

BAB 2

LANDASAN TEORI

2.1 Landasan Teori

Landasan teori bertujuan memberikan gambaran dari teori yang terkait dalam pembangunan aplikasi. Landasan teori yang dibahas yaitu pengertian GPS Tracker, Android, GPS, *Geotagging*, *Java*, *Web Service*, *PHP*, *MySQL*, Metode yang digunakan dan bahasa pemrograman yang digunakan dalam pembangunan aplikasi.

2.1.1 GPS Tracker

GPS Tracker adalah suatu sistem yang menentukan posisi kendaraan, armada, maupun personal secara realtime. GPS tracker ini memanfaatkan teknologi GSM dan GPS untuk menentukan titik kordinat dan menterjemahkannya ke dalam bentuk peta seperti google maps atau lainnya. [2]

2.1.2 Konfigurasi GPS Tracker

Setelah proses pemasangan GPS pada kendaraan dilakukan hal selanjutnya yaitu melakukan konfigurasi pada GPS Tracker sesuai kebutuhan sistem. Konfigurasi dilakukan pertama kali via sms, dengan mengirimkan perintah ke nomor GSM pada GPS. Setiap perintah akan mendapatkan balasan dari unit GPS Tracking, perintah yang benar segera dijalankan setelah GPS mengirim balasan sms. Pesan konfigurasi server GPS Tracker TR02 adalah sebagai berikut : format pesan yang dikirimkan pada GPS SERVER,0,IP,PORT,0#. Data pertama (SERVER) merupakan perintah untuk men-set server. Data kedua men-set menggunakan IP. Data ketiga merupakan IP dari server yang akan dituju. Data keempat merupakan port yang dibuka oleh server. IP dan Port merupakan alaman server tempat GPS mengirimkan data. Data kelima untuk menghubungkan ke server dengan mengguan TCP Protocol. Jika konfigurasi berhasil dilakukan, pengirim akan mendapatkan balasan "OK" kemudian GPS menjalankan perintah yang diinstruksikan.[3]

2.1.3 Definisi GSM

Global System for Mobile Communication disingkat (GSM) adalah sebuah teknologi komunikasi selular yang bersifat digital. Teknologi GSM banyak

diterapkan pada komunikasi bergerak, khususnya telepon genggam. Teknologi ini memanfaatkan gelombang mikro dan pengiriman sinyal yang dibagi berdasarkan waktu, sehingga sinyal informasi yang dikirim akan sampai pada tujuan. GSM dijadikan standar global untuk komunikasi selular sekaligus sebagai teknologi selular yang paling banyak digunakan orang di seluruh dunia. GSM beroperasi pada frekuensi range 900MHz dan range 1800 MHz.[4]

2.1.4 Google Maps API

Google Maps adalah layanan pemetaan berbasis web service yang disediakan oleh Google dan bersifat gratis, yang memiliki kemampuan terhadap banyak layanan pemetaan berbasis web. Google Maps juga memiliki sifat server side, yaitu peta yang tersimpan pada server Google dapat dimanfaatkan oleh pengguna. Berikut adalah logo dari google maps.

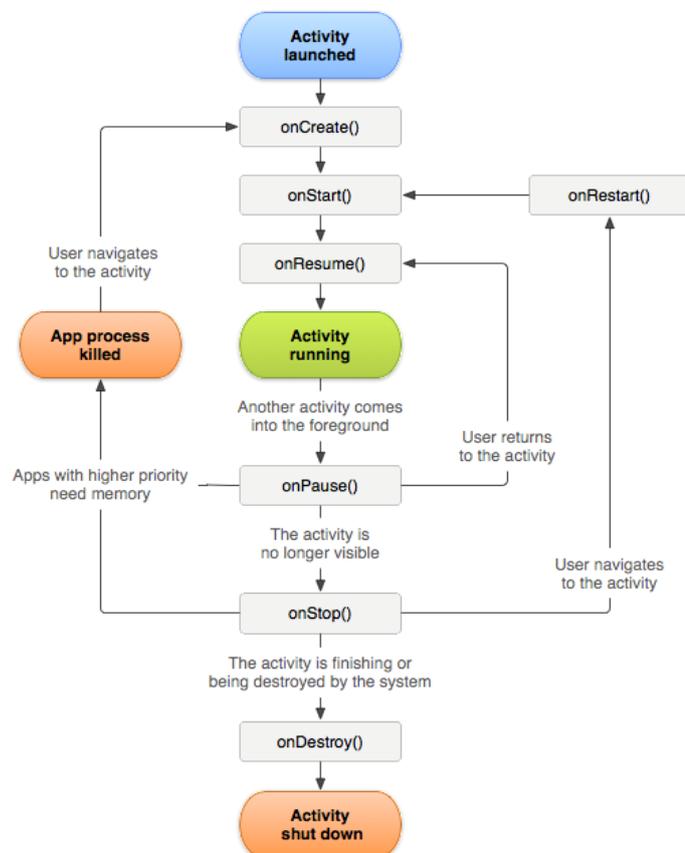
Google Maps API adalah suatu library yang berbentuk javascript yang berguna untuk memodifikasi peta yang ada di Google Maps sesuai kebutuhan. Untuk membangun aplikasi yang memanfaatkan Google Maps di desktop dan mobile device maka akan digunakan Google Maps Javascript API v3 yang memiliki keunggulan lebih cepat dari versi sebelumnya.[5]

2.1.5 Android

Android adalah sebuah sistem operasi untuk perangkat mobile berbasis linux yang mencakup sistem operasi, *middleware*, dan aplikasi. Android adalah sistem operasi untuk telepon seluler yang berbasis Linux. Android menyediakan *platform* terbuka bagi para pengembang untuk membuat aplikasi mereka sendiri. Pada awalnya dikembangkan oleh *Android Inc*, sebuah perusahaan pendatang baru yang membuat perangkat lunak untuk ponsel yang kemudian dibeli oleh *Google Inc*. Untuk pengembangannya, dibentuklah Open Handset Alliance (OHA), konsorsium dari 34 perusahaan perangkat keras, perangkat lunak, dan telekomunikasi termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, dan Nvidia [6].

