

BAB I. PENDAHULUAN

1.1 Latar Belakang

Di era globalisasi saat ini, perkembangan teknologi informasi yang cepat telah memberikan kemudahan dan dukungan yang signifikan bagi unit kerja, instansi pemerintahan, maupun entitas swasta. Kemajuan ini memungkinkan akses yang efisien terhadap informasi yang tersedia dengan tingkat keakuratan yang tinggi. Manajemen arsip menjadi aspek krusial dalam organisasi, baik itu bersifat bisnis maupun pemerintahan. Dengan bertambahnya kebutuhan pekerjaan dalam suatu organisasi, jumlah arsip yang dihasilkan juga meningkat. Kuantitas arsip ini terus bertambah dari waktu ke waktu, sehingga memerlukan penanganan yang lebih efektif dan efisien daripada sebelumnya.

PT. XYZ berdiri sejak tanggal 22 Mei 1991 berdasarkan dari No. 046.K/023/DIR/ 1991. PT. XYZ merupakan bagian dari salah satu BUMN atau biasa disebut dengan Badan Usaha Milik Negara. PT. XYZ ini bertanggung jawab dalam ketersediaan pasokan listrik yang ada didaerah Lampung dan Sekitarnya. Unit Kendali Pembangkitan Lampung lebih memilih menggunakan energi terbarukan yang murah untuk menghasilkan listrik ketika menggunakan PLTA seperti diketahui, biaya bahan bakar saat ini relatif mahal, karena yang digunakan juga dibatasi. Selain itu, generator panasnya yang sudah cukup tua, jadi mengurangi fungsinya (Anggataru, 2021)

Ada beberapa masalah pendataan pegawai pada PT. XYZ yang saat ini masih memakai cara yang manual dengan semua data masih berbentuk *hardcopy* tanpa adanya *backup* data yang berbentuk *softcopy* sehingga memungkinkan kehilangan data atau kerusakan pada data tersebut karena adanya bencana alam. Beberapa tahun ini proses penyimpanan data pegawai masih di kelompokkan menurut tahun masuk pegawai sampai pensiun. Hal tersebut menyebabkan proses pencarian data menjadi tidak efisien dan cenderung lama.

Berdasarkan permasalahan tersebut maka dibutuhkannya aplikasi sistem manajemen pegawai agar membantu staf pada bagian SDM dalam *input* data dan mengakses data seperti data pribadi, mutasi, SK dan lain-lain. Tujuan dari pembuatan aplikasi ini adalah mempermudah pegawai yang ada dibidangnya dalam

mencari dokumen dengan cara terkomputerisasi tanpa harus mencari manual lagi. Sistem manajemen pegawai memudahkan staf dalam *backup* data agar lebih fleksibel, efisien, dan lebih terjamin keamanannya. Pengecekan dapat secara berkala agar dapat menghindari kehilangan data dan dapat menunjang data lain untuk ke depannya. Pada penulisan laporan tugas akhir ini penulis hanya merujuk pada pembuatan *Database*.

