

BAB II

LANDASAN TEORI

2.1 Tinjauan Pustaka

Dalam penelitian ini dibutuhkan tinjauan pustaka sebagai panduan komprehensif yang ditinjau dari beberapa jurnal penelitian terdahulu, berkaitan dengan judul penelitian dan pokok pembahasan pada penelitian yang akan dilakukan. Adapun jurnal penelitian terdahulu yang digunakan dalam acuan pada penelitian ini terlihat pada tabel 2.1.

Tabel 2.1 Tinjauan Pustaka

No	Literatur	Jenis Aplikasi/Bahasa Pemrograman	Metode Pengembangan Sistem	Teknologi	Hasil
1	Pengembangan Aplikasi berbasis Android Sistem Presensi dengan menggunakan Validasi Koordinat Lokasi dan Foto <i>Selfie</i> di PT. Bintang Mas Glassolutions (Buana et al., 2022)	<i>Android</i> /Tidak Disebutkan	<i>Waterfall</i>	Lokasi dan Foto <i>Selfie</i>	Memberikan kemudahan akses dalam presensi karyawan
2	Rancang Bangun Sistem	<i>Android/Java</i>	<i>Waterfall</i>	Lokasi dan Foto <i>Selfie</i>	Membantu pihak manajemen

	Informasi Presensi Karyawan Lapangan Menggunakan Foto Dan GPS Berbasis Android Pada PT. Asando Karya (Devia, 2022)				dalam melakukan monitoring dan rekapitulasi data kehadiran penugasan lapangan untuk pegawai survei
3	Aplikasi Presensi Online Menggunakan Verifikasi Foto <i>Selfie</i> Dan <i>Ip Server</i> Berbasis <i>Android</i> (Wahyuni et al., 2023)	<i>Android (Web View)/PHP</i>	Tidak disebutkan	<i>Ip Server</i> dan Foto <i>Selfie</i>	Mempermu dah proses presensi masuk dan pulang dosen/pegawai dengan memanfaatkan gawai masing-masing
4	Sistem Informasi Absensi Menggunakan Foto <i>Selfie</i> dan <i>Geotagging</i> (Prayogo & Sanjaya, 2021)	<i>Android/Tidak</i> Disebutkan	<i>Waterfall</i>	Lokasi dan Foto <i>Selfie</i>	Mempermu dah pegawai dalam melakukan absensi dengan cara foto <i>selfie</i> dilokasi bertugas sehingga pegawai tidak perlu datang ke kantor

					untuk melakukan absensi yang akan memakan waktu dan jarak tempuh
5	Perancangan Presensi <i>Online</i> Menggunakan Foto dan <i>Geolocation</i> Guna Meningkatkan Kedisiplinan Pegawai Berbasis <i>Website</i> (Studi Kasus: PT. A Logistic International) (Turmudi & Rosyani, 2023)	<i>Web/PHP</i>	<i>Waterfall</i>	Lokasi dan foto <i>Selfie</i>	Memudahkan pegawai pada saat absensi diluar kantor lebih efisien, lebih disiplin dan mengurangi resiko kerusakan dan kehilangan pada data absensi
6	Aplikasi Absensi <i>Mobile</i> Berbasis <i>Mapping</i> Koordinat Lokasi (Studi Kasus : Lorus Celluler) (I. Gunawan et al., 2022)	Android (Web View)/PHP	<i>Prototyping</i>	Lokasi dan Foto <i>Selfie</i>	Memperudahkan pengelolaan absensi karyawan dan dapat mencegah kecurangan pengambilan absensi oleh karyawan

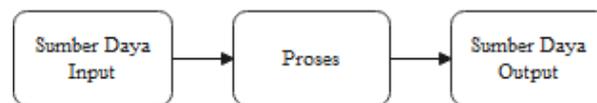
7	Pengembangan Sistem Presensi dan Kedisiplinan Dosen Terhadap Biaya Operasioal Perguruan Tinggi (Wantoro, 2016)	<i>Desktop/Tidak Disbeutkan</i>	<i>Waterfall</i>	-	Mengurangi resiko manipulasi data dan kesalahan presensi yang dilakukan oleh dosen
8	Rancang Bangun Aplikasi Presensi SMS Gateway Berbasis Web dengan Framework Codeingniter Pada SMKN 1 Trimurjo (Wulandari & Sulistiani, 2020)	<i>Web/PHP</i>	<i>Waterfall</i>	<i>SMS Gateway</i>	Memberikan informasi kehadiran siswa kepada orang tua/wali murid via SMS
9	Aplikasi Monitoring Absensi Karyawan Ditlantas Dengan Penerapan Teknologi GPS (Studi Kasus: Ditlantas Polda Lampung) (Kurnia	<i>Android/Tidak Disebutkan</i>	<i>Waterfall</i>	Lokasi dan Foto <i>Selfie</i>	Memudahkan dalam pemantauan lokasi kerja dan presensi Ditlantas

	Adam et al., 2023)				
10	Sistem Informasi Manajemen Presensi Siswa Berbasis <i>Mobile</i> Studi Kasus SMAN 1 Sungkai Utara Lampung Utara (Putra Setiawan & Muhaqiqin, 2021)	<i>Mobile</i> /Tidak Disebutkan	<i>Extreme Programming</i>	-	Memberikan informasi kehadiran siswa berupa notifikasi kepada orang tua siswa
11	Peneliti	<i>Android Native/Kotlin</i>	Prototipe	Foto <i>Selfie</i>	Diharapkan dapat memudahkan proses presensi karyawan diseluruh perusahaan CBP Group, mengurangi kerumitan administratif serta meningkatkan kedisiplinan

2.2 Sistem Informasi

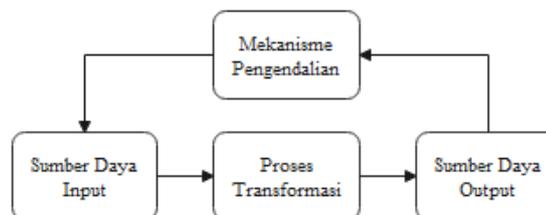
Sistem merupakan sebuah dasar pergerakan dalam seluruh kegiatan, keberadaan sistem dalam segala bidang sangat diperlukan, tanpa adanya konsep dari sistem, kegiatan atau pekerjaan akan berjalan tanpa kendali. Sistem akan berjalan baik apabila keseluruhan karakteristik sistem saling bersinergi untuk mencapai tujuan sesuai yang ditetapkan pada tahap awal (Ridwan et al., 2021).

Sistem memiliki pendekatan yang ditekankan dalam sebuah prosedur jaringan kerja secara saling hubung, mengelompok serta bekerja bersama untuk mendapatkan pencapaian sasaran yang diinginkan. Dalam prosedur terdapat instruksi dengan tahapan-tahapan yang berurutan dimana apa yang dikerjakan, siapa yang melakukan pekerjaan, kapan pengerjaannya dan bagaimana cara kerjanya (Dedy, 2020). Pengelompokan sistem dibagi menjadi dua bagian dimana yang pertama sistem terbuka dan sistem tertutup. Sistem terbuka merupakan hubungan proses sistem dengan lingkungan melalui arus sumber daya seperti pada gambar 2.1.



Gambar 2.1 Sistem Terbuka

Sistem tertutup adalah sistem yang tidak dipengaruhi oleh pihak luar dengan mekanisme pengendalian dapat dilihat pada gambar 2.2.



Gambar 2.2 Sistem Tertutup

Sistem informasi adalah suatu sistem didalam suatu organisasi yang merupakan kombinasi dari orang-orang, fasilitas, teknologi, media, prosedur-prosedur dan pengendalian yang ditujukan untuk mendapatkan jalur komunikasi penting, memproses tipe transaksi rutin tertentu, memberi sinyal kepada manajemen dan yang lainnya terhadap kejadian-kejadian internal dan eksternal yang penting dan menyediakan suatu dasar informasi untuk pengambilan keputusan yang cerdas. Dalam sistem informasi, arus informasi harus diklasifikasikan. Hal ini dikarenakan keragaman kebutuhan informasi oleh pengguna informasi.

2.3 Presensi

Absensi atau dengan kata lain presensi adalah suatu kegiatan yang bertujuan untuk mengetahui tingkat keikutsertaan kegiatan dan disiplin anggota suatu instansi, lembaga atau perusahaan pemerintah (R. Gunawan et al., 2021). Presensi pegawai merupakan alat untuk menghitung kehadiran pegawai dengan tujuan untuk mengetahui data yang berkaitan dengan kehadiran masuk dan keluar pegawai secara periodik baik harian maupun bulanan (Sikumbang et al., 2020).

Presensi *online* merupakan pencatatan kehadiran menggunakan sistem *cloud* yang terhubung menggunakan *database* secara *realtime*. Dengan aplikasi berbasis *cloud* pada manajemen data presensi karyawan ini, presensi *online* bisa dilakukan dengan mengunggah foto *selfie* oleh karyawan, dan sanggup mencatat jam kerja secara *realtime* pula memantau kegiatan kehadiran karyawan menggunakan efektif dan efisien (Buana et al., 2022).

2.4 *Android*

Android adalah sistem operasi berbasis *Linux* untuk perangkat seluler yang mencakup sistem operasi, *middleware*, dan aplikasi. *Android* menawarkan pengembang *platform* terbuka untuk membangun aplikasi. *Android* dapat berjalan di beberapa perangkat dari produsen yang berbeda. *Android* mencakup pengembangan perangkat lunak untuk menulis kode asli dan merakit modul perangkat lunak untuk membuat aplikasi bagi pengguna *Android*. *Android* juga menawarkan pasar untuk distribusi aplikasi. Secara umum, *Android* adalah ekosistem aplikasi seluler. Versi *Android* yang kompatibel dan digunakan dalam penelitian ini adalah dimulai dari versi 8.0 sampai versi *Android* terbaru saat ini (Karman et al., 2019). Perangkat *Android* memiliki banyak fitur perangkat keras di dalamnya. Inilah yang dapat dimanfaatkan *developer* dalam membangun aplikasi, diantaranya adalah :

- a. *Touchscreen*. Perangkat *Android* memiliki fitur layar sentuh yang memberikan beberapa kemungkinan bagi pengguna untuk berinteraksi dengan aplikasi menggunakan jari.
- b. *GPS*. Sistem operasi *Android* mendukung *GPS*, yang memungkinkan pengembang mengakses lokasi pengguna.
- c. *Accelerometer*. *Android* mendukung *accelerometer*, yaitu perangkat yang digunakan untuk mengatur percepatan.
- d. *SD Card*. *Android* memiliki fitur yang memungkinkan pengguna atau aplikasi untuk mengakses (menyimpan atau membuka) *file* pada *SD Card*.

Android memiliki banyak fitur perangkat lunak yang dapat digunakan pengembang untuk mengembangkan aplikasi seperti :

- a. Internet. Kemampuan akses internet pada *Android* memiliki banyak keunggulan. Berbagai informasi secara *real-time* dapat diperoleh dengan mudah dengan internet.
- b. Dukungan Audio dan Video. Sistem operasi *Android* memungkinkan *developer* menyertakan audio dan video dalam aplikasi dengan mudah.
- c. *Contact*. *Android* memungkinkan akses ke kontak yang tersimpan pada perangkat.
- d. *Security*. *Android* memungkinkan aplikasi melakukan banyak hal. *Android* juga menyiapkan keamanan berupa *permission* berkaitan dengan beberapa tugas.
- e. *Google APIs*. Sistem operasi *Android* memungkinkan dengan tidak terbatas membuat panggilan telepon, mengorganisasi kontak dan instal aplikasi. *Developer* juga dapat mengintegrasikan peta ke dalam suatu aplikasi dengan menggunakan *Maps API*.

2.5 Kotlin

Kotlin adalah bahasa yang kuat yang pasti cocok untuk pengembang untuk membuat aplikasi *Android*. Karena *Kotlin* adalah bahasa nomor satu untuk mengembangkan aplikasi *Android*. Kelebihan *Kotlin* adalah (Aljundi & Akbar, 2018) :

- a. *Concise* : *Kotlin* mampu mengurangi *boilerplate of code* atau tingkat kerumitan dari kode yang biasa kita tulis, ketika menggunakan bahasa *Java*.
- b. *Safe* : *Kotlin* mampu menjamin bahwa setiap *syntax* yang kita tulis secara proses kompilasi dapat mencegah kemungkinan terjadinya *error*, misalnya mampu mencegah terjadinya *NullPointerException* ketika kita *coding* menggunakan bahasa *Java*.

- c. *Versatile* : *Kotlin* sebenarnya sama dengan *Java* karena *Kotlin* sendiri berasal dari bahasa induknya yaitu *Java*. Artinya, *Kotlin* juga dapat digunakan dalam pengembangan situs web dan aplikasi seluler.
- d. *Interoperable* : *Kotlin* tidak sama dengan bahasa *Java* turunan lainnya (misalnya *Scala* atau *Clojure*), yang tidak bisa berjalan bersamaan dengan kode yang kita tulis di *Java*. *Kotlin* dapat membaca kode lama atau *library* yang kita gunakan atau tulis di *Java* dan sebaliknya.

2.6 Database

Database adalah kumpulan data yang dimuat ke komputer sehingga dapat dikelola oleh program komputer. Menurut konsepnya, *database* adalah kombinasi data yang mengatur repositori atau *file* yang berinteraksi satu sama lain dengan beberapa metode untuk memuat data atau informasi baru (Subandi & Syahidi, 2018). Konsep lain dari basis data adalah kumpulan data yang berinteraksi satu sama lain dan diatur menurut model atau sistem tertentu (Rosa & Shalahuddin, 2018). Adapun struktur *database* terdiri dari :

a. Data

Sekumpulan fakta mengenai objek tertentu, orang dan lain-lain yang dinyatakan dengan angka, huruf, gambar, film, suara dan sebagainya yang relevan dan belum mempunyai arti.

b. Informasi

Hasil pengolahan data yang konkrit dan sudah mempunyai arti untuk mencapai suatu tujuan tertentu.

c. Tabel

Merupakan hal yang paling mendasar dalam hal penyimpanan data yang terdiri dari *field* dan *record*.

d. *Field*

Field adalah elemen tabel yang berisi informasi spesifik dari judul tabel elemen data. Syarat untuk membentuk nama *field* dalam tabel adalah :

1. Harus unik atau spesifik
2. Boleh disingkat

Pemisah sebagai pengganti spasi dalam pembentuk *field* adalah tanda lambang “_”. Contoh: Kode Barang menjadi KdBarang, KodeBrg, Kd_Brg, Kd_Barang.

Ada tiga jenis bidang dalam sistem manajemen basis data :

1. Harus diisi (wajib)
2. Bisa dihilangkan (pilihan)
3. Perhitungan dari *field* lain (dihitung). Pengguna tidak dapat memasukkan data di bidang terakhir (dihitung).

e. *Query*

Kueri adalah pertanyaan atau permintaan untuk menerima informasi spesifik dari *database* dalam format tertentu. Ada tiga metode utama untuk membuat *query*, yaitu :

1. *Query by example* (QBE) adalah metode *query* yang disediakan oleh sistem dalam bentuk record kosong dan pengguna dapat menentukan *field* dan nilai tertentu yang akan digunakan dalam *query*.

2. Bahasa kueri adalah bahasa khusus yang digunakan untuk membuat kueri terhadap basis data. Metode ini adalah yang paling kompleks, tetapi juga paling fleksibel.

f. *Record*

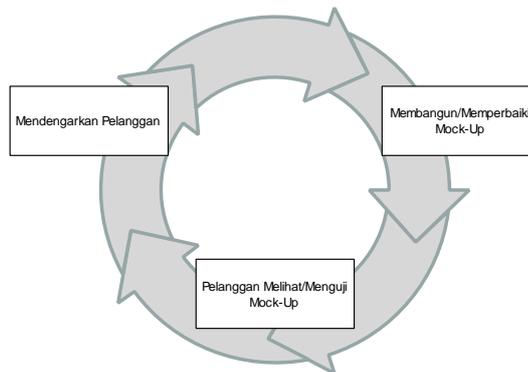
Sekumpulan data yang saling berkaitan tentang sebuah subjek tertentu, misalnya data seorang siswa akan disimpan dalam *record* yang terdiri dari beberapa kolom/*field*. Struktur *database* adalah cara data diorganisasi agar pemrosesan data menjadi lebih efisien. Sebuah sistem manajemen basis data (DBMS) adalah perangkat lunak yang menyimpan struktur basis data itu sendiri, hubungan antara data dalam basis data, dan nama formulir, tipe data, desimal, jumlah karakter, nilai *default*, dan semua deskripsi lapangan lainnya.

2.7 Metode Prototipe

Prototipe merupakan metode yang efektif dalam merancang perangkat lunak. Prototipe dimulai dengan mengumpulkan kebutuhan yang akan di rancang. Pengembang mendefinisikan objek keseluruhan dari perangkat lunak, mengidentifikasi segala aktifitas yang diketahui dan kemudian melakukan “perancangan kilat”. Dalam desain kilat, fokusnya adalah mewakili bidang perangkat lunak yang akan terlihat oleh klien atau pengguna (misalnya, metode masukan dan format keluaran).

Metode prototipe merupakan metode pengembangan sistem dimana hasil analisa perbagian sistem langsung diterapkan kedalam sebuah model tanpa menunggu seluruh sistem selesai. *Mockup* adalah sesuatu yang digunakan sebagai *template* gambar untuk pengajaran, demonstrasi, evaluasi desain, periklanan, atau tujuan lain. Sebuah model disebut prototipe perangkat lunak ketika menyediakan

atau dapat mendemonstrasikan sebagian besar fungsionalitas sistem perangkat lunak dan memungkinkan pengujian desain sistem perangkat lunak. Iterasi terjadi pada pembuatan prototipe sampai sesuai dengan keinginan pelanggan atau pengguna (Rosa & Shalahuddin, 2018).



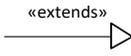
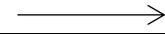
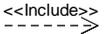
Gambar 2.3 Pemodelan Prototipe

2.8 Pemodelan Sistem

2.8.1 Use Case Diagram

Use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu. Syarat penamaan pada *use case* adalah nama didefinisikan sesimpel mungkin dan dapat dipahami. Ada dua hal utama pada *use case* yaitu pendefinisian apa yang dibuat aktor dan *use case*. Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi, walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang (Rosa & Shalahuddin, 2018).

Tabel 2.2 Simbol *Use Case Diagram*

Simbol	Deskripsi
<i>Use case</i> 	Fungsionalitas yang disediakan sistem sebagai unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja diawal frase nama <i>use case</i>
Aktor 	Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar itu sendiri.
Asosiasi 	Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.
Ekstensi 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> , dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan.
Generalisasi 	Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
<i>Include</i> 	<i>Include</i> berarti <i>use case</i> yang ditambahkan akan selalu dipanggil saat <i>use case</i> tambahan dijalankan

2.8.2 Activity Diagram

Diagram aktivitas menggambarkan alur kerja atau fungsi dari suatu sistem atau proses bisnis. Perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem (Rosa & Shalahuddin, 2018).

Tabel 2.3 Simbol *Activity Diagram*

Simbol	Deskripsi
Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
Percabangan 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
Penggabungan 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
<i>Swimlane</i> 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.
Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.

2.9 Pengujian ISO/IEC 25010

ISO/IEC 25010 adalah standar evaluasi perangkat lunak yang diterbitkan oleh Canadian Standards Association pada tahun 2011. ISO/IEC 25010 adalah model baru dalam seri ISO/IEC 250n, yang merupakan pengembangan dari versi ISO/IEC 9126. Model 25010 memiliki fitur kualitas yang dapat dipertimbangkan saat mengevaluasi perangkat lunak (Mulyawan et al., 2021). Secara umum, ISO/IEC 25010 memiliki delapan karakteristik yang mengukur kualitas perangkat lunak secara keseluruhan, termasuk portabilitas, efisiensi, keandalan, kegunaan keamanan, pemeliharaan, kompatibilitas, dan kesesuaian fungsional. Beberapa definisi karakteristik ISO/IEC 25010 adalah sebagai berikut :

- a. *Functional suitability* adalah produk aplikasi yang menyediakan fungsionalitas yang memenuhi persyaratan penggunaan produk dalam kondisi tertentu.
- b. *Reliability* mengacu pada sejauh mana produk aplikasi dapat mempertahankan tingkat kinerja tertentu bila digunakan dalam kondisi tertentu.
- c. *Performance efficiency* adalah tingkat di mana produk aplikasi berperforma baik dengan sumber daya yang digunakan.
- d. *Usability* berarti produk aplikasi mudah dipahami, mudah digunakan, dan menarik untuk digunakan.
- e. *Security* adalah tingkat produk aplikasi menyediakan layanan untuk melindungi akses, penggunaan, modifikasi, pengrusakan, atau pengungkapan yang berbahaya.
- f. *Compatibility* adalah kemampuan satu atau lebih komponen aplikasi untuk bertukar data.
- g. *Maintainability* adalah tingkat dimana produk aplikasi dapat dimodifikasi. Modifikasi yang dilakukan dapat meliputi perbaikan, pengembangan atau adaptasi perangkat lunak untuk menyesuaikan dengan lingkungan, serta modifikasi pada kriteria dan spesifikasi fungsi.
- h. *Portability* mengacu pada sejauh mana produk aplikasi dapat dipindahkan dari satu keadaan ke keadaan lain.