2.1.5.1 Android Life Cycle

Android memiliki paradigma pemrograman lain tidak seperti paradigma pemrograman biasa di mana aplikasi yang dijalankan pada fungsi main(), sistem android menjalankan kode dalam method Activity dengan menerapkan metode callback tertentu yang sesuai dengan tahap tertentu dari siklus hidup. Setiap aplikasi yang berjalan dalam sistem operasi android memiliki siklus hidup yang berbeda dengan aplikasi desktop atau web. Hal ini dikarenakan aplikasi mobile memiliki tingkat interupsi proses yang lumayan tinggi seperti ketika handling panggilan masuk aplikasi diharuskan menghentikan proses sementara. Penerapan siklus hidup juga berguna untuk memastikan aplikasi tidak menghabiskan sumber daya baterai pengguna [3].



Sumber :

https://developer.android.com/guide/components/images/activity_lifecycle.png

Gambar 2. 1 Android Life Cycle

Terdapat beberapa state dalam siklus hidup android yang terjadi seperti diilustrasikan pada Gambar 2.1 Siklus Hidup Android, akan tetapi hanya beberapa dari state tersebut yang menjadi statis diantaranya[3]:

1. **Resumed**

Resumed terjadi ketika aplikasi berjalan setelah state *paused* . State ini akan menjalankan perintah program yang ditulis pada method `onResume()` [3].

2. **Paused**

Dalam keadaan ini aktivitas yang terjadi dihentikan secara sementara tetapi masih terlihat oleh pengguna karena terdapat proses yang memiliki prioritas lebih tinggi seperti panggilan telepon. Aplikasi tidak dapat menjalankan perintah apapun ataupun menampilkan apapun dalam state ini [3].

3. **Stopped**

Dalam keadaan ini, aplikasi benar-benar tidak ditampilkan dan tidak terlihat oleh pengguna tetapi masih meninggalkan service dibackground [3].

State lain seperti `Created` dan `Started` bersifat sementara dan sistem dengan cepat menjalankan state berikutnya dengan memanggil metode *life cycle callback* berikutnya. Artinya, setelah sistem `OnCreate()` dipanggil, dengan cepat sistem akan memanggil method `OnStart()`, kemudian diikuti oleh `onResume()`.

2.1.6 Flowchart

Flowchart adalah penggambaran secara grafik dari langkah-langkah dan urutan-urutan prosedur dari suatu program. Flowchart menolong analis dan programmer untuk memecahkan masalah kedalam segmen-segmen yang lebih kecil dan menolong dalam menganalisis alternatif-alternatif lain dalam pengoperasian. Flowchart biasanya mempermudah penyelesaian suatu masalah khususnya masalah yang perlu dipelajari dan dievaluasi lebih lanjut [8].

Bila seorang analis dan programmer akan membuat flowchart, ada beberapa petunjuk yang harus diperhatikan, seperti [8]:

1. Flowchart digambarkan dari halaman atas ke bawah dan dari kiri ke kanan.
2. Aktivitas yang digambarkan harus didefinisikan secara hati-hati dan definisi ini harus dapat dimengerti oleh pembacanya.
3. Kapan aktivitas dimulai dan berakhir harus ditentukan secara jelas.

4. Setiap langkah dari aktivitas harus diuraikan dengan menggunakan deskripsi kata kerja..
5. Setiap langkah dari aktivitas harus berada pada urutan yang benar.
6. Lingkup dan range dari aktifitas yang sedang digambarkan harus ditelusuri dengan hati-hati. Percabangan-percabangan yang memotong aktivitas yang sedang digambarkan tidak perlu digambarkan pada flowchart yang sama. Simbol konektor harus digunakan dan percabangannya diletakan pada halaman yang terpisah atau hilangkan seluruhnya bila percabangannya tidak berkaitan dengan sistem.
7. Gunakan simbol-simbol flowchart yang standar.

2.1.7 Global Positioning System (GPS)

Global Positioning System adalah sistem navigasi berbasis satelit terdiri dari jaringan 24 satelit ditempatkan ke orbit oleh Departemen Pertahanan Amerika Serikat yang pertama kali diperkenalkan mulai tahun 1978. Layanan GPS dahulu hanya dipergunakan untuk keperluan militer namun mulai terbuka untuk publik. 24 satelit GPS tersebut berada sekitar 12.000 mil di atas bumi bergerak mengelilingi bumi 12 jam dengan kecepatan 7.000 mil per jam. Satelit GPS berkekuatan energi sinar matahari, memiliki baterai cadangan untuk menjaga agar tetap berjalan pada saat gerhana matahari atau pada saat tidak ada energi matahari dan memiliki roket penguat kecil pada masing-masing satelit agar dapat mengorbit tepat pada tempatnya [4].