1.2 Tujuan

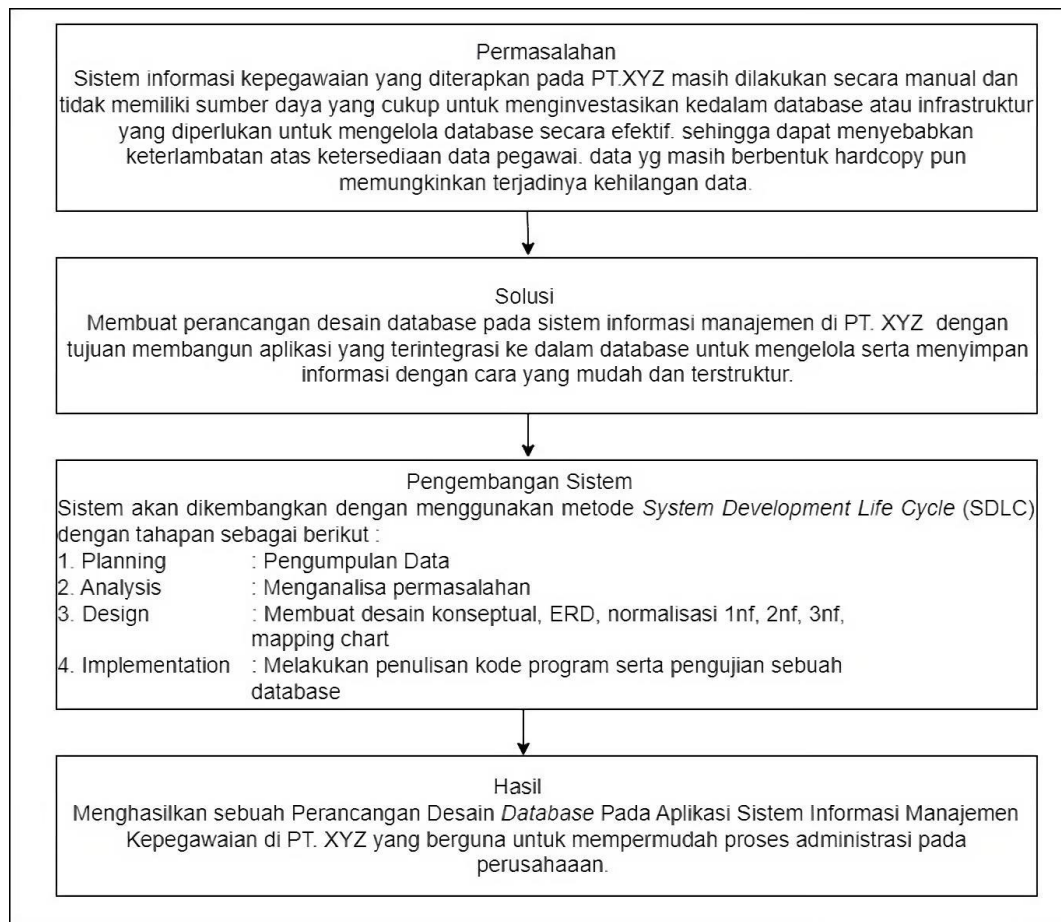
Tujuan dari penulisan Tugas Akhir ini adalah untuk menghasilkan Perancangan Desain *Database* Pada Aplikasi Sistem Informasi Manajemen Kepegawaian Di PT. XYZ dengan menggunakan *Postgresql*. Desain *database* ini untuk mempermudah dan berguna untuk pendataan pegawai serta meningkatkan kecepatan dalam pencariannya.

1.3 Kerangka Pemikiran

Berdasarkan pengamatan yang didapat pada permasalahan yang dihadapi saat ini adalah penggunaan data dalam bentuk fisik, yang rentan terhadap kerusakan, kehilangan, dan ketidakamanan. Bahkan, sistem yang digunakan saat ini belum mampu menyediakan kenyamanan, akurasi, efisiensi, dan keamanan yang optimal untuk data di PT. XYZ. Sehingga menyebabkan banyak dokumen yang tertumpuk banyak tempat serta penyimpanan data – data arsip tentang pegawai saat ini masih diurutkan secara tahun pertama masuk sampai pensiun jadi menyebabkan kehilangan atau kerusakan data, serta memperlambat proses pencarian karena belum ada aplikasi yang mendukung. Atas dasar tersebut maka diperlukannya solusi agar dapat mengatasi permasalahan tersebut secara efisien dan dapat menghambat hilang atau rusaknya sebuah dokumen yaitu dengan membuat “Perancangan Desain *Database* Pada Aplikasi Sistem Informasi Manajemen Kepegawaian Di PT. XYZ”.

Setelah dibuatnya Aplikasi Sistem Informasi Manajemen Kepegawaian PT. XYZ ini diharapkan bisa memudahkan pegawai yang berwenang untuk mendata pegawai, *input* data pegawai, mencari data pegawai dengan waktu yang lebih efektif dan efisien. Membangun sebuah “Perancangan Desain *Database* Pada Aplikasi Sistem Informasi Manajemen Kepegawaian Di PT. XYZ” ini membutuhkan beberapa tahapan yaitu perencanaan, analisis, desain, implementasi dengan metode

System Development Life Cycle (SDLC). Bagan kerangka pemikiran dapat dilihat pada Gambar 1.



Gambar 1. Kerangka Pemikiran

1.4 Kontribusi

Kontribusi yang diharapkan dari pengerjaan Perancangan Desain *Database* Pada Aplikasi Sistem Informasi Manajemen Kepegawaian adalah sebagai berikut:

1. Diharapkan dapat membantu perusahaan agar mempermudah dan mempercepat dalam pengelolaan data pegawai.
2. Diharapkan dapat pengetahuan baru tentang adanya Perancangan Desain *Database* Pada Aplikasi Sistem Informasi Manajemen Kepegawaian PT. XYZ menggunakan *Postgresql*.
3. Meningkatkan kemampuan mahasiswa Politeknik Negeri Lampung dalam bidang desain *database*.
4. Bahan acuan untuk penyusunan tugas akhir mahasiswa Program Studi Manajemen Informatika Politeknik Negeri Lampung.

BAB II. TINJAUAN PUSTAKA

2.1 Perancangan *Database*

Perancangan merupakan elemen kunci dalam pembuatan basis data, dengan tantangan utamanya adalah bagaimana memastikan bahwa basis data yang dibuat dapat memenuhi kebutuhan saat ini dan pada masa depan. Oleh karena itu, diperlukan perancangan database yang komprehensif, mencakup aspek fisik dan konseptual. Metodologi desain database merupakan serangkaian teknik sistematis yang digunakan untuk merancang struktur database. Teknik perancangan basis data dibagi menjadi tiga level yaitu : (*conseptual database design*), (*logical database design*) dan (*physical database design*).

Desain database merupakan tahap dalam siklus hidup *micro lifecycle*. Meskipun kegiatan yang terlibat dalam proses tersebut mencakup pengumpulan dan analisis data serta perancangan database konseptual, pemilihan DBMS, desain *database* logis (pemetaan model data), desain *database* fisik, dan implementasi sistem *database*. Beberapa metode digunakan dalam perancangan basis data, salah satunya adalah metode *Entity Relationship Diagram* atau disingkat ERD (Umar dkk., 2019).

Ada tiga tahapan dalam perancangan *database*, yaitu :

2.1.1 Desain *Database* Konseptual

Perancangan konseptual database melibatkan pembuatan model yang berasal dari data yang dikumpulkan selama analisis kebutuhan. Model ini mencakup semua aspek informasi yang diperlukan dalam perancangan sistem basis data dan berfungsi sebagai sumber informasi selama tahap desain logis (Umar dkk., 2019).

2.1.2 Desain *Logikal*

Desain *logikal* (Efendy, 2018) adalah proses pembuatan model berdasarkan pola dan kumpulan data tertentu yang digunakan oleh perusahaan. Deskripsi implementasi basis data berdasarkan hasil desain logis dalam diagram hubungan entitas (ERD) dari sistem manajemen basis data dan DBMS.

2.1.3 Desain Fisikal


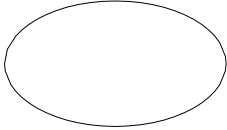
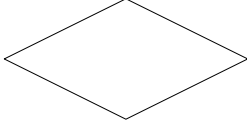

Perancangan *fisikal* adalah pembuatan deskripsi implementasi database dalam memori sekunder melibatkan penjelasan mengenai keterkaitan dasar, pemanfaatan indeks untuk mencapai akses data yang optimal, struktur organisasi file, serta semua batasan integritas dan tindakan keamanan yang terkait. Database logis dibentuk dan ditempatkan secara fisik di hard disk sesuai dengan sistem manajemen basis data (DBMS) yang sedang digunakan. Salah satunya DBMS yaitu *My Structured Query Language (MySQL)* (Efendy, 2018).

2.1.4 Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) adalah suatu metode pendekatan yang menjelaskan atau menggambarkan keterkaitan dalam suatu model tertentu. Model ini mengilustrasikan data dalam basis data berdasarkan objek data dan menunjukkan hubungan antara relasi. ERD mencerminkan objek data (Entitas) dan hubungan (Relationship) yang ada dalam entitas tersebut (Eka Wida Fridayanthie, 2016)

Adapun simbol-simbol ERD (*Entity Relationship Diagram*) tertera pada Tabel 1.

Tabel 1. Simbol *Entity Relationship Diagram*

Simbol	Nama	Deskripsi
	Entitas	<i>Entitas</i> merujuk pada inti data yang disimpan di dalam komputer agar dapat diakses melalui aplikasi. Umumnya, entitas mengacu pada objek atau konsep tanpa mencakup nama dalam tabel..
	Atribut	<i>Field</i> , yang sering disebut sebagai kolom data, perlu ditempatkan di dalam suatu entitas.
	Relasi	<i>Relasi</i> adalah pengait antar entitas dan umumnya dimulai dengan kata kerja terlebih dahulu.
	Asosiasi	<i>Asosiasi</i> adalah ikatan antara hubungan dan entitas, di mana kedua ujungnya memiliki tingkat kelimpahan yang diakumulasikan dalam penerapannya.

Sumber : (Lutfi Rahman, 2019)

2.1.4 Normalisasi

Normalisasi adalah salah satu pilihan meminimalkan duplikasi data (redundansi data), Normalisasi diperlukan jika ada referensinya Tabel yang kami buat kurang bagus (ada duplikat data, kemungkinan inkonsistensi data di tempat kerja mengubah, menyembunyikan informasi tertentu dan lain-lain) dan diperlukan agar ketika menggabungkan tabel *hash* kita dapat menggabungkannya kembali buat *array* asli sebelum dekomposisi, mendapatkan meja yang bagus. Hasil normalisasi adalah kumpulan data (tabel) dalam bentuk normal yang menggunakan normalisasi minimalis replikasi data (redundansi data) dan Memfasilitasi identifikasi entitas objek (Susanto, 2019).

Menurut (Khomsy Pane dkk., 2022) Berikut ada beberapa tahapan bentuk normal dalam normalisasi :

1. *First Normal Form* (1NF)

Bentuk normalisasi pertama (1NF) membutuhkan pemenuhan untuk beberapa kondisi dalam database dan fungsinya termasuk menghilangkan kolom duplikat dari tabel yang serupa, membuat tabel yang berbeda di setiap kumpulan data terkait, dengan mengidentifikasi setiap baris menggunakan nilai unik (kunci utama atau *primary key*).

2. *Second Normal Form* (2NF)

Penerapan bentuk normalisasi kedua menuntut bahwa data harus berada dalam bentuk 1NF, dan beberapa fungsi dari normalisasi 2NF termasuk menghilangkan subkumpulan data dari larik dan menempatkannya dalam larik terpisah, membentuk hubungan antara tabel baru dan yang lama melalui pembuatan kunci asing, serta memastikan bahwa tabel tidak memiliki atribut yang secara fungsional bergantung pada kunci kandidat tabel.


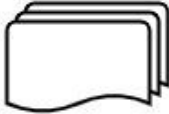

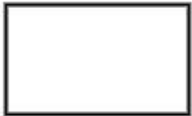

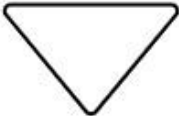
3. *Third Normal Form* (3NF)

Normalisasi database dalam format tingkat ketiga (3NF) berkonsentrasi pada penghapusan *field* atau atribut yang tidak memiliki hubungan dengan kunci utama. Oleh karena itu, tidak diperkenankan adanya ketergantungan transitif pada setiap kunci kandidat. Syarat yang harus dilaksanakan untuk mencapai tingkat normalisasi ketiga atau 3NF melibatkan semua kriteria bentuk normal kedua dan mengeliminasi kolom yang tidak bergantung pada kunci utama.

2.1.5 Mapping Chart

Mapping Chart adalah gabungan antara peta dan bagan alir yang menunjukkan alur proses atau penelitian dari satu proses ke proses lainnya. *Mapping Chart* membantu peneliti menganalisis proses penelitian agar terpecahkan masalahnya. Menganalisis segmen atau bagian yang lebih kecil serta membantu menguraikan alternatif dalam sebuah penelitian. Fungsi *mapping chart* yaitu untuk menggambarkan, menyederhanakan beberapa proses atau proses sehingga mudah dipahami dan mudah divisualisasikan berdasarkan urutan langkah proses (Mukodimah and Muslihudin, 2019)

Tabel 2. Simbol *mapping chart* disajikan pada tabel :

Simbol	Keterangan
	Dokumen
	Dokumen Rangkap
	Input / Output
	Proses yang dikerjakan oleh komputer
	Pengelolaan secara manual
	Pengarsipan dengan urutan A : Alfabet N : Angka T : Tanggal

