BAB II
TINJAUAN PUSTAKA

2.1 Penelitian yang Relevan

Penelitian yang sudah dipublikasikan secara jurnal atau penelitian sains digunakan untuk memberikan perspektif agar terlihat sistem yang dulu diimplementasikan dan pengembangan sistem saat ini yang dilakukan untuk manajemen karyawan pada perusahaan PT Neurogs Inovasi Teknologi.

Penelitian terdahulu menunjukan bahwa penggunaan pendekatan visual seperti kanban board dan gantt chart dapat meningkatkan koordinasi dan produktivitasi masing masing personil tim dalam mengerjakan job desk nya (Rahman et al., 2022) dan (Hidayat & Prasetyo, 2021). Dalam rangka menilai kinerja karyawan diperlukan suatu modul yang mencatat seluruh proses dan riwayat pekerjaan agar masing masing personil tim dalam suatu proyek dapat dievaluasi baik secara individu atau secara tim agar performance manajemen dapat terjaga dan termonitor dengan baik seperti penelitian yang dilakukan oleh (Armstrong, 2021) dan (Susanto et al., 2020).

Penelitian yang dilakukan oleh (Nugroho & Sari, 2021) adalah sebuah penelitian yang menginspirasi guna untuk melakukan standarisasi absensi di perusahaan sehingga kedisiplinan karyawan dan kehadiran serta ketepatan waktu hadir di perusahaan oleh karyawan semuanya dapat tercatat dan dapat direkapitulasi untuk penilaian KPI masing masing karyawan sebagai bahan evaluasi oleh tim *human resource* untuk pertimbangan kenaikan gaji dan bonus karyawan. Penelitian dari (Wibowo, 2022) mengindikasikan bahwa sistem yang bagus dan terkelola dengan baik dapat melakukan integrasi modul penilaian karyawan untuk efisiensi dan apresiasi masing masing karyawan sehingga operasional *human resource* dapat diintegrasikan dengan sistem yang sudah dimiliki oleh perusahaan atau ingin diintegrasikan di masa mendatang.

Dalam rangka meningkatkan transparansi dan akuntabilitas sebuah perusahaan, diperlukan sistem yang memiliki tingkatan hierarki dan persetujuan dari masing masing supervisor untuk menjaga pengeluaran anggaran perusahaan dan pengambilan keputusan yang bijak sehingga dapat dihasilkan kesimpulan yang matang dan professional guna keberlangsungan perusahaan secara kebijakan ketenagakerjaan seperti yang dijelaskan oleh (Pratama & Kurniawan, 2021). Penelitian yang dipublikasikan oleh (Sandhu et al., 2020) menjelaskan terkait otorisasi sistem dengan pengelolaan hak akses demi

meningkatkan keamanan data didalam sistem, mengingat sistem yang digunakan oleh perusahaan memiliki data data sensitif karyawan seperti NIK, penghasilan dan NPWP serta pelaporan pajak seperti pph 21 sehingga sistem tersebut harus menggunakan *Role-Based Accesss Control* agar sesuai dengan kebijakan perusahaan yang sudah terstandarisasi mengenai keamanan data perusahaan.

Proses perancangan sistem yang dilakukan sesuai dengan penelitian RESTful API oleh (Fielding, 2020) yang memberikan penjelasan terkait interoperabilitas sistem dalam pertukaran data dan skalabilitas ketika sistem digunakan secara masif pada kurun waktu tertentu seperti awal dan akhir bulan serta pengembangan aplikasi dimana mendatang ketika perusahaan membutuhkan fitur lainnya. Dalam proses perancangan sistem juga ada proses yang perlu dilakukan yaitu pengujian perangkat lunak atau yang dikenal dengan istilah teknis *Software Testing* guna memastikan sistem yang diimplementasi ini memenuhi standarisasi keamanan data dan meningkatkan keamanan sistem berdasarkan mekanisme kontrol akses serta pengujian tersebut dilakukan secara *Static Application Security Testing (SAST)* dan *qualitative analysis (QA).*