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:[4]

- a. Waktu. GPS receiver menerima informasi waktu dari jam atom yang mempunyai keakurasian sangat tinggi.
- b. Lokasi. GPS memberikan informasi lokasi dalam tiga dimensi:
 1. Latitude

2. Longitude
 3. Elevasi
- c. Kecepatan. Ketika berpindah tempat, GPS dapat menunjukkan informasi kecepatan berpindah tersebut.
 - d. Arah perjalanan. GPS dapat menunjukkan arah tujuan.
 - e. Simpan lokasi. Tempat-tempat yang sudah pernah atau ingin dikunjungi bisa disimpan oleh GPS receiver.
 - f. 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. [4]

2.1.8 JavaScript Object Notation (JSON)

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 [5].

JSON terbuat dari dua struktur [5]:

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 [5].

2.1.9 Java

Java menurut definisi dari Sun adalah nama untuk sekumpulan teknologi untuk membuat dan menjalankan perangkat lunak pada komputer standalone ataupun pada lingkungan jaringan. Java dikembangkan pada bulan Agustus 1991, dengan nama semula Oak. Pada Januari 1995, karena nama Oak dianggap kurang komersial, maka diganti menjadi Java. Pada Desember 1998, Sun memperkenalkan nama “Java 2” (J2) sebagai generasi kedua dari java platform. Konvensi nama baru ini diterapkan untuk semua edisi Java yaitu Standard Edition (J2SE), Enterprise Edition (J2EE), dan Micro Edition (J2ME) [6].

Ada tiga platform Java yang masing-masing diarahkan untuk tujuan tertentu dan untuk lingkungan komputasi yang berbeda-beda [6]:

1. Standard Edition (J2SE) : J2SE merupakan inti dari bahasa pemrograman Java. J2SE didesain untuk jalan pada komputer desktop dan komputer workstations.
2. Enterprise Edition (J2EE): Dengan built-in mendukung untuk servlets, JSP, dan XML, edisi ini ditujukan untuk aplikasi berbasis server.
3. Micro Edition (J2ME) : Didesain untuk piranti dengan memori terbatas, layar display terbatas dan power pemrosesan yang juga terbatas.

2.1.10 Web services

Web service adalah suatu sistem perangkat lunak yang dirancang untuk mendukung interoperabilitas dan interaksi antar sistem pada suatu jaringan. Web service digunakan sebagai suatu fasilitas yang disediakan oleh suatu web site untuk menyediakan layanan (dalam bentuk informasi) kepada sistem lain, sehingga sistem lain dapat berinteraksi dengan sistem tersebut melalui layanan (service) yang disediakan oleh suatu sistem yang menyediakan web service. Web service menyimpan data informasi dalam format XML, sehingga data ini dapat diakses oleh

sistem lain walaupun berbeda platform, sistem operasi, maupun bahasa compiler [7].

Web service bertujuan untuk meningkatkan kolaborasi antar pemrogram dan perusahaan, yang memungkinkan sebuah fungsi di dalam Web Service dapat dipinjam oleh aplikasi lain tanpa perlu mengetahui detail pemrograman yang terdapat di dalamnya [7].

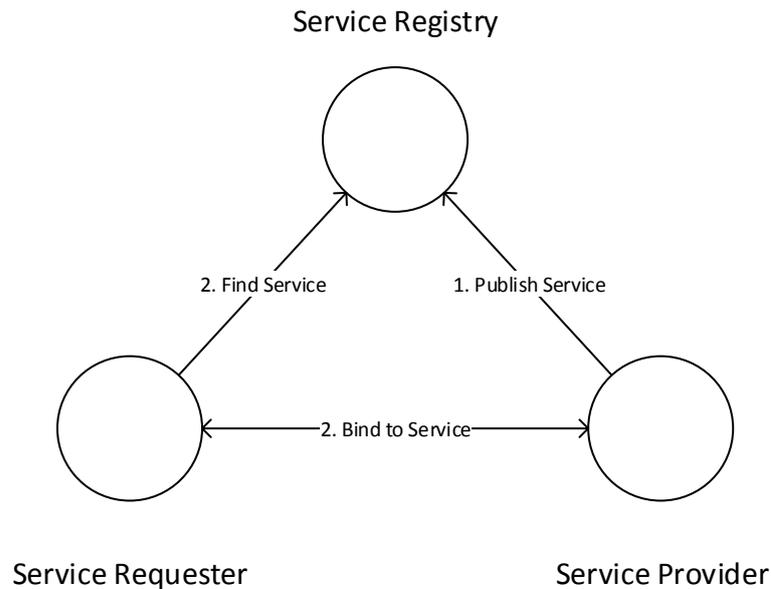
Beberapa alasan mengapa digunakannya web service adalah sebagai berikut:[7]

1. Web service dapat digunakan untuk mentransformasikan satu atau beberapa bisnis logic atau class dan objek yang terpisah dalam satu ruang lingkup yang menjadi satu, sehingga tingkat keamanan dapat ditangani dengan baik [7].
2. Web service memiliki kemudahan dalam proses deployment-nya, karena tidak memerlukan registrasi khusus ke dalam suatu sistem operasi. Web service cukup di-upload ke web server dan siap diakses oleh pihak-pihak yang telah diberikan otorisasi [7].
3. Web service berjalan di port 80 yang merupakan protokol standar HTTP, dengan demikian web service tidak memerlukan konfigurasi khusus di sisi firewall [7].

