

BAB II

TINJAUAN PUSTAKA

2.1 Landasan Teori

Landasan teori adalah penjelasan berbagai teori dan konsep dasar pendukung yang berkaitan dengan pembangunan aplikasi mobile android untuk . Landasan teori ini akan menjadi dasar yang kuat dalam penelitian yang akan dilakukan, terkait teori-teori dalam mendukung penelitian ini akan dibahas pada bab ini.

2.2 Rumah Sakit

Rumah sakit adalah Institusi pelayanan kesehatan publik yang menyelenggarakan pelayanan kesehatan secara paripurna, penyembuhan penyakit dan pencegahan penyakit untuk masyarakat. Rumah sakit mempunyai tugas untuk menyediakan pelayanan berupa rawat inap, rawat jalan dan gawat darurat.

Pengaturan penyelenggaraan rumah sakit bertujuan untuk :

- a. mempermudah akses masyarakat untuk mendapatkan pelayanan kesehatan
- b. memberikan perlindungan terhadap keselamatan pasien, masyarakat, lingkungan rumah sakit dan sumber daya manusia di rumah sakit
- c. meningkatkan mutu dan mempertahankan standar pelayanan rumah sakit
- d. memberikan kepastian hukum kepada pasien, masyarakat, sumber daya manusia rumah sakit, dan Rumah Sakit.

Untuk menjalankan tugas sebagaimana dimaksud dalam Pasal 4, Rumah Sakit mempunyai fungsi:

- a. penyelenggaraan pelayanan pengobatan dan pemulihan kesehatan sesuai dengan standar pelayanan rumah sakit
- b. pemeliharaan dan peningkatan kesehatan perorangan melalui pelayanan

- kesehatan yang paripurna tingkat kedua dan ketiga sesuai kebutuhan medis
- c. penyelenggaraan pendidikan dan pelatihan sumber daya manusia dalam rangka peningkatan kemampuan dalam pemberian pelayanan kesehatan
 - d. penyelenggaraan penelitian dan pengembangan serta penapisan teknologi bidang kesehatan dalam rangka peningkatan pelayanan kesehatan dengan memperhatikan etika ilmu pengetahuan bidang kesehatan [9].

2.3 Ambulan

Ambulan adalah alat transportasi untuk membawa orang yang sakit ataupun terluka menuju rumah sakit. Kata ambulan digunakan untuk mendeskripsikan alat transportasi yang memiliki peralatan medis untuk pasien yang ada di luar rumah sakit atau untuk membawa pasien ke rumah sakit untuk mendapatkan perawatan lebih lanjut. Jadi ambulan adalah alat transportasi yang digunakan untuk memindahkan orang sakit trauma ataupun nontrauma ke rumah sakit, baik dalam keadaan *emergency* ataupun *non emergency*, yang dilengkapi dengan peralatan medis yang memadai [10].

Emergency Ambulance (Ambulan Gawat Darurat) adalah unit transportasi medis yang didesain khusus dan berbeda dengan model transportasi lainnya. Ambulans gawat darurat didesain agar dapat menangani pasien gawat darurat, memberikan pertolongan pertama, dan melakukan perawatan intensif selama dalam perjalanan menuju rumah sakit rujukan.

2.4 Android

Android merupakan sistem operasi untuk perangkat *mobile* berbasis linux yang mencakup sistem operasi, *middleware* dan aplikasi (Safaat, 2011). Android menyediakan *platform* terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri untuk digunakan oleh bermacam piranti bergerak. Awalnya, Google Inc. membeli Android Inc., pendatang baru yang membuat piranti lunak untuk ponsel. Kemudian untuk mengembangkan Android, dibentuklah *Open Handset Alliance*, konsorsium dari 34 perusahaan piranti keras, piranti lunak, dan telekomunikasi, termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, dan Nvidia [11].

Android juga menjadi pilihan bagi perusahaan teknologi yang menginginkan sistem operasi berbiaya rendah, dapat dikustomisasi dan ringan untuk perangkat berteknologi tinggi tanpa harus mengembangkannya dari awal. Kemudahan-kemudahan dan keistimewaan yang dimilikinya, mengakibatkan meskipun pada awalnya sistem operasi Android dirancang khusus untuk *smartphone* dan *tablet PC*, kini telah kembali beranjak berkembang, Android juga dikembangkan menjadi aplikasi tambahan di televisi, konsol permainan, kamera digital, dan perangkat elektronik lainnya. Keberadaan Android yang bersifat terbuka telah mendorong munculnya berbagai komunitas pengembang aplikasi untuk menggunakan kode sumber terbuka sebagai dasar proyek pengembang dan pembuatan aplikasi, dengan cara menambahkan fitur-fitur baru dan lebih menarik bagi pengguna tingkat lanjut atau mengoperasikan Android pada berbagai macam *platform* yang secara resmi dirilis dengan menggunakan sistem operasi lain.

2.4.1 Versi Android

1. Apple Pie / Alpha v1.0

Untuk OS pertama yang dimiliki oleh Android adalah Apple Pie, OS ini dirilis pada tanggal 23 September 2008.

2. Banana Bread v1.1

Untuk OS Android yang kedua adalah Banana Bread yang dirilis pada bulan Februari 2009 yang digunakan oleh Smartphone pengadopsi Android, HTC dengan merk Dream.

3. CupCake v1.5

Untuk OS yang selanjutnya adalah Cupcake, versi Android CupCake mulai menunjukkan kecanggihannya sebagai Sistem Operasi Android, dan ini kali pertama Google menjadikan nama makanan.

4. Donut v1.6

Versi Donut ini dirilis pada bulan September 2009, di versi ini Google mulai memperbaiki masalah/bug yang membuat Android ini lebih sempurna ditambah lagi dengan fitur navigasi turn-by-turn

5. Eclair v2.0-2.1

Dirilis pada Oktober 2009 di versi 2.0 dengan nama Eclair yang menjadi awal diaplikasikan Bluetooth 2.1 di Android. Setelah dirilis versi 2.0 diupdate lagi ke versi 2.1 yang mendapat fitur tambahan seperti multi-touch, Live Wallpaper dan juga Blitz kamera.

6. Froyo/Forzen Yogurt v2.2

Pada bulan Mei 2010 Paduan antara Google Chrome dengan Android mulai nampak dengan ditambahkan script Chrome dengan upaya untuk meningkatkan kecepatan kinerja. Forzen Yogurt sendiri yakni memiliki makna Yogurt yang dibekukan, jadi seperti Ice Cream.

7. Gingerbread v2.3

Bulan Desember 2010 seiring dirilisnya versi Gingerbread, Android kedatangan salah satu merk Platform raksasa Smartphone yakni Samsung yang meluncurkan Samsung Nexus. Di android versi 2.3 ini Google menambahkan fitur baru yaitu NFC, internet Calling, Download Manager

8. Honeycomb v3.0-3.2

Lanjutan dari Huruf berabjad H adalah Honeycomb yang dirilis pada Februari 2014 , untuk android versi ini diperuntukkan dan spesialis Tablet dimana era Tablet sedang menjadi-jadi di tahun ini, Update tak hanya berhenti di versi 3.0 saja namun juga dikembangkan ke versi 3.1 dan 3.2.

9. Ice Cream Sandwich v4.0

Revormasi Android mulai terlihat dengan adanya Ice Cream Sandwich dimana OS versi ini mulai bekerja disemua jenis platform dan smartphone seperti tablet , fiturnya juga menjadi bertambah diantaranya adalah ekstra multitasking dan notifikasi yang lebih banyak.

10. Jelly bean v4.1.2

Android versi Jelly Bean ini dirilis pada 27 Juni 2014 lewat konferensi I/O Google. Jelly Bean menjadi versi Android yang mendapatkan banyak update, tercatat 2 kali sudah mendapatkan update di Jelly Bean yakni versi 4.1.2 dimana berbeda dibandingkan versi sebelumnya adalah segi User Interface yang lebih elegan seta penambahan fitur Google Search.

11. KitKat v4.4.2

Android Versi 4.42 atau bernama KitKat ini paling banyak dipakai pada smartphone masa kini.

12. Lolipop v5.0

Android Lolipop adalah Android versi terbaru yang diluncurkan Google pada tahun 2014. Versi Lollipop pertama kali ditanamkan di Smartphone Google Nexus 6.

13. Marsmelow v6.0

Google secara resmi mengeluarkan Android versi 6.0 yang diberi nama yaitu Marshmallow. Selain itu Android Marshmallow juga mendapatkan fitur fitur terbarunya.

14. Nougat v7.0

Nougat adalah versi Android termutakhir yang baru diperkenalkan pada ajang kumpul developer Google I/O, pertengahan 2016 ini. Setelah beberapa lamanya, Google menghadirkan OS Nougat secara resmi untuk

publik. Pembaruan yang paling mendasar pada versi Nougat adalah kehadiran Google Assistant yang menggantikan Google Now. Asisten tersebut lebih bisa diandalkan untuk menjalankan berbagai fungsi.

Fitur-fitur baru lainnya mencakup layar split-screen saat dipakai multitasking, serta fitur Doze yang telah dikenalkan sebelum di versi Android Marshmallow namun telah ditingkatkan. Android versi 7.0 atau Nougat juga memiliki dukungan terhadap platform virtual reality terbaru Google.

15. Oreo v8.0

Versi android ini resmi diperkenalkan oleh Google pada tanggal 22 Agustus 2017 yang lalu dan juga sudah secara resmi bisa diluncurkan langsung ke lapangan, tapi sebelum diresmikan versi android ini oleh google, nama “Oreo” tersebut sudah terendus sejak Android O pertama kali diperkenalkan di ajang Google I/O 2017 pada Mei 2017 lalu.

2.4.2 Arsitektur Android

Android terdiri dari beberapa lapisan perangkat lunak atau software, yang diantaranya adalah lapisan applications and widget, application framework, libraries, android RunTime, dan Linux Kernel.

Berikut adalah penjelasan dari tiap lapisan tersebut :

1. Applications dan Widget

Application dan Widgets ini adalah layer dimana kita berhubungan dengan aplikasi saja, dimana biasanya kita download aplikasi kemudian kita lakukan instalasi dan jalankan aplikasi tersebut. Di layer terdapat aplikasi inti termasuk klien email, program SMS, kalender, peta, browser, kontak, dan lain-lain. Hampir semua aplikasi ditulis menggunakan bahasa pemrograman Java.

2. Applications Framework

Android adalah “Open Development Platform” yaitu Android menawarkan kepada pengembang atau memberi kemampuan kepada pengembang untuk membangun aplikasi yang bagus dan inovatif. Pengembang bebas untuk mengakses perangkat keras, akses informasi resource, menjalankan service background, mengatur alarm, dan menambah status notifications, dan sebagainya. Pengembang memiliki akses penuh menuju API framework seperti yang dilakukan oleh aplikasi kategori inti. Arsitektur aplikasi dirancang supaya kita dengan mudah dapat menggunakan kembali komponen yang sudah digunakan (reuse). Sehingga bisa kita simpulkan Application Frameworks ini adalah layer dimana para pembuat aplikasi melakukan pengembangan/pembuatan aplikasi yang akan dijalankan di sistem operasi Android, karena pada layer inilah aplikasi dapat dirancang dan dibuat, seperti content providers yang berupa sms dan panggilan telepon. (Nazruddin, 2011). Komponen-komponen yang termasuk di dalam Application Frameworks adalah sebagai berikut:

1. Views
2. Content Provider
3. Resource Manager
4. Notification Manager
5. Activity Manager

3. Layer Libraries

Libraries ini adalah layer dimana fitur-fitur Android berada, biasanya para pembuat aplikasi mengakses libraries untuk menjalankan aplikasinya. Berjalan di atas Kernel, layer ini meliputi berbagai library C/C++ inti seperti Libc SSL, serta:

1. Libraries media untuk pemutaran media audio dan video
2. Libraries untuk manajemen tampilan

3. Libraries Graphics mencakup SGL dan OpenGL untuk grafis 2D dan 3D
4. Libraries SQLite untuk dukungan database
5. Libraries SSL dan WebKit terintegrasi dengan web browser dan security
6. Libraries LiveWebcore mencakup modern web browser dengan engine embedded web view
7. Libraries 3D yang mencakup implementasi OpenGL ES1.0 API's

4. Android RunTime

Layer yang membuat aplikasi Android dapat dijalankan dimana dalam prosesnya menggunakan Implementasi Linux. Dalvik Virtual Machine (DVM) merupakan mesin yang membentuk dasar kerangka aplikasi Android. Di dalam Android Run Time dibagi menjadi dua bagian yaitu:

1. Core Libraries: Aplikasi Android dibangun dalam bahasa Java, sementara Dalvik sebagai virtual mesinnya bukan Virtual Machine Java, sehingga diperlukan sebuah libraries yang 3 berfungsi untuk menterjemahkan bahasa Java/C yang ditangani oleh Core Libraries.
2. Dalvik Virtual Machine: Virtual mesin berbasis register yang dioptimalkan untuk menjalankan fungsi-fungsi secara efisien, dimana merupakan pengembangan yang mampu membuat Linux Kernel untuk melakukan threading dan manajemen tingkat rendah.

5. Linux Kernel

Linux Kernel adalah layer dimana inti dari sistem operasi Android itu berada. Berisi file-file sistem yang mengatur sistem processing, memory, resource, drivers, dan sistem-sistem operasi Android lainnya. Linux Kernel yang digunakan Android adalah Linux Kernel release 2.6 [12].

2.5 Ionic Framework

Ionic merupakan kerangka ponsel HTML5 dengan fokus pada kinerja yang memanfaatkan akselerasi *hardware* dan tidak memerlukan pihak ketiga seperti JS *library*. Ionic bekerja berbarengan dengan Angular.js untuk membangun sebuah aplikasi interaktif. Aplikasi *hybrid* pada dasarnya ialah *website* kecil yang berjalan di *shell* browser, sebuah aplikasi yang mempunyai akses ke lapisan platform asli dari sebuah *device*. Aplikasi *hybrid* mempunyai sangat banyak manfaat jika dibandingkan dengan aplikasi *native*, khususnya dalam hal mendukung platform dan kecepatan pengembangan, Ionic dilengkapi dengan elemen UI *mobile* dan *layout* yang mirip dengan SDK asli pada ios atau android. Ionic memerlukan Apache Cordova untuk menjalankan aplikasi.

Dalam pengembangan aplikasi *hybrid*, juga banyak pihak yang menawarkan *framework* dengan segala kemampuannya yang akan sangat membantu pembuatan aplikasi menjadi lebih mudah. Salah satu yang paling populer dan banyak digunakan oleh para *programmer* di dunia yaitu *framework* Ionic. Ionic terintegrasi dengan Angular sehingga sangat cocok digunakan untuk membuat aplikasi yang berbasis Angular.

Sejak versi 2, Angular sudah banyak berbeda dibanding dengan versi pertamanya. Untungnya, Ionic merupakan *framework* yang selalu *update* dan mengikuti perkembangan Angular.

Beberapa keunggulan yang dimiliki Ionic dibanding dengan para pesaingnya di antaranya sebagai berikut:

1. Ionic berbasis *open source* sehingga bebas mengembangkan aplikasi baik untuk keperluan sendiri maupun aplikasi komersial menggunakan Ionic.
2. Ionic terintegrasi dengan Angular, bahkan gaya pengkodeannya juga persis mengikuti gaya pengkodean Angular.
3. Ionic memiliki UI *default* yang sangat cantik dan mudah untuk di *customize*.
4. Ionic dapat di instal melalui GUI maupun CLI, serta menyediakan *Tool* dan *Service* yang memudahkan pengguna Ionic bagi *programmer*.

5. Ionic dapat diintegrasikan dengan mudah dengan fitur-fitur *smartphone* seperti kamera, galeri, kontak, *geolocation*, dan sebagainya.
6. Tim pengembang Ionic sangat aktif di media sosial.
7. Ionic juga memiliki komunitas yang sangat besar aktif di media sosial.

Apache Cordova adalah *framework* untuk membangun sebuah aplikasi pada berbagai macam platform seperti Android, Blackberry, IPhone ataupun Windows Phone menggunakan HTML5, JQuery, JQuery Mobile dan CSS3. Membangun aplikasi untuk *device* yang berbeda seperti Android, iPhone, Windows Mobile dan lainnya dibutuhkan *framework* dan bahasa pemrograman yang berbeda, seperti pada Android menggunakan bahasa pemrograman Jawa, blackberry dengan bahasa pemrograman Jawa, iPhone dengan Basic C dan Windows Phone dengan C# [14].

2.6 GPS (Global Positioning System)

GPS adalah singkatan dari Global Positioning System yang merupakan sistem untuk menentukan posisi dan navigasi secara global dengan menggunakan satelit dan metode Triangulasi. Sistem tersebut merupakan sistem yang pertama kali dikembangkan oleh Departemen Pertahanan Amerika yang awalnya diperuntukan bagi kepentingan militer. NAVSTAR GPS (Navigation Satellite Timing and Ranging Global Positioning System) adalah nama asli dari Sistem GPS, yang mempunyai tiga segmen yaitu: satelit (Space Segment), pengendali (Control Segment), dan penerima/pengguna (User Segment) [15].

