

## **BAB 2**

### **TINJAUN PUSTAKA**

#### **2.1. Tinjauan Tempat Penelitian**

Dinas Pekerjaan Umum dan Tata Ruang Kabupaten Majalengka dibentuk berdasarkan Peraturan Daerah Kabupaten Majalengka Nomor 14 Tahun 2016 tentang Pembentukan dan Susunan Perangkat Daerah Kabupaten Majalengka.

Susunan dan tipologi Dinas Pekerjaan Umum dan Tata Ruang Kabupaten Majalengka sebagaimana tertuang dalam Pasal (4) Peraturan Bupati Majalengka Nomor 19 Tahun 2016 tentang Kedudukan, Susunan Organisasi, Tugas Pokok Dan Fungsi Dinas Di Lingkungan Pemerintah Kabupaten Majalengka, bahwa susunan dan tipologi Dinas Pekerjaan Umum dan Tata Ruang Kabupaten Majalengka dengan Tipe B yaitu menyelenggarakan urusan pemerintahan bidang pekerjaan umum dan penataan ruang.

Dinas Pekerjaan Umum dan Tata Ruang Kabupaten Majalengka dipimpin oleh seorang Kepala Dinas yang berkedudukan dan bertanggungjawab kepada Bupati melalui Sekretaris Daerah dan mempunyai tugas pokok membantu Bupati melaksanakan urusan pemerintahan dibidang pekerjaan umum meliputi Pekerjaan Umum dan Tata Ruang, dan Bangunan yang menjadi kewenangan daerah dan tugas pembantuan yang diberikan kepada Daerah [4].

##### **2.1.1. Visi dan Misi**

###### **1. Visi**

Terwujudnya Pemenuhan Prasarana Kebinamargaan Keciptakaryaan yang Berkelanjutan Untuk Mendukung Majalengka Makmur

###### **2. Misi**

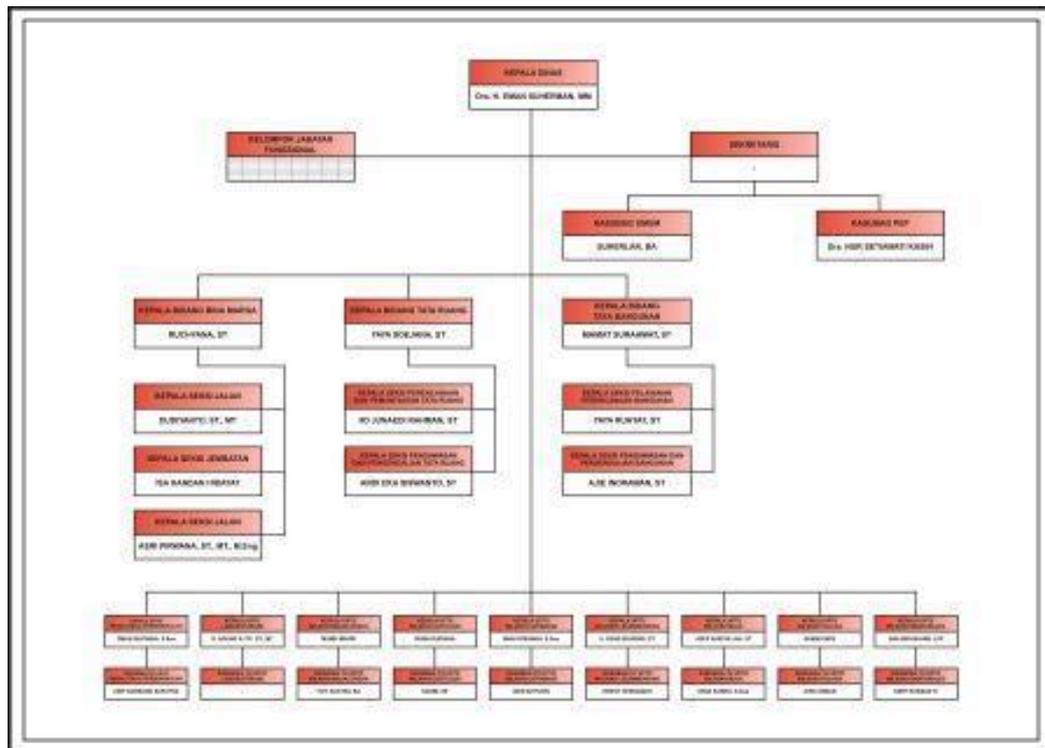
- a. Meningkatkan aksesibilitas dan mobilitas wilayah dalam mendukung pertumbuhan ekonomi untuk meningkatkan kesejahteraan masyarakat dengan penyediaan jaringan jalan yang andal, terpadu dan berkelanjutan

- b. Meningkatkan pembangunan infrastruktur kebinamargaan dan keciptakaryaan berbasis penataan ruang sebagai acuan pembangunan Kabupaten Majalengka berkelanjutan
- c. Meningkatkan Kualitas Sumberdaya Manusia serta Sarana Kebinamargaan dan Keciptakaryaan

### **2.1.2. Struktur Organisasi**

Struktur organisasi merupakan suatu kerangka dalam manajemen organisasi agar dapat berjalan sebagaimana mestinya sesuai dengan tujuan pertama kali berdirinya perusahaan/organisasi tersebut. Struktur organisasi memuat tatanan kerja yang ada di suatu perusahaan tersebut yang menyangkut tugas, tanggung jawab dan wewenang. Semuanya ada agar tercipta hubungan yang baik lebih dinamis dan masing – masing mempunyai tugas dan tanggung jawab sendiri juga mempunyai rasa hormat terhadap yang lain baik itu berkedudukan diatas, dibawah atau setingkat. Tidak berjalannya struktur organisasi dengan baik akan mengarahkan perusahaan tersebut pada kehancuran maka dari itu struktur organisasi merupakan faktor penting dalam suatu perusahaan.

Struktur organisasi merupakan suatu bentuk susunan keanggotaan yang membedakan jabatan masing – masing personil didalam suatu perusahaan, manfaat lainnya dengan dibuatnya struktur organisasi dalam proses kerja. Pegawai perlu mengetahui struktur organisasi perusahaan, agar dapat melakukan kegiatan yang sesuai dengan fungsi dan tidak menyimpang dari yang telah ditentukan, sehingga pegawai mengetahui hak dan kewajibannya didalam perusahaan tersebut. Berikut adalah gambar struktur organisasi Dinas Pekerjaan Umum dan Tata Ruang Kabupaten Majalengka.



**Gambar 2.1 Struktur Organisasi**

## 2.2. Landasan Teori

Pada bab ini akan dijelaskan konsep dasar dan teori-teori yang mendukung pembangunan aplikasi. Selain itu, pada bab ini juga dijelaskan tools-tools yang dipakai dalam pembangunan Aplikasi deteksi kerusakan jalan berbasis android.

### 2.2.1. Aplikasi

Aplikasi adalah suatu subkelas dari suatu perangkat lunak komputer yang memanfaatkan kemampuan komputer secara langsung untuk melakukan suatu tugas yang diinginkan pengguna (Wikipedia, 2012). Aplikasi dapat juga dikatakan sebagai penerjemah perintah-perintah yang dijalankan pengguna komputer untuk diteruskan ke atau diproses oleh perangkat keras. Menurut Marimin dkk. (2011:43) Aplikasi merupakan program yang secara langsung dapat melakukan proses-proses yang digunakan dalam komputer oleh pengguna. Aplikasi merupakan kumpulan

dari file-file tertentu yang berisi kode program yang menghubungkan antara pengguna dan perangkat keras Komputer [5].

### **2.2.2. Sistem Informasi**

Sistem adalah seperangkat elemen-elemen yang membentuk suatu kumpulan dari berbagai prosedur atau berbagai bagan pengolahan untuk mencari sebuah tujuan bersama dengan cara mengoperasikan data maupun barang untuk menghasilkan suatu informasi [6].

