pengumpulan persyaratan sistem dan perencanaan pengembangan. Sementara fase analisis berusaha memahami kebutuhan pengguna dan bisnis melalui wawancara, survei, dan analisis dokumen. Kemudian, fase desain berperan dalam merancang

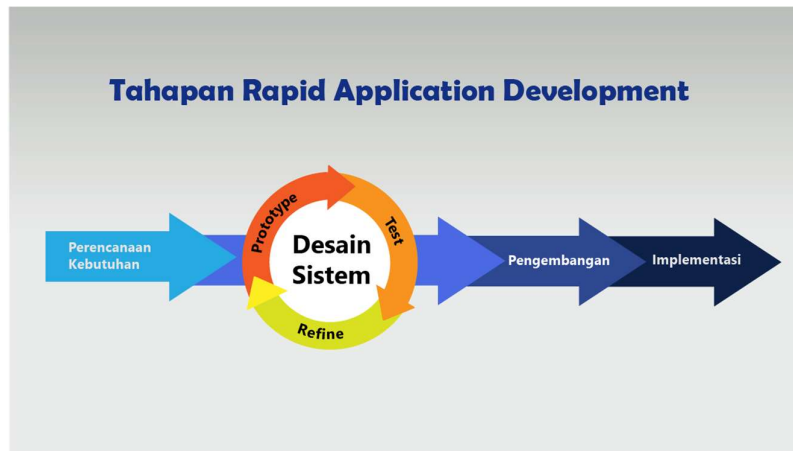
sistem secara teknis, menghasilkan dokumen desain yang menggambarkan arsitektur dan interaksi sistem. Fase implementasi menitikberatkan pada pembangunan dan instalasi sistem, melibatkan aktivitas pengkodean, pengujian, dan konfigurasi. Terakhir, fase operasi dan pemeliharaan menekankan pada pengoperasian dan pemeliharaan sistem, mencakup pelatihan pengguna, dukungan, dan peningkatan sistem. Keseluruhan, fase-fase ini membentuk landasan kokoh dalam siklus hidup pengembangan sistem informasi.

SDLC memiliki sejumlah manfaat yang signifikan dalam pengembangan sistem informasi. Pertama-tama, SDLC membantu memastikan bahwa sistem informasi yang dikembangkan dapat memenuhi kebutuhan pengguna dan kebutuhan bisnis secara efektif. Melalui proses analisis dan perencanaan yang cermat, SDLC juga berperan dalam meningkatkan kualitas keseluruhan dari sistem informasi yang dihasilkan. Selain itu, SDLC berkontribusi dalam mengurangi risiko kegagalan sistem informasi dengan menyediakan kerangka kerja terstruktur untuk pengembangan. Terakhir, dengan mengikuti tahapan-tahapan yang terdefinisi dengan baik, SDLC membantu dalam meningkatkan efisiensi dan efektivitas selama proses pengembangan sistem informasi, menciptakan hasil yang lebih terukur dan dapat diandalkan. Keseluruhan, manfaat-manfaat ini menjadikan SDLC sebagai pendekatan yang penting dan bermanfaat dalam siklus hidup pengembangan sistem informasi.

2.1.9 Metode *Rapid Application Development (RAD)*

Rapid Application Development (RAD) merupakan model proses pengembangan perangkat lunak yang mengikuti pendekatan sekuensial linier dengan penekanan pada siklus pengembangan yang sangat cepat (Priscila & Zulfachmi, 2023). Model RAD ini merupakan adaptasi "kecepatan tinggi" dari model sekuensial linier, di mana pengembangan yang cepat dicapai melalui penerapan pendekatan konstruksi berbasis komponen. Tujuan dari RAD adalah untuk memperpendek waktu yang

biasanya diperlukan dalam siklus hidup pengembangan sistem tradisional, terutama antara tahap perancangan dan implementasi suatu sistem informasi.



