

## BAB 2

### TINJAUAN PUSTAKA DAN DASAR TEORI

#### 2.1 Tinjauan Pustaka

Tinjauan kepustakaan menjadi acuan utama dalam beberapa studi yang telah dilakukan terkait dengan penelitian ini. Terdapat beberapa penelitian yang digunakan sebagai rujukan dalam penelitian yang tercantum dalam tabel 2.1.

**Table 2.1** Perbandingan Penelitian terdahulu

<b>Nama Penulis</b>	<b>Judul</b>	<b>Teknologi</b>	<b>Hasil</b>
Visa Soininen (2021)	<i>Jetpack Compose vs React Native – Differences in UI Development</i>	<i>Jetpack Compose, React Native</i>	Hasil pengujian kinerja menunjukkan bahwa <i>Jetpack Compose</i> memiliki kinerja yang lebih baik daripada <i>React Native</i> dalam beberapa kasus. Namun, kedua kerangka kerja ini memiliki kelebihan dan kekurangan masing-masing.
Beslam Asefa (2022)	<i>Building Android Component Library Using Jetpack Compose</i>	<i>Jetpack Compose</i>	Penelitian ini mengungkapkan keuntungan <i>Jetpack Compose</i> dalam membangun <i>library</i> komponen, seperti penghematan waktu dan biaya pengembangan. Selain itu, dipaparkan arsitektur <i>Jetpack Compose</i> dan komponen-komponen UI yang disediakan oleh <i>library</i> tersebut.
Nandang Hermanto, Nurfaizah dan Nur Rahmat Dwi Riyanto (2019)	Aplikasi sistem Presensi Mahasiswa berbasis Android	<i>QR Code</i>	Aplikasi presensi mahasiswa berbasis Android yang memungkinkan penggunaannya untuk merekam data kehadiran mahasiswa dengan cepat. Dosen dapat mengetahui rekapitulasi kehadiran mahasiswa setelah perkuliahan selesai. Sistem

Nama Penulis	Judul	Teknologi	Hasil
			ini juga dapat memberikan rekapitulasi kehadiran mahasiswa secara keseluruhan.
Arafat Febriandirza (2020)	Perancangan Aplikasi Absensi Online dengan Menggunakan Bahasa pemrograman Kotlin	GPS	Aplikasi absensi berbasis android yang dirancang untuk melakukan absensi karyawan dengan fitur yang memanfaatkan GPS pada masing-masing <i>smartphone</i> karyawan.
Muhammad Himyar, Muhammad Femy Mulya, Johny Hizkia Siringo Ringo (2021)	Aplikasi Absensi Karyawan berbasis Android dengan penerapan <i>QR Code</i> disertai foto diri dan lokasi sebagai validasi: Studi Kasus PT.Selindo Alpha	<i>QR Code, GPS</i>	Aplikasi absensi karyawan berbasis Android yang memungkinkan pengguna untuk melakukan absensi melalui <i>QR Code</i> disertai dengan foto diri dan menggunakan LBS ( <i>Location Based System</i> ) secara online. Aplikasi ini dikembangkan menggunakan bahasa pemrograman Java dan menggunakan <i>MySql</i> sebagai <i>database</i> .
Bayu Surachmad, Haris Yuana, Ni'ma Kholila (2021)	Perancangan Aplikasi Absensi Karyawan Berdasarkan <i>QR Code</i> Berbasis Android Pada Pt. Mandiri Utama Finance	<i>QR Code, GPS</i>	Aplikasi absensi karyawan berbasis Android yang dirancang menggunakan <i>QR Code</i> serta dilengkapi dengan adanya penggunaan <i>location based service</i> (LBS).
Muhammad Rakha, Mercy Hermawati, Nurfidah Dwitiyanti (2022)	Sistem Absensi Menggunakan <i>QR Code Scanner</i> Berbasis Android Pada Pt. Indobara Bahana	<i>QR Code</i>	Aplikasi Absensi berbasis Android dengan kotlin sebagai Bahasa pemrograman utamanya dan terdapat fitur <i>QR Code</i> untuk memudahkan karyawan dalam proses absensi serta admin dapat membuat laporan absensi dengan mudah dan efisien agar lebih tepat dan akurat.

Penelitian yang dilakukan oleh Visa Soininen (2021) adalah membuat “*Jetpack Compose vs React Native – Differences in UI Development*”. Tujuan penelitian ini untuk mengetahui perbedaan antara kedua kerangka kerja tersebut

dalam hal pengembangan aplikasi dan kinerja. Hasil dari penelitian menunjukkan bahwa *Jetpack Compose* memiliki kinerja yang lebih baik daripada *React Native* dalam beberapa kasus. Namun, kedua kerangka kerja ini memiliki kelebihan dan kekurangan masing-masing. Kesimpulannya, penggunaan *Jetpack Compose* atau *React Native* tergantung pada kebutuhan dan preferensi pengembang. Penelitian ini dapat membantu pengembang dalam memilih kerangka kerja yang tepat untuk pengembangan aplikasi mereka.

Penelitian yang dilakukan oleh Beselam Asefa (2022) adalah membuat "*Building Android Component Library Using Jetpack Compose*". Tujuan dari penelitian ini adalah untuk membangun sebuah komponen *library* untuk Android menggunakan *Jetpack Compose*. Penelitian ini bertujuan untuk menunjukkan keuntungan menggunakan *Jetpack Compose* dalam membangun *library* komponen, seperti penghematan waktu dan biaya pengembangan. Selain itu, penelitian ini juga bertujuan untuk menunjukkan bagaimana komponen *library* yang dibangun dapat digunakan oleh Android developers di Elisa, sebuah perusahaan telekomunikasi terkenal di Finlandia. Penelitian ini juga bertujuan untuk menjelaskan tentang arsitektur *Jetpack Compose* dan pendekatan yang dapat digunakan untuk membangun *library* komponen menggunakan *Jetpack Compose*.

Penelitian yang dilakukan oleh Nandang Hermanto, Nurfaizah, dan Nur Rahmat Dwi Riyanto (2019) adalah membuat "Aplikasi Sistem Presensi Mahasiswa Berbasis Android". Tujuan penelitian ini untuk merancang sistem presensi mahasiswa menggunakan Android dengan memanfaatkan *QR Code*. *QR Code* digunakan dalam proses presensi untuk memberikan dan memperoleh informasi

secara cepat. Mahasiswa dapat menggunakan *QR Code Reader* untuk membaca *QR Code*. Penggunaan teknologi ini diharapkan menjadi terobosan dalam perkuliahan. Sistem cerdas ini dibuat untuk mengatasi masalah mahasiswa yang sering menitipkan absen, sehingga penilaian presensi dalam penilaian perkuliahan menjadi lebih baik.

Penelitian yang dilakukan oleh Arafat Febriandirza (2020) adalah membuat “Perancangan Aplikasi Absensi Online dengan Menggunakan Bahasa pemrograman Kotlin”. Tujuan penelitian ini untuk mengatasi beberapa kekurangan sistem absensi secara konvensional dan cenderung tidak efisien. Dalam hal ini, karyawan dapat melakukan absensi menggunakan *smartphone* mereka jika sudah didalam lingkungan perusahaan. Maka akan dibuat sistem absensi karyawan dengan memanfaatkan GPS yang terdapat pada masing-masing *smartphone* karyawan. Diharapkan dengan adanya sistem absensi ini dapat memudahkan perusahaan dalam memantau data kehadiran pegawai dan sistem absensi ini juga dapat memberikan laporan kehadiran sesuai dengan kebutuhan secara efektif dan tepat. Selain itu, aplikasi ini juga dapat meningkatkan disiplin karyawan, mengurangi potensi kecurangan, meningkatkan efisiensi dan akurasi, dapat memantau karyawan yang sering absen, dapat memantau karyawan dalam penugasan, mampu menciptakan lingkungan kerja yang produktif, dan mengetahui posisi karyawan yang sedang bekerja.

Penelitian yang dilakukan oleh Muhammad Himyar, Muhamad Femy Mulya, Johny Hizkia Siringo Ringo (2020) adalah membuat “Aplikasi Absensi Karyawan berbasis Android dengan penerapan *QR Code* disertai foto diri dan lokasi sebagai

validasi: Studi Kasus PT.Selindo Alpha”. Tujuan penelitian ini untuk mempermudah proses absensi karyawan pada PT Selindo Alpha dengan menggunakan teknologi *QR Code*, foto diri, dan lokasi sebagai validasi. Selain itu, aplikasi ini juga bertujuan untuk meringankan pekerjaan HRD dalam rekap absensi bulanan, meminimalisir terjadinya titip absen pada pegawai, dan mempercepat waktu yang dibutuhkan pegawai untuk input absen. Aplikasi ini juga memiliki kelebihan dalam menentukan *range* lokasi absen sehingga user tidak bisa absen pada sembarang lokasi. Namun, aplikasi ini juga memiliki kekurangan seperti harus menambahkan lokasi secara manual dalam database ketika ada lokasi kantor cabang baru, tidak adanya *alert/notifikasi* ketika user terlambat untuk input absen, dan tidak ada fitur pengajuan cuti berbasis android dalam aplikasi. validasi kehadiran karyawan di PT.Selindo Alpha dan MySQL sebagai penyimpanan datanya.

Penelitian yang dilakukan oleh Bayu Surachmad, Haris Yuana, Ni'ma Kholila (2021) adalah membuat “Perancangan Aplikasi Absensi Karyawan Berdasarkan *Qr Code* Berbasis Android Pada Pt. Mandiri Utama Finance”. Tujuan penelitian ini untuk meningkatkan efisiensi dan efektivitas kegiatan presensi karyawan di perusahaan. Dalam kondisi pandemi *Covid-19*, penggunaan *finger print* untuk kegiatan presensi dianggap tidak efisien dan berpotensi menimbulkan penyebaran virus. Selain itu, kegiatan presensi manual dengan menggunakan buku daftar hadir karyawan juga rentan terjadi kecurangan dan kesalahan. Oleh karena itu, perusahaan memanfaatkan kemajuan teknologi dengan menggunakan aplikasi absensi melalui *QR Code* pada smartphone android milik masing-masing karyawan. Dengan aplikasi ini, diharapkan pengolahan data absensi karyawan menjadi lebih

efektif dan efisien, serta dapat mengurangi risiko kecurangan dan kesalahan dalam kegiatan presensi.

Penelitian yang dilakukan oleh Muhammad Muhammad Rakha, Mercy Hermawati, Nurfidah Dwitiyanti (2022) adalah membuat “Sistem Absensi Menggunakan *Qr Code Scanner* Berbasis Android Pada Pt. Indobara Bahana”. Tujuan penelitian ini untuk memudahkan proses absensi karyawan dan mengelola data absensi menjadi sebuah laporan yang terkomputerisasi. Dalam masa pandemi seperti sekarang, aplikasi ini juga memungkinkan karyawan untuk melakukan absensi menggunakan smartphone mereka sendiri, sehingga dapat menjaga jarak dan mengurangi risiko penularan virus *Covid-19*. Selain itu, aplikasi ini juga diharapkan dapat meningkatkan efektivitas dan efisiensi pekerjaan rekapitulasi absen pada karyawan.

## **2.2 Dasar Teori**

Dasar teori merupakan landasan atau pemahaman yang menjadi dasar dalam suatu studi atau penelitian. Pada dasar teori, akan dijelaskan pemahaman yang terkait dengan penelitian dengan sumber yang jelas.

### **2.2.1 Absensi**

Absensi adalah suatu bentuk pendataan presensi atau kehadiran seseorang atau pegawai yang merupakan bagian pelaporan dari suatu institusi yang berisi data – data status kehadiran yang disusun dan diatur secara rapi dan mudah untuk dicari, dan digunakan apabila sewaktu-waktu diperlukan oleh pihak yang berkepentingan (Aji Afriansyah & Ari Syaripudin, 2022). Absensi juga merupakan pola rutinitas ketidakhadiran dari tugas atau kewajiban. Absensi dapat dianggap sebagai masalah

administratif jika dilihat dari sudut pandang ekonomi (Haryo Kusumo , Moh. Muthohir & Sindhu Rakasiwi, 2022).

Absensi disusun dan diatur sehingga mudah untuk dicari dan dipergunakan ketika diperlukan oleh pihak yang berkepentingan. Secara umum, jenis-jenis absensi menurut cara penggunaannya dapat dikelompokkan menjadi dua, yaitu:

1. Absensi secara manual: Menginput absensi atau kedatangan menggunakan pena melalui tanda tangan dan kertas.
2. Absensi Non Manual : Cara memasukkan absensi atau kedatangan menggunakan perangkat atau sistem komputerisasi seperti menggunakan kartu dengan kode batang (*barcode*) dan pengambilan sidik jari (*finger print*).

Absensi adalah hal terpenting dalam mendukung aktivitas dalam pekerjaan atau kuliah. Sebelum mulai melakukan sesuatu atau setelah itu, setiap orang atau karyawan diminta mengisi kehadiran atau kehadiran sebagai bukti bahwa mereka telah hadir di sebuah kegiatan (Brillyan Putra Pradana, Mohammad Farizd, Dwi Shahita, Seftin Fitri Ana Wati, 2022).

### **2.2.2 QR Code**

*QR Code* merupakan teknik yang mengubah data tertulis menjadi kode-kode 2 dimensi yang tercetak kedalam suatu media yang lebih ringkas. *QR Code* adalah *barcode* 2 dimensi yang diperkenalkan pertama kali oleh perusahaan Jepang Denso-Wave pada tahun 1994. *Barcode* ini pertama kali digunakan untuk pendataan inventaris produksi suku cadang kendaraan dan sekarang sudah digunakan dalam berbagai bidang. *QR* adalah singkatan dari *Quick Response* karena ditujukan untuk diterjemahkan isinya dengan cepat. *QR Code* merupakan

pengembangan dari *barcode* satu dimensi, *QR Code* salah satu tipe dari *barcode* yang dapat dibaca menggunakan kamera *handphone* (Anita Rahmawati & Arif Rahman, 2011).

*QR Code* mampu menyimpan semua jenis data, seperti data angka/numerik, alphanumerik, biner, kanji/kana. Selain itu *QR Code* memiliki tampilan yang lebih kecil daripada *barcode*. Hal ini dikarenakan *QR Code* mampu menampung data secara horizontal dan vertikal, jadi secara otomatis ukuran dari tampilannya gambar *QR Code* bisa hanya sepersepuluh dari ukuran sebuah *barcode*. Tidak hanya itu *QR Code* juga tahan terhadap kerusakan, sebab *QR Code* mampu memperbaiki kesalahan sampai dengan 30% tergantung dengan ukuran atau versinya. Oleh karena itu, walaupun sebagian simbol *QR Code* kotor ataupun rusak, data tetap dapat disimpan dan dibaca. Tiga tanda berbentuk persegi di tiga sudut memiliki fungsi agar simbol dapat dibaca dengan hasil yang sama dari sudut manapun. *QR Code* memiliki kemampuan mengoreksi kesalahan untuk mengembalikan data jika kode kotor atau rusak. Empat tingkat kesalahan koreksi yang tersedia bagi pengguna, tingkatan ini mampu mengoreksi kesalahan pada *QR Code*. Faktor lingkungan dan ukuran *QR Code* perlu dipertimbangkan untuk mengetahui tingkat kesalahan. Tingkat Q atau H dapat dipilih jika kondisi lingkungan kotor yang akan menyebabkan *QR Code* mengalami kerusakan, sedangkan tingkat L dapat dipilih jika kondisi lingkungan bersih dengan jumlah data yang besar. Tingkat M adalah tingkat yang paling sering digunakan (Anita Rahmawati & Arif Rahman, 2011).

### 2.2.3 Android Studio

Android studio adalah IDE (*Integrated Development Environment*) resmi untuk pengembangan aplikasi Android dan bersifat *open source* atau gratis. Peluncuran Android Studio ini diumumkan oleh Google pada 16 mei 2013 pada event Google I/O Conference untuk tahun 2013. Sejak saat itu, Android Studio menggantikan Eclipse sebagai IDE resmi untuk mengembangkan aplikasi Android. Android studio sendiri dikembangkan berdasarkan IntelliJ IDEA yang mirip dengan Eclipse disertai dengan ADT plugin (Android Development Tools), Android studio memiliki fitur (Andi Juansyah, 2015):

- a. Projek berbasis pada *Gradle Build*
- b. *Refactory* dan pembenahan bug yang cepat
- c. *Tools* baru yang bernama *Lint* diklaim dapat memonitor kecepatan, kegunaan, serta kompetibelitas aplikasi dengan cepat.
- d. Mendukung *Proguard And App-signing* untuk keamanan.
- e. Memiliki GUI aplikasi android lebih mudah
- f. Didukung oleh Google Cloud Platfrom untuk setiap aplikasi yang dikembangkan.

#### 2.2.3.1 Library Zxing (Zebra Crossing)

*Library Zebra crossing* atau sering juga disebut dengan *zxing* adalah sebuah *library open source* yang berfungsi untuk pemrosesan berbagai jenis *barcode* 1 dimensi dan 2 dimensi. *Library zxing* ini berfokus kepada penggunaan kamera pada *smartphone* sebagai pengganti dari alat *scanner*, namun *library* ini juga dapat dipakai pada desktop dan server. *Zxing* dapat melakukan proses *encode* maupun

*decode barcode*. Library *zxing* mendukung berbagai macam format *barcode* antara lain *upc-a*, *upc-e*, *ean-8*, *ean-13*, *code 39*, *data matrix*, *QR Code*, dan *code bar* (Derry Susilo & Marcel, 2015).

### **2.2.3.2 Kotlin**

Kotlin adalah bahasa pemrograman berbasis *Java Virtual Machine* (JVM) yang dikembangkan oleh JetBrains. Kotlin merupakan bahasa pemrograman yang pragmatis untuk android yang mengkombinasikan *object oriented* (OO) dan pemrograman fungsional. Kotlin juga bahasa pemrograman yang interoperabilitas yang membuat bahasa ini dapat digabungkan dalam satu project dengan bahasa pemrograman Java. Bahasa pemrograman ini juga dapat digunakan untuk pengembangan aplikasi berbasis desktop, web dan bahkan untuk backend. Beberapa keuntungan yang mungkin akan didapatkan jika pengembangan aplikasi beralih menggunakan Kotlin untuk mengembangkan aplikasi diatas platform JVM adalah sebagai berikut (Niko Sumanda Sibarani, Ghifari Munawar & Bambang Wisnuadhi, 2018):

1. Dapat mengatasi *NullPointerException* yang umumnya terdapat pada Java.
2. Penulisan kode lebih ringkas dan mudah dibaca dibandingkan kode yang ditulis dengan menggunakan bahasa Java.
3. Mudah dipelajari.
4. Dukungan IDE untuk mempermudah dalam pemrograman.

### **2.2.3.3 Jetpack Compose**

*Jetpack Compose* adalah *framework* yang membantu pengembang menggunakan pengembangan modern dengan memanfaatkan komponen yang dapat digunakan kembali dan menyediakan opsi bawaan untuk

mengimplementasikan tema gelap dan animasi dalam proyek. *Compose* dibangun di atas arsitektur *Jetpack* asli yang memungkinkan pengembang tetap mendapatkan manfaat dari semua fungsionalitas yang tersedia dalam pengembangan Android asli (Visa Soininen, 2021).