Pendekatan SSDLC yang digunakan untuk aspek keamanan dalam setiap tahapan pengembangan aplikasi mulai dari tahap analisis, perancangan, implementasi sampai ke tahap pengujian menggunakan penerapan SSDLC agar potensi kerentanan keamanan bisa diidentifikasi lebih dahulu agar resiko eksploitasi pada code dan sistem yang telah digunakan secara operasional bisa diminimalisir (McGraw, 2018)

Penelitian terdahulu oleh Tiarto, Rizky Dwi dan kawan-kawan (2024) mengkaji deteksi kerentanan keamanan pada website dengan menggunakan metode penetration testing berbasis ISSAF Framework. Penelitian ini fokus pada identifikasi berbagai celah keamanan yang dapat dimanfaatkan oleh peretas, serta memberikan pemahaman tentang pentingnya penetration testing sebagai salah satu tahap krusial dalam pengujian keamanan sistem. Temuan dalam penelitian tersebut menegaskan bahwa penetration testing efektif dalam mengungkap kerentanan yang tersembunyi sehingga dapat dijadikan bagian integral dari proses Secure Software Development Lifecycle (SSDLC), khususnya pada fase security testing untuk meningkatkan keamanan aplikasi secara menyeluruh.

Penelitian terdahulu dari Mahdy, Prayitno, dan Rifai (2024) meneliti kerentanan keamanan pada platform aplikasi web dengan menggunakan *Open Web Application Security Project Risk Rating Methodology*. Penelitian ini menganalisis tingkat risiko

keamanan berdasarkan *OWASP Top 10* terhadap tiga platform berbeda, yaitu WordPress, PHP Native, dan Laravel, dan mengukur skor risiko dari masing-masing platform serta merekomendasikan perbaikan prioritas pada kelemahan kritis untuk mengurangi kerentanan aplikasi web secara umum .

Selanjutnya, penelitian oleh Antony, Fitriani, dan Sudirman (2024) pada website PT Intercloud Digital Inovasi menunjukkan bahwa meskipun perusahaan telah menerapkan sistem keamanan, masih terdapat celah keamanan seperti SQL Injection, Brute Force, dan DDoS yang dapat dimanfaatkan oleh pihak tidak bertanggung jawab. Penelitian ini menggunakan metode *Footprinting* dan *Vulnerability Scanning* untuk memetakan kelemahan sistem dan menyatakan perlunya upaya yang lebih sistematis untuk menanggulangi ancaman keamanan siber guna meningkatkan kepercayaan pengguna .

Penelitian lain yang relevan adalah studi mengenai keamanan *Content Management System (CMS)* berbasis WordPress yang dilakukan oleh Fadhlurrohman, Sudirman, dan Putra (2024). Penelitian ini menggunakan pendekatan uji penetrasi sesuai panduan OWASP Web Security Test Guide dan menemukan sejumlah kerentanan penting yang berpotensi dieksploitasi, serta memberikan rekomendasi perbaikan konfigurasi keamanan untuk meminimalkan risiko eksploitasi pada situs berbasis CMS .

Selain itu, Pratama (2024) meneliti implementasi *JSON Web Token (JWT)* dalam pengembangan aplikasi web untuk *Personal Assessment Website* di PT Engineering Career Center. Penelitian ini menggarisbawahi pentingnya penggunaan JWT sebagai standar autentikasi yang aman dalam aplikasi web modern dan menunjukkan bagaimana implementasi JWT dapat meningkatkan keamanan serta pengelolaan data pengguna secara profesional dalam konteks pengembangan sistem informasi berbasis web .

Dari hasil telaah referensi tersebut, penelitian yang sudah dipublikasikan banyak menekankan pada identifikasi dan evaluasi kerentanan serta rekomendasi perbaikan keamanan pada aplikasi web, baik melalui risk rating, footprinting, scanning, maupun uji penetrasi. Namun, sebagian besar penelitian tersebut masih fokus pada aspek pengujian atau satu teknik keamanan tertentu, seperti pengujian kerentanan atau implementasi JWT pada konteks tertentu, tanpa mengintegrasikan seluruh aspek keamanan sejak tahapan awal pengembangan sistem secara menyeluruh dan terpusat.