2.1.10.1 Arsitektur Web Service

Web service memiliki tiga entitas dalam arsitekturnya, yaitu: [8]

1. Service Requester (peminta layanan)
2. Service Provider (penyedia layanan)
3. Service Registry (daftar layanan)



Gambar 2. 2 Arsitektur Web Service

- Service Provider berfungsi untuk menyediakan layanan/service dan mengolah sebuah registry agar layanan-layanan tersebut dapat tersedia [8].
- Service Registry berfungsi sebagai lokasi central yang mendeskripsikan semua layanan/service yang telah di-register [8].
- Service Requestor meminta layanan yang mencari dan menemukan layanan yang dibutuhkan serta menggunakan layanan tersebut [8].

2.1.11 Processor Hypertext Protocol (PHP)

PHP merupakan salah satu pemrograman server-side di antara beberapa pemrograman yang ada. Sejak di luncurkan , PHP mendapat respon yang sangat baik dari kalangan pengembang aplikasi web kemudahannya untuk di pahami, serta sintaksnya yang mirip dengan Bahasa C menjadikan pemrograman ini cepat di kenal di kalangan luas [8].

Tanpa mengurangi kemampuan-kemampuan yang sudah ada, PHP 5 hadir dengan fitur lebih kompleks , dan merupakan rilis terbaru yang di dikeluarkan oleh pengembangnya secara besar besaran pengembang PHP merombak bug yang sering di jumpai daam PHP 4, hal ini akan sangat jelas terasa ketika anda membuat aplikasi berbasis objek [8].

Dengan tujuan untuk lebih meningkatkan kinerja aplikasi, tanpa bermaksud membuang fitur yang ada, PHP 5 menambahkan fitur – fitur baru seperti dukungan terhadap MYSQL. Dukungan ini di wujudkan dalam bentuk fungsi- fungsi lama dan di tulis ulang [8].

2.1.12 HTML

Hyper Text Markup Language (HTML) adalah sebuah bahasa markah yang digunakan untuk membuat sebuah halaman web. Weibsite yang dibuat dengan HTML ini, dapat di lihat oleh semua orang yang terkoneksi dengan internet . Tentunya dengan menggunakan aplikasi penjelajah atau browser [9].

Tiap kali kita mengakses dokumen web, maka sesungguhnya kita mengakses dokumen seseorang yang ditulis dengan menggunakan format HTML. Beberapa orang merasa keberatan jika dikatakan HTML adalah sebuah bahasa pemrograman karena struktur yang dimilikinya dianggap terlalu sederhana, kode-kode dibaca oleh browser baris per baris, dari atas ke bawah. HTML juga tidak memiliki ‘looping’ seperti bahasa pemrograman lain [9].

Pada HTML dipergunakan hypertext link atau hubungan antara teks dan dokumen lain. Dengan demikian pembaca dokumen bisa melompat dari satu dokumen ke dokumen yang lain dengan mudah [9].

2.1.13 MySQL

MySQL adalah salah satu jenis database server yang sangat terkenal dan banyak digunakan untuk membangun aplikasi web yang menggunakan database sebagai sumber dan pengolahan datanya. MySQL dikembangkan oleh perusahaan swedia bernama MySQL AB yang pada saat ini bernama Tcx DataKonsult AB sekitar tahun 1994-1995, namun cikal bakal kodenya sudah ada sejak tahun 1979. Awalnya Tcx merupakan perusahaan pengembang software dan konsultan database, dan saat ini MySQL sudah diambil alih oleh Oracle Corp [10].

Kepopuleran MySQL antara lain karena MySQL menggunakan SQL sebagai bahasa dasar untuk mengakses databasenya sehingga mudah untuk digunakan, kinerja query cepat, dan mencukupi untuk kebutuhan database perusahaan-perusahaan yang berskala kecil sampai menengah, MySQL juga

bersifat open source (tidak berbayar). MySQL merupakan database yang pertama kali didukung oleh bahasa pemrograman script untuk internet (PHP dan Perl). MySQL dan PHP dianggap sebagai pasangan software pembangun aplikasi web yang ideal. MySQL lebih sering digunakan untuk membangun aplikasi berbasis web, umumnya pengembangan aplikasinya menggunakan bahasa pemrograman script PHP [10].

2.1.14 Unified Modelling Language (UML)

Unified Modeling Language (UML) adalah keluarga notasi grafis yang didukung oleh model-model tunggal, yang membantu pendeskripsian dan desain sistem perangkat lunak, khususnya sistem yang dibangun menggunakan pemrograman berorientasi objek (OO). UML merupakan standar yang relatif terbuka yang dikontrol oleh Object Management Group (OMG) [10].

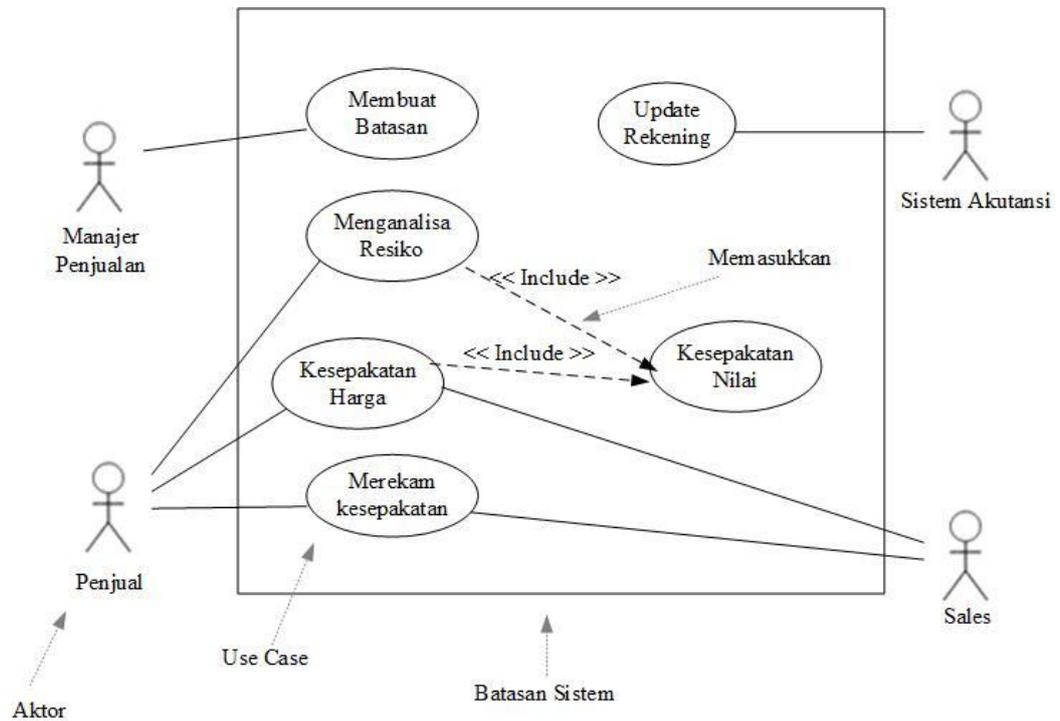
2.1.14.1 Use Case Diagram

Use case adalah teknik untuk merekam persyaratan fungsional sebuah sistem. *Use case* mendeskripsikan interaksi tipikal antara para pengguna sistem dengan sistem itu sendiri, dengan memberi sebuah narasi tentang bagaimana sistem tersebut digunakan. Sebuah *use case* adalah serangkaian skenario yang dikemas menjadi satu oleh tujuan pengguna umum. Dalam bahasa *use case*, para pengguna disebut sebagai aktor. Aktor merupakan sebuah peran yang dimainkan oleh seorang pengguna dalam kaitannya dengan sistem. aktor tidak harus manusia. Jika sebuah sistem melakukan layanan untuk sebuah sistem komputer lain, sistem lain tersebut merupakan aktor [10].

Use case diagram menampilkan aktor, *use case*, dan hubungan antara mereka [10]:

- a. Aktor mana yang menggunakan *use case* mana.
- b. *Use case* mana yang memasukkan *use case* lain.

Berikut ini adalah contoh dari diagram *use case* :

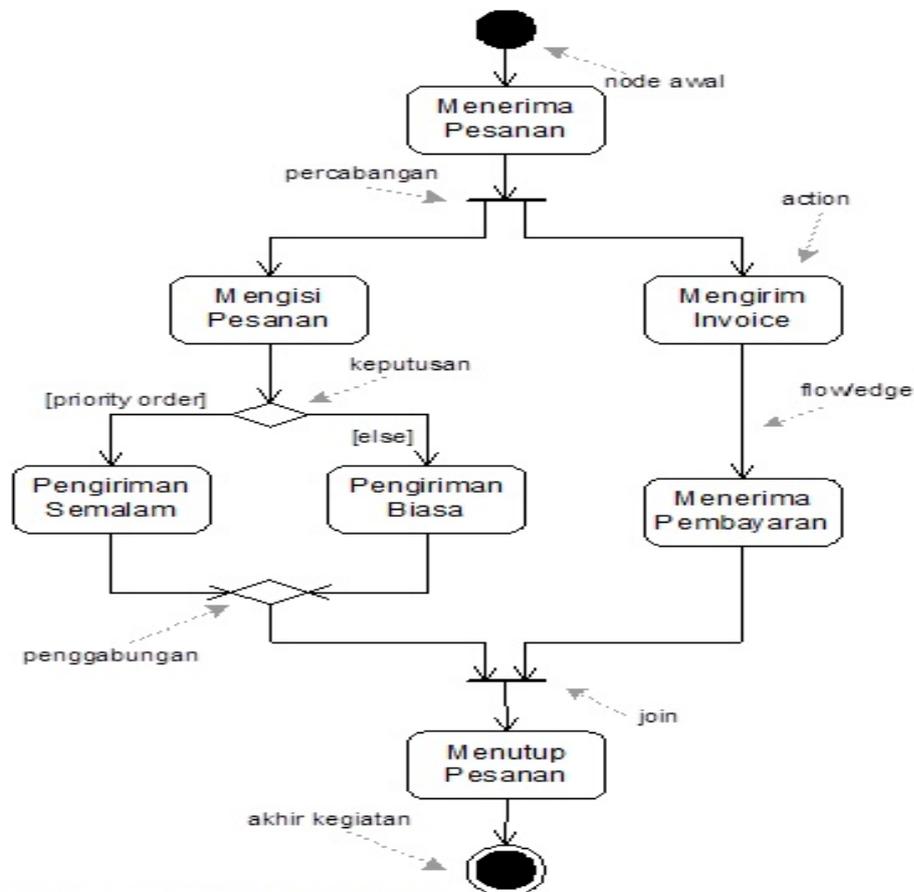


Gambar 2. 3 Contoh diagram Use Case

[Cockburnm use case] menjelaskan sebuah skema tingkatan use case. Inti use case berada pada tingkat sea level. Use case sea level khususnya mewakili sebuah interaksi diskrit antara aktor utama dan sistem. Use case yang ada disana hanya karena mereka dimasukkan oleh use case sea level adalah fish level. Lebih tinggi, use case kite level menampilkan bagaimana use case sea level sesuai dengan interaksi bisnis yang lebih luas. Use case kite level biasanya merupakan use case bisnis, sedangkan se case sea dan fish level merupakan use case sistem. Kebanyakan use case Anda berada pada tingkatan sea level [10]

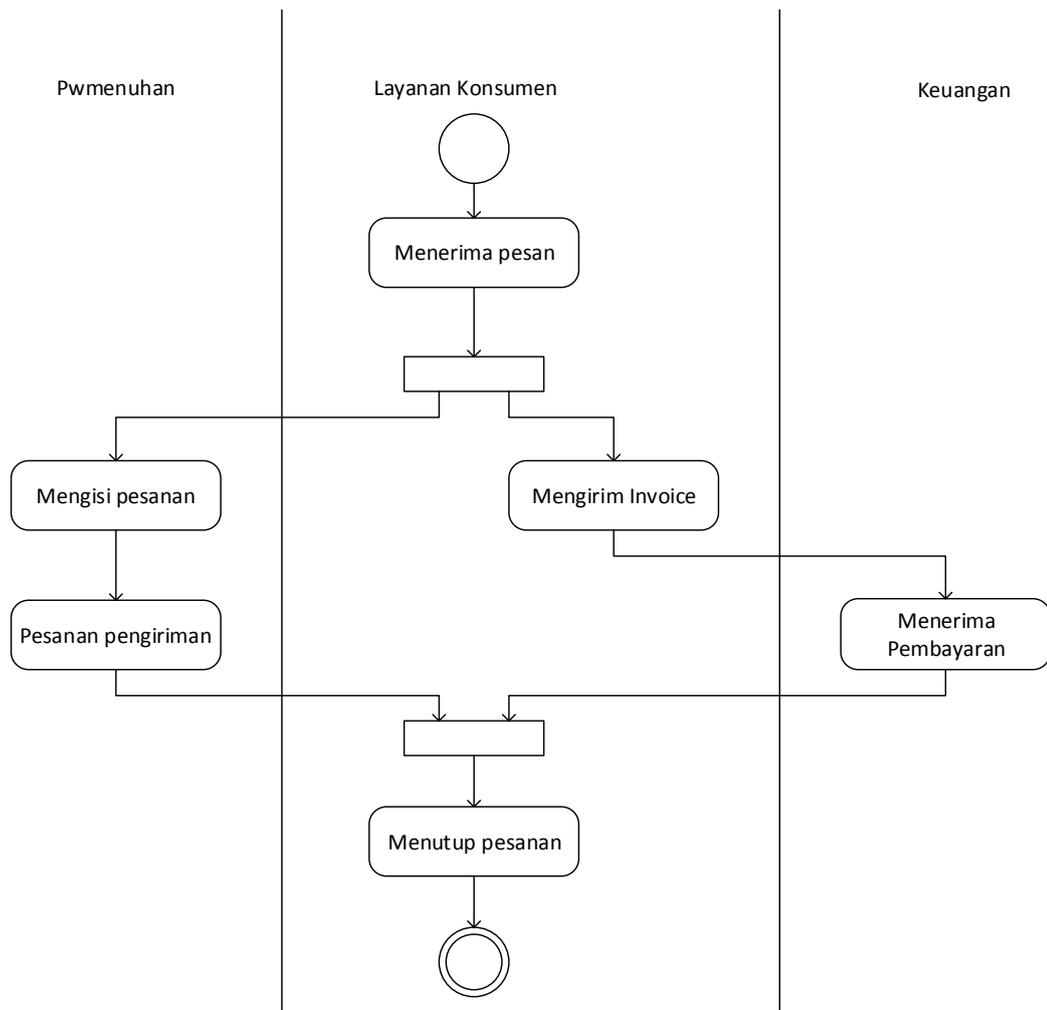
2.1.14.2 Activity Diagram

Diagram *activity* adalah teknik untuk menggambarkan logika prosedural, proses bisnis, dan jalur kerja. [10].



Gambar 2. 4 Contoh Diagram Aktivitas

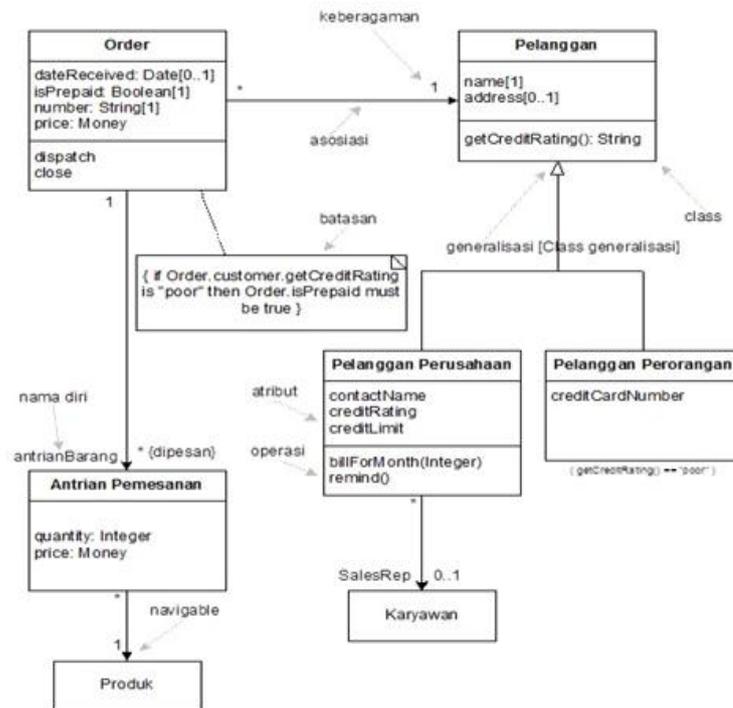
Activity diagram memberi tahu anda tentang apa yang terjadi, tetapi diagram ini tidak memberi tahu anda tentang siapa yang melakukan apa. Jika anda ingin menunjukkan siapa melakukan apa, anda dapat memisahkan activity diagram ke dalam partisi-partisi yang menampilkan action mana yang dilakukan oleh sebuah class atau organisasi. Pembuatan partisi pada Gambar 2.5 merupakan pembuatan partisi sederhana satu dimensi. Model ini sering disebut sebagai swim lanes.



Gambar 2. 5 Partisi Pada Activity Diagram

2.1.14.3 Diagram Kelas

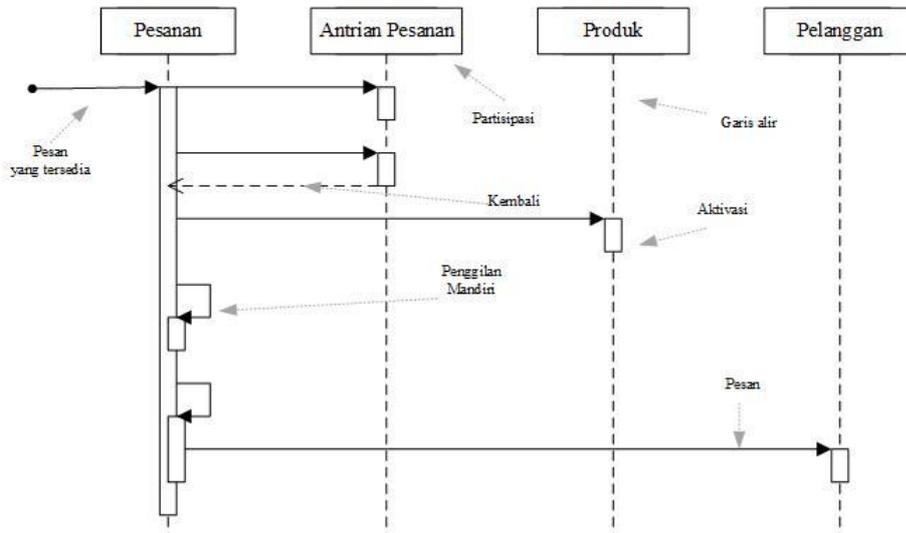
Diagram *class* mendeskripsikan jenis-jenis objek dalam sistem dan berbagai macam hubungan statis yang terdapat diantara mereka. Diagram *class* juga menunjukkan properti dan operasi sebuah *class* dan batasan-batasan yang terdapat pada sebuah hubungan-hubungan objek. UML menggunakan istilah fitur sebagai istilah umum yang meliputi properti dan operasi sebuah *class*. Didalam *class* dibagi menjadi tiga bagian. Nama *class*, atributnya, dan operasinya. [10].



Gambar 2. 6 Contoh Diagram Class

2.1.14.4 Sequence Diagram

Diagram *sequence* secara khusus menjabarkan *behavior* sebuah skenario tunggal. Diagram tersebut menunjukkan sejumlah objek contoh dan pesan-pesan yang melewati objek-objek di dalam *use case*. Diagram *sequence* menunjukkan interaksi dengan menampilkan setiap partisipan dengan garis alir secara vertikal dan pengurutan pesan dari atas ke bawah [10].



Gambar 2. 7 Contoh Diagram Sequence

2.1.15 Diagram Konteks

Diagram Context, adalah data flow diagram tingkat atas (DFD Top Level), yaitu diagram yang paling tidak detail, dari sebuah sistem informasi yang menggambarkan aliran-aliran data ke dalam dan ke luar sistem dan ke dalam dan ke luar entitas-entitas eksternal. (CD menggambarkan sistem dalam satu lingkaran dan hubungan dengan entitas luar. Lingkaran tersebut menggambarkan keseluruhan proses dalam sistem)

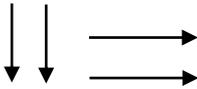
2.1.16 Data Flow Diagram

Data Flow Diagram (DFD) adalah representasi grafik dari sebuah sistem. DFD menggambarkan komponen-komponen sebuah sistem, aliran-aliran data di mana komponen-komponen tersebut, dan asal, tujuan, dan penyimpanan dari data tersebut. DFD terdiri dari empat komponen utama diantaranya adalah

1. Entitas.
2. Proses.
3. Data Store.
4. Data Flow.

DFD dapat dikatakan merupakan breakdown dari diagram konteks. Analisis mengenai sistem yang lebih detail dapat dilakukan dengan melakukan breakdown pada DFD level 1, 2 dan seterusnya. Tabel 2.1 adalah daftar simbol DFD.

Tabel 2. 1 Daftar Simbol DFD

| Simbol | Nama Simbol | Fungsi |
|--|---------------------------------|---|
|  | Entitas Luar | Menggambarkan entitas eksternal yang berhubungan dengan sistem |
|  | Sistem (Konteks) / Proses (DFD) | Menggambarkan proses yang ada dalam suatu sistem |
|  | Aliran Data/Informasi | Menggambarkan aliran data antar proses, data store dan entitas luar |
|  | Data Storage | Menggambarkan tempat penyimpanan data di dalam sistem |

2.1.17 Spesifikasi Proses

Spesifikasi Proses adalah tabel yang berisi keterangan atau deskripsi dari semua proses yang terdapat pada DFD. Logika proses yang ditulis pada spesifikasi proses harus dituliskan secara jelas baik menggunakan bahasa deskriptif atau *pseudocode* tetapi tidak boleh mengkombinasikan kedua bahasa tersebut.

Tabel 2. 2 Logika Proses

| No | Proses | Keterangan |
|----|-----------------|---|
| 1. | No Proses | Menyatakn no proses |
| 2. | Nama Proses | Menyatakan nama proses |
| 3. | Source (Sumber) | (Menyatakan sumber data input menuju proses) |
| 4. | Input | (menyatakan isi data yang masuk ke prose) |
| 5. | Output | (menyatakan informasi yang keluar dari proses) |
| 6. | Destination | (menatakan tujuan informasi output dari proses) |
| 7. | Logika Proses | (menyatakan algoritma dari proses) |

2.1.18 Kamus Data

Kamus data adalah tabel yang berisi deskripsi dari data yang mengalir pada DFD, Penjelasan struktur data dapat berupa field tiap data harus sama dengan yang sudah dimodelkan di ERD.

Tabel 2. 3 Rincian Kamus Data

| | |
|---------------------|--|
| Nama aliran data | Data Siswa Nama dari data yang digunakan |
| Where used/How used | Daftar pada poin proses apa data digunakan |
| Keterangan | Uraian singkat dari data yang digunakan |
| Struktur Data | Daftar komponen data yang ada pada data yang digunakan |
| Deskripsi | Jenis data dalam representasi komputer untuk masing-masing data. |

2.1.19 Pengujian Aplikasi

Pengujian bertujuan untuk mencari kesalahan. Pengujian yang baik adalah pengujian yang memiliki kemungkinan besar dalam menentukan kesalahan. Berikut adalah penjelasan mengenai pengujian white box dan black box.

2.1.19.1 White Box Testing

Pengujian white box didasarkan pada pemeriksaan yang teliti terhadap detail prosedural. Jalur logis diseluruh perangkat lunak dan kolaborasi antar komponen diuji dengan menguji serangkaian kondisi dan atau loop spesifik. Pengujian white box akan membawa pada kebenaran program, yang perlu dilakukan adalah mendefinisikan semua jalur logis, mengembangkan test case untuk menguji dan mengevaluasi hasilnya. Test case menjamin bahwa semua jalur independen didalam modul telah dieksekusi sedikitnya satu kali, melaksanakan semua keputusan logis pada sisi benar dan yang salah, melaksanakan semua loop pada batas dan dalam batas-batas operasional mereka, dan melakukan struktur data internal.

2.1.19.2 Black Box Testing

Pengujian black box disebut juga dengan pengujian perilaku, berfokus pada persyaratan fungsional perangkat lunak. Pengujian black box bukan teknik

alternatif untuk white box. Sebaliknya, black box merupakan pendekatan pelengkap yang mungkin dilakukan untuk mengungkap kelas kesalahan yang berbeda dari yang diungkap oleh metode white box. Pengujian black box berupaya untuk menemukan kesalahan dalam kategori seperti berikut:

1. Fungsi yang salah atau hilang.
2. Kesalahan antarmuka.
3. Kesalahan dalam struktur data atau akses basis data eksternal.
4. Kesalahan perilaku atau kinerja.
5. Kesalahan inisialisasi dan penghentian.

Tidak seperti pengujian white box, yang dilakukan pada awal proses pengujian, pengujian black box cenderung diterapkan selama tahap-tahap pengujian selanjutnya. Pengujian black box dirancang untuk menjawab pertanyaan berikut:

1. Bagaimana validitas fungsional diuji?
2. Bagaimana perilaku dan kinerja sistem diuji?
3. Kelas-kelas masukan apakah yang akan membentuk test case yang baik?
4. Apakah sistem sangat sensitif terhadap nilai masukan tertentu?
5. Bagaimana batas-batas kelas data diisolasi?
6. Beberapa kecepatan dan volume data yang dapat ditolerir oleh sistem?
7. Apakah pengaruh kombinasi spesifik data pada operasi sistem?

2.1.20 Pengujian Beta

Pengujian beta merupakan pengujian yang dilakukan secara objektif dimana diuji secara langsung ke lapangan, dengan menggunakan kuesioner mengenai tanggapan pengguna bengkel Blackjack terhadap aplikasi yang telah dibangun. Pengujian beta terdiri dari kuesioner.

2.1.21 Skala Likert

Skala data yang digunakan untuk pengukuran variabel independen adalah skala likert. skala likert adalah skala yang digunakan untuk mengukur sikap, pendapat dan persepsi seseorang atau sekelompok orang tentang fenomena sosial. Data yang telah terkumpul melalui angket, kemudian penulis olah kedalam bentuk kuantitatif, yaitu dengan cara menetapkan skor jawaban dari pertanyaan yang telah

dijawab oleh responden, dimana pemberian skor tersebut didasarkan pada ketentuan Sugiyono menyatakan bahwa variabel penelitian pada dasarnya adalah segala sesuatu yang berbentuk apa saja yang ditetapkan oleh peneliti untuk dipelajari sehingga diperoleh informasi tentang hal tersebut, kemudian ditarik kesimpulannya.