1. GPS Segmen Satelit

Satelit GPS bisa dianalogikan sebagai stasiun radio di angkasa yang dilengkapi dengan antena-antena untuk mengirim dan menerima sinyal-sinyal. Sinyal ini selanjutnya diterima oleh GPS receiver di permukaan bumi atau dekat permukaan bumi untuk menentukan informasi posisi, kecepatan maupun waktu.

2. GPS Segmen Kontrol/Pengendali

Segmen sistem kontrol berfungsi mengontrol dan memantau operasional satelit dan memastikan satelit berfungsi sebagaimana mestinya.

Satelit GPS tersebut dimonitor dan dikontrol oleh segmen sistem kontrol, yang terdiri dari beberapa stasiun pemonitor dan pengontrol yang tersebar di seluruh dunia. Di samping memonitor dan mengontrol kondisi seluruh satelit beserta komponennya. Segmen kontrol juga berfungsi menentukan orbit dari satelit-satelit GPS.

3. GPS Segmen Pengguna

Segmen pengguna terdiri dari para pengguna satelit GPS, baik di darat, laut maupun di angkasa. Dalam hal ini alat penerima sinyal satelit GPS (GPS receiver) diperlukan untuk menerima dan memproses sinyal-sinyal dari satelit GPS yang akan digunakan dalam penentuan posisi, kecepatan, maupun waktu, kemudian menampilkan hasilnya pada suatu tampilan (layar monitor). Komponen utama receiver GPS secara umum adalah antena dengan preamplifier, yang berfungsi untuk menerima sinyal yang dipancarkan dari satelit GPS dan mengirimkannya ke bagian radio frequency.

Cara kerja GPS secara logik ada 5 langkah:

1. Memakai perhitungan “triangulation” dari satelit.
2. Untuk perhitungan “triangulation”, GPS mengukur jarak menggunakan travel time sinyal radio.
3. Untuk mengukur travel time, GPS memerlukan akurasi waktu yang tinggi.
4. Untuk perhitungan jarak, kita harus tahu dengan pasti posisi satelit dan ketinggian pada orbitnya.
5. Terakhir harus mengoreksi delay sinyal waktu perjalanan di atmosfer sampai diterima receiver.

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. Sebuah GPS receiver harus mengunci sinyal minimal tiga satelit untuk menghitung posisi 2D (latitude dan longitude) dan track pergerakan. Jika GPS receiver dapat menerima empat atau lebih satelit, maka dapat menghitung posisi 3D (latitude, longitude dan altitude). Jika sudah dapat menentukan posisi user, selanjutnya GPS dapat menghitung informasi lain, seperti kecepatan, arah yang dituju, jalur, tujuan perjalanan, jarak tujuan, matahari terbit dan matahari terbenam dan lainlain. Sinyal yang dikirimkan oleh satelit ke GPS akan digunakan untuk menghitung waktu perjalanan (travel time). Waktu perjalanan ini sering juga disebut sebagai Time of Arrival (TOA). Sesuai dengan prinsip fisika, bahwa untuk mengukur jarak dapat diperoleh dari waktu dikalikan dengan cepat rambat sinyal.

Dari beberapa pemakaian GPS di atas dikategorikan menjadi:

a. Waktu.

GPS receiver menerima informasi waktu dari jam atom yang mempunyai keakurasian sangat tinggi.

b. Lokasi.

GPS memberikan informasi lokasi:

1. Latitude
2. Longitude
3. Altitude

c. Kecepatan.

Ketika berpindah tempat, GPS dapat menunjukkan informasi kecepatan berpindah tersebut.

d. Arah perjalanan.

GPS dapat menunjukkan arah tujuan. Simpan lokasi. Tempat-tempat yang sudah pernah atau ingin dikunjungi bisa disimpan oleh GPS receiver.

e. Komulasi data.

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.

f. Tracking.

Membantu untuk memonitoring pergerakan obyek. Membantu memetakan posisi tertentu, dan perhitungan jaringan terdekat [16].

2.6.1 Akurasi GPS

Tingkat Akurasi dari GPS terdiri dari akurasi jarak dan akurasi waktu. Akurasi Jarak:

1. Tergantung dari kualitas GPS unit yang digunakan (militer atau sipil) akurasi berkisar antara 20m s/d 1mm
2. Sebuah GPS sipil yg berkualitas medium (misalnya: Garmin E-track) dapat memberikan akurasi dari 12m s/d 3m. Sedangkan akurasi waktu: sebuah GPS unit (Baik militer maupun sipil) akan memberikan 60 nano second (detik) s/d 5 nano second akurasi waktu (time accuracy).

Kelemahan GPS antara lain adalah sinyal mudah terdistorsi oleh benda padat seperti bangunan, pohon, manusia dan sebagainya, serta hal ini menyebabkan blank spot atau kesalahan lokasi [17].

2.7 Google Maps

Google Maps adalah layanan gratis yang diberikan oleh Google dan sangat populer dan merupakan suatu peta dunia yang dapat kita gunakan untuk melihat suatu daerah. Dengan kata lain, Google Maps merupakan suatu peta yang dapat dilihat dengan menggunakan suatu browser. Kita dapat menambahkan fitur Google Maps dalam web yang telah kita buat atau pada blog kita yang berbayar maupun gratis sekalipun dengan Google Maps API [18].

Google mempunyai banyak fungsi, seperti pencarian lokasi dengan cara memasukkan kata kunci, menampilkan gambar lokasi jalanan, perhitungan rute perjalanan dari satu tempat ke tempat lainnya dan tingkat kemacetan lalu lintas pada suatu daerah.

2.8 Geotagging

Geotagging adalah sebuah proses penambahan informasi posisi data pada GPS berupa informasi latitude dan longitude dalam sebuah foto digital. Dengan adanya fitur geotagging dalam informasi sebuah foto maka letak pengambilan foto tersebut dapat dengan mudah diketahui. Terdapat tiga metode yang dapat dilakukan untuk melakukan geotagging pada suatu media, antara lain:

1. Geocoding manual, merupakan metode dimana informasi mengenai lokasi ditambahkan secara manual dengan menginputkan koordinat tertentu atau memilih lokasi pada saat melakukan upload suatu media ke dalam internet. Tingkat akurasi dari metode geotagging ini tergantung pada tools yang digunakan ataupun GPS receiver untuk mendapatkan koordinat.
2. Kamera digital dengan fitur GPS, pada beberapa kamera digital telah dilengkapi dengan fitur GPS secara langsung. Hasil foto maupun video akan secara otomatis memiliki informasi lokasi pada EXIF header dari data tersebut.
3. Kamera digital dengan GPS terpisah, pada kamera digital yang belum dilengkapi dengan fitur GPS geotagging dilakukan dengan menyingkronkan kamera dan GPS melalui sebuah software. Koordinat yang terekam oleh GPS pada waktu yang bersamaan dengan waktu pengambilan foto atau video akan ditambahkan sebagai informasi lokasi pada EXIF foto.

Mekanisme geotagging pada foto dimulai ketika foto diambil menggunakan kamera yang telah memiliki fitur geotagging, kamera tersebut dapat mencatat informasi yang lebih banyak dibandingkan dengan sebuah foto yang diambil dengan kamera

biasa. Informasi yang dapat dicatat diantaranya waktu dan lokasi dimana foto diambil, apakah menggunakan lampu flash, orientasi dari kamera (landscape atau portrait), dan detil kamera lainnya yang digunakan seperti Apertur, Local Length, dan Exposure. Semua data ini disimpan dalam suatu tempat yang disebut EXIF Headers.

Format Exchangeable Image File format (EXIF) merupakan spesifikasi industri yang diterbitkan untuk format file gambar yang digunakan oleh kamera digital. Exchangeable Image File Format (EXIF) headers dapat berisi informasi mengenai foto dengan jenis data yang dapat dibaca oleh perangkat lunak manajemen foto [19].

2.9 Firebase Cloud Messaging

Firebase Cloud Messaging adalah suatu service atau layanan yang digunakan oleh pengembang aplikasi dan disediakan oleh firebase untuk mendapatkan push notification pada perangkat mobile. Ini bisa menjadi pesan ringan yang memberitahu aplikasi android bahwa ada data baru yang akan diambil dari server. Keuntungan yang didapat dari menggunakan FCM ini adalah bisa menangani semua aspek antrian pesan dan pengiriman ke aplikasi target android berjalan pada perangkat target [20], gratis, dan firebase cloud messaging juga hemat daya dan penggunaannya cukup mudah jika dibandingkan dengan raw socket yang cepat menghabiskan daya dan jauh lebih rumit.

Penggunaan firebase cloud messaging sangat dibutuhkan dalam pembangunan aplikasi rumah sakit Sentot Patrol karena dengan fitur yang diberikan firebase ini dapat berjalan secara *real-time*.

2.10 API (Application Programming Interface)

Application programming interface (API) merupakan suatu dokumentasi yang terdiri dari interface, fungsi, kelas, struktur dan sebagainya untuk membangun sebuah perangkat lunak. Dengan adanya API ini, maka memudahkan programmer untuk “membongkar” suatu software, kemudian dapat dikembangkan atau diintegrasikan dengan perangkat lunak yang lain. API dapat dikatakan sebagai penghubung suatu

aplikasi dengan aplikasi lainnya yang memungkinkan programmer menggunakan sistem function. Proses ini dikelola melalui sistem operasi [21].

2.11 JSON (JavaScript Object Notation)

JSON (JavaScript Object Notation) adalah format pertukaran data yang ringan, mudah dibaca dan ditulis oleh manusia, serta mudah diterjemahkan dan dibuat (generate) oleh komputer. Format ini dibuat berdasarkan bagian dari Bahasa Pemrograman JavaScript, Standar ECMA-262 Edisi ke-3 - Desember 1999. JSON merupakan format teks yang tidak bergantung pada bahasa pemrograman apapun karena menggunakan gaya bahasa yang umum digunakan oleh programmer keluarga C termasuk C, C++, C#, Java, JavaScript, Perl, Python dll. Oleh karena sifat-sifat tersebut, menjadikan JSON ideal sebagai bahasa pertukaran data. JSON terbuat dari dua struktur:

1. Kumpulan pasangan nama/nilai. Pada beberapa bahasa, hal ini dinyatakan sebagai objek (object), rekaman (record), struktur (struct), kamus (dictionary), tabel hash (hash table), daftar berkunci (keyed list), atau associative array.
2. Daftar nilai terurutkan (an ordered list of values). Pada kebanyakan bahasa, hal ini dinyatakan sebagai larik (array), vektor (vector), daftar (list), atau urutan (sequence).

Struktur-struktur data ini disebut sebagai struktur data universal. Pada dasarnya, semua bahasa pemrograman moderen mendukung struktur data ini dalam bentuk yang sama maupun berlainan. Hal ini pantas disebut demikian karena format data mudah dipertukarkan dengan bahasa-bahasa pemrograman yang juga berdasarkan pada struktur data ini.

JSON menggunakan bentuk sebagai berikut:

1. Objek adalah sepasang nama/nilai yang tidak terurutkan. Objek dimulai dengan { (kurung kurawal buka) dan diakhiri dengan } (kurung kurawal

tutup). Setiap nama diikuti dengan : (titik dua) dan setiap pasangan nama/nilai dipisahkan oleh , (koma).