Metode pengujian keamanan menggunakan Static Application Security Testing ini dilakukan dengan analisis *source code* secara statis untuk mendeteksi potensi kerentanan sistem sebelum sistem digunakan secara *production,* hal ini dilakukan untuk menunjukan bahwa penerapan SAST ini dapat memberitahukan *developer* bahwa ada kekurangan dan kesalahan pada proses implementasi keamanan, hal ini dilakukan dari awal tahap pengembangan hingga tahap akhir *production* untuk meningkatkan kualitas keamanan aplikasi sebelum akhirnya dipublish (Shahriar & Zulkernine, 2020).

Di sisi lain, aspek keamanan sistem sejak tahap awal pengembangan melalui pendekatan Secure Software Development Life Cycle (SSDLC), serta penggunaan pengujian keamanan statis sebagai bagian dari proses evaluasi aplikasi, masih jarang dibahas secara komprehensif pada penelitian sebelumnya.

Berdasarkan hasil terhadap penelitian terdahulu, dapat disimpulkan bahwa pengembangan sistem manajemen karyawan dan manajemen proyek berbasis web telah banyak dilakukan dengan pendekatan pengembangan perangkat lunak yang beragam, penggunaan kanban dan gantt chart untuk manajemen tugas, sistem absensi digital serta modul KPI. Penelitian lain juga membahas aspek keamanan melalui penerapan RBAC serta arsitektur RESTful API untuk mendukung skalabilitas dan interoperabilitas sistem. Namun sebagian besar penelitian tersebut masih berfokus pada modul atau fungsi tertentu secara terpisah tanpa mengintegrasikan seluruh proses operasional karyawan dalam satu sistem terpusat. Selain itu, penelitian terdahulu tidak membahas implementasi sistem sebagai pengganti langsung platform berbasis subscription serta belum menekankan aspek otomatisasi KPI yang terintegrasi langsung dengan absensi, aktivitas proyek dan audit log sistem. Di sisi lain, aspek keamanan sistem sejak tahap awal pengembangan dengan SSDLC serta penerapan sistem secara on-premises untuk kontrol penuh terhadap data dan biaya operasional perusahaan masih jarang dibahas secara komprehensif pada penelitian sebelumnya.

Novelty pada penelitian ini terletak pada pengembangan sistem manajemen karyawan terintegrasi berbasis web yang tidak hanya mengotomatisasi perhitungan dan evaluasi Key Performance Indicator (KPI) berdasarkan data operasional aktual, seperti absensi, aktivitas proyek, dan audit log, tetapi juga dirancang dengan mempertimbangkan aspek keamanan aplikasi sejak tahap awal pengembangan. Sistem dikembangkan menggunakan model Waterfall yang dilaksanakan secara bertahap dan terstruktur,

meliputi analisis kebutuhan, perancangan, implementasi, pengujian, dan pemeliharaan, dengan tetap mengintegrasikan prinsip Secure Software Development Life Cycle (SSDLC) pada setiap fase pengembangan. Penerapan pengaturan hak akses berbasis peran atau Role-Based Access Control (RBAC) digunakan sebagai mekanisme utama dalam proses autentikasi dan otorisasi pengguna. Selain itu, sistem diimplementasikan secara on-premises sebagai alternatif terhadap platform Software as a Service (SaaS) berbasis subscription, sehingga perusahaan memiliki kendali yang lebih baik terhadap keamanan data, fleksibilitas pengembangan, serta pengelolaan operasional sistem sesuai dengan kebutuhan internal.

2.2 Sistem Informasi Manajemen

Sistem Informasi Manajemen merupakan sistem terintegrasi yang digunakan untuk mendukung pengelolaan data, proses operasional, serta pengambilan keputusan dalam organisasi. Sistem ini berperan dalam mengubah data menjadi informasi yang terstruktur sehingga mendukung efektivitas dan efisiensi operasional perusahaan (Laudon & Laudon, 2020).

