

## **BAB 2**

### **TINJAUAN PUSTAKA**

#### **2.1 Pengertian Orang Hilang**

Istilah orang ‘dinyatakan hilang’ sudah sering terdengar. Biasanya jika terjadi musibah seperti bencana alam, kecelakaan pesawat atau kapal laut, hilang dalam tugas, atau hilang karena peristiwa kriminalitas. Seseorang yang dinyatakan hilang tak berarti bahwa orang tersebut sudah pasti meninggal dunia. Ada beberapa contoh yang bisa ditelusuri dalam kasus tsunami di Aceh. Ada yang sampai bertahun-tahun dinyatakan hilang, kemudian muncul kembali dan bisa bertemu dengan keluarganya. Bisa juga terjadi orang yang hilang di tengah hutan, lalu kembali beberapa pekan kemudian [8].

##### **2.1.1 Orang Hilang Berdasarkan Hukum**

Khususnya di Indonesia ada beberapa pasal yang menjelaskan kategori-kategori yang menjadikan seseorang disebut sebagai orang hilang. diantaranya adalah

1. Pasal 44 ayat (4) UU No. 23 Tahun 2006 tentang Administrasi Kependudukan menyebutkan satu kondisi yang relevan, yaitu frasa ‘ketidakjelasan keberadaan seseorang karena hilang’. Selengkapnya dinyatakan begini: “Dalam hal terjadi ketidakjelasan keberadaan seseorang karena hilang atau mati tetapi tidak ditemukan jenazahnya, pencatatan oleh Pejabat Pencatatan Sipil baru dilakukan setelah adanya penetapan pengadilan”.
2. Pasal 45 ayat (3) Undang-Undang yang sama mengatur tentang warga negara Indonesia yang hilang di luar wilayah Indonesia. Namun Undang-Undang ini tak mendefinisikan sama sekali apa yang dimaksud orang yang dinyatakan hilang. Dalam definisi peristiwa kependudukan pun tak disinggung tentang ‘hilang’. Peristiwa kependudukan adalah kejadian yang dialami penduduk yang harus dilaporkan karena membawa akibat terhadap penerbitan atau perubahan Kartu Keluarga, Kartu Tanda Penduduk dan/atau surat keterangan kependudukan lainnya meliputi pindah datang, perubahan alamat, serta status tempat tinggal terbatas menjadi tinggal tetap.

3. Dalam Undang-Undang No. 12 Tahun 2006 tentang Kewarganegaraan, dikenal juga frasa ‘dinyatakan hilang’ meskipun dalam konteks kehilangan kewarganegaraan. Jadi, dalam hal ini bukan orangnya yang dinyatakan hilang, melainkan kewarganegaraannya. [8]

### **2.1.2 Contoh Kasus Orang Hilang**

Seseorang yang dinyatakan hilang tak berarti bahwa orang tersebut sudah pasti meninggal dunia. Namun hilang ini sendiri tidak ada aturan kapan waktu minimal agar seseorang bisa dinyatakan hilang, kecuali anggota kepolisian, khususnya pada kasus hilang dalam tugas. Maksudnya adalah pada saat keadaan anggota Polri melaksanakan tugas Kepolisian namun tidak diketahui keberadaannya dan tidak diketahui apakah masih hidup atau telah meninggal dunia dan status hilang akan diberikan setelah dilakukan upaya pencarian selama satu tahun, namun tidak diketemukan. [8]

Selain itu ada juga beberapa contoh yang bisa ditelusuri dalam kasus tsunami di Aceh. Ada yang sampai bertahun-tahun dinyatakan hilang, kemudian muncul kembali dan bisa bertemu dengan keluarganya. Bisa juga terjadi orang yang hilang di tengah hutan, lalu kembali beberapa pekan kemudian. Seperti kasus Nining Sunarsih. Warga Kampung Cibunar Desa Gede Pangrango Kecamatan Kadudampit, Nining Sunarsih itu ditemukan di Sukabumi setelah menghilang 18 bulan. Ia sudah dianggap hilang, sebelum kemudian ditemukan masih hidup. Belakangan, polisi menengarai Nining bukan hilang, tetapi sengaja bersembunyi karena terbelit utang. [8]