2. Larik adalah kumpulan nilai yang terurutkan. Larik dimulai dengan [(kurung kotak buka) dan diakhiri dengan] (kurung kotak tutup). Setiap nilai dipisahkan oleh , (koma). [22]

2.12 PHP

PHP dikenal secara umum dikenal sebagai bahasa pemrograman *script-script* yang membuat dokumen HTML secara *on the fly* yang dieksekusi di server web, dokumen yang dihasilkan dari suatu aplikasi bukan dokumen HTML yang dibuat dengan menggunakan editor teks atau editor HTML, dikenal juga sebagai bahasa pemrograman server side. PHP diciptakan oleh Rasmus Lerdorf pertama kali tahun 1994. Pada awalnya PHP adalah singkatan dari "*Personal Home Page Tools*". Selanjutnya diganti menjadi FI ("*Forms Interpreter*"). Sejak versi 3.0, nama bahasa ini diubah menjadi "*PHP: Hypertext Preprocessor*" dengan singkatannya "PHP".

Beberapa kelebihan PHP sebagai bahasa pemrograman web antara lain sebagai berikut:

1. Bahasa pemrograman PHP adalah sebuah bahasa *script* yang tidak melakukan sebuah kompilasi dalam penggunaannya.
2. Web Server yang mendukung PHP dapat ditemukan di mana - mana dari mulai apache, IIS, Lighttpd, hingga Xitami dengan konfigurasi yang relatif mudah.
3. Dalam sisi pengembangan lebih mudah, karena banyaknya milis - milis dan developer yang siap membantu dalam pengembangan.
4. Dalam sisi pemahaman, PHP adalah bahasa *scripting* yang paling mudah karena memiliki referensi yang banyak.

PHP adalah bahasa open *source* yang dapat digunakan di berbagai mesin (Linux, Unix, Macintosh, Windows) dan dapat dijalankan secara *runtime* melalui *console* serta juga dapat menjalankan perintah-perintah sistem [23].

2.13 MySQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL atau yang dikenal dengan DBMS (*Database Management System*), *database* ini *multithread*, *multi-user*. MySQL AB membuat MySQL tersedia sebagai perangkat lunak gratis di bawah lisensi GNU *General Public License* (GPL), tetapi mereka juga menjual di bawah lisensi komersial untuk kasus-kasus yang bersifat khusus.

Kekuatan MySQL tidak ditopang oleh sebuah komunitas, seperti Apache, yang dikembangkan oleh komunitas umum, dan hak cipta untuk kode sumber dimiliki oleh pemilik masing-masing, tetapi MySQL didukung penuh oleh sebuah perusahaan profesional dan komersial, yakni MySQL AB dari Swedia.

MySQL adalah *Relational Database Management System* (RDBMS) yang didistribusikan secara gratis di bawah lisensi GPL (*General Public License*). Di mana setiap orang bebas untuk menggunakan MySQL, namun tidak boleh dijadikan produk turunan yang bersifat *closed source* atau komersial. MySQL sebenarnya merupakan turunan salah satu konsep utama dalam *database* sejak lama, yaitu SQL (*Structured Query Language*). SQL adalah sebuah konsep pengoperasian *database*, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan mudah secara otomatis [24].

Berikut adalah contoh sintak sederhana pada MySQL:

1. Membuat *database*

```
f.1 CREATE DATABASE rental;
```

2. Melihat *database*

```
f.2 SHOW DATABASES;
```

3. Membuat tabel

```
f.3 CREATE TABLE customer (  
f.4     id_customer VARCHAR (8) NOT NULL,  
f.5     nama_customer VARCHAR (20) NOT NULL,  
f.6     tgl_rental DATETIME NOT NULL,
```

```
1.7         tgl_kembali DATETIME NOT NULL,  
1.8         PRIMARY KEY (id)  
1.9     );
```

4. Melihat tabel

```
1.10    SHOW TABLES;
```

5. Menghapus tabel

```
1.11    DROP TABLE customer;
```

6. Menghapus database

```
1.12    DROP DATABASE rental;
```

Dalam menggunakan *database* MySQL, setiap perintah yang diketikkan disebut *query*. Perintah MySQL dapat dikategorikan menjadi 3 sub perintah, yaitu DDL (*Data Definition Language*), DML (*Data Manipulation Language*), dan DCL (*Data Control Language*). Berikut adalah pemaparan dari setiap kategori:

1. DDL (*Data Definition Language*)

Data Definition Language yang kalau di singkat DDL merupakan perintah SQL yang berhubungan dengan pendefinisian suatu struktur *database*, dalam hal ini *database* dan *table*. Beberapa perintah dasar yang termasuk DDL ini antara lain:

- a. CREATE berfungsi untuk membuat *database* baru, tabel baru, *view* baru dan kolom.
- b. ALTER berfungsi untuk mengubah struktur tabel. Seperti mengganti nama tabel, menambah kolom, mengubah kolom, menghapus kolom maupun memberikan atribut pada kolom.
- c. DROP berfungsi untuk menghapus *database* dan tabel.
- d. TRUNCATE berfungsi untuk menghapus semua catatan dari tabel.
- e. COMMENT berfungsi untuk menambahkan komentar pada data.
- f. RENAME berfungsi untuk mengubah nama objek.

2. DML (*Data Manipulation Language*)

Kategori selanjutnya adalah *Data Manipulation Language* (DML). DML ialah perintah yang digunakan untuk mengelola/memanipulasi data dalam *database*. Terdapat beberapa perintah DML pada MySQL sebagai berikut:

- a. SELECT berfungsi untuk mengambil/menampilkan data dari *database*.
- b. INSERT berfungsi untuk memasukkan data ke dalam tabel.
- c. UPDATE berfungsi untuk memperbarui data dalam tabel.
- d. DELETE berfungsi untuk menghapus data dari tabel.
- e. CALL berfungsi untuk memanggil sub program PL / SQL atau Java.
- f. EXPLAIN PLAN berfungsi untuk menjelaskan jalur akses ke data.
- g. LOCK TABLE berfungsi untuk mengunci tabel.

3. DCL (*Data Control Language*)

Kategori yang terakhir adalah *Data Control Language* (DCL). DCL merupakan perintah yang digunakan untuk melakukan pengontrolan data dan server *databasenya*. Terdapat beberapa perintah DCL pada MySQL sebagai berikut:

- a. GRANT berfungsi untuk memberikan hak akses pengguna ke *database*.