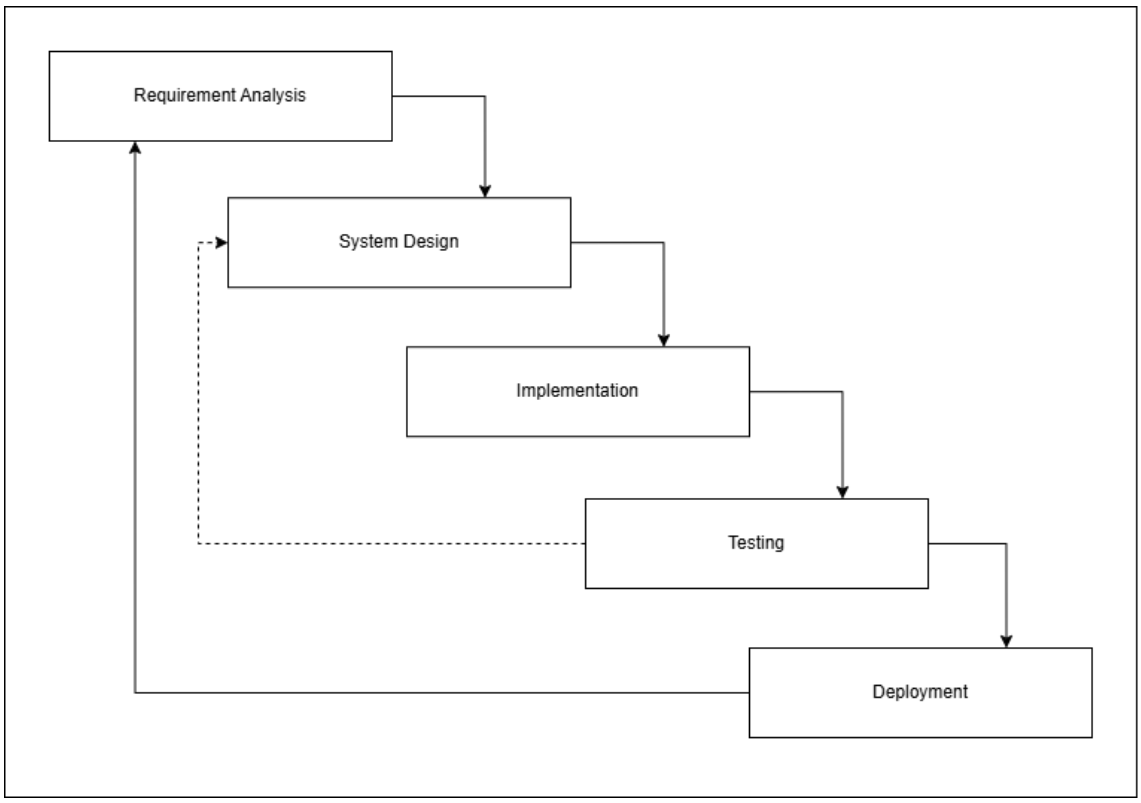
Dalam konteks penelitian ini, sistem manajemen karyawan digunakan untuk mengelola data proyek, kinerja (KPI), absensi, dan reimbursement secara terpusat sehingga proses administrasi dan pemantauan kinerja dapat terdokumentasi secara sistematis.

2.3 Key Performance Indicator (KPI)

Key Performance Indicator (KPI) merupakan indikator terukur yang digunakan untuk mengevaluasi tingkat pencapaian kinerja individu maupun organisasi terhadap target yang telah ditetapkan. KPI membantu organisasi dalam melakukan pengukuran performa secara objektif dan terstruktur (Parmenter, 2020).

Pada penelitian ini, KPI digunakan sebagai komponen evaluasi kinerja karyawan yang terintegrasi dengan data absensi dan penyelesaian tugas proyek dalam sistem.

2.4 Metode Waterfall



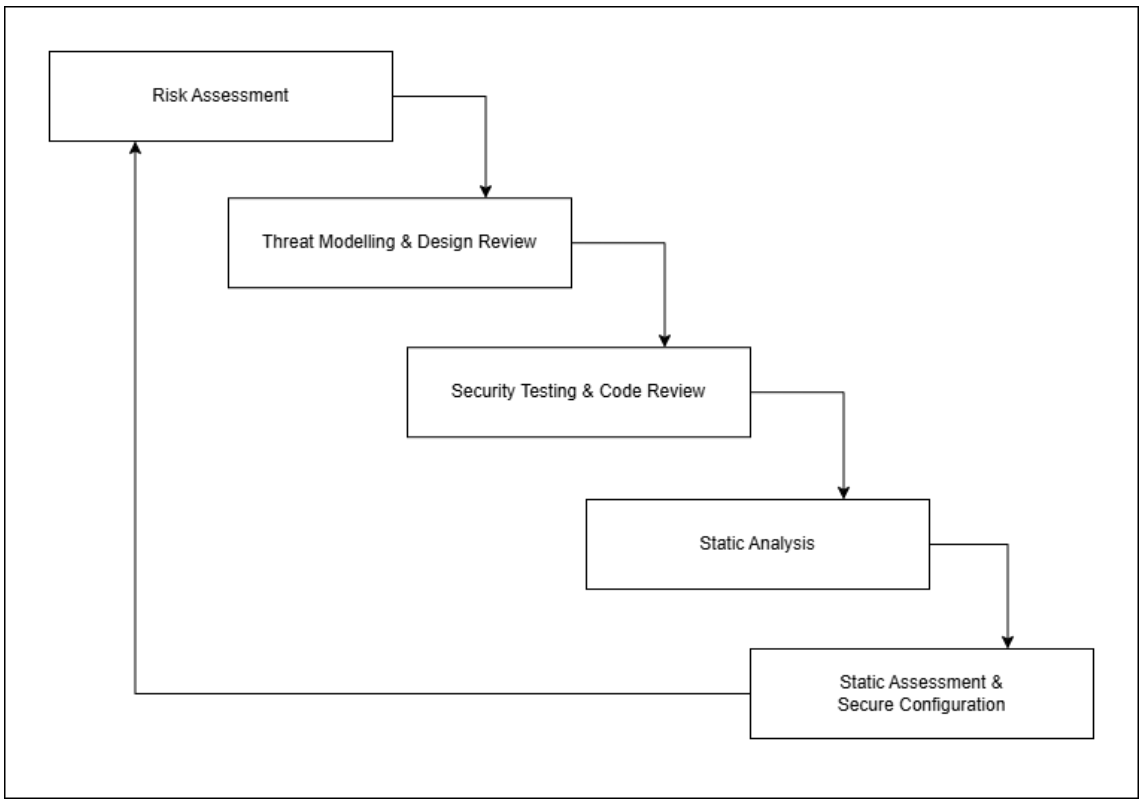
Gambar 2. 1 Waterfall Diagram

Model Waterfall merupakan model pengembangan perangkat lunak yang bersifat linear dan sekuensial, di mana setiap tahapan diselesaikan terlebih dahulu sebelum melanjutkan ke tahap berikutnya. Model ini masih relevan untuk proyek dengan kebutuhan yang terstruktur dan dokumentasi formal yang jelas (Alshamrani & Bahattab, 2021; Suryanto et al., 2022; Rahman et al., 2023).

Tahapan utama dalam model Waterfall meliputi Requirement Analysis, System Design, Implementation, Testing, serta Deployment and Maintenance. Setiap fase menghasilkan dokumentasi yang menjadi dasar bagi tahap berikutnya.

Dalam penelitian ini, model Waterfall digunakan sebagai kerangka tahapan pengembangan sistem, yang kemudian diperkaya dengan penerapan kontrol keamanan melalui pendekatan Secure Software Development Life Cycle (SSDLC).

2.5 Secure Software Development Life Cycle (SSDLC) Berbasis Waterfall



Gambar 2. 2 SSDLC Berbasis Waterfall Diagram

Secure Software Development Life Cycle (SSDLC) merupakan pendekatan pengembangan perangkat lunak yang mengintegrasikan aspek keamanan pada setiap tahapan siklus pengembangan sistem. Pendekatan ini menekankan bahwa pengendalian risiko keamanan harus dilakukan sejak tahap awal perancangan hingga tahap pengujian dan pemeliharaan sistem (McGraw, 2018).

Menurut Almorsy, Grundy, dan Müller (2020), penerapan keamanan secara sistematis pada seluruh tahapan pengembangan mampu mengurangi potensi kerentanan secara signifikan. Penelitian lain menunjukkan bahwa integrasi keamanan sejak tahap awal dapat menurunkan risiko eksploitasi serta mengurangi biaya perbaikan dibandingkan pendekatan yang hanya melakukan pengujian di akhir proses pengembangan (Khan et al., 2021).

Dalam penelitian ini, SSDLC diterapkan dengan model tahapan berbasis Waterfall, sehingga setiap fase pengembangan—Requirement Analysis, System Design,

Implementation, Testing, serta Deployment and Maintenance—mengandung kontrol keamanan yang terstruktur. Integrasi keamanan dilakukan melalui:

- Identifikasi kebutuhan autentikasi dan otorisasi pada tahap analisis kebutuhan.
- Perancangan mekanisme kontrol akses berbasis peran (Role-Based Access Control / RBAC) serta arsitektur autentikasi berbasis JSON Web Token (JWT) pada tahap desain sistem.
- Penerapan praktik secure coding dan pengamanan konstruksi query pada tahap implementasi.
- Pengujian keamanan menggunakan Static Application Security Testing (SAST) pada tahap testing sebelum sistem dioperasikan.

Dengan pendekatan ini, keamanan tidak hanya diposisikan sebagai tahap tambahan di akhir pengembangan, melainkan menjadi bagian integral dalam setiap fase pengembangan sistem.

2.6 Arsitektur Sistem Informasi

2.6.1 Arsitektur MVC

Arsitektur Model-View-Controller (MVC) merupakan pola perancangan perangkat lunak yang memisahkan sistem menjadi tiga komponen utama, yaitu model sebagai pengelola data, view sebagai antarmuka pengguna, dan controller sebagai pengatur logika aplikasi. Pemisahan ini bertujuan untuk meningkatkan modularitas, maintainability, serta kemudahan pengembangan sistem (Necula, 2024).

Dalam penelitian ini, arsitektur MVC digunakan untuk mendukung struktur pengembangan yang terorganisir dan mempermudah penerapan kontrol keamanan pada lapisan logika aplikasi.

2.6.2 RESTful API

RESTful API merupakan arsitektur layanan web yang memungkinkan komunikasi antara client dan server melalui protokol HTTP dengan pendekatan stateless. Arsitektur ini menekankan interaksi berbasis resource serta efisiensi pertukaran data (Fielding, 2000).

Pada penelitian ini, RESTful API digunakan untuk menghubungkan frontend dan backend sistem serta mendukung implementasi autentikasi berbasis JSON Web Token (JWT) dan kontrol akses RBAC pada setiap endpoint layanan.

2.7 Keamanan Sistem Informasi

2.7.1 Role-Based Access Control (RBAC)

Untuk keamanan sistem akan diterapkan *Role-Based Access Control (RBAC)* agar sistem digunakan hanya oleh personil yang memiliki tanggung jawab terhadap penggunaan sistem yang diimplementasikan sehingga tidak dapat dimanipulasi oleh karyawan dalam manajerial perusahaan. Menurut (Sandhu, 2020) sistem RBAC diciptakan agar keamanan informasi didalam sistem terjaga dan hanya dapat diakses oleh personil yang memiliki *authorization* yang diberikan oleh *super admin* guna menjaga keamanan data dan dapat melakukan pembuatan data baru untuk kandidat karyawan yang akan masuk dan penghapusan data lama untuk karyawan yang sudah *resign* atau mengundurkan diri dari perusahaan. secara umum *RBAC* memiliki beberapa elemen seperti *User* yaitu pengguna sistem aplikasi tersebut, *Role* yang dimana memiliki tanggung jawab pada strukturisasi jabatan kantor, *Permission* dimana memiliki akses ke seluruh sistem, sebagian atau hanya beberapa fitur tertentu serta *session* dimana menghubungkan antara *user* dan *role* dalam proses penggunaan aplikasi

2.7.2 JSON Web Token (JWT)

JSON Web Token (JWT) adalah sebuah format token berbasis JSON yang digunakan sebagai mekanisme otentikasi dengan sifat *stateless* dimana JWT merupakan sebuah penyampaian klaim antar sistem bahwa ada request ke server dan harus dijawab dan diberikan datanya oleh sistem akan tetapi perlu adanya token sehingga data tersebut hanya bisa diakses oleh otoritas pengguna yang terdaftar (Fielding, 2020).

2.7.3 Static Application Security Testing (SAST)

Static Application Security Testing (SAST) adalah metode pengujian sistem aplikasi dengan menganalisis *source code* aplikasi secara statis tanpa menjalankan aplikasi tersebut. Testing SAST ini bertujuan untuk mengidentifikasi potensi kerentanan

keamanan, kesalahan implementasi dan coding yang tidak sesuai dengan standar keamanan dari awal tahap pengembangan. SAST lebih efektif digunakan sebagai bagian dari awal proses pengembangan aplikasi karena mampu membantu pengembang dalam mendeteksi kerentanan lebih awal sebelum sistem dirilis *production* (Shahriar dan Zulkernine, 2020).

2.8 Prinsip Dasar Keamanan Informasi

Keamanan informasi merupakan aspek fundamental dalam pengembangan sistem informasi, khususnya pada sistem yang mengelola data organisasi. ISO/IEC 27001:2022 menjelaskan bahwa pengelolaan keamanan informasi bertujuan untuk melindungi informasi dari ancaman guna menjaga keberlangsungan operasional organisasi (ISO/IEC, 2022).

Konsep dasar keamanan informasi dikenal dengan Confidentiality, Integrity, dan Availability (CIA). Menurut Alharbi dan Tassaddiq (2021), ketiga prinsip tersebut menjadi fondasi dalam perancangan sistem yang aman, terutama pada aplikasi berbasis web.

- Confidentiality (kerahasiaan) berkaitan dengan pembatasan akses terhadap informasi agar hanya dapat diakses oleh pihak yang berwenang. Penerapan autentikasi dan otorisasi dalam sistem merupakan bentuk implementasi dari prinsip ini.
- Integrity (integritas) mengacu pada upaya menjaga keutuhan dan konsistensi data agar tidak mengalami perubahan tanpa izin. Mekanisme kontrol akses, validasi input, serta pencatatan aktivitas sistem berperan dalam menjaga integritas data yang dikelola.
- Availability (ketersediaan) memastikan bahwa sistem dan informasi dapat diakses oleh pengguna yang sah ketika dibutuhkan. Perancangan arsitektur sistem yang stabil serta pengelolaan infrastruktur yang baik mendukung terjaganya ketersediaan layanan.

Dalam penelitian ini, prinsip CIA menjadi dasar dalam penerapan autentikasi berbasis JWT, pengelolaan hak akses menggunakan Role-Based Access Control (RBAC), serta pencatatan audit log untuk mendukung keamanan sistem secara menyeluruh.

2.10 Integrasi Keamanan pada Setiap Tahapan SSDLC

Secure Software Development Life Cycle (SSDLC) merupakan pendekatan pengembangan perangkat lunak yang mengintegrasikan aspek keamanan pada setiap tahapan pengembangan sistem. Menurut Almorsy, Grundy, dan Müller (2020), keamanan perangkat lunak yang efektif harus diterapkan sejak tahap analisis kebutuhan hingga tahap pengujian untuk mengurangi potensi kerentanan.

Penelitian oleh Khan et al. (2021) menjelaskan bahwa integrasi keamanan sejak awal proses pengembangan dapat menurunkan risiko eksploitasi serta mengurangi biaya perbaikan dibandingkan pendekatan yang hanya melakukan pengujian di akhir.

Pendekatan SSDLC umumnya mencakup identifikasi kebutuhan keamanan pada tahap analisis, perancangan arsitektur yang mempertimbangkan kontrol keamanan, penerapan praktik secure coding pada tahap implementasi, serta pengujian keamanan sebelum sistem dirilis.

Dalam penelitian ini, integrasi keamanan dilakukan melalui identifikasi kebutuhan autentikasi dan otorisasi, penerapan Role-Based Access Control (RBAC), implementasi JSON Web Token (JWT) dan refresh token, serta pengujian keamanan menggunakan Static Application Security Testing (SAST) sebelum sistem dioperasikan.