Informasi adalah sekumpulan data yang telah diolah dan digunakan untuk memberikan pengetahuan atau pengambilan keputusan bagi penerimanya. Informasi berfungsi untuk memberikan gambaran tentang suatu permasalahan, sehingga pengambil keputusan dapat menentukan keputusan lebih tepat.

Sistem informasi dapat diartikan sebagai perangkat lunak yang digunakan untuk mengolah dan menganalisa kumpulan data. Mengolah data dapat berupa mengumpulkan, menyimpan, lalu memproses data hingga menjadi informasi. Tujuan dari sistem informasi adalah untuk mengubah data mentah menjadi informasi yang bermanfaat bagi penerimanya.

### **2.2.3. Basis Data**

Database atau basis data merupakan sebuah koleksi atau kumpulan dari data yang bersifat mekanis, terbagi, terdefinisi secara formal serta terkontrol. Pengontrolan dari sistem database tersebut adalah terpusat, yang biasanya dimiliki dan juga dipegang oleh suatu organisasi [7].

Pemanfaatan basis data untuk pengolahan data, juga memiliki tujuan – tujuan lain. Secara lengkap tujuan pemanfaatan basis data adalah sebagai berikut :

1. Kecepatan dan Kemudahan (*speed*).

Dengan menggunakan basis data pengambilan informasi dapat dilakukan dengan cepat dan mudah. Basis data memiliki kemampuan dalam mengelompokan, mengurutkan bahkan perhitungan dengan matematika.

2. Kebebasan (*data independence*).

Jika sebuah program telah selesai dibuat, dan ternyata ada perubahan isi atau struktur data. Maka dengan basis data, perubahan ini hanya perlu dilakukan pada level DBMS tanpa harus membongkar kembali program aplikasinya.

3. Keakuratan (*accuracy*).

Penerapan secara ketat aturan tipe data, domain data, keunikan data, hubungan antar data, dan lain-lain menekan ketidakakuratan dalam pemasukan atau penyimpanan data.

4. Ketersediaan (*availability*).

Dengan basis data kita dapat mem-backup data, memilah-milah data mana yang masih diperlukan dan data mana yang perlu disimpan ke tempat lain. Hal ini mengingat pertumbuhan transaksi suatu organisasi dari waktu ke waktu membutuhkan media penyimpanan yang semakin besar.

5. Keamanan (*security*).

Kebanyakan DBMS dilengkapi dengan fasilitas manajemen pengguna. Pengguna diberi hak akses yang berbeda-beda sesuai dengan kepentingan dan posisinya. Basis data bisa diberikan password untuk membatasi orang yang mengaksesnya.

6. Kebersamaan Pemakaian (*sharability*).

Karena cukup dengan satu basis data untuk banyak keperluan, pengontrolan terhadap data juga cukup dilakukan disatu tempat saja. Jika ada perubahan data alamat mahasiswa misalnya, maka tidak perlu kita memperbarui semua data di masing-masing bagian, tetapi cukup hanya disatu basis data.

#### **2.2.4. Sistem Basis Data**

Sistem basis data pada dasarnya dapat dianggap sebagai tempat atau lokasi untuk sekumpulan berkas data yang sudah terkomputerisasi dengan tujuan untuk memelihara informasi, dan juga memuat informasi tersebut, terutama apabila informasi tersebut sedang dibutuhkan [7]. Komponen utama sistem basis data adalah:

1. Perangkat Keras (*Hardware*).
2. Sistem Operasi (*Operating System*).
3. Basis data (*Database*).
4. Sistem Pengelola Basis Data (*Database Management System* atau disingkat *DBMS*).
5. Pemakai (*User*).
6. Aplikasi yang lainnya.

Adapun fungsi dari sistem basis data adalah untuk mengatasi masalah-masalah pemrosesan data yang sering dialami dengan cara konvensional, diantaranya :

1. Redudansi data dan inkonsistensi data.
2. Kesulitan pengaksesan data.
3. Data Isolation.
4. Konkurensi pengaksesan.
5. Masalah keamanan.
6. Masalah Integritas.

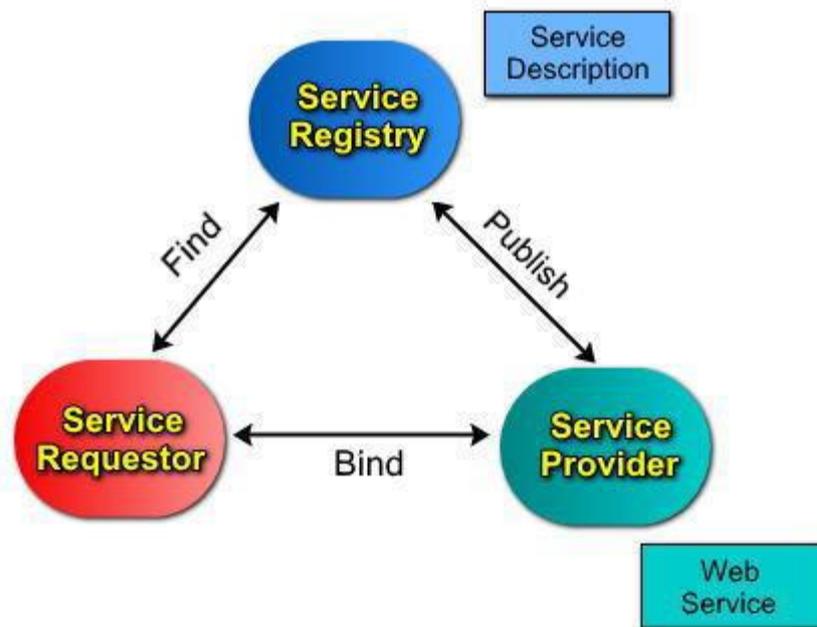
### **2.2.5. Web Service**

*Web service* adalah salah satu sistem perangkat lunak yang didesain untuk mendukung interaksi mesin-mesin melalui jaringan. *Web service* memiliki interface yang dideskripsikan dalam format yang dapat dibaca oleh mesin. *Web service* menyimpan data informasi dalam format *XML*, sehingga data ini dapat diakses oleh sistem lain walaupun berbeda platform, sistem operasi, maupun bahasa compiler. Namun dalam pengembangannya *XML* tidak lagi menjadi satu-satunya format pertukaran data, *JSON(Java Script Object Notation)* merupakan salah satu format pertukaran data yang dapat diimplementasikan pada *web service* selain *XML* [8].

#### **2.2.5.1. Arsitektur Web Service**

*Web service* dibangun dari tiga komponen utama yaitu *service provider*, *service registry*, dan *service requestor*. Komponen-komponen tersebut saling berinteraksi melalui artifak *web service*, yang berupa deskripsi dan implementasi

layanan. Terdapat macam tiga operasi yang memungkinkan komponen-komponen tersebut untuk dapat saling berinteraksi, yaitu publish, find, dan bind.



**Gambar 2.2** Arsitektur Web Service

Pada Gambar Di atas terdapat komponen utama, Yaitu:

1. *Service Provider*, merupakan komponen yang menyediakan akses operasi layanan *web service*.
2. *Service Requestor*, platform yang bertindak sebagai client dari *web service* dan memberikan akses layanan.
3. *Service Registry*, merupakan tempat dimana *service provider* mempublikasikan layanannya. Pada arsitektur *web service*, *service registry* bersifat optional. Teknologi *web service* memungkinkan kita dapat menghubungkan berbagai jenis software yang memiliki platform dan sistem operasi yang berbeda.

### 2.2.6. Unified Modeling Language

*Unified Modeling Language (UML)* adalah bahasa spesifikasi standar untuk mendokumentasikan, menspesifikasikan, dan membangun sistem perangkat lunak. *UML* merupakan himpunan struktur dan teknik untuk pemodelan desain program berorientasi objek (*OOP*) serta aplikasinya. *UML* adalah metodologi untuk

mengembangkan sistem *OOP* dan sekelompok perangkat (*tool*) untuk mendukung pengembangan sistem tersebut [9]. *UML* merupakan sistem arsitektur yang bekerja dalam *OOAD* dengan satu bahasa yang konsisten untuk menentukan, visualisasi, konstruksi dan mendokumentasikan artifact yang terdapat dalam sistem. Artifact adalah sepotong informasi yang digunakan atau dihasilkan dalam suatu proses rekayasa *software*. Artifact dapat berupa model, deskripsi atau software.

#### **2.2.6.1. Use Case Diagram**

Use case diagram merupakan interaksi antara pengguna dengan sistem atau sistem eksternal lainnya yang dapat membantu untuk pemetaan sistem yang akan dibangun. Use case merupakan titik awal yang dilakukan untuk setiap aspek pengembangan, desain serta pengujian dan dokumentasi sistem berorientasi objek. Use case menggambarkan persyaratan yang ada dalam sebuah sistem. Use case juga membantu dalam pembangunan tes untuk sebuah sistem [9].

Diagram yang menggambarkan *actor*, *use case* dan relasinya sebagai suatu urutan tindakan yang memberikan nilai terukur untuk aktor. Sebuah *use case* digambarkan sebagai elips horizontal dalam suatu diagram UML use case.

#### **2.2.6.2. Class Diagram**

*Class diagram* merupakan kumpulan class dari sebuah sistem. Sebagai permulaan, class bisa memiliki hubungan satu sama lainnya. Sebuah class bisa menjadi generalisasi dari class lain atau bisa juga berisi objek-objek dari kelas lain tergantung pada seberapa kuat hubungan antara sebuah class dengan class lainnya [9].

*Class diagram* menggambarkan struktur statis dari kelas dalam sistem anda dan menggambarkan atribut, operasi dan hubungan antara kelas. *Class diagram* membantu dalam memvisualisasikan struktur kelas-kelas dari suatu sistem dan merupakan tipe diagram yang paling banyak dipakai.

### **2.2.6.3. Sequence Diagram**

*Sequence diagram* merupakan diagram penting dalam UML yang dikenal sebagai diagram interaksi. *Sequence diagram* adalah diagram interaksi yang paling terkenal. Hal ini dikarenakan *sequence diagram* dapat menunjukkan jenis informasi secara tepat [9].

*Sequence diagram* menjelaskan interaksi objek yang disusun berdasarkan urutan waktu. Secara mudahnya *sequence diagram* adalah gambaran tahap demi tahap, termasuk kronologi (urutan) perubahan secara logis yang seharusnya dilakukan untuk menghasilkan sesuatu sesuai dengan *use case diagram*.

### **2.2.6.4. Activity Diagram**

*Activity diagram* menggambarkan bagaimana sistem yang dibangun dapat mencapai tujuannya. *Activity diagram* menunjukkan tingkat dari tindakan yang dihubungkan bersama untuk mewakili proses yang terjadi pada sebuah sistem. *Activity diagram* merupakan satu-satunya diagram di dalam UML yang menampilkan model proses sebuah sistem [9].

*Activity Diagram* sangat baik dalam pemodelan proses bisnis. Proses bisnis merupakan serangkaian tugas yang terkoordinasi untuk mencapai tujuan bisnis seperti pengiriman pesan pelanggan. *Activity diagram* memungkinkan pengguna untuk menentukan bagaimana sebuah sistem akan mencapai tujuannya.

### **2.2.7. Smartphone**

Smartphone atau ponsel cerdas merupakan salah satu wujud realisasi ubiquitous computing (ubicomp) di mana teknologi tersebut memungkinkan proses komputasi dapat terintegrasi dengan berbagai aktivitas keseharian manusia dengan jangkauannya yang tidak dibatasi dalam satu wilayah atau suatu scope area [10].

Disebut smartphone karena ponsel ini dapat bekerja dengan kemampuan yang mirip dengan komputer. Ponsel cerdas ini biasanya dilengkapi dengan kamera, layanan internet, pemutar musik, layanan untuk membaca dan mengedit sebuah dokumen dan lain-lain yang pada umumnya didukung oleh berbagai macam aplikasi yang dapat diunduh oleh pengguna ponsel. Terdapat berbagai macam

sistem operasi yang dapat ditemukan pada ponsel cerdas. Contoh sistem operasi 21 yang biasa ditemukan pada ponsel cerdas atau smartphone adalah Windows Mobile, Android, serta iOS. Smartphone menjadi dasar perangkat yang digunakan dalam penelitian ini dikarenakan smartphone memiliki banyak keunggulan yang dapat dimanfaatkan bagi pengembang aplikasi.

### **2.2.8. Android**

Android adalah sistem operasi berbasis Linux yang dirancang untuk perangkat seluler layer sentuh seperti komputer pintar dan komputer tablet. Android awalnya dikembangkan oleh Android, Inc., dengan dukungan finansial dari Google yang kemudian membelinya pada tahun 2005. Sistem operasi ini kemudian di rilis pada tahun 2007.

Beberapa fitur utama dari Android antara lain WiFi hotspot, Multi-touch, Multitasking, GPS, accelerometers, support java, mendukung banyak jaringan (GSM/EDGE, IDEN, CDMA, EV-DO, UMTS, Bluetooth, Wi-Fi, LTE & WiMAX) serta juga kemampuan dasar handphone pada umumnya [11].

Banyak smartphone dan PC Tablet menggunakan sistem operasi dengan versi yang berbeda. Semakin tinggi versi, fiturnya semakin canggih dan banyak. Telepon pertama yang memakai sistem operasi Android adalah HTC Dream yang 9 dirilis pada tanggal 22 Oktober 2008. Beberapa uraian versi android seperti dibawah ini :

#### **1. Android Versi 1.1**

Pada 9 Maret 2009, Google merilis Android versi 1.1. Android versi ini dilengkapi dengan pembaruan estetis pada aplikasi, jam alarm, voice search (pencarian suara), pengiriman pesan dengan Gmail, dan pemberitahuan Email.

#### **2. Android Versi 1.5 (Cupcake)**

Pada pertengahan Mei 2009, Google kembali merilis telepon seluler dengan menggunakan Android dan SDK (Software Development Kit) dengan versi 1.5 (Cupcake). Terdapat beberapa pembaruan termasuk juga penambahan beberapa fitur dalam seluler versi ini yakni kemampuan merekam dan menonton video dengan modus kamera, mengunggah video ke Youtube dan gambar ke Picasa langsung dari telepon, dukungan Bluetooth A2DP, kemampuan terhubung secara

otomatis ke headset bluetooth, animasi layar dan keyboard pada layar yang dapat disesuaikan dengan sistem.

### 3. Android Versi 1.6 (Donut)

Donut (versi 1.6) dirilis pada September dengan menampilkan proses pencarian yang lebih baik dibanding sebelumnya, penggunaan baterai indikator dan kontrol applet VPN. Fitur lainnya adalah galeri yang memungkinkan pengguna untuk memilih foto yang akan dihapus pada kamera, camcorder dan galeri yang diintegrasikan pada CDMA / EVDO, 802.1x, VPN, Gestures, dan Text-to-speech engine. Kemampuan dial kontak teknologi text to change speech (tidak tersedia pada semua ponsel).

### 4. Android Versi 2.0/2.1 (Éclair)

Pada 3 Desember 2009 kembali diluncurkan ponsel Android dengan versi 2.0/2.1 (Eclair), perubahan yang dilakukan adalah mengoptimalkan hardware, peningkatan Google Maps 3.1.2, perubahan UI dengan browser baru dan dukungan HTML 5, daftar kontak yang baru, dukungan flash untuk kamera 3.2 MP, digital zoom, dan Bluetooth 2.1. Untuk bergerak cepat dalam persaingan perangkat 10 generasi berikut, Google melakukan investasi dengan mengadakan kompetisi aplikasi mobile terbaik (killer apps-aplikasi unggulan). Kompetisi ini berhadiah \$25,000 bagi setiap pengembang aplikasi terpilih. Kompetisi diadakan selama dua tahap yang tiap tahapnya dipilih 50 aplikasi terbaik. Dengan semakin berkembangnya dan semakin bertambahnya jumlah handset Android, semakin banyak pihak ketiga yang berminat untuk menyalurkan aplikasi mereka kepada sistem operasi Android. Aplikasi terkenal yang diubah ke dalam sistem operasi Android adalah Shazam, Backgrounds dan WeatherBug. Sistem operasi Android dalam situs internet juga dianggap penting untuk menciptakan aplikasi Android asli, contohnya oleh MySpace dan Facebook.

### 5. Android Versi 2.2 (Froyo : Frozen Yoghurt)

Pada 20 Mei 2010, Android versi 2.2 (Froyo) diluncurkan. Perubahan-perubahan umumnya terhadap versi-versi sebelumnya antara lain dukungan Adobe Flash 10.1, kecepatan kinerja dan aplikasi 2 sampai 5 kali lebih cepat, integrasi dengan V8 JavaScript engine yang dipakai Google Chrome yang

mempercepat kemampuan rendering pada browser, pemasangan aplikasi dalam SD Card, kemampuan WiFi Hotspot portabel dan kemampuan auto update dalam aplikasi Android Market.

#### 6. Android Versi 2.3 (Gingerbread)

Pada 6 Desember 2010, Android versi 2.3 (Gingerbread) diluncurkan. Perubahan-perubahan umum yang didapat dari versi ini antara lain peningkatan kemampuan permainan (gaming), peningkatan fungsi copy paste, layar antarmuka (User Interface) didesain ulang, dukungan format video VP8 dan WebM. efek audio baru (reverb, equalization, headphone virtualization, dan bass boost), dukungan kemampuan Near Field Communication (NFC) dan dukungan jumlah kamera yang lebih dari satu.

#### 7. Android Versi 3.1 (Honeycomb)

Android Honeycomb dirancang khusus untuk tablet. Android versi ini mendukung ukuran layar yang lebih besar. User Interface pada Honeycomb juga berbeda karena sudah didesain untuk tablet. Honeycomb juga mendukung multi prosesor dan juga akselerasi perangkat keras (hardware) untuk grafis. Tablet peratam yang dibuat dengan menjalankan Honeycomb adalah Motorola Xoom.

#### 8. Android Versi 4.0 (Ice Cream Sandwich)

Diumumkan pada tanggal 19 Oktober 2011, membawa fitur Honeycomb untuk smartphone dan menambahkan fitur baru termasuk membuka kunci dengan pengenalan wajah, jaringan data pemantauan penggunaan dan kontrol, terpadu kontak jaringan sosial, perangkat tambahan fotografi, mencari email secara offline dan berbagi informasi dengan menggunakan NFC.

#### 9. Android Versi 4.1 (Jelly Bean)

Pada tanggal 27 Juni 2012 Google mengumumkan Android versi 4.1 (Jelly Bean) dalam konferensi Google I/O berdasarkan kernel Linux 3.0.31. Jelly Bean adalah pembaruan penting yang bertujuan untuk meningkatkan fungsi dan kinerja antarmuka pengguna (User Interface). Pembaruan ini diwujudkan dalam proyek Butter, perbaikan ini termasukantisipasi sentuh, triple buffering, perpanjangan waktu vsync dan peningkatan frame rate hingga 60fps untuk menciptakan UI yang

lebih halus. Perangkat pertama yang menggunakan sistem operasi ini adalah tablet Nexus 7, yang dirilis pada 13 Juli 2012.

#### 10. Android Versi 4.4 (KitKat)

Android versi 4.4 (KitKat) direncanakan akan dirilis pada bulan Oktober 2013. Setelah sebelumnya beredar rumor bahwa Android versi berikutnya setelah Jelly Bean diperkirakan akan diberi nomor 5.0 dan dinamai Key Lime Pie [11].

#### 11. Android Versi 5.0 (Lollipop)

Pertama kali diperkenalkan pada Mei 2014. Android lollipop merupakan perancangan ulang terbesar untuk Android. Smartphone Google Nexus 6, bersama dengan tablet Nexus 9- nya, merupakan perangkat pertama yang memiliki Lollipop yang telah terpasang sebelumnya. Peningkatan terbesar yang dilakukan oleh Lollipop adalah pengenalan Material Design yang dengan cepat menjadi bahasa desain terpadu yang diterapkan di seluruh produk Google.

#### 12. Android Versi 6.0 (Marshmallow)

Di rilis pada tahun 2015. Ini Perangkat pertama yang dikirim bersama Marshmallow yang telah terpasang sebelumnya adalah smartphone Google Nexus 6P dan Nexus 5X, dengan tablet Pixel C-nya. Tujuan marshmallow memoles sudut kasar dan membuat versi Lollipop lebih baik lagi.

#### 13. Android Versi 7.0 (Nougat)

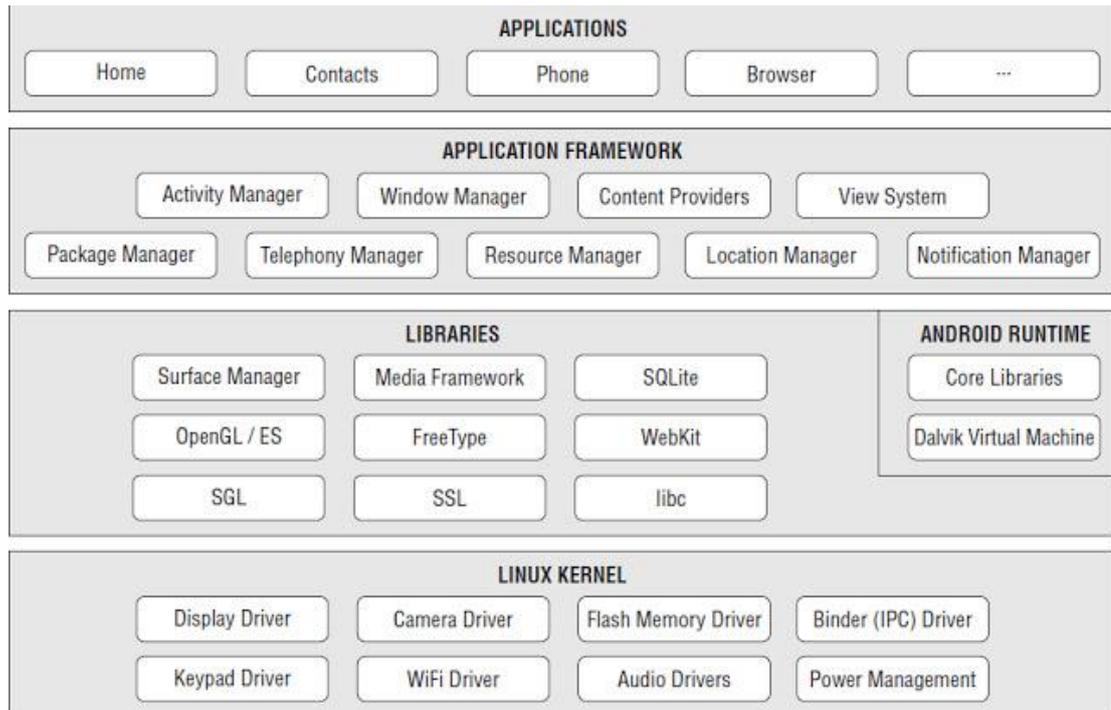
Android 7.0 (Nougat) Dirilis pada Tahun ,2016. Sebelum Nougat terungkap "Android N" dirujuk secara internal oleh Google sebagai "New York Cheesecake".

#### 14. Android Versi 8.0 (Oreo)

Pada bulan Maret 2017, Google Rilis Android 8.0 Oreo, bulan Agustus, Google mengkonfirmasi Oreo akan menjadi nama publik untuk Android 8.0. Sperti yang kita ketahui Ini adalah kedua kalinya Google memilih nama merek dagan untuk Android (Oreo dimiliki oleh Nabisco).

### 2.2.8.1. Arsitektur Android

Arsitektur pada sistem Android memiliki 5 tingkat lapisan layer. Setiap lapisan layer tersebut menghimpun beberapa program yang mendukung fungsi-fungsi spesifik dari sistem operasi. Pemisahan layer tersebut bertujuan untuk memberikan abstraksi sehingga memudahkan pengembang aplikasi [12].



**Gambar 2.3 Arsitektur Android**

#### 1. Application

Merupakan lapisan teratas. Segala macam aplikasi yang dibuat akan ditempatkan pada lapisan ini.

#### 2. Application Framework

Memperlihatkan berbagai kemampuan dari sistem android. Disediakan untuk pengembang aplikasi, sehingga dapat digunakan dalam aplikasi yang dibuat.

#### 3. Android Runtime

Berada pada lapisan yang sama sebagai library. Android runtime menyediakan inti libraries yang memungkinkan pengembang untuk menulis aplikasi Android dengan bahasa pemrograman Java. Android runtime juga

menyertakan Dalvik virtual machine (aplikasi Android di kompilasi kedalam file Dalvik yang dapat dieksekusi). Dalvik merupakan mesin virtual spesial yang di rancang secara khusus untuk android dan dioptimalkan untuk perangkat mobile sumber batere dengan memory dan CPU yang terbatas.

#### 4. Libraries

Meliputi seluruh kode yang merupakan fitur utama dari system Android. Contohnya Library SQLite yang merupakan library untuk yang menyediakan layanan database sehingga aplikasi dapat menggunakannya sebagai tempat penyimpanan data. Contoh lainnya Library Webkit yang menyediakan segala fungsi yang berkaitan dengan web browsing.

#### 5. Linux Kernel

Merupakan kernel (inti) dari sistem operasi Android. Lapisan ini meliputi seluruh driver pada tingkatan device untuk berbagai jenis komponen hardware dari perangkat Android.

### **2.2.9. Android Studio**

Android Studio adalah Lingkungan Pengembangan Terpadu Integrated Development Environment (IDE) untuk pengembangan aplikasi Android, berdasarkan IntelliJ IDEA [13]. Selain merupakan editor kode IntelliJ dan alat pengembang yang berdaya guna, Android Studio menawarkan fitur lebih banyak untuk meningkatkan produktivitas Anda saat membuat aplikasi Android, misalnya:

1. Sistem versi berbasis Gradle yang fleksibel.
2. Emulator yang cepat dan kaya fitur.
3. Lingkungan yang menyatu untuk pengembangan bagi semua perangkat Android.
4. Instant Run untuk mendorong perubahan ke aplikasi yang berjalan tanpa membuat APK baru.
5. Template kode dan integrasi GitHub untuk membuat fitur aplikasi yang sama dan mengimpor kode contoh.
6. Alat pengujian dan kerangka kerja yang ekstensif.

7. Alat Lint untuk meningkatkan kinerja, kegunaan, kompatibilitas versi, dan masalah-masalah lain.
8. Dukungan C++ dan NDK.
9. Dukungan bawaan untuk Google Cloud Platform, mempermudah pengintegrasian Google Cloud Messaging dan App Engine.

Setiap proyek di Android Studio berisi satu atau beberapa modul dengan file kode sumber dan file sumber daya. Jenis-jenis modul mencakup:

1. Modul aplikasi Android.
2. Modul Pustaka.
3. Modul Google App Engine.

#### **2.2.10. Global Positioning System (GPS)**

GPS adalah singkatan dari *Global Positioning System*, yang merupakan sistem navigasi dengan menggunakan teknologi satelit yang dapat menerima sinyal dari satelit. Sistem ini menggunakan 24 satelit yang mengirimkan sinyal gelombang mikro ke bumi. Sinyal ini diterima oleh alat penerima (receiver) di permukaan, dimana GPS receiver ini akan mengumpulkan informasi dari satelit GPS, seperti:

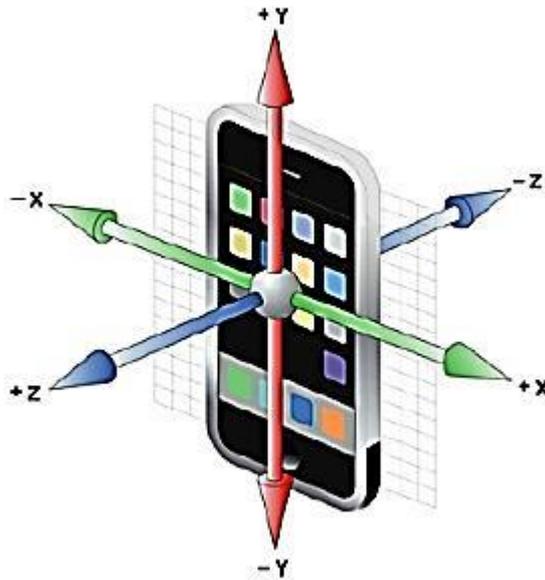
1. Waktu. GPS receiver menerima informasi waktu dari jam atom yang mempunyai keakurasian sangat tinggi.
2. Lokasi. GPS memberikan informasi lokasi dalam tiga dimensi:
  - a) Latitude
  - b) Longitude
  - c) Elevasi
3. Kecepatan. Ketika berpindah tempat, GPS dapat menunjukkan informasi kecepatan berpindah tersebut.
4. Arah perjalanan. GPS dapat menunjukkan arah tujuan.
5. Simpan lokasi. Tempat-tempat yang sudah pernah atau ingin dikunjungi bisa disimpan oleh GPS receiver. GPS receiver dapat menyimpan informasi track, seperti total perjalanan yang sudah pernah dilakukan, kecepatan rata-rata, kecepatan paling tinggi, kecepatan paling rendah, waktu/jam sampai tujuan, dan sebagainya [14].

### **2.2.11. Accelerometer**

Accelerometer adalah sebuah sensor/transduser yang berfungsi untuk mengukur percepatan, mendeteksi dan mengukur getaran, ataupun untuk mengukur percepatan akibat gravitasi bumi. Pada penelitian ini, Accelerometer digunakan sebagai mengambil nilai sudut kemiringan dari suatu objek. Accelerometer juga dapat digunakan untuk mengukur getaran yang terjadi pada kendaraan, bangunan, mesin, dan juga bisa digunakan untuk mengukur getaran yang terjadi di dalam bumi, getaran mesin, jarak yang dinamis, dan kecepatan dengan ataupun tanpa pengaruh gravitasi bumi.

Percepatan merupakan suatu keadaan berubahnya kecepatan terhadap waktu. Bertambahnya suatu kecepatan dalam suatu rentang waktu disebut juga percepatan (acceleration). Jika kecepatan semakin berkurang daripada kecepatan sebelumnya, disebut deceleration. Percepatan juga bergantung pada arah/orientasi karena merupakan penurunan kecepatan yang merupakan besaran vektor. Berubahnya arah pergerakan suatu benda akan menimbulkan percepatan pula.

Pada smartphone, Accelerometer merupakan sensor yang bisa membaca pergerakan sehingga dapat mengubah tampilan layar dari posisi landscape ke portrait atau sebaliknya dengan cukup memiringkan badan ponsel secara otomatis. Pada smartphone berbasis android terdapat sensor manager yang berfungsi untuk mengaktifkan sensor accelerometer dalam mencari nilai koordinat x,y,z dengan memiringkan handphone[15].



**Gambar 2.4 Sumbu Accelerometer**

#### **2.2.11.1. Prinsip Kerja Accelerometer**

Prinsip kerja dari transduser ini berdasarkan hukum fisika bahwa apabila suatu konduktor digerakkan melalui suatu medan magnet, atau jika suatu medan magnet digerakkan melalui suatu konduktor, maka akan timbul suatu tegangan induksi pada konduktor tersebut. Accelerometer yang diletakan di permukaan bumi dapat mendeteksi percepatan  $1g$  (ukuran gravitasi bumi) pada titik vertikalnya, untuk percepatan yang dikarenakan oleh pergerakan horizontal maka accelerometer akan mengukur percepatannya secara langsung ketika bergerak secara horizontal. Hal ini sesuai dengan tipe dan jenis sensor accelerometer yang digunakan karena setiap jenis sensor berbeda-beda sesuai dengan spesifikasi yang dikeluarkan oleh perusahaan pembuatnya. Saat ini hampir semua sensor/transduser accelerometer sudah dalam bentuk digital (bukan dengan sistem mekanik) sehingga cara kerjanya hanya berdasarkan temperatur yang diolah secara digital dalam satu chip [15].

### 2.2.11.2. Tipe Accelerometer

#### a. Capacitive

Lempengan metal pada sensor memproduksi sejumlah kapasitansi, perubahan kapasitansi akan mempengaruhi percepatan.

#### b. Piezoelectric

Kristal piezoelectric yang terdapat pada accelerometer jenis ini mengeluarkan tegangan yang selanjutnya dikonversi menjadi percepatan.

#### c. Piezoresistive

Lempengan yang secara resistan akan berubah sesuai dengan perubahan percepatan.

#### d. Hall effect

Percepatan yang dirubah menjadi sinyal elektrik dengan cara mengukur setiap perubahan pergerakan yang terjadi pada daerah yang terinduksi magnet.

#### e. Magnetoresistive

Perubahan percepatan diketahui berdasarkan resistivitas material karena adanya daerah yang terinduksi magnet.

#### f. Heat Transfer

Percepatan dapat diketahui dari lokasi sebuah benda yang dipanaskan dan diukur ketika terjadi percepatan dengan sensor temperature

### 2.2.12. Google Maps API

*Google Maps API* adalah salah satu Application Programming Interface (API) yang dimiliki Google. API ini mempunyai fitur untuk melakukan aktivitas-aktivitas yang berkaitan dengan Google Maps, antara lain menampilkan peta, mencari rute terdekat antara dua tempat, dan lain sebagainya. Google Maps API tersedia untuk platform android, iOS, web, dan juga web service. Google maps API juga menyediakan layanan seperti direction, geocoding, distance matrix API, dan elevation API.

Google maps merupakan sebuah layanan gratis yang diberikan oleh google dan sangat populer. Google maps adalah suatu peta dunia yang dapat kita gunakan untuk dapat untuk dapat melihat suatu daerah. Dengan kata lain, google maps

merupakan suatu peta yang dapat dilihat dengan menggunakan suatu web browser. Kita dapat menambahkan fitur google maps dalam sistem atau aplikasi yang kita buat. Google maps API adalah suatu library yang berbentuk javascript dimana kita dapat mengubah dan menambahkan variabel-variabel tertentu sehingga bisa dibuat sesuai dengan keinginan kita. Berkas yang mengandung bytecode kemudian akan dikonversikan oleh java interpreter menjadi bahasa mesin sesuai dengan jenis dan platform yang digunakan [15].

Dalam perkembangannya google maps API diberikan kemampuan untuk mengambil peta statis. Melakukan geocoding dan memberikan penuntun arah. Kekurangan pada google maps API yaitu jika ingin melakukan akses harus terdapat layanan internet pada perangkat yang digunakan, sedangkan kelebihan yang dimiliki yaitu :

1. Dukungan penuh yang dilakukan google sehingga terjamin dan fitur yang bervariasi pada google maps API.
2. Banyak pengembang yang menggunakan google maps API sehingga mudah dalam mencari referensi dalam pengembangan aplikasi.

### **2.2.13. Maps SDK for Android Utility Library**

Maps SDK for Android Utility Library merupakan sebuah Class Library Open-Source, yang dapat memberikan beberapa fitur menarik kedalam Google Maps. Diantaranya:

1. Import GeoJSON  
Sebuah fitur yang bisa digunakan untuk membaca data dalam format GeoJSON dan menampilkannya dalam bentuk layer pada map.
2. Import KML  
Utilitas ini bisa mengubah object KML menjadi bentuk informasi geografis dan menampilkannya dalam bentuk layer pada map.
3. Heatmaps  
Heatmaps memudahkan pengguna dalam melihat seberapa intensitasnya data dalam suatu lokasi.
4. Icon Generator

Icon Generator berfungsi untuk menampilkan potongan informasi pada marker. Utilitas ini menyediakan cara untuk membuat ikon marker terlihat seperti jendela info, karena marker itu sendiri dapat berisi teks dan konten lainnya.

5. Cluster Manager

Clustering mampu membantu mengelola tampilan marker sesuai dengan tingkatan zoom level.

6. Encode dan Decode Polyline atau Polygon

PolyUtil berguna untuk mengubah Polyline dan Poligon yang dikodekan menjadi koordinat latitude / longitude, dan sebaliknya.

7. Spherical Geometry

8. Dengan menggunakan utilitas SphericalUtil, Pengguna dapat menghitung jarak, luas berdasarkan latitude dan longitude.

#### 2.2.14. Google Directions API

Google Direction API adalah layanan yang menghitung arah antar lokasi menggunakan permintaan HTTP. Google Direction API memiliki beberapa parameter yang di butuhkan untuk meminta lama perjalanan. Daftar dari parameter yang ada dapat dilihat pada tabel dibawah Daftar parameter yang dibutuhkan Google Direction API [16].

**Tabel 2.1 Parameter Google Direction API**

<b>Nama Parameter</b>	<b>Masukkan</b>
Origin	Koordinat atau Nama Lokasi
Destination	Koordinat atau Nama Lokasi
Key	Kunci API

Parameter yang Terdapat pada Tabel diatas merupakan parameter yang dibutuhkan untuk Google Maps Direction API untuk dapat digunakan parameter Origin merupakan parameter untuk Titik Awal perhitungan, parameter Destination Merupakan parameter untuk titik akhir perhitungan, parameter Key merupakan kunci API yang digunakan pada aplikasi, Google Maps Direction API sendiri memiliki beberapa parameter Opsional yang diberikan daftar dari parameter

opsional yang ada dapat dilihat pada tabel dibawah Daftar parameter opsional Google Maps Direction API

**Tabel 2.2 Parameter Opsional Google Direction API**

<b>Nama Parameter</b>	<b>Masukkan</b>
Mode	Driving, walking
Waypoints	Koordinat atau Nama Lokasi
Alternatives	True atau False
Language	Bahasa yang digunakan
Region	Country code
Avoid	Tolls, Highways
Units	Satuan yang digunakan
Arrival_time	Unix Time
Departure_time	Unix Time, now
Traffic_model	Best_guess
Transit_model	Bus, subway, train
Transit_routing_preferences	Less_walking

Penjelasan mengenai parameter parameter pada Tabel diatas adalah sebagai berikut:

9. Mode

Merupakan parameter untuk menentukan mode transportasi yang akan digunakan saat menghitung arah.

10. Waypoints

Merupakan parameter untuk menentukan susunan lokasi perantara untuk disertakan di sepanjang rute antara titik asal dan tujuan sebagai melewati atau menghentikan lokasi. Titik jalan mengubah rute dengan mengarahkannya melalui lokasi yang ditentukan.

11. Alternatives

menetapkan bahwa layanan Arah dapat memberikan lebih dari satu alternatif rute dalam respons.

4. Language

Merupakan parameter bahasa yang akan di gunakan untuk hasil dari Google Maps Direction API.

5. Region

Merupakan Parameter area untuk membatasi hasil dari geocoder.

6. Avoid

Merupakan Parameter yang membatasi jalur yang digunakan seperti tidak melewati toll.

7. Units

merupakan Parameter yang menentukan satuan jarak yang akan di gunakan pada hasil.

8. Arrival Time

Merupakan parameter yang digunakan untuk menspesifikasi waktu Sampai yang di inginkan namun parameter ini hanya dapat digunakan bila berkendara menggunakan transit.

9. Departure Time

Merupakan parameter yang digunakan untuk menspesifikasi waktu keberangkatan yang di inginkan parameter ini membutuhkan masukan Unix Time yaitu detik semenjak 1970 januari 1 UTC(Coordinated universal Time).

10. Traffic model

Merupakan Parameter yang dipakai untuk asumsi perhitungan prediksi lama Perjalanan.

11. Transit model

Merupakan Parameter yang dipakai untuk asumsi perhitungan prediksi lama Perjalananan transit.

12. Transit routing preference

Merupakan parameter untuk memilih jalur transit apakah lebih banyak jalan atau berkendara

### **2.2.15. Firebase**

Firebase merupakan sebuah layanan infrastruktur Backend-as-a-service (BaaS) yang diakuisisi oleh Google pada oktober 2014. Firebase menawarkan kemudahan kepada para pengembang perangkat lunak dalam membangun aplikasi

yang lebih baik serta bisnis yang sukses melalui seluruh fitur komplementernya. Firebase dibangun diatas tiga pilar yang meliputi Develop, Grow, dan Earn, dapat dilihat pada gambar 2.3 Pilar Firebase [17].



**Gambar 2.5 Logo Firebase**

Firebase memiliki fitur yang dirancang untuk meningkatkan pengalaman pengembang perangkat lunak dalam mengembangkan aplikasinya. Berikut adalah beberapa teknologi tersebut :

#### **2.2.15.1. Firebase Authentication**

Firebase Authentication merupakan layanan siap pakai yang dimiliki oleh Firebase SDK. Firebase Authentaction memungkinkan aplikasi untuk melakukan autentikasi yang aman, sekaligus meningkatkan pengalaman login dan pengalaman aktivasi bagi end-user. Metode autentikasi yang digunakan meliputi email and password based authentication, federated identity provider idintegration (authentication menggunakan akun Google, Facebook, Twitter atau Github), custom authentication system integration hingga anonymous authentication. Firebase User Authentication ini bekerja dengan cara mengirimkan server response dari Firebase Server berdasarkan credential yang dikirimkan oleh client ke Firebase Server. Credential tersebut dapat berupa alamat email dan password ataupun sebuah token OAuth dari sebuah federated identity provider. Melalui server response yang diterima dari Firebase Server, aplikasi dapat mengakses informasi dasar profil pengguna dan mengontrol akses pengguna terhadap produk atau layanan Firebase yang terdapat pada aplikasi. Firebase Authentication dibangun untuk memberikan API yang mudah kepada pengembang yang dapat digunakan untuk proses sign-in dari federated providers dengan skema email dan password, atau yang sudah terintegrasi dengan autentikasi yang sudah ada. Firebase Authentication telah

terintegrasi dengan Firebase Realtime Database sehingga administrator dapat mengontrol siapa yang dapat mengakses data.

Untuk membuat pengguna login ke aplikasi, dapatkan kredensial autentikasi dari pengguna terlebih dahulu. Kredensial ini dapat berupa alamat email dan sandi pengguna atau token OAuth dari penyedia identitas tergabung. Kemudian, teruskan kredensial ini ke Firebase Authentication SDK. Layanan backend kami selanjutnya akan memverifikasi kredensial tersebut dan menampilkan respons ke klien. Setelah berhasil login, Anda dapat mengakses informasi profil dasar pengguna dan Anda dapat mengontrol akses pengguna ke data yang disimpan di produk Firebase lainnya. Anda juga dapat menggunakan token autentikasi yang disediakan untuk memverifikasi identitas pengguna di layanan backend Anda sendiri [17].

#### **2.2.15.2. Firebase Realtime Database**

Firebase Realtime Database merupakan sebuah layanan NoSQL cloudhosted database yang dimiliki oleh Firebase SDK. Firebase Realtime Database adalah database event-driver yang cara kerjanya berbeda dari database SQL. Tidak ada kode sisi server dan tingkat akses database semua pengkodean dilakukan di klien. Layanan ini menawarkan layanan penyimpanan data yang dapat disinkronisasikan secara realtime terhadap seluruh klien yang terhubung. Maksud dari realtime adalah jika terdapat perubahan pada data pada database, maka seluruh client yang terhubung akan secara otomatis mendapatkan perubahannya dalam hitungan milidetik. Kemudian offline, yaitu aplikasi yang menggunakan Firebase Realtime Database akan tetap responsif bahkan saat offline. Hal ini disebabkan karena Firebase SDK dapat mempertahankan data dan perubahannya pada media penyimpanan klien. Pada saat klien terhubung ke jaringan internet, maka Firebase SDK akan melakukan penyesuaian otomatis atas catatan perubahan data yang disimpan pada media penyimpanan klien dengan kondisi terkini dari Firebase server. Kemampuan inti yang terakhir adalah accessible from client devices. Layanan ini menawarkan kemudahan untuk mengakses Firebase Realtime Database secara langsung dari sebuah perangkat mobile atau sebuah peramban web tanpa membutuhkan server application.

Firestore Realtime Database memungkinkan Anda untuk membuat aplikasi kolaboratif dan kaya fitur dengan menyediakan akses yang aman ke database, langsung dari kode sisi klien. Data disimpan di drive lokal. Bahkan saat offline sekalipun, peristiwa realtime terus berlangsung, sehingga pengguna akhir akan merasakan pengalaman yang responsif. Ketika koneksi perangkat pulih kembali, Realtime Database akan menyinkronkan perubahan data lokal dengan update jarak jauh yang terjadi selama klien offline, sehingga setiap perbedaan akan otomatis digabungkan.

Realtime Database menyediakan bahasa aturan berbasis ekspresi yang fleksibel, atau disebut juga Aturan Keamanan Firestore Realtime Database, untuk menentukan metode strukturisasi data dan kapan data dapat dibaca atau ditulis. Ketika diintegrasikan dengan Firestore Authentication, developer dapat menentukan siapa yang memiliki akses ke data tertentu dan bagaimana mereka dapat mengaksesnya.

Realtime Database adalah database NoSQL, sehingga memiliki pengoptimalan dan fungsionalitas yang berbeda dengan database terkait. API Realtime Database dirancang agar hanya mengizinkan operasi yang dapat dijalankan dengan cepat. Hal ini memungkinkan Anda untuk membangun pengalaman realtime yang luar biasa dan dapat melayani jutaan pengguna tanpa mengorbankan kemampuan respons. Oleh karena itu, perlu dipikirkan bagaimana pengguna mengakses data, kemudian buat struktur data sesuai dengan kebutuhan tersebut [17].

### **2.2.15.3. Firestore Cloud Messaging**

FCM adalah sebuah layanan yang digunakan untuk melakukan pemberitahuan (notifications) pada aplikasi berbasis Android, iOS maupun aplikasi web. Sebelumnya Firestore Cloud Messaging ini bernama Google Cloud Messaging atau GCM, namun sekarang sudah berubah dan menjadi lebih besar di Firestore. Dengan FCM, pengembang aplikasi dapat mengirim pemberitahuan ke aplikasi yang didukung FCM dengan mudah. Dalam hal responsif, GCM / FCM memiliki hasil

yang lebih baik (lebih cepat) dari pada SMS [18]. Dengan menggunakan FCM memungkinkan terjadinya pengiriman dua jenis pesan ke klien :

1. Notification messages, terkadang diangg Display messages ditangani oleh FCM SDK secara otomatis. Digunakan FCM pengembang ingin menangani pemberitahuan pada aplikasi klien.
2. Data messages, yang ditangani oleh aplikasi klien. Digunakan jika pengembang ingin memproses pesan di aplikasi klien.

Pesan pemberitahuan berisi serangkaian kunci yang dapat dilihat pengguna yang telah ditentukan sebelumnya. Pesan data, sebaliknya, hanya berisi pasangan nilai kunci khusus yang ditetapkan pengguna. Pesan pemberitahuan dapat berisi muatan data opsional. Payload maksimum untuk kedua jenis pesan adalah 4KB, kecuali saat mengirim pesan dari Firebase console, yang memberlakukan batas 1024 karakter.

FCM dapat mengirim pesan pemberitahuan termasuk muatan data opsional. Dalam kasus semacam itu, FCM menangani tampilan payload notifikasi, dan aplikasi klien menangani payload data. Implementasi FCM mencakup dua komponen utama untuk mengirim dan menerima:

1. Lingkungan terpercaya seperti Cloud Functions untuk Firebase atau server aplikasi yang digunakan untuk membuat, menargetkan, dan mengirim pesan. Aplikasi klien iOS, Android, atau web (JavaScript) yang menerima pesan.
2. Anda dapat mengirim pesan melalui SDK Admin atau API HTTP dan XMPP. Untuk menguji atau mengirim pesan pemasaran atau keterlibatan dengan penargetan dan analitik internal yang kuat, Anda juga dapat menggunakan komposer Notifications [19].

#### **2.2.16. Java**

Java adalah bahasa pemrograman yang berorientasi objek (OOP) dan dapat dijalankan pada berbagai platform sistem operasi. Perkembangan java tidak hanya terfokus pada satu sistem operasi, tetapi juga dikembangkan untuk berbagai sistem operasi dan bersifat open source [20]. Bahasa ini awalnya dibuat oleh James Gosling saat masih bergabung di Sun Microsystems saat ini merupakan bagian dari

Oracle dan dirilis tahun 1995. Bahasa ini banyak mengadopsi sintaksis yang terdapat pada C dan C++ namun dengan sintaksis model objek yang lebih sederhana. Aplikasi-aplikasi berbasis Java umumnya dikompilasi ke dalam p-code (bytecode) dan dapat dijalankan pada berbagai Mesin Virtual Java (JVM).

Java merupakan bahasa pemrograman yang bersifat umum/non-spesifik (general purpose), dan secara khusus didisain untuk memanfaatkan dependensi implementasi seminimal mungkin. Karena fungsionalitasnya yang memungkinkan aplikasi Java mampu berjalan di beberapa platform sistem operasi yang berbeda, Java dikenal pula dengan slogannya, "Tulis sekali, jalankan di mana pun". Saat ini Java merupakan bahasa pemrograman yang paling populer digunakan, dan secara luas dimanfaatkan dalam pengembangan berbagai jenis perangkat lunak aplikasi ataupun aplikasi berbasis web.

Java menggunakan model pengamanan tiga lapis (three-layer security model) untuk melindungi sistem dari untrusted Java code. Pertama, bytecode verifier membaca bytecode sebelum dijalankan dan menjamin bytecode memenuhi aturan-aturan dasar bahasa Java. Kedua, class loader menangani pemuatan kelas Java ke runtime interpreter. Ketiga, manajer keamanan menangani keamanan tingkat aplikasi dengan mengendalikan apakah program berhak mengakses sumber daya seperti sistem file, port jaringan, proses eksternal dan sistem window.

Java termasuk bahasa Multithreading. Thread adalah untuk menyatakan program komputer melakukan lebih dari satu tugas di satu waktu yang sama. Java menyediakan kelas untuk menulis program multithreaded, program mempunyai lebih dari satu thread eksekusi pada saat yang sama sehingga memungkinkan program menangani beberapa tugas secara konkuren.

Program Java melakukan garbage collection yang berarti program tidak perlu menghapus sendiri objek-objek yang tidak digunakan lagi. Fasilitas ini mengurangi beban pengelolaan memori oleh pemrogram dan mengurangi atau mengeliminasi sumber kesalahan terbesar yang terdapat di bahasa yang memungkinkan alokasi dinamis.

Java mempunyai mekanisme exception-handling yang ampuh. Exceptionhandling menyediakan cara untuk memisahkan antara bagian

penanganan kesalahan dengan bagian kode normal sehingga menuntun ke struktur kode program yang lebih bersih dan menjadikan aplikasi lebih tegar. Ketika kesalahan yang serius ditemukan, program Java menciptakan exception. Exception dapat ditangkap dan dikelola program tanpa resiko membuat sistem menjadi turun.

Program Java mendukung native method yaitu fungsi ditulis di bahasa lain, biasanya C/C++. Dukungan native method memungkinkan pemrogram menulis fungsi yang dapat dieksekusi lebih cepat dibanding fungsi ekivalen di java. Native method secara dinamis akan di-link ke program java, yaitu diasosiasikan dengan program saat berjalan.

Adapun kelebihan dari Java yaitu :

Java mempunyai beberapa keunggulan dibandingkan dengan Bahasa pemrograman lainnya. Keunggulan bahasa pemrograman Java antara lain:

1. Berorientasi objek

Java adalah bahasa pemrograman yang berorientasi pada objek. Java membagi program menjadi objek - objek serta memodelkan sifat dan tingkah laku masing-masing dalam menyelesaikan suatu masalah.

2. Java bersifat multiplatform

Java dirancang untuk mendukung aplikasi yang dapat beroperasi di lingkungan jaringan berbeda. Untuk mengakomodasi hal tersebut, Java compiler membangkitkan bytecodes (sebuah format yang tidak tergantung pada arsitektur tertentu yang didesain untuk mengirimkan kode ke banyak platform perangkat keras dan perangkat lunak secara efisien). Java dapat dijalankan oleh banyak platform seperti Linux, Unix, Windows, Solaris, maupun Mac.

3. Java bersifat multithread

Multithreading adalah kemampuan suatu program komputer untuk mengerjakan beberapa proses dalam suatu waktu. Thread dalam Java memiliki kemampuan untuk memanfaatkan kelebihan multi prosesor apabila sistem operasi yang digunakan mendukung multi prosesor.

4. Dapat didistribusi dengan mudah

Java memiliki library rutin yang lengkap untuk dirangkai pada protocol TCP/IP (seperti HTTP dan FTP) dengan mudah. Kemampuan networking Java lebih

kuat dan lebih mudah digunakan. Java memudahkan tugas pemrograman jaringan yang sulit seperti membuka dan mengakses sebuah soket koneksi. Java juga memudahkan pembuatan CGI (Common Gateway Interface).

#### 5. Bersifat dinamis

Java dirancang untuk beradaptasi dengan lingkungan yang sedang berkembang. Java bersifat dinamis dalam tahap linking. Class yang ada dapat di-link sebatas yang diperlukan, apabila diperlukan modul kode yang baru dapat di-link dari beberapa sumber, bahkan dari sumber dalam jaringan Internet.

Kekurangan Java yaitu:

##### 1. Tulis sekali

Masih ada beberapa hal yang tidak kompatibel antara platform satu dengan platform lain. Untuk J2SE, misalnya SWT-AWT bridge yang sampai sekarang tidak berfungsi pada Mac OS X.

##### 2. Mudah didekompilasi

Dekompilasi adalah proses membalikkan dari kode jadi menjadi kod sumber. Ini dimungkinkan karena kode jadi Java merupakan bytecode yang menyimpan banyak atribut bahasa tingkat tinggi, seperti nama-nama kelas, metode, dan tipe data. Hal yang sama juga terjadi pada Microsoft .NET Platform. Dengan demikian, algoritma yang digunakan program akan lebih sulit disembunyikan dan mudah dibajak/di-reverse-engineer.

##### 3. Penggunaan memori yang banyak

Penggunaan memori untuk program berbasis Java jauh lebih besar daripada bahasa tingkat tinggi generasi sebelumnya seperti C/C++ dan Pascal (lebih spesifik lagi, Delphi dan Object Pascal). Biasanya ini bukan merupakan masalah bagi pihak yang menggunakan teknologi terbaru (karena trend memori terpasang makin murah), tetapi menjadi masalah bagi mereka yang masih harus berkuat dengan mesin komputer berumur lebih dari 4 tahun.

### **2.2.17. Javascript**

JavaScript adalah bahasa scripting yang kuat, berbasis objek; Program JavaScript dapat disematkan langsung di halaman web HTML. Ketika

dikombinasikan dengan Document Object Model (DOM) yang ditentukan oleh browser web, JavaScript memungkinkan Anda untuk membuat konten HTML Dinamis dan aplikasi web sisi-klien interaktif[21].

Sintaks JavaScript didasarkan pada bahasa pemrograman populer C, C ++, dan Java, yang membuatnya akrab dan mudah dipelajari untuk programmer berpengalaman. Pada saat yang sama, JavaScript adalah bahasa scripting yang ditafsirkan, menyediakan lingkungan pemrograman yang fleksibel dan memaafkan di mana programmer baru dapat belajar.