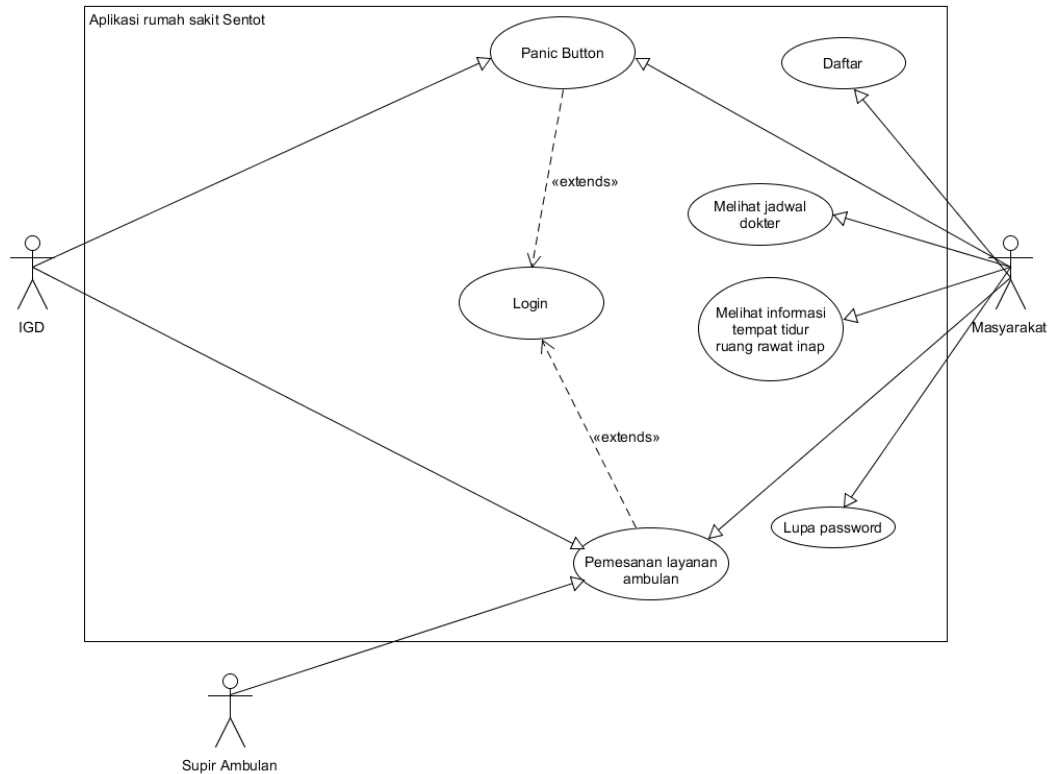
REVOKE berfungsi untuk menghilangkan hak akses yang telah diberikan dengan perintah GRANT.

2.14 UML (**Unified Modeling Language**)

Unified Modeling Language (UML) adalah sebuah bahasa yang berdasarkan gambar untuk memvisualisasikan, menspesifikasikan, membangun dan pendokumentasian dari sebuah sistem pengembangan perangkat lunak berbasis Objek. Unified Modeling Language (UML) bukanlah merupakan bahasa pemrograman tetapi model-model yang tercipta berhubungan langsung dengan berbagai macam bahasa pemrograman, sehingga memungkinkan melakukan pemetaan (mapping) langsung dari model-model yang dibuat dengan Unified Modeling Language (UML) dengan bahasa-bahasa pemrograman berorientasi obyek, seperti Java [25].

2.13.1 Use Case Diagram

Use Case Diagram merupakan serangkaian langkah yang menggambarkan interaksi antar user dan sistem yang saling terkait. Use Case digunakan untuk menunjukkan hubungan (relationship) antar aktor sebagai



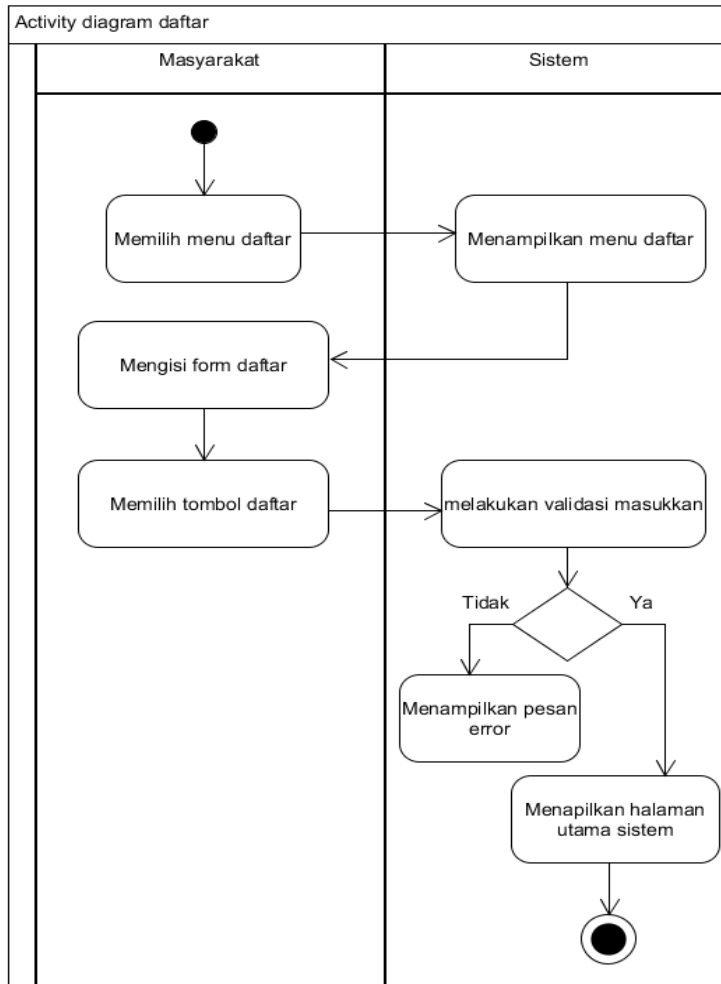
pengguna sistem dengan Use Case yang digunakan. Contoh use case diagram dapat dilihat pada gambar 2.2 berikut:

Gambar 2.1 Contoh Use Case Diagram

2.13.2 Activity Diagram

Activity diagram adalah diagram menggambarkan berbagai alur aktivitas atau proses bisnis dalam sistem yang sedang dirancang, bagaimana

masing-masing alur berawal, kondisi yang mungkin terjadi, dan bagaimana mereka berakhir.

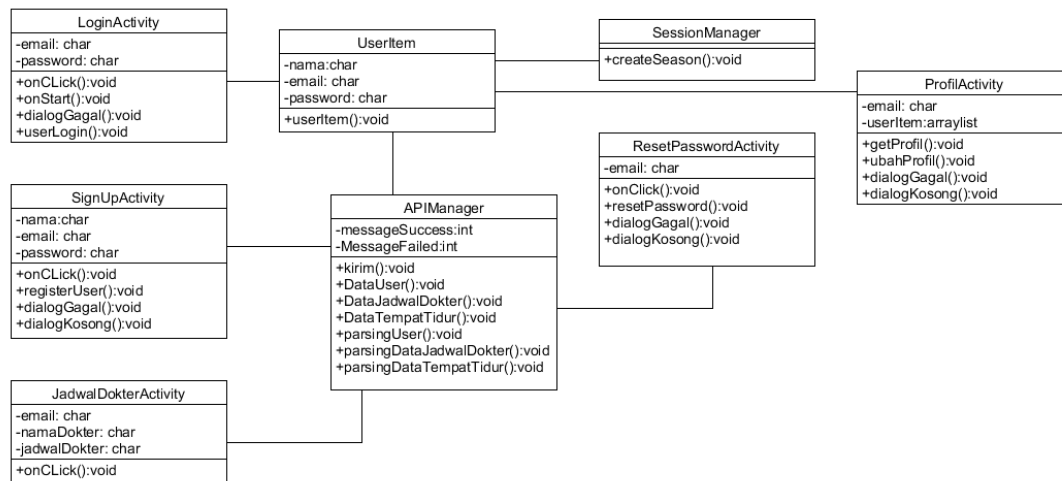


Gambar 2.2 Contoh Activity Diagram

2.13.3 Class Diagram

Class Diagram adalah sebuah spesifikasi yang jika diinstansiasi maka akan menghasilkan objek yang merupakan inti dari pengembangan dan desain

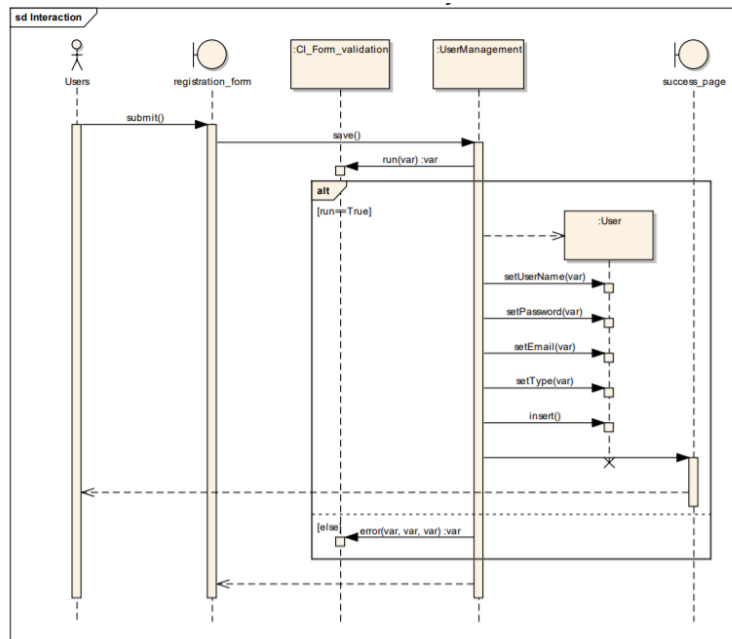
berorientasi objek. Kelas menggambarkan atribut atau properti dari sebuah system sekaligus menawarkan layanan apa saja yang bisa dilakukan dengan objek tersebut (method/fungsi).



Gambar 2.3 Contoh Class Diagram

2.13.4 Sequence Diagram

Sequence Diagram adalah gambaran interaksi antar objek di dalam dan disekitar sistem. Sequence diagram juga digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah apa yang seharusnya terjadi sebagai respon dari sebuah event untuk menghasilkan sesuatu didalam use case sebagai output.



Gambar 2.4 Contoh Sequence Diagram

2.15 Pengujian

Pengujian alpha adalah pengujian yang bertujuan untuk memastikan bahwa aplikasi yang diuji dapat berjalan dengan lancar tanpa gangguan error atau bug. Pada tahap pengujian ini penguji yang dilibatkan adalah sesama pengembang aplikasi yang mengerti cara kerja aplikasi dan mengetahui sumber error atau bug secara teknis [26].

2.16 Pengujian Beta

Pengujian beta adalah pengujian yang lebih mengutamakan kesiapan aplikasi yang dikembangkan sebelum diluncurkan ke pengguna yang sesungguhnya. Pengujiannya dilakukan secara langsung terhadap pengguna dengan menggunakan kuisisioner mengenai kepuasan pengguna atas aplikasi yang telah dibangun. Kuisisioner yang di sebarakan menggunakan teknik simple random sampling dimana anggota sample dari populasi dilakukan secara acak tanpa memperhatikan strata yang ada dalam populasi. [27].