

## BAB II

### TINJAUAN PUSTAKA DAN DASAR TEORI

#### 2.1 Tinjauan Pustaka

Pada penelitian ini digunakan tinjauan pustaka sebagai referensi dalam pembangunan rancangan hingga implementasi *platform* yang akan dibuat. Semua jurnal menggunakan *tools* dan metode yang menjadi acuan pembangunan platform yang digunakan pada tabel di bawah ini :

**Tabel 2.1 Tinjauan Pustaka**

Penulis (Tahun)	Topik Penelitian	Pendekatan/Metode	Hasil
Syaputra, Reno dan Dini Faktasari (2022)	Implementasi Aplikasi Berbasis <i>Android</i> Berbasis <i>Mobile</i> GraphQL API	Studi Literatur	Pertukaran data antara aplikasi klien dan sistem <i>server</i>
Adi Mardian, Thomas Budiman, Rachmawaty Haroen dan Verdi Yasin (2020)	Perancangan Aplikasi Pemantauan Kinerja Karyawan Berbasis <i>Android</i> di PT. Salestrade Corp. Indonesia	Wawancara, Studi Pustaka dan Observasi	Aplikasi yang dapat membantu dan mempermudah bagi Karyawan dalam melakukan laporan pekerjaan dengan cepat dan efisien.
Akhiruddin Pulungan dan Alfa Saleh (2020)	Perancangan Aplikasi Absensi Menggunakan QR Code Berbasis <i>Android</i>	Studi Literatur	Sistem absensi yang dirancang untuk mencatat kehadiran menggunakan <i>smartphone android</i> dan membaca QR Code yang berfungsi sebagai penanda kehadiran.
Titih Aji Kurniawan, Agung Triayudi dan Sari Ningsih (2023)	Implementasi Aplikasi Human Resource Management System (HRMS) Berbasis <i>Web</i> dan <i>Android</i> Menggunakan Metodologi <i>Agile</i>	Observasi, Wawancara dan Studi Literatur	Menu administrator untuk keperluan manajemen dari <i>website</i> dan <i>android</i> bagi pengguna.

**Tabel 2.2 Tinjauan Pustaka (Lanjutan)**

Penulis (Tahun)	Topik Penelitian	Pendekatan/Metode	Hasil
Alam Rahmatulloh, Heni Sulastri dan Rizal Nugroho (2018)	Keamanan RESTful <i>Web Service</i> Menggunakan JSON <i>Web Token</i> (JWT) HMAC SHA-512	Studi Literatur	Penerapan JSON Web Token pada Resfull

## 2.2 Dasar Teori

### 2.2.1 Bahasa Pemrograman Go

Golang (*Google Language*) adalah bahasa pemrograman yang dikembangkan oleh Google pada tahun 2009 bersama dengan Ken Thompson, Robert Griesemer dan Rob Pike. Tujuan pengembangannya adalah untuk menciptakan bahasa yang memiliki keunggulan dalam hal kecepatan, kehandalan, skalabilitas, dan kesederhanaan (Sari & Hidayat, 2022).

Golang merupakan bahasa pemrograman yang menggabungkan keamanan dan efisiensi untuk mengembangkan sistem *open source*. Golang mendukung konkurensi dalam sistem pemrograman melalui implementasi yang sangat sederhana. Golang juga memiliki sistem *garbage collection* yang baik dengan memanfaatkan bantuan *built-in garbage collector process* (Goroutines). Golang termasuk bahasa pemrograman yang andal dan cepat dalam skala besar dan bersifat *clean code* agar tidak membebani sistem (Ari Kristanto et al., 2020).

## 2.2.2 Hypertext Transfer Protokol

HTTP (*Hypertext Transfer Protocol*) merupakan sebuah protocol atau standarisasi pengiriman data antar dua entitas yang sering digunakan Ketika mengakses sebuah *website* (Wire Bagye, 2021). HTTP menerapkan *protocol* dengan format data yang dikirimkan memiliki dua bagian, yaitu *header* dan *message body*. Pada bagian *header* terdapat informasi tambahan mengenai data yang dikirimkan seperti tipe data, status data dan lainnya. Kemudian pada bagian *message body* adalah isi data sesungguhnya yang dikirimkan.