## **2.2 React Native**

React Native adalah framework open source besutan facebook yang dibuat untuk membuat aplikasi mobile dengan menggunakan javascript. Dengan kata lain, react native memungkinkan untuk para developer web untuk membuat aplikasi mobile yang mirip seperti aplikasi native. React native juga membuat pembuatan ini lebih mudah karena sebagian besar kode yang dibuat atau dituliskan dapat dipakai pada multiplatform, hal tersebut dikarenakan react native bersifat

multiplatform sehingga dapat memudahkan dalam pembuatan atau pengembangan aplikasi secara bersamaan untuk android maupun iOS. [8]

React native sendiri ditulis dengan campuran JavaScript dan markup XML-esque, yang dikenal sebagai JSX yang Kemudian React Native "bridge" memanggil API rendering asli di Objective-C (untuk iOS) atau Java (untuk Android). Dengan demikian, aplikasi yang telah dibuat atau dituliskan akan dirender menggunakan komponen UI aslinya dan bukan sebuah tampilan web sehingga akan terlihat dan terasa seperti aplikasi mobile lainnya. Selain itu, React Native juga mengekspos antarmuka JavaScript untuk API platform, sehingga aplikasi react native dapat mengakses fitur platform seperti kamera atau lokasi pengguna. Proyek React Native inti mendukung penulisan aplikasi seluler untuk iOS dan Android [9]

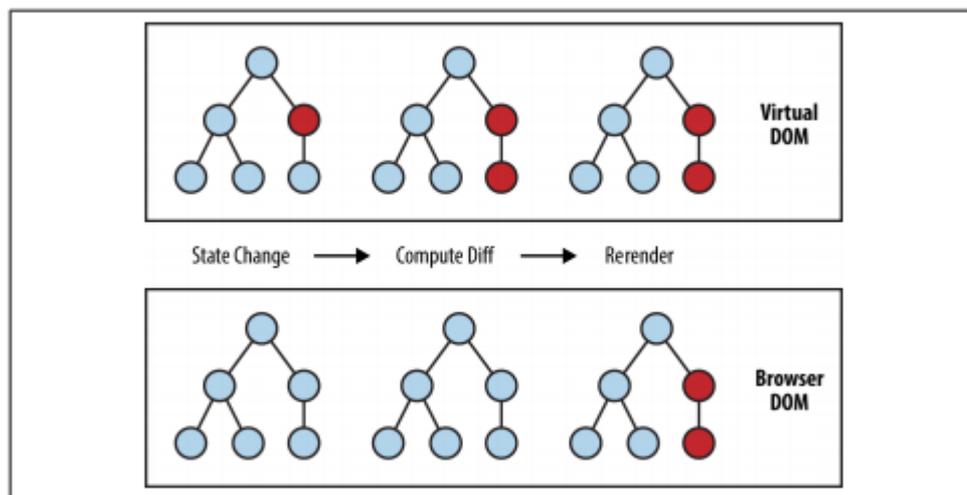
### **2.2.1 Keuntungan React Native**

Fakta bahwa React Native merender menggunakan API rendering standar platform dapat membuat react native berbeda dari kebanyakan framework pengembangan multiplatform lainnya seperti Cordova ataupun Ionic. Framework aplikasi mobile yang mengkombinasikan JavaScript, HTML, dan CSS biasanya disajikan dengan menggunakan tampilan web. Meskipun hal ini dapat bekerja namun hal ini juga terdapat kelemahan, terutama di kinerja Selain itu, framework tersebut biasanya tidak memiliki akses ke kumpulan elemen UI yang asli. Ketika framework tersebut mencoba meniru UI asli, hasilnya biasanya terasa kurang. Selain itu, reverse-engineering pada detail baik dari hal-hal seperti animasi pun membutuhkan banyak usaha, sehingga aplikasi dengan framework tersebut dapat dengan cepat menjadi usang. Sebaliknya, React Native sebenarnya menerjemahkan markup yang dibuat ke elemen UI asli, sehingga memberikan tampilan yang serupa dengan aslinya pada platform apa pun yang sedang dikerjakan. Siklus pembaharuan di React Native sama dengan React. Ketika props atau status berubah, React Native merender view. Perbedaan utama antara react native dan reactjs di browser adalah bahwa React Native melakukan ini dengan memanfaatkan library UI dari platform aslinya (android ataupun iOS) daripada menggunakan markup HTML dan CSS. Sehingga untuk developer yang terbiasa bekerja di web dapat merasakan

pengalaman membuat aplikasi mobile dengan kinerja dan tampilan dan nuansa aplikasi pada platform mobile aslinya dengan menggunakan tools dan bahasa yang sudah dikuasai. [9]

### 2.2.2 Cara Kerja React Native

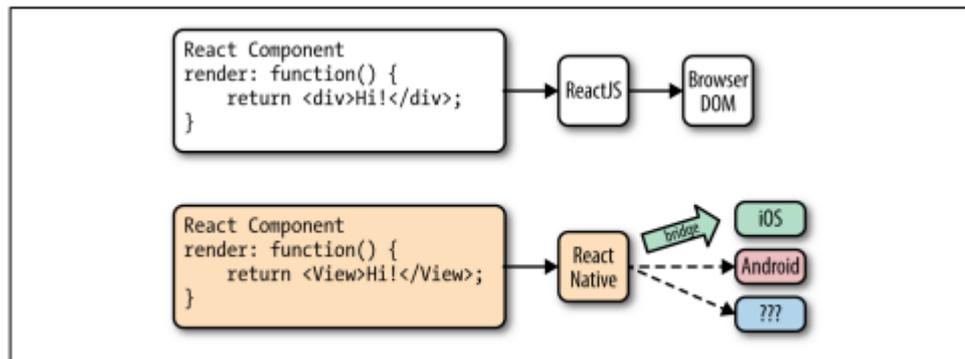
Untuk memahami teknis yang mendasari React Native, pertama-tama diperlukan untuk memahami salah satu konsep React: Virtual-DOM .pada react native, virtual DOM bertindak sebagai penyambung antara hal-hal yang harus ditampilkan dan fungsi yang harus dilakukan untuk benar-benar membuat sebuah web. Untuk membuat antarmuka pengguna interaktif di browser, developer harus mengedit DOM atau Document Object Model. Ini adalah step yang sulit menulis karna akan berdampak signifikan terhadap kinerja. Gambar 2.1 akan menunjukkan cara kerjanya.



Gambar 2.1 Cara Kerja DOM

Dalam react di web, sebagian besar developer menganggap Virtual DOM sebagai optimasi kinerja. Virtual DOM tentu memiliki manfaat kinerja, tetapi potensi sesungguhnya terletak pada kekuatan abstraksinya. Menempatkan lapisan abstraksi yang bersih antara kode pengembang dan rendering. Ini adalah cara React Native bekerja, seperti yang ditunjukkan pada Gambar 2-2. Alih-alih rendering ke DOM browser, React Native memanggil API Objective-C untuk dirender ke

komponen iOS, atau Java API untuk dirender ke komponen Android. Ini membedakan React Native dari opsi pengembangan aplikasi lintas platform lainnya, yang sering berakhir dengan rendering tampilan berbasis web.

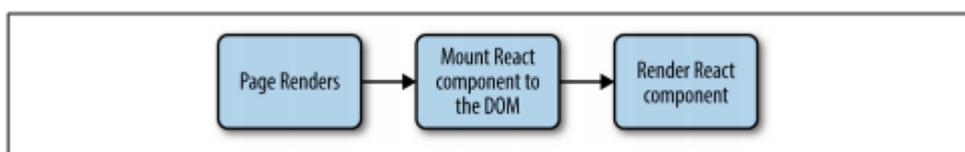


Gambar 2.2 Perbedaan Rendering React Native

Hal ini sangat memungkinkan karena adanