

BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1 Pengertian Rancang Bangun

Berdasarkan KBBI, kata "rancang" merupakan akar kata dari "merancang", memiliki arti menyusun segala sesuatu sebelum melakukan tindakan, menjalankan suatu aktivitas, atau merencanakan sesuatu. Menurut Pressman (2016) Hasil analisis sistim diterjemahkan ke dalam bahasa pemrograman melalui prosedur yang dikenal sebagai proses desain. Metode ini memberikan penjelasan menyeluruh mengenai implementasi komponen-komponen sistem. Namun, pengembangan sistem mengacu pada proses pengembangan sistem baru atau melakukan modifikasi dan penyempurnaan terhadap sistem yang sudah ada, baik seluruhnya maupun sebagian.

Pengembangan sistem, sering juga dikenal sebagai pembangunan sistem, adalah proses mengembangkan sistem baru atau melakukan perbaikan menyeluruh terhadap sistem yang sudah ada. Oleh karena itu, rancang bangun dipahami sebagai proses yang menggambarkan, merencanakan, dan mengatur elemen-elemen terpisah menjadi unit yang sepenuhnya berfungsi. Akibatnya, proses desain mencakup pembuatan paket perangkat lunak yang menafsirkan hasil penelitian sebelum membuat sistem baru atau menyempurnakan sistem lama. (Pressman, 2019)

2.1.2 Aplikasi

Perangkat lunak yang diinstal pada komputer dan dirancang untuk menjalankan berbagai tugas sesuai dengan instruksi pengguna disebut aplikasi. Aplikasi ini dapat berupa program sederhana yang melakukan fungsi dasar hingga perangkat lunak kompleks yang mampu menangani berbagai pekerjaan spesifik, dari pengolahan

data hingga pengelolaan sistem informasi yang besar. Dengan antarmuka yang user-friendly, aplikasi memungkinkan pengguna untuk mengoperasikan komputer secara lebih efisien dan efektif, memfasilitasi berbagai kegiatan mulai dari tugas administratif, kreatif, hingga teknis. Pengguna dapat berinteraksi dengan aplikasi melalui berbagai input, seperti keyboard, mouse, atau layar sentuh, untuk menyelesaikan pekerjaan mereka dengan lebih cepat dan akurat. (Santoso dan Rahman, 2015:79).

Menurut Jogiyanto (2015:79) Program yang berisi instruksi untuk mengolah data disebut aplikasi. Berbagai atribut dalam aplikasi ini, yang terdiri dari beberapa kolom formulir yang disusun dengan baik, menciptakan tampilan yang menarik dan memudahkan pengguna dalam mengoperasikannya.

● 2.1.3 **Presensi**

Menurut Indrajit (2017) Dalam “Manajemen Sistem Informasi dan Teknologi Informasi, bukunya,” presensi dianggap sebagai aspek penting dalam instansi pemerintah dan universitas. Sistem presensi yang baik memungkinkan pengendalian proses penyelesaian pekerjaan menjadi lebih efektif, menghasilkan output sehingga tujuan-tujuan tersebut dapat terpenuhi semaksimal mungkin. Penerapan teknologi informasi, termasuk teknologi komputer, telekomunikasi, dan teknologi lainnya, diperlukan untuk mencapai sistem informasi presensi yang optimal, memberikan nilai tambah dalam pengelolaan sistem tersebut.

2.1.4 **Guru**

Menurut Suharsimi Arikunto (2016:239) Guru harus memiliki kompetensi sosial yang memungkinkan komunikasi efektif antara mereka dengan murid, pendidik lain, kepala sekolah, staf administrasi, serta anggota masyarakat. Beberapa indikator yang dapat digunakan untuk mengamati hal ini khususnya: (1) interaksi dengan siswa; (2) interaksi dengan kepala sekolah; (3) interaksi dengan rekan kerja; (4) interaksi dengan orang tua siswa; dan (5)

interaksi dengan masyarakat umum.

Menurut Nana Sudjana (2017: 61) Partisipasi siswa dalam menjalankan tugas, Aktivitas siswa dapat diamati ketika mereka menyelesaikan masalah sendiri, berkonsultasi dengan guru atau siswa lain, melakukan diskusi kelompok, mencari informasi sendiri, berlatih memecahkan masalah, dan menggunakan apa yang Anda pelajari untuk menyelesaikan tugas. Untuk menjamin siswa dapat menerima dan memahami transfer informasi guru secara efektif, maka sangat penting bagi siswa untuk berpartisipasi dalam proses pembelajaran.

2.1.5 Geolokasi

Proses geolokasi atau geolocation melibatkan menyediakan data lokasi yang akurat dari komputer, perangkat jaringan, atau alat lainnya dengan cara menemukan dan menentukan lokasi. Teknologi ini memungkinkan identifikasi daerah menggunakan data koordinat geografis yang sangat tepat. Proses tersebut biasanya menggunakan satelit dan teknologi lainnya untuk memastikan lokasi yang akurat. Geolocation sering disebut sebagai GPS (Global Positioning System), salah satu aplikasi teknologi geolokasi yang paling umum. GPS memungkinkan pelacakan dan penentuan posisi secara real-time, yang sangat berguna dalam berbagai bidang seperti navigasi, logistik, dan pelacakan perangkat. (Halifa dan Hasna 2019:85)

2.1.6 Swafoto

Menurut KBBI (Kamus Besar Bahasa Indonesia) online, swafoto adalah gambar diri sendiri yang diambil pribadi menggunakan kamera HP atau kamera digital. Swafoto, yang lebih familiar Selfie adalah sejenis foto potret diri yang diambil secara mandiri dengan kamera. Secara umum, selfie merupakan foto yang diambil seseorang sendiri. Definisi swafoto itu sendiri merupakan *a self-taken picture, usually captured using a webcam or smartphone and posted on a social networking platform*, biasanya menggunakan ponsel pintar atau webcam dan diunggah ke situs sosial media.

(Syahbana, 2014:9).

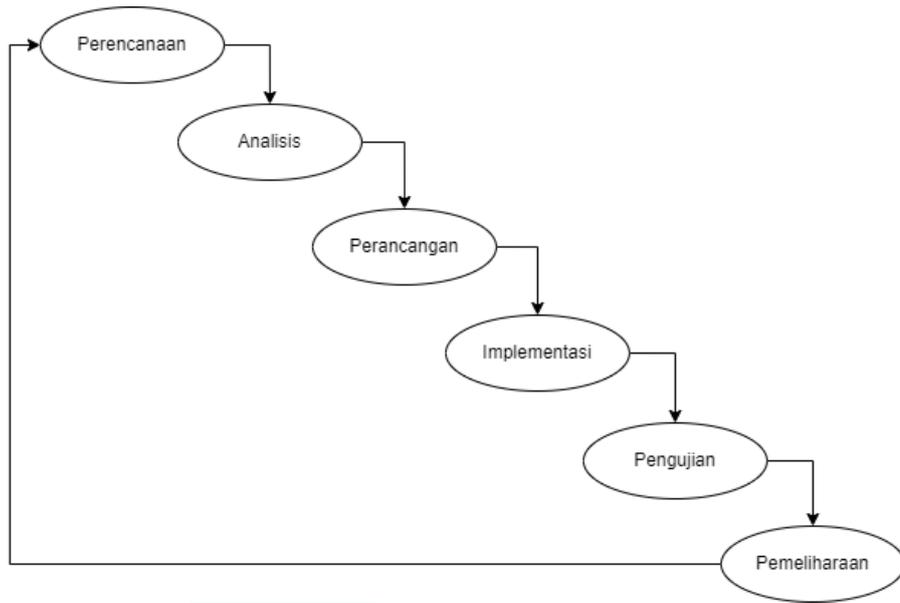
Psikolog Diana Parkisan mengatakan (dalam Syahbana, 2014:87) Swafoto diakui sebagai *update* yang populer untuk berkomunikasi, dan menjadi bentuk modern dari upaya menarik perhatian, khususnya di era di mana komunikasi online semakin mendominasi. Dengan begitu, swafoto tidak hanya sekadar menggambarkan diri kita, tetapi juga menjadi cara untuk menunjukkan keberadaan dan ekspresi diri. Pengunggahan swafoto ke media sosial juga bertujuan untuk berinteraksi dengan orang lain dan menilai seberapa banyak orang yang mengapresiasi hasil karya itu.

2.2 Teori Khusus

• 2.2.1 *System Development Life Cycle (SDLC)*

SDLC (Siklus Hidup Pengembangan Sistem) adalah model diterapkan menyelesaikan masalah yang muncul dari pendekatan sistem, dengan mengembangkan solusi sistem informasi untuk menghadapi tantangan bisnis. (Parwati, 2015).

Berikut adalah uraian mengenai proses tahap-tahap dalam SDLC, yang dimulai dari perencanaan dan berakhir pada tahap pemeliharaan. Tahap-tahap ini membentuk kerangka kerja lengkap untuk pengembangan aplikasi.



Gambar 2. 1 Tahapan Fase SDLC

Sumber : (Andi, Anton, Rusydi., 2019)

Langkah-langkah dalam SDLC menurut Rosa dan Shalahuddin (2016:26),:

1. Perencanaan
Dokumen perencanaan lainnya disiapkan, termasuk rencana manajemen proyek. Dasar untuk mendapatkan sumber daya yang diperlukan untuk mencapai solusi disediakan.
2. Analisis
Pastikan persyaratan pengguna sistem perangkat lunak dan buat persyaratan tersebut. Buat dokumentasi yang menguraikan persyaratan fungsional.
3. Tahap Perancangan
Ubah persyaratan khusus menjadi persyaratan umum dengan menggunakan dokumen desain sistem yang menekankan penyelesaian tugas yang diperlukan.
4. Tahap Implementasi
Melaksanakan desain informasi sistem secara lengkap, yang mencakup pengumpulan dan pengaturan sistem lingkungan yang diperlukan, menghasilkan data dasar, menyiapkan

protokol kasus uji, mengumpulkan file pengujian, menulis dan menyusun kode, memperbaiki dan mengatur program, dan menilai prosedur pengujian.

5. Tahap Pengujian

Di bawah bimbingan pengguna dan tim penjaminan mutu, pastikan sistem perangkat lunak memenuhi spesifikasi yang tercantum dalam dokumen persyaratan fungsional. Hasilnya, laporan analisis pengujian dihasilkan.

6. Tahap Pemeliharaan

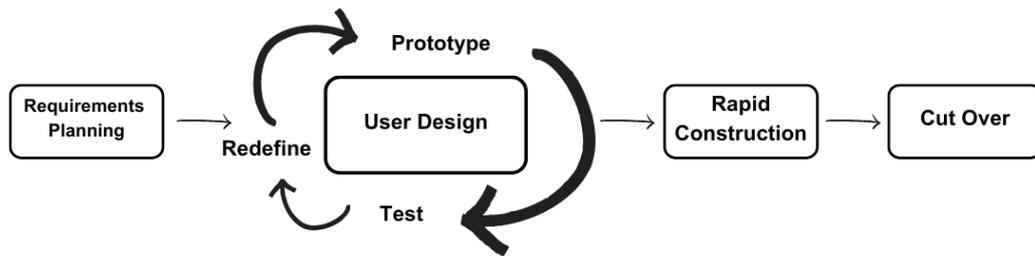
menjelaskan apa yang diperlukan untuk menjalankan dan mengelola sistem informasi dalam lingkungan produksi.

2.2.2 Rapid Application Development (RAD)

Pressman (2018:32) RAD didefinisikan sebagai metodologi pengembangan perangkat lunak yang menekankan pada kecepatan dan pembuatan prototipe. Metode ini menggunakan pendekatan iteratif, di mana prototipe perangkat lunak dikembangkan secara bertahap dan diuji berulang kali bersama pengguna. Dengan cara ini, pengembang dapat memperoleh umpan balik dari pengguna dengan cepat dan melakukan perubahan yang diperlukan secara efisien.

Sedangkan menurut (Sikumbangetal.,2020) Metodologi Rapid Application Development (RAD) menggabungkan konsep berorientasi objek ke dalam proses pengembangan perangkat lunak. Dibandingkan dengan metode konvensional, tujuan utama pendekatan ini adalah untuk mempercepat perencanaan, pengembangan, dan implementasi sistem.

RAD DESIGN



Gambar 2.2 Tahapan Fase RAD

Sumber : (Agustinus Noertjahyana. 2002)

Berikut adalah ringkasan singkat dari proses pelaksanaan RAD menurut Afrizal Zein (2023:95), sebagai berikut:

1. *Requirements Planning*

Tahap awal dalam Model RAD adalah tahap perencanaan kebutuhan. Pada tahap ini, tim pengembang perangkat lunak bertemu dengan pelanggan untuk mengidentifikasi dan memprioritaskan kebutuhan fungsional dan non-fungsional.

2. *User Design*

Tahap kedua adalah tahap perancangan antarmuka pengguna. Dalam tahap ini, tim pengembang perangkat lunak akan merancang antarmuka berdasarkan kebutuhan pelanggan. Peran pelanggan sangat penting pada tahap ini, karena mereka memberikan masukan langsung untuk memastikan antarmuka memenuhi kebutuhan mereka.

3. *Rapid Construction*

Tahap ketiga adalah tahap konstruksi cepat. Pada tahap ini, pengembang perangkat lunak akan membuat prototipe atau model awal dari perangkat lunak tersebut. Prototipe ini digunakan untuk memperoleh umpan balik dari pelanggan, sehingga pengembang dapat memperbaiki perangkat lunak sesuai dengan masukan yang diterima.

4. *Cutover*

Tahap akhir dalam Model RAD adalah tahap implementasi atau Cutover. Pada tahap ini, implementasi perangkat lunak yang sudah selesai dibangun dapat dilakukan. Tim pengembang perangkat lunak akan menginstal perangkat lunak tersebut pada sistem pelanggan dan melakukan pengujian sebelum perangkat lunak dapat dioperasikan sepenuhnya.

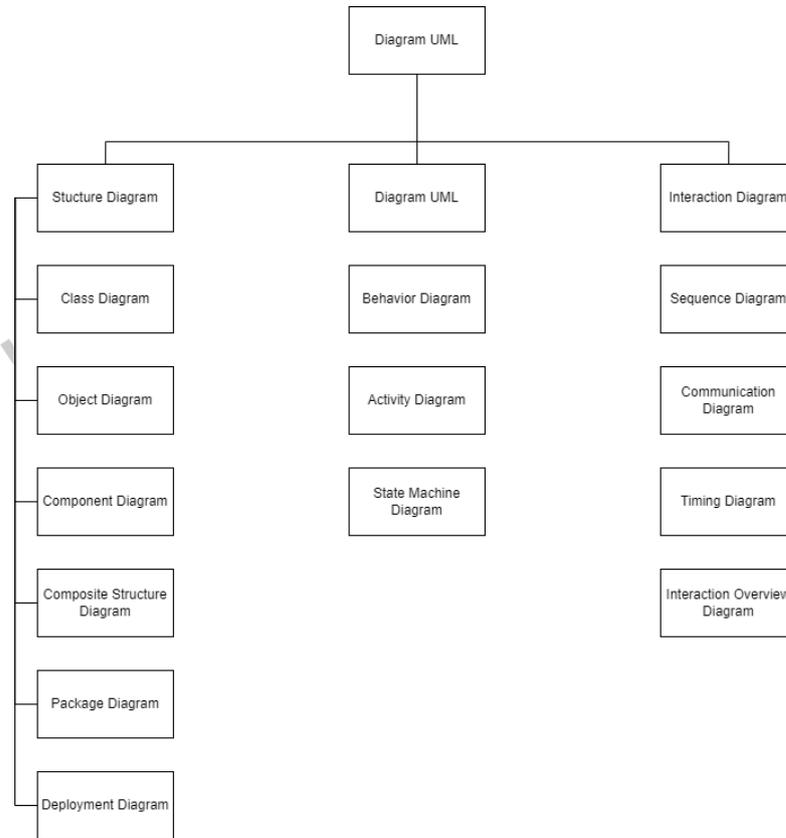
2.2.3 *Object Oriented Analysis and Design (OOAD)*

Mathiassen (dalam Purwaningtias, 2018) Objek diartikan sebagai entitas yang memiliki identitas yang unik, kondisi yang menggambarkan keadaan atau atributnya, dan perilaku yang merujuk pada aktivitas atau operasi yang dapat dilakukan objek tersebut. Ini memungkinkan objek-objek dalam sistem untuk berinteraksi, memfasilitasi komunikasi dan pertukaran informasi di antara mereka. Pendekatan Object Oriented and Design (OOAD) adalah sebuah metodologi yang melibatkan serangkaian kegiatan seperti analisis kebutuhan sistem dan desain sistem dengan mempertimbangkan perspektif kelas-kelas yang terlibat serta arsitektur keseluruhan dari sistem yang akan dibangun. Tujuan dari metodologi ini adalah membuktikan bahwa semua aspek kebutuhan fungsional dan non-fungsional sistem terdokumentasi dengan baik sebelum proses implementasi dimulai. (Hasanuddin, 2016).

2.2.4 *Unified Modelling Language (UML)*

Rosa dan Shalahuddin (2016:139), Bahasa model yang tidak terdefinisi terdiri dari kombinasi beberapa diagram yang berjumlah antara delapan hingga sembilan, yang digabungkan menjadi diagram interaksi. UML adalah bahasa pemodelan penting untuk pengembangan perangkat lunak yang membantu dalam pemahaman, pembuatan, pendokumentasian, dan visualisasi dan mengkomunikasikan desain sistem perangkat lunak. UML mencakup 13 jenis diagram yang dikelompokkan ke dalam tiga kategori. Berikut adalah pembagian dari ketiga kategori tersebut

beserta macam-macam diagram yang termasuk di dalamnya:



Gambar 2.3 Diagram UML

Sumber: (Sukamto dan Shalahuddin., 2016:140)

Berikut ringkasan pembagian dari kategori-kategori tersebut menurut Rosa dan Shalahuddin (2016:141):

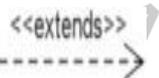
1. *Structure Diagram* adalah sekumpulan diagram yang menunjukkan struktur statis sistem yang dimodelkan.
2. *Behavior Diagram* adalah kumpulan diagram yang digunakan untuk menjelaskan perubahan atau perilaku sistem.
3. *Interaction Diagram* adalah sekumpulan diagram yang menjelaskan bagaimana suatu sistem berinteraksi dengan sistem lain atau dengan subsistemnya.

2.2.5 Usecase Diagram

Menurut Eri Yuniarto dan Ammar Fauzan (2022) Use case

diagram ialah alat visualisasi agar memperlihatkan hubungan antara pengguna dan sistim yang dirancang. Diagram ini dibuat dengan sederhana agar informasi mudah dipahami. Notasi dasar dari Use case diagram mencakup berbagai sintaksi yang dipakai untuk melihat interaksi antara objek atau entitas dalam sebuah sistem secara berurutan berdasarkan waktu. Sintaksi-sintaksi ini meliputi:

Table 2. 1 Notasi Usecase Diagram

Simbol	Keterangan
	Aktor: mewakili peran orang, sistem yang lain, atau alat ketika berkomunikasi dengan <i>use case</i>
	<i>Use Case</i> : abstraksi dan interaksi antara <i>sistem</i> dan <i>actor</i>
	<i>Association</i> : abstraksi dari penghubung antara <i>actor</i> dengan <i>use case</i>
	Generalisasi: menunjukkan spesialisasi actor untuk berpartisipasi dengan <i>use case</i>
	<i>Include</i> : menunjukkan bahwa suatu <i>use case</i> seluruhnya merupakan fungsionalitas dari <i>use case</i> lainnya
	<i>Extends</i> : Menunjukkan bahwa suatu <i>use case</i> merupakan tambahan fungsionalitas dari <i>use case</i> lainnya jika suatu kondisi terpenuhi

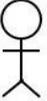
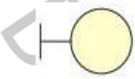
Sumber : (Ery dan Ammar., 2022)

2.2.6 Sequence Diagram

Menurut Ery dan Ammar (2022) Diagram urutan menggambarkan bagaimana item-item yang terhubung berinteraksi satu sama lain dan menunjukkan bagaimana objek-objek tersebut berkomunikasi. Diagram Ini menguraikan perilaku dalam skenario tertentu dan menggambarkan pesan yang dipertukarkan antara entitas dan sistem.

Menurut Andri, Gunawan, Faiza (2016) Diagram urutan menunjukkan interaksi—pesan yang disusun secara kronologis—antara pengguna, tampilan, dan objek lain di dalam dan di luar sistem. Dua dimensi diagram sekuens adalah vertikal, yang melambangkan waktu, dan horizontal, yang melambangkan benda-benda yang terhubung. Diagram ini biasanya digunakan untuk menggambarkan suatu situasi atau serangkaian tindakan yang dilakukan sebagai reaksi terhadap suatu kejadian untuk mencapai hasil tertentu.

Table 2. 2 Notasi Sequence Diagram

Simbol	Keterangan
	Aktor: mempresentasikan <i>entitas</i> berada di luar sistem dan berinteraksi dengan system
	<i>Lifeline</i> : menghubungkan objek selama <i>sequence</i> (<i>message</i> dikirim atau diterima dan aktifitasnya)
 Nama objek	<i>General</i> : mempresentasikan entitas tunggal dalam <i>sequence</i> diagram
	<i>Boundary</i> : berupa tepi dari sistem, seperti user <i>interface</i> atau suatu alat yang berinteraksi dengan <i>sistem</i> yang lain
	<i>Control element</i> : mengatur aliran dari informasi untuk sebuah skenario. Objek ini umumnya mengatur perilaku.
	<i>Entitas</i> : elemen yang bertanggung jawab menyimpan data atau informasi.

 Waktu aktif	<i>Activation</i> : suatu titik di mana sebuah objek mulai berpartisipasi di dalam sebuah <i>sequence</i> yang menunjukkan kapan sebuah objek mengirim atau menerima objek. Panjang waktu yang dibutuhkan.
 Pesan masuk	<i>Message Entry</i> : berfungsi untuk menggambarkan pesan/hubungan antar objek yang menunjukkan urutan kejadian yang terjadi.
 Keluaran	<i>Message to self</i> : Menggambarkan pesan/ hubungan objek itu sendiri, yang menunjukkan urutan kejadian yang terjadi.
 Hasil pengiriman	<i>Message Return</i> : menggambarkan hasil dari pengiriman message.

Sumber : (Ery dan Ammar., 2022)

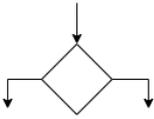
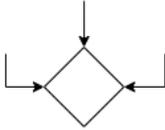
2.2.7 Activity Diagram

Nova Indrayana Yusman (2019) Alur kerja dan aktivitas dalam sistem, aktor, atau proses bisnis digambarkan menggunakan diagram aktivitas. Skema ini menunjukkan aktivitas atau alur kerja dalam sebuah sistem atau proses bisnis. Diagram Aktivitas menampilkan keadaan proses, aliran, dan banyak aktivitas menggunakan berbagai simbol. (Siregar, Siregar, Melani., 2019)

Temuan penelitian, diagram aktivitas adalah representasi skematis dari urutan di mana proses bisnis tertentu diselesaikan dalam suatu sistem, dengan setiap aktivitas bertanggung jawab kepada unit organisasi tertentu.

Table 2. 3 Notasi Activity Diagram

Simbol	Keterangan
--------	------------

	<p>Status awal: sebuah diagram aktivitas memiliki sebuah status awal</p>
	<p>Aktivitas: aktivitas yang dilakukan <i>sistem</i>, aktivitas biasanya diawali dengan kata kerja</p>
	<p>Pecabangan/<i>Decision</i> : Percabangan di mana ada pilihan aktivitas yang lebih dari satu</p>
	<p>Penggabungan/<i>Join</i> : Penggabungan di mana yang mana lebih dari satu aktivitas lalu di gabungkan jadi satu</p>
	<p>Status akhir (<i>end state</i>) : Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki status akhir</p>
	<p><i>Swimlane</i> : Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi</p>
	<p><i>Merge</i> : untuk menyatukan kembali <i>decision path</i> yang dibuat dengan menggunakan <i>decision node</i></p>
	<p><i>Final Flow-node</i> : untuk menghentikan <i>control flow</i> atau <i>object flow</i> tertentu</p>

Sumber : (Ery dan Ammar., 2022)

2.2.8 Database

Menurut Hasrul dan Lamro (2016) *Database* adalah serangkaian item data yang saling terkait, disusun berdasarkan suatu Perangkat keras komputer menyimpan skema atau struktur tertentu, yang kemudian dikelola dengan perangkat lunak khusus untuk manipulasi guna memenuhi kebutuhan spesifik. Suatu sistem komputer untuk penyusunan dan manajemen catatan, bertujuan untuk menyimpan, merekam, dan menjaga data secara menyeluruh dalam sebuah organisasi atau perusahaan. Sistem ini dirancang untuk memberikan informasi optimal kepada pengguna dan mendukung proses pengambilan keputusan. Lubis (2016)

Sistem basis data terdiri dari sekelompok tabel data dan program yang terhubung satu sama lain, Pengguna atau program lain dapat mengakses, menggunakan, dan memodifikasi tabel data tersebut. (Fikry, 2019).

Menurut Ikhsan, Susilo, Abdillah (2023) dalam buku “Konsep Dasar Membangun Database” Beberapa komponen terdapat dalam sistem *database*, yaitu:

1. Perangkat Keras (*Hardware*)
Komponen hardware yang biasanya ada dalam sistem basis data meliputi server database, hard disk, dan lainnya.
2. Sistem Operasi (*Operating System*)
Komponen hardware yang biasanya ada dalam sistem basis data meliputi server database, hard disk, dan lainnya.
3. *Management System* (DBMS)
Pengguna tidak secara langsung melakukan operasi database yang sebenarnya; sebaliknya, penyimpanan, pembaruan, dan pengambilan data dikelola oleh perangkat lunak yang dikenal sebagai sistem manajemen basis data, atau DBMS.
4. Pemakai (*User*)
Program yang ditulis dalam bahasa pemrograman memungkinkan pengguna untuk mengelola data dan

berinteraksi dengan database.

5. Aplikasi atau Perangkat Lain

Aplikasi ini bervariasi tergantung kebutuhan, pengguna basis data dapat membuat program khusus yang sederhana untuk memasukkan, mengubah, dan mengambil data. Program-program ini dapat berintegrasi langsung dengan DBMS atau dibuat menggunakan bahasa pemrograman tertentu.

2.3 *Literature Review*

1. Jurnal yang ditinjau adalah "**Aplikasi Sistem Informasi Absensi Mahasiswadan Dosen**" yang ditulis oleh Firliana, Rina dan Rhohman, Fatkur (2019) Artikel jurnal ini membahas pentingnya sistem absensi dalam dunia pendidikan dan perkuliahan. Penelitian menunjukkan bahwa banyak institusi pendidikan masih menggunakan metode manual untuk absensi, di mana dosen memanggil satu per satu mahasiswa untuk mencatat kehadiran atau mahasiswa menandatangani daftar hadir secara bergilir. Hal ini meningkatkan kemungkinan siswa tidak hadir atau menggunakan tanda tangan fiktif untuk melakukan penipuan. Administrator juga harus memberikan data kehadiran secara manual kepada direktur program, dan instruktur bertanggung jawab untuk melakukan pemeriksaan independen terhadap data tersebut untuk menilai kehadiran siswa pada semester tersebut. Sebuah aplikasi web untuk sistem informasi kehadiran dosen dan mahasiswa sedang diusulkan oleh peneliti. Tinjauan terhadap penelitian sebelumnya tentang sistem absensi dari buku dan jurnal, analisis melalui wawancara pihak terkait, perancangan menggunakan data flow diagram (DFD), dan pemrograman PHP untuk implementasi merupakan beberapa tahapan dalam proyek penelitian ini. Melalui fase-fase ini, kami mengembangkan sistem kehadiran online untuk studi ini yang memungkinkan instruktur dan siswa untuk segera memasukkan kehadiran untuk setiap perkuliahan, sehingga memudahkan manajemen untuk melaporkan

2. Tinjauan studi dari jurnal berjudul **“RANCANG BANGUN SISTEM APLIKASI ABSENSI PEGAWAI GURU TETAP DAN GURU EKSKUL BERBASIS WEBSITE PADA SD IT AL-IKHLAS”** oleh Fazel Junia priyaditama (2022) Penelitian ini membahas pentingnya sistem absensi dalam pendidikan atau perkuliahan. Hasilnya menunjukkan bahwa banyak institusi pendidikan masih memakai sistem absensi manual. Dosen memanggil mahasiswa satu per satu untuk mencatat kehadiran atau mahasiswa menandatangani daftar hadir secara bergiliran. Hal ini memungkinkan mahasiswa terlewat atau melakukan kecurangan dengan menitip tanda tangan palsu. Selain itu, admin harus melaporkan data absensi ke ketua program studi secara manual, dan dosen harus memeriksa data absensi satu per satu untuk menilai kehadiran selama satu semester. Oleh karena itu, peneliti mengusulkan pengembangan aplikasi sistem informasi absensi mahasiswa dan dosen secara online. Metode penelitian yang digunakan meliputi studi pustaka dari buku dan jurnal terkait sistem absensi, analisis melalui wawancara dengan pihak terkait, perancangan dengan Data Flow Diagram (DFD), dan implementasi program menggunakan PHP. Dengan tahapan-tahapan ini, penelitian menghasilkan sistem absensi online di mana mahasiswa dan dosen dapat langsung menginput kehadiran setiap perkuliahan, dan admin dapat dengan mudah merekap jumlah kehadiran untuk pelaporan.
3. Tinjauan studi dari jurnal berjudul **“Aplikasi Absensi Karyawan Studi Kasus Cakra Entertainment Berbasis Web”** yang ditulis oleh Wahyu Eko Setiawan, Andi Rahman Putera, Abdul Rozaq (2022) Penelitian ini menyajikan perancangan aplikasi berbasis web yang berupaya meningkatkan efisiensi kehadiran dalam bisnis. Penelitian untuk penelitian ini dilakukan di PT Cakra Entertainment. Observasi, wawancara, dan tinjauan pustaka dengan menggunakan pendekatan pengembangan sistem Rapid Application Development (RAD) merupakan beberapa teknik pengumpulan data yang digunakan. Pada penelitian ini penulis menggunakan MySQL sebagai sistem manajemen

database dan bahasa pemrograman PHP yang dibuat dengan framework Codeigniter.

4. Jurnal yang peneliti tinjau adalah “**Sistem Informasi Absensi Pegawai Menggunakan Metode RAD dan Metode LBS Pada Koordinat Absensi**” ditulis oleh Mohammad Arya Rosyd Sikumbang , Roni Habibi , Syafrial Fachri Pane Pada 2020, Perancangan sistem absensi online dibahas dalam makalah jurnal ini. Pencatatan kehadiran yang merupakan salah satu komponen penting dalam pelaporan suatu lembaga disebut dengan kehadiran. Kebijakan disiplin karyawan sangat penting bagi bisnis karena kebijakan ini berfungsi sebagai tolok ukur utama untuk mengevaluasi kinerja karyawan sehubungan dengan kehadiran mereka. Oleh karena itu, dunia usaha perlu memiliki sistem kehadiran yang dapat mengontrol kehadiran pekerja, beserta aturan, batasan, dan sanksi jika melanggar atau tidak melaksanakan tugas. Salah satu permasalahan yang sering dihadapi Badan Pusat Statistik Kota Bandung adalah pegawai yang melakukan perjalanan dinas terkadang lalai untuk tiba di tempat kerja lebih awal. Metode Rapid Application Development (RAD) adalah teknik pengembangan perangkat lunak berorientasi objek yang digunakan dalam penelitian ini yang mencoba mempercepat perencanaan, perancangan, dan implementasi sistem dibandingkan dengan teknik yang lebih konvensional. Selain itu, layanan informasi geografis diakses melalui teknik Layanan Berbasis Lokasi (LBS), yang memungkinkan pengguna perangkat seluler memetakan lokasi mereka. Penelitian ini dimaksudkan agar organisasi dan pekerja dapat melaksanakan pekerjaannya dengan lebih efisien tanpa terbebani oleh tekanan atau beban kerja yang berlebihan.