## 2.2.3 RESTful API

RESTful API merupakan suatu pengembangan API (*Application Programming Interface*) mengimplementasikan arsitektur REST. REST kependekan dari *Representational State Transfer*. REST bertujuan untuk menyederhanakan metode dalam melakukan transmisi data pada suatu jaringan (Kristanto, 2020).

Arsitektur RESTful mempunyai beberapa komponen sebagai berikut (Perdana, 2018) :

### 1. URI *Design*

RESTful API menggunakan protokol HTTP dalam mengakses *resource*.

Untuk mengakses *resource* pada API, diperlukan *Uniform Resource Identifier* (URI) atau pada API biasanya disebut *endpoint*.

### 2. HTTP Verbs

Komponen ini bisa disebut juga sebagai metode ketika *client* melakukan *request* pada suatu *endpoint* terhadap *server*. Beragam metode yang terdapat

pada HTTP, namun ada beberapa metode yang sering digunakan yaitu GET, POST, PUT, dan DELETE.

### 3. HTTP *Response Code*

Komponen ini merupakan suatu kode yang sudah menjadi standar pada protokol HTTP ketika *client* melakukan *request*. Terdapat 3 jenis kode yang sering digunakan pada suatu RESTful API diantaranya sebagai berikut:

- a. 2XX, menandakan *request* yang diminta *client* berhasil.
- b. 4XX, menandakan terjadi kesalahan pada sisi *client* saat melakukan *request*.
- c. 5XX, menandakan terjadi kesalahan pada sisi *server* saat melakukan *request*.

### 4. Format *Response*

Setiap *request* yang dilakukan oleh *client*, *server* akan mengembalikan response berupa data atau informasi. Ada 2 format *response* yang umum digunakan pada REST API yaitu *Extensible Markup Language* (XML) ataupun *JavaScript Object Notation* (JSON).

#### 2.2.4 Framework Gin

Gin adalah mikro kerangka kerja berperforma tinggi yang dapat digunakan untuk membangun aplikasi web. Memungkinkan pengembang untuk membuat *Middleware* yang dapat terhubung ke satu atau lebih banyak menangani permintaan atau grup penanganan permintaan (Simeunović, 2023).

### 2.2.5 PostgreSQL

PostgreSQL adalah sistem database *objek-relasional* sumber terbuka yang kuat yang menggunakan dan memperluas bahasa SQL yang dikombinasikan dengan banyak fitur yang menyimpan dan menskalakan beban kerja data yang paling rumit dengan aman. PostgreSQL telah mendapatkan reputasi yang kuat untuk arsitektur, keandalan, integritas data, rangkaian fitur yang kuat, ekstensibilitas, dan dedikasi komunitas *open source* di belakang perangkat lunak untuk secara konsisten memberikan solusi yang berkinerja dan inovatif (Putra, 2020).

### 2.2.6 GORM

*Go Object Relational Mapping* (GORM) adalah sebuah *library* (pustaka) *Object Relational Mapping* (ORM) yang populer dalam bahasa pemrograman Go. GORM menyediakan fitur-fitur yang memudahkan pengembang dalam melakukan operasi-operasi basis data dengan menggunakan bahasa Go. GORM memungkinkan pengembang untuk melakukan pemetaan objek ke tabel (*Object-to-Table*) dan relasi-relasi antar tabel (*Relationships*) secara otomatis, sehingga mengurangi jumlah kode yang perlu ditulis secara manual untuk berinteraksi dengan basis data.

### 2.2.7 JWT

*JSON Web Token* (JWT) adalah sebuah token berbentuk string JSON yang sangat padat (ukurannya), informasi mandiri yang gunanya sendiri untuk melakukan sistem autentikasi dan pertukaran informasi. Karena bentuknya kecil, token JWT dapat dikirim melalui URL, parameter HTTP POST atau di dalam *Header* HTTP, dan juga karena ukurannya yang kecil maka dapat ditransmisikan

dengan lebih cepat. Disebut informasi mandiri karena isi dari token yang dihasilkan memiliki informasi dari pengguna yang dibutuhkan, sehingga tidak perlu *query* ke basis data lebih dari satu kali. Token tersebut dapat diverifikasi dan dipercaya karena sudah di-sign secara digital (Alam Rahmatuloh, 2018).