Pada sisi lain, XML adalah bahasa markup yang dipakai juga untuk membuat tata letak antarmuka pengguna dalam pengembangan Android. Dalam pendekatan yang mengandalkan XML, antarmuka pengguna dijelaskan melalui hierarki elemen-elemen XML yang merepresentasikan komponen-komponen UI. Tiap elemen XML diberkahi atribut-atribut yang mengatur properti serta perilaku komponen UI. Perbedaan utama antara *Jetpack Compose* dan XML adalah pendekatan yang digunakan dalam mendefinisikan antarmuka pengguna. *Jetpack Compose* menggunakan pendekatan deklaratif dengan menggunakan fungsi-fungsi komposabel, sementara XML menggunakan pendekatan imperatif dengan menggunakan hierarki elemen XML. Beberapa keunggulan yang dimiliki oleh *Jetpack Compose* dibandingkan XML meliputi (Pölkki, 2023):

1. *Less code*: Dalam *Jetpack Compose*, pengembang dapat membuat antarmuka pengguna dengan lebih sedikit kode dibandingkan dengan pendekatan berbasis XML.
2. *Reusability*: Dengan pendekatan ini memungkinkan elemen-elemen antarmuka pengguna untuk lebih mudah digunakan kembali di berbagai bagian aplikasi.
3. *Loose Coupling*: *Jetpack Compose* memastikan bahwa antarmuka pengguna tidak terikat secara ketat dengan lapisan data. Ini mempermudah perawatan

karena perubahan pada data tidak langsung mempengaruhi antarmuka pengguna.

#### **2.2.3.4 Clean Architecture**

Prinsip *clean architecture* dapat meningkatkan tingkat *maintainability* dari suatu sistem. Hal ini terjadi karena ada pemisahan antara komponen pada suatu sistem menjadi beberapa komponen kecil yang independen dan lebih modular sehingga jika terjadi sebuah bug pada sebuah komponen, pengembang dapat fokus memperbaiki bug pada komponen tersebut tanpa memengaruhi komponen lainnya. Penambahan sebuah fitur pada sistem yang menerapkan prinsip *clean architecture* juga sangat mudah dilakukan. Penggunaan *Clean Architecture* pada proyek Android pada umumnya dibagi menjadi tiga layer, yaitu *presentation*, *domain*, dan *data*. Berikut adalah penjelasan mengenai komponen dari setiap layer yang terdapat pada Android *Clean Architecture* (Arief Rahman Fajri & Septia Rani, 2022).

1. *Presentation Layer* berisi UI dan *ViewModel* yang akan mengatur tampilan berdasarkan *update* data terbaru. UI akan bergantung kepada *Use Case*.
2. *Domain Layer* berisi *Entities*, *Use Case*, dan *Repository Interface*. Layer ini merupakan layer inti yang berkaitan dengan bisnis model.
3. *Data Layer* berisi implementasi *Repository* dan *Data Source* yang bisa berasal dari *local data source* atau *remote data source*.

#### **2.2.4 REST API**

REST API adalah sebuah implementasi dari API (*Application Programming Interface*). REST (*Representational State Transfer*) adalah suatu arsitektur metode komunikasi yang menggunakan protokol HTTP untuk pertukaran

data. REST API terdiri dari beberapa komponen yaitu (Muhammad Angga Kawa Perdana, 2018):

#### 1. URL design

REST API diakses menggunakan protokol HTTP, oleh karena itu diperlukan penamaan dan struktur URL yang baik dan mudah dimengerti dalam penggunaannya. URL API biasa disebut sebagai endpoint dalam pemanggilannya.

#### 2. HTTP Verbs

Sebutan metode yang dilakukan ketika melakukan *request* sehingga server mengetahui apa yang ingin *client* dapatkan. Metode ini ada beberapa macam namun yang paling sering dipakai adalah: *GET, POST, PUT, DELETE*.

#### 3. HTTP Response Code

kode yang telah menjadi standar dalam menginformasikan hasil *request* kepada klien. Secara umum terdapat 3 kelompok kode yang paling sering digunakan di REST API yaitu:

- a. 200, yang menandakan request yang dilakukan berhasil.
- b. 400, yang menandakan bahwa request mengalami kesalahan pada sisi klien.
- c. 500, yang menandakan bahwa request mengalami kesalahan pada sisi server.

#### 4. Format response

Setiap *request* yang dilakukan klien akan menerima data response dari server, response tersebut biasanya berupa data XML ataupun JSON. Setelah

mendapatkan data response tersebut barulah client bisa menggunakannya dengan cara parsing (mengurai) data tersebut dan diolah sesuai kebutuhan.