Sumber : (Liksha, 2018)

2.2 Database

Database merupakan sekelompok informasi yang ditempatkan secara sistematis dalam komputer dengan teratur sehingga dapat dilihat memakai program komputer agar mendapatkan informasi dari basis data tersebut. Fungsi utama dari *database* yaitu pengguna dapat mengatur pandangan abstraksi data. Kegiatan ini bermaksud untuk memudahkan hubungan antara pengguna dan sistemnya beserta basis data yang dapat mengeskpresikan pandangan yang berbeda kepada pengguna lainnya, *programmer*, dan administrator. Sistem yang digunakan untuk mengelola dan menjalankan *query* dalam basis data disebut sebagai perangkat lunak manajemen basis data (DBMS). Pada dasarnya *database* yaitu sekumpulan data yang membentuk berkas (*file*), yang dihubungkan oleh operasi (relasional) khusus agar membentuk data atau informasi baru. (Umar dkk., 2019)

Keamanan *database* merupakan perlindungan *database* terhadap berbagai ancaman dan gangguan, baik teknis maupun administratif, Ini penting karena *database* sering kali terganggu dengan yang bervariasi, mencakup perangkat keras perangkat lunak, data dan orang. Secara keseluruhan insiden *database* fisik dan non fisik mencakup dari pencurian data, kehilangan kerahasiaan data, kehilangan integritas data dan kehilangan volume data. Ada beberapa cara untuk mengamankan *database*, termasuk mengotorisasi pengguna untuk mengakses objek *database*.

Aspek layanan keamanan informasi *database* adalah sebagai berikut:

- a) *Confidentiality* yaitu Layanan mencegah orang yang tidak berhak membuka atau membaca pesan yang ada di dalamnya
- b) *Data integrity* yaitu Layanan yang pesannya asli/utuh dan tidak rusak selama pengiriman.
- c) *Authentication* yaitu layanan yang terkait dengan verifikasi kebenaran terhadap pihak yang berkomunikasi, seperti autentikasi pengguna, dan mengkonfirmasi keaslian sumber pesan, yang disebut autentikasi asal data.
- d) *Non-repudation* yaitu Layanan yang melindungi entitas yang berkomunikasi dari penolakan, yaitu pengirim pesan menolak mengirim pesan ataupun penerima pesan.

2.2.1 PostgreSQL

PostgreSQL yang dikenal sebagai sistem manajemen basis data yang sangat

terkenal di seluruh dunia adalah sistem basis data yang telah dipublikasikan secara bebas sesuai dengan perjanjian lisensi dan telah mengalami pengembangan lebih dari 15 tahun. Dengan reputasi yang terbukti, PostgreSQL atau disebut juga Postgres-SQL merupakan sistem manajemen basis data relasional (RDBMS) *open source* yang dikembangkan oleh sekelompok sukarelawan di berbagai penjuru dunia.

Kelebihan yang ada pada *PostgreSQL* :

- a. *PostgreSQL* mengadopsi arsitektur *multi-proses (forking)*, yang berarti memiliki tingkat kestabilan yang tinggi karena satu proses yang mengalami kegagalan tidak akan menyebabkan kegagalan keseluruhan, meskipun sebelumnya hal ini sering terjadi.
- b. Dalam situasi beban tinggi, di mana terdapat banyak koneksi simultan, *PostgreSQL* menunjukkan kestabilan dan kepercayaan yang lebih baik daripada *MySQL*, terutama dalam menangani *query* dengan pernyataan gabung yang kompleks. Keunggulan ini disebabkan oleh dukungan *PostgreSQL* terhadap penguncian pada tingkat baris yang lebih rendah.

PostgreSQL menyajikan fitur berorientasi objek dengan kemampuan untuk memahami tabel dan tipe data, termasuk tipe data *array*, yang dapat bermanfaat dalam menyimpan beberapa item data dalam satu *entry*. Fungsi OO ini memungkinkan kita untuk mendefinisikan tabel di *PostgreSQL* yang mewarisi definisi tabel lain (Wibowo and Wiguna, 2019)

2.2.2 *Supabase*

Supabase adalah alternatif *database open source* untuk *Firebase*. Selain itu, *Supabase* merupakan layanan *database* yang menawarkan skalabilitas dan fitur yang dapat bersaing dengan *Firebase*. *Supabase* bukanlah layanan yang 100% dipetakan ke *Firebase*. *Supabase* menawarkan banyak fitur yang ditawarkan *Firebase*, tetapi tidak dengan cara yang sama. Teknologi yang dipilih oleh *supabase* sangat berbeda dengan *Firebase*. Semua yang digunakan *Supabase* berupa *open source*. *Supabase* menggunakan dan mendukung alat yang ada di ekosistem daripada membangun dari awal (Jemal Rashidi, Brata, and Brata 2022)

2.2.3 *Database Management System (DBMS)*

Database Management System (DBMS) adalah aplikasi sistem yang

digunakan untuk menyimpan, mengarsipkan, mengelola, dan menampilkan data. Tujuan dari DBMS adalah untuk mempermudah proses pemrosesan data, dan sistem ini dirancang khusus untuk tujuan tersebut. Adapun beberapa bahasa yang ada di DBMS yaitu : *Data Definition Language (DDL)*, *Data Manipulation Language (DML)* dan *Data Control Language (DCL)* (Yanuardi and Permana, 2019)

Keuntungan menggunakan sistem manajemen basis data atau DBMS meliputi:

1. Independensi data

DBMS dapat mengelola independensi representasi terperinci dan penyimpanan data aplikasi dengan menghapus data untuk mengisolasi kode program dalam aplikasi yang menggunakan data tersebut.

2. Efisiensi data

DBMS dapat memberikan akses yang efisien ke data, terutama jika data berada di penyimpanan eksternal, seperti server *database*.

3. Keamanan Data dan Integritas

Data yang diproses oleh *Database Management System* harus mengikuti peraturan yang telah ditentukan, tentu saja ini memastikan integritas data yang diperlukan dalam aplikasi, serta hak akses data yang berbeda untuk setiap jenis pengguna data DBMS , mengatur data, keamanan dari informasi yang dikelola.

4. Administrasi Data

Ketika data dihasilkan oleh banyak pengguna, diperlukan proses manajemen data yang benar-benar terpusat. Mengelola manajemen data membutuhkan seorang profesional yang memahami pentingnya manajemen data dan bagaimana kelompok pengguna menggunakannya untuk menghindari agregasi data dan memenuhi data . berbagai pengaturan untuk meningkatkan data.

5. Pemulihan Kerusakan dan Akses Konkuren

DBMS mengelola akses data bersamaan, sehingga pengguna merasa seperti mengakses data yang digunakan hanya oleh 1 dari pengguna. Selain itu, DBMS dapat melindungi pengguna dari kegagalan sistem.

6. Mengurangi Waktu Pengembangan Perangkat Lunak

Sistem Manajemen Basis Data memiliki sejumlah fungsi yang umumnya digunakan oleh aplikasi untuk mengakses data yang tersimpan. Kemudahan ini

tanpa keraguan membuat pengembangan perangkat lunak menjadi lebih efisien dan dapat diandalkan karena sejumlah operasi pemrosesan data dilakukan oleh DBMS sendiri.

2.2.4 SQL (*Structured Query Language*)

Structured Query Language merupakan bahasa standar yang digunakan di banyak basis data yang ada, oleh karena itu mudah digunakan bahkan saat bermigrasi dari satu basis data ke basis data lainnya. SQL menggunakan bahasa yang berurutan untuk mengambil data dari *database* dan entitas *database* tersebut. SQL terdiri dari satu atau lebih dari pernyataan SQL yang secara efektif mengontrol tugas server *database* (Pamungkas, 2018)

2.2.5 DDL (*Data Definition Language*)

Data Definition Language merupakan suatu bahasa yang dipakai untuk menetapkan skema dalam basis data dalam bentuk serangkaian pernyataan DDL. Pernyataan DDL ini mencakup pengumpulan dan pembuatan tabel yang ditempatkan dalam file yang disebut kamus data, yang juga mengandung meta data (Maanari and Sengkey, 2013)

2.2.6 DML (*Data Manipulation Language*)

Data Manipulation Language merupakan bahasa yang dipakai untuk memanipulasi *database* seperti mengambil data, memasukkan data, perubahan data dan menghapus data. Yang bertujuan untuk memudahkan orang berinteraksi dengan sistem. Ada dua jenis DML, yaitu prosedural yang dimana pengguna menentukan data apa yang dibutuhkan dan dengan cara apa mengambil data tersebut, dan non prosedural dimana pengguna hanya memastikan data apa saja yang dibutuhkan. Bahasa *query* adalah bagian dari *Data Manipulation Language* yang hanya mengikut sertakan pengambilan data. Aturan DML dan bahasa *query* biasanya hampir sama (Maanari and Sengkey, 2013)

2.3 Aplikasi

Sebuah aplikasi adalah perangkat lunak yang diciptakan untuk memungkinkan pengguna menjalankan perintah mereka dan mendapatkan hasil yang lebih akurat sesuai dengan tujuan pembuatan aplikasi tersebut. Tujuan dari suatu aplikasi adalah untuk menyelesaikan masalah dengan memanfaatkan berbagai teknik komputasi, yang umumnya bersaing dengan komputasi dan pemrosesan data

yang diperlukan atau diharapkan. (Wahyuni and Irawan, 2020).

Salah satu aplikasi yang digunakan untuk pembuatan database ini yaitu *PgAdmin*. *PgAdmin* adalah antarmuka aplikasi untuk PostgreSQL yang menyediakan solusi terkait perencanaan dan administrasi secara menyeluruh. Aplikasi ini dapat digunakan pada sistem operasi *Windows* dan *Linux*. *PgAdmin* tetap menggunakan Lisensi Seni yang gratis untuk penggunaan dan distribusi. Selain itu, *PgAdmin* juga merupakan sub-aplikasi yang mendukung sejumlah pengembangan dengan fitur-fitur kaya dan alat manajemen untuk DBMS *PostgreSQL*. (Aminullah, Suprayogi, and Sukmono., 2018).

2.4 Sistem Informasi Manajemen Kepegawaian

Secara teknis, sistem informasi dapat didefinisikan sebagai serangkaian komponen yang saling berkaitan, yang mengumpulkan atau memperoleh, memproses, menyimpan, dan menyebarkan informasi untuk mendukung pengambilan keputusan dan pemantauan di dalam suatu organisasi.

Manajemen Sistem Informasi merupakan rangkaian sub-sistem informasi yang utuh dan terkoordinasi secara logis yang bekerja bersama dengan koordinasi untuk mengubah data menjadi informasi dengan berbagai metode, dengan tujuan meningkatkan produktivitas sesuai dengan karakteristik dan model sistem. Ini dilakukan berdasarkan standar kualitas data yang telah divalidasi. Sistem informasi manajemen mencerminkan suatu sistem yang bertujuan untuk menyampaikan seluruh informasi yang memengaruhi berbagai kegiatan dalam organisasi.

Manajemen Sumber Daya Manusia secara umum merujuk pada semua aspek terkait jabatan, tanggung jawab, hak, dan pengembangan karyawan. Karyawan adalah individu yang menyumbangkan tenaga kerja, baik secara mental maupun fisik, yang sangat penting untuk kelangsungan suatu entitas. Karyawan juga merupakan aset utama suatu organisasi yang bekerja menuju pencapaian tujuan tertentu. Oleh karena itu, karyawan dapat diartikan sebagai individu yang bekerja untuk suatu entitas pemerintah atau perusahaan. Tujuan utama dari peningkatan kualitas karyawan adalah agar mereka dapat memberikan kontribusi yang signifikan terhadap kemajuan perusahaan atau lembaga negara. Sehubungan dengan hal ini, salah satu metode untuk meningkatkan kinerja manajemen di bidang sumber daya manusia adalah melalui partisipasi dalam pelatihan (Saputra, 2017)

2.5 Metode Pengembangan Sistem

Ada variasi metode dalam pengembangan sistem informasi, mulai dari yang terstruktur hingga yang berbasis objek. Dengan bantuan metode pengembangan ini, kita dapat mengetahui perangkat mana yang digunakan oleh sistem informasi. Salah satu metode pengembangan sistem yang umum digunakan adalah Siklus Hidup Pengembangan Sistem (SDLC), yang merupakan metode klasik yang banyak diterapkan dalam fase pengembangan, pemeliharaan, dan pemanfaatan sistem informasi. Pendekatan yang digunakan oleh SDLC adalah pendekatan sistematis, yang sering disebut sebagai pendekatan, yang melibatkan serangkaian tahapan pengembangan sistem (Aswati dkk., 2017).

Ada beberapa tahapan yang diperoleh pada metode *System Development Life Cycle* dijelaskan sebagai berikut :

1. *Planning*

Planning merupakan langkah awal dalam pengembangan sistem yang mengidentifikasi gambaran umum untuk kebutuhan sumber daya, termasuk peralatan fisik, tenaga kerja, metode (teknik dan tindakan), dan anggaran pada tingkat yang umum.

2. *Analysis*

Analisis *system* merupakan langkah kedua dalam pengembangan sistem, yang melibatkan pemeriksaan dan penelitian tahap sistem yang telah ada. Tujuannya adalah untuk mengembangkan sistem baru atau memperbaiki sistem yang sudah ada.

3. *Design*

Design atau sering disebut sebagai tahap rancangan, merupakan langkah yang mengikuti analisis sistematis untuk menetapkan proses dan informasi yang diperlukan oleh sistem yang baru. Desain sistem terdiri dari dua kategori, yaitu perancangan sistem secara umum dan desain sistem secara terperinci.

4. *Implementation*

Implementation adalah fase di mana desain sistem diubah menjadi kode (program) yang siap untuk diimplementasikan atau diterapkan.

2.6 Artikel Ilmiah

Artikel ilmiah merupakan teori dari berbagai sumber riset yang terkait dengan penulisan yang dilakukan oleh penulis. Keterkaitan yang dimaksud dapat berupa kesamaan dalam metode pengembangan sistem, maupun studi kasus yang diangkat oleh penulis. Ada beberapa referensi karya ilmiah yang serupa dan didapat dijadikan panduan dalam penulisan Tugas Akhir ini adalah sebagai berikut:

1. Kraugusteeliana, (2020) dengan judul “*Desain Database Aplikasi Monitoring Peralatan dan Bahan Praktikum pada Lab Patologi Klinik FK UPNVJ*” Penulisan karya ilmiah ini dilakukan dengan tujuan mengembangkan sebuah aplikasi yang dapat mendukung pengelolaan laboratorium secara sistematis. Aplikasi ini bertujuan membantu pengelola laboratorium dalam pendataan yang teratur, memonitor kondisi alat laboratorium, serta melaporkan kondisi tersebut. Selain itu, aplikasi ini juga memfasilitasi pengelola laboratorium dalam memantau ketersediaan materi pelatihan untuk kegiatan belajar mengajar pegawai laboratorium, metode yang digunakan yaitu *Database Life Cycle (DBLC)* dalam penelitiannya.
2. Hardiansyah and Fatmawati, (2020) judul “*Perancangan Basis Data Sistem Informasi Perwira Tugas Belajar (Sipatubel) Pada Kementerian Pertahanan*” karya ilmiah ini dibuat untuk menghasilkan aplikasi yang berfungsi menyimpan semua informasi terkait pelatihan personel Kemhan, misalnya sekitar personel yang dapat rekomendasi pelatihan di luar lembaga Kementerian, semuanya ini tersimpan di Sipatubel ini dalam urusan pendataan menggunakan *database* dan untuk memudahkan pengiriman data melalui sistem *database* yang dibuat oleh penulis. Metode yang digunakan adalah *System Development Life Cycle (SDLC)*.
3. Handayani and Putri, (2017) dengan judul “*Desain Database dan Hypertext Untuk Website Penelusuran Minat Peserta Didik Sebagai Layanan Bimbingan Konseling*” karya ilmiah ini dibuat untuk menghasilkan halaman web layanan bimbingan dan konseling sebanyak sekolah, khususnya untuk program kegiatan *tracking* yang menarik bagi siswa. jadi tujuan sebuah program ini adalah dapat menentukan berapa jumlah siswa yang memahami minat dan potensinya. Metode yang digunakan adalah metode *Database Life Cycle (DBLC)*.
4. Setyabudhi, (2017) dengan judul “*perancangan database relational pada toko*

buku *online*” Tujuan dari karya ilmiah ini adalah merancang basis data yang dapat mempermudah proses pembelian dan penjualan online bagi para konsumen. Proses pengembangan perangkat lunak ini diterapkan dengan menggunakan pendekatan model *waterfall*.