Gambar 2. 2 Tahap *Rapid Application Development*

Sumber : (Nurman Hidayat & Kusuma Hati, 2021)

Tahap-tahap dalam pendekatan *Rapid Application Development* (RAD) mencakup perencanaan, desain, pengembangan, dan implementasi. Fase perencanaan merupakan awal dari siklus, di mana persyaratan sistem dikumpulkan dengan cermat dan rencana pengembangan disusun. Setelah itu, tahap desain fokus pada perancangan teknis sistem, menghasilkan dokumen yang menggambarkan arsitektur dan interaksi komponen. Proses pengembangan berlangsung dengan cepat, menerapkan konsep konstruksi berbasis komponen dan memungkinkan pengembang untuk melakukan iterasi pada sistem. Fase implementasi melibatkan pengujian, pengkodean, dan konfigurasi, dan sistem yang telah dikembangkan diintegrasikan ke dalam lingkungan produksi. Keseluruhan, pendekatan RAD memberikan penekanan pada pengembangan yang cepat, responsif, dan adaptif terhadap perubahan kebutuhan pengguna.

Rapid Application Development (RAD) mencirikan dirinya dengan beberapa karakteristik kunci. Pendekatan iteratif dan incremental menjadi ciri khas RAD, di mana pengembangan sistem dilakukan secara bertahap dan dengan penambahan fungsionalitas secara bertingkat. Keterlibatan pengguna menjadi fokus utama, dengan tim pengembang RAD menekankan kerjasama dan umpan balik pengguna selama seluruh siklus pengembangan. Selain itu, RAD memanfaatkan teknologi pemrograman yang cepat untuk mempercepat proses pengembangan, memungkinkan hasil yang responsif dan efisien. Keseluruhan karakteristik-karakteristik ini membuat RAD menjadi pendekatan yang sangat adaptif dan terbuka terhadap perubahan, dengan fokus pada penerapan sistem secara cepat dan berorientasi pada kebutuhan pengguna.

2.1.10 Service Level Agreement (SLA)

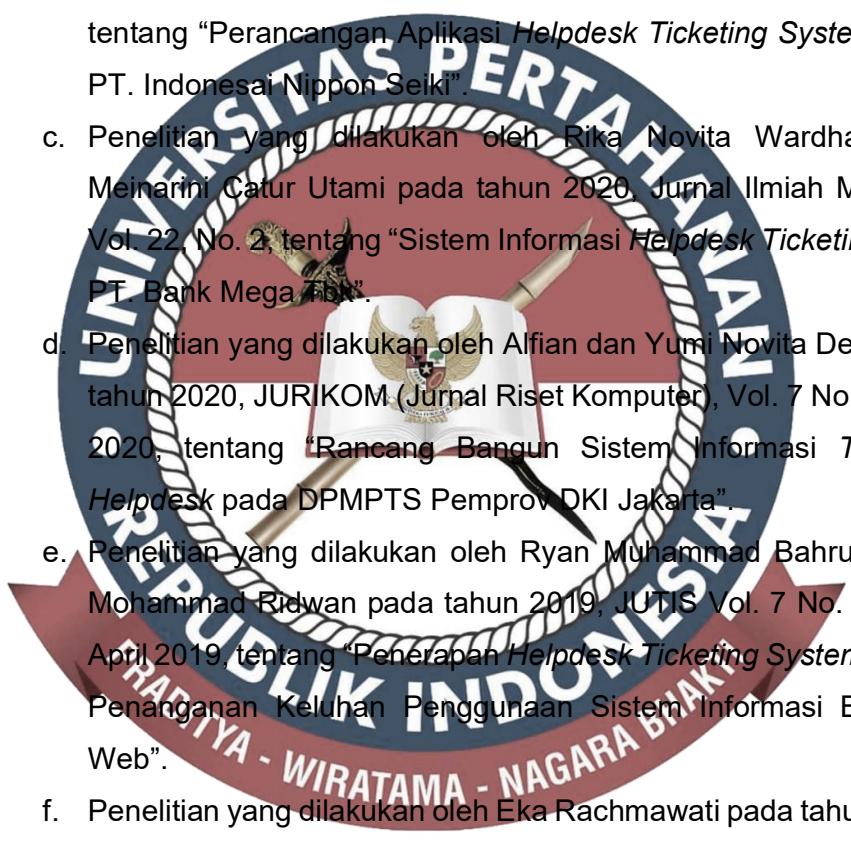
Service Level Agreement (SLA) adalah sebuah pernyataan tertulis dari harapan dan kewajiban yang dalam hubungan bisnis antara dua organisasi, yaitu penyedia layanan dan pelanggan (Supono, 2020). Perjanjian Layanan Level Layanan (SLA) mencakup beberapa aspek penting untuk memastikan bahwa penyedia layanan memberikan layanan yang sesuai dengan kebutuhan dan harapan pelanggan. Pertama, ketersediaan layanan menjadi fokus utama, mengukur persentase waktu di mana layanan tersebut tersedia untuk digunakan oleh pelanggan. Selanjutnya, waktu respons menjadi faktor kunci yang menentukan responsibilitas penyedia layanan dalam menanggapi permintaan atau masalah dari pelanggan dengan cepat dan efisien. Kemudian, waktu penyelesaian menjadi indikator penting untuk mengukur efektivitas dalam menyelesaikan permintaan atau masalah yang muncul dari pelanggan. Selain itu, kualitas layanan juga merupakan aspek yang tidak bisa diabaikan, mencakup sejauh mana layanan tersebut sesuai dengan kebutuhan dan harapan pelanggan. Terakhir, cakupan layanan menentukan lingkup layanan yang disediakan oleh penyedia, memastikan bahwa semua kebutuhan dan area yang diharapkan oleh pelanggan

tertutup dengan baik. Dengan memperhatikan aspek-aspek ini, SLA dapat menjadi instrumen yang kuat dalam memastikan penyedia layanan memberikan layanan yang berkualitas dan responsif kepada pelanggan mereka.

Manfaat dari Perjanjian Layanan Level Layanan (SLA) sangat luas dan berdampak positif pada berbagai aspek dalam hubungan antara penyedia layanan dan pelanggan. Pertama-tama, SLA dapat secara signifikan meningkatkan kepuasan pelanggan dengan memastikan bahwa layanan yang diberikan sesuai dengan harapan mereka dan memenuhi standar kualitas yang telah ditetapkan. Dengan demikian, SLA membantu membangun hubungan yang kuat antara penyedia layanan dan pelanggan, yang merupakan fondasi penting dalam bisnis yang berkelanjutan. Selain itu, SLA juga memberikan manfaat bagi penyedia layanan dengan memberikan kerangka kerja yang jelas untuk mengukur dan meningkatkan kinerja operasional mereka. Dengan memantau kinerja mereka sesuai dengan parameter SLA, penyedia layanan dapat mengidentifikasi area di mana mereka perlu meningkatkan efisiensi dan kualitas layanan mereka. Selanjutnya, SLA juga merupakan alat yang efektif dalam pengelolaan risiko, karena membantu mengidentifikasi, mengukur, dan mengelola risiko yang terkait dengan penyediaan layanan, seperti kerusakan peralatan, gangguan jaringan, atau kesalahan manusia. Dengan demikian, SLA tidak hanya memperkuat hubungan antara penyedia layanan dan pelanggan, tetapi juga memastikan kelancaran operasional dan mengurangi risiko dalam lingkungan bisnis yang kompleks.

2.2 Hasil Penelitian Terdahulu

Dalam penyusunan proposal skripsi ini, penulis menelaah terhadap penelitian-penelitian terdahulu yang berfokus pada manajemen jaringan dan penggunaan teknologi berbasis website dalam manajemen tiket jaringan. Berikut ini penelitian terdahulu yang melandasi penelitian ini sebagai berikut:

- 
- a. Penelitian yang dilakukan oleh Husnil Fadhilah Effendi dan Setiawan Assegaf pada tahun 2022, Jurnal Manajemen Sistem Informasi, Vol. 7, No. 3, tentang “*IT Helpdesk Ticketing System Berbasis Web Pada Bagian ERP (Enterprise Resource Planning) PT Perkebunan Nusantara VI*”.
- b. Penelitian yang dilakukan oleh Rehulina Tarigan dan Imam Kusosi pada tahun 2022, Jurnal SISFOKOM, Vol. 11, No. 1, Hal. 9-18, tentang “*Perancangan Aplikasi Helpdesk Ticketing System Pada PT. Indonesai Nippon Seiki*”.
- c. Penelitian yang dilakukan oleh Rika Novita Wardhani dan Meinarini Catur Utami pada tahun 2020, Jurnal Ilmiah MATRIK, Vol. 22, No. 2, tentang “*Sistem Informasi Helpdesk Ticketing Pada PT. Bank Mega Tbk*”.
- d. Penelitian yang dilakukan oleh Alfian dan Yumi Novita Dewi pada tahun 2020, JURIKOM (Jurnal Riset Komputer), Vol. 7 No. 2, April 2020, tentang “*Rancang Bangun Sistem Informasi Ticketing Helpdesk pada DPMPTS Pemprov DKI Jakarta*”.
- e. Penelitian yang dilakukan oleh Ryan Muhammad Bahrudin dan Mohammad Ridwan pada tahun 2019, JUTIS Vol. 7 No. 1 Bulan April 2019, tentang “*Peherapan Helpdesk Ticketing System Dalam Penanganan Keluhan Penggunaan Sistem Informasi Berbasis Web*”.
- f. Penelitian yang dilakukan oleh Eka Rachmawati pada tahun 2018, *International Journal of Computer Science and Mobile Computing*, Vol. 7, Issue. 12, Hal. 29-41, tentang “*Web-Based Ticketing System Helpdesk Application Using CodeIgniter Framework (Case Study: PT Commonwealth Life)*”.

Berikut merupakan penelitian terdahulu yang melandasi penelitian ini, dapat dilihat pada tabel dibawah ini.

Tabel 2. 4 Penelitian Terdahulu

No	Judul Penelitian	Metode	Kesimpulan	Persamaan / Perbedaan
1	<i>IT Helpdesk Ticketing System</i> Berbasis Web Pada Bagian ERP (<i>Enterprise Resource Planning</i>) PT Perkebunan Nusantara VI (Fadillah Effendi & Assegaff, 2022)	Membangun aplikasi <i>ticketing system</i> berbasis web menggunakan model <i>prototype</i> .	 - Perancangan <i>Helpdesk Ticketing System</i> bertujuan memudahkan setiap unit menyampaikan kendala dalam penggunaan sistem SAP. - Aplikasi ini mempermudah <i>key user</i> PT Perkebunan Nusantara VI dalam mengelola pengaduan terstruktur dari setiap unit usaha dan bagian. - Analisis masalah di unit usaha dan bagian PT	Persamaan: Membangun aplikasi berbasis <i>website</i>. Metode pengumpulan data yang sama yaitu studi literatur, wawancara, dan observasi. Perbedaan: Pada penelitian penulis, melakukan analisa lanjutan terhadap data yang diinput oleh <i>user</i> sehingga ada pembelajaran pada teknisi terhadap aset jaringan (<i>Executive Summary Dashboard</i>). Pada penelitian penulis juga

No	Judul Penelitian	Metode	Kesimpulan	Persamaan / Perbedaan
			Perkebunan Nusantara VI lebih efisien dengan data terintegrasi dalam sistem.	menggunakan konsep PERN (Postgresql, Express, React, NodeJs) yang pada penelitian ini belum dilakukan. Menggunakan Metode <i>Rapid Application Development (RAD)</i> .
2	Perancangan Aplikasi <i>Helpdesk Ticketing System</i> Pada PT. Indonesia Nippon Seiki (Tarigan et al., 2022)	Membangun aplikasi <i>ticketing system</i> berbasis web menggunakan model <i>prototype</i> .	- Aplikasi <i>Helpdesk Ticketing</i> memperbaiki layanan dengan memungkinkan pengguna dan pimpinan IT memonitor secara <i>real-time</i> status penyelesaian tiket oleh teknisi. - Peningkatan kinerja teknisi dan pembagian tugas yang merata terjadi karena pimpinan departemen IT yang menetapkan tugas	Persamaan: Membangun aplikasi berbasis <i>website</i>. Mengumpulkan data dengan cara studi literatur, observasi, dan wawancara. Perbedaan: Pada penelitian penulis, melakukan analisa lanjutan terhadap data yang diinput oleh <i>user</i> sehingga ada pembelajaran pada teknisi terhadap aset jaringan (<i>Executive Summary Dashboard</i>). Pada

No	Judul Penelitian	Metode	Kesimpulan	Persamaan / Perbedaan
			untuk setiap tiket melalui sistem.	penelitian penulis juga menggunakan konsep PERN (Postgresql, Express, React, NodeJs) yang pada penelitian ini belum dilakukan. Menggunakan Metode <i>Rapid Application Development (RAD)</i> .
3	Sistem Informasi <i>Helpdesk Ticketing</i> Pada PT. Bank Mega Tbk (Wardhani et al., 2020)	Membangun aplikasi <i>ticketing system</i> berbasis web menggunakan model <i>Rapid Application Development (RAD)</i> .	Pencatatan masalah dilakukan secara sistematis tanpa duplikasi data. Dengan <i>Helpdesk Ticketing System</i> kepala divisi IT dapat efisien melihat laporan bulanan masalah dari unit lain termasuk koneksi internet dan kerusakan <i>hardware</i> .	Persamaan: Menggunakan metode yang sama yaitu <i>Rapid Application Development (RAD)</i> dan membangun aplikasi berbasis <i>website</i> . Perbedaan: Pada penelitian penulis, melakukan analisa lanjutan terhadap data yang diinput oleh <i>user</i> sehingga ada pembelajaran pada teknisi

No	Judul Penelitian	Metode	Kesimpulan	Persamaan / Perbedaan
			- Proses perbaikan keluhan unit dan penanganan kerusakan <i>hardware</i> terkontrol dan terkoordinasi dengan baik. - Sistem <i>Helpdesk Ticketing</i> mempermudah staff <i>helpdesk</i> dalam membuat laporan masalah, memproses data, dan menyimpan informasi.	terhadap aset jaringan (<i>Executive Summary Dashboard</i>). Pada penelitian penulis juga menggunakan konsep PERN (Postgresql, Express, React, NodeJs) yang pada penelitian ini belum dilakukan.
4	Rancang Bangun Sistem Informasi <i>Ticketing Helpdesk</i> pada DPMPTS Pemprov DKI Jakarta (Alfian et al., 2020)	Membangun aplikasi <i>ticketing</i> system berbasis web menggunakan model <i>prototype</i> .	- Penggunaan manual Sistem Informasi Pelaksana Teknologi dan Informasi (PSTIK), Kearsipan terutama dalam <i>Ticketing Helpdesk</i>, berpotensi menyebabkan risiko. Oleh	Persamaan: Membangun aplikasi berbasis <i>website</i>. Mengumpulkan data dengan cara studi literatur dan observasi. Perbedaan: Pada penelitian penulis, melakukan analisa

No	Judul Penelitian	Metode	Kesimpulan	Persamaan / Perbedaan
			karena itu, dibuatnya sistem informasi <i>Ticketing Helpdesk</i> bertujuan meningkatkan efisiensi operasional dan kualitas proses bisnis di bidang PSTIK. Sistem ini diharapkan dapat memberikan bantuan yang lebih baik, mendukung peningkatan kinerja, dan menyajikan informasi yang akurat.	lanjutan terhadap data yang diinput oleh <i>user</i> sehingga ada pembelajaran pada teknisi terhadap aset jaringan (<i>Executive Summary Dashboard</i>). Pada penelitian penulis juga menggunakan konsep PERN (Postgresql, Express, React, NodeJs) yang pada penelitian ini belum dilakukan.
5	Penerapan <i>Helpdesk Ticketing System</i> Dalam Penanganan Keluhan Penggunaan Sistem Informasi Berbasis Web	Membangun aplikasi <i>ticketing system</i> berbasis web menggunakan	- Aplikasi <i>Helpdesk Ticketing</i> memudahkan admin verifikasi tiket sesuai kategori, membantu teknisi	Persamaan: Membangun aplikasi berbasis website. Mengumpulkan data dengan cara studi literatur, observasi, dan wawancara.

No	Judul Penelitian	Metode	Kesimpulan	Persamaan / Perbedaan
	(Bahrudin et al., 2019)	model <i>prototype</i> .	IT memberikan solusi dengan lebih efisien. - Aplikasi <i>Helpdesk Ticketing</i> membantu teknisi dalam menanggapi laporan-laporan yang diajukan oleh pengguna dan memberikan solusi. - Aplikasi <i>Helpdesk</i> memfasilitasi pelaporan bagi pengguna tanpa perlu menghadap langsung kepada teknisi.	Perbedaan: Pada penelitian penulis, melakukan analisa lanjutan terhadap data yang diinput oleh <i>user</i> sehingga ada pembelajaran pada teknisi terhadap aset jaringan (<i>Executive Summary Dashboard</i>). Pada penelitian penulis juga menggunakan konsep PERN (Postgresql, Express, React, NodeJs) yang pada penelitian ini belum dilakukan.
6	<i>Web-Based Ticketing System Helpdesk Application Using CodeIgniter Framework</i>	Membangun aplikasi <i>ticketing</i> system berbasis web menggunakan	- Aplikasi ini mempermudah penanganan keluhan pengguna terkait perbaikan data dan kerusakan	Persamaan: Membangun aplikasi berbasis <i>website</i>. Metode pengumpulan data dengan studi literatur.

No	Judul Penelitian	Metode	Kesimpulan	Persamaan / Perbedaan
	(Case Study: PT Commonwealth Life) (Rachmawati, 2018)	model <i>waterfall</i> dan metode analisa SWOT.	perangkat keras oleh Departemen IT. - Aplikasi ini mampu mencatat masalah secara sistematis, sehingga data yang tercatat tetap terjaga dan tidak hilang. - Aplikasi ini dapat digunakan sebagai monitoring dalam penanganan permasalahan.	Perbedaan: Pada penelitian penulis, melakukan analisa lanjutan terhadap data yang diinput oleh <i>user</i> sehingga ada pembelajaran pada teknisi terhadap aset jaringan (<i>Executive Summary Dashboard</i>). Pada penelitian penulis juga menggunakan konsep PERN (Postgresql, Express, React, NodeJs) yang pada penelitian ini belum dilakukan. Menggunakan Metode <i>Rapid Application Development (RAD)</i>.

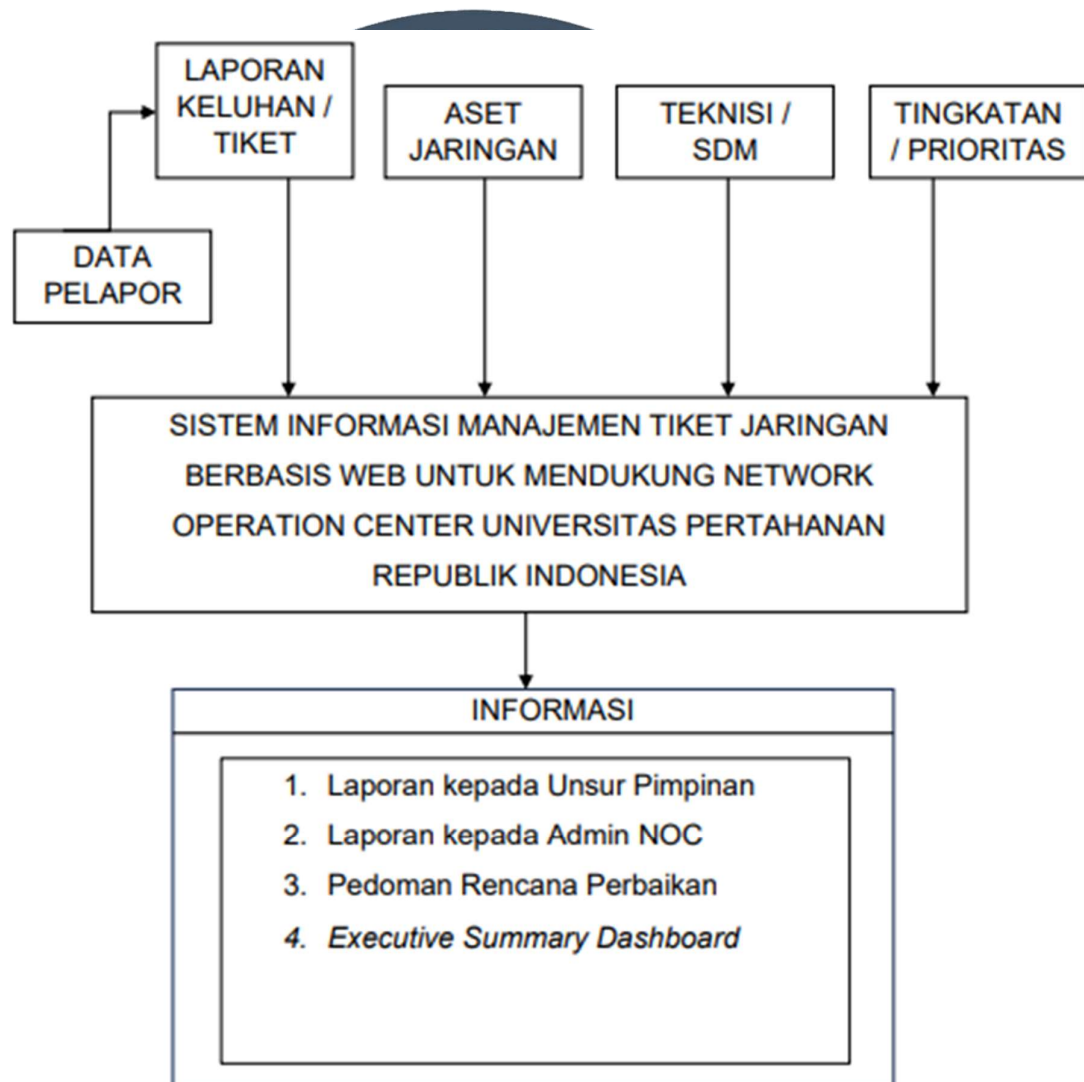
Sumber: Diolah oleh Penulis

Pada Studi literatur penelitian terdahulu, terdapat beberapa kesamaan unsur dari penelitian yang akan dilaksanakan oleh penulis seperti metode pengumpulan data dan identifikasi masalah. Namun, terdapat perbedaan yang signifikan yang penulis bangun pada proyek penelitian ini yaitu adanya *Executive Summary Dashboard* yang akan menampilkan hasil dari analisa data-data keluhan yang dialami pengguna yang berdampak pada aset jaringan yang terkait. Nantinya analisa yang ada pada *Executive Summary Dashboard* akan menjadi pertimbangan oleh teknisi ataupun pengambil Keputusan untuk melakukan perbaikan ataupun pengembangan terhadap sistem dan aset jaringan yang terkait.



2.3 Kerangka Berpikir

Pada subbab ini akan dijabarkan kerangka berpikir dari proyek penelitian ini yang terdiri dari beberapa unsur atau variabel yang terlibat. Variabel yang terlibat seperti data pelapor, laporan keluhan atau tiket, aset jaringan, teknisi, dan tingkatan atau prioritas. Variabel ini akan diproses pada proyek penelitian ini dan akan menghasilkan output berupa informasi untuk memenuhi tujuan penelitian ini.



Gambar 2. 3 Kerangka Berpikir

Sumber: Diolah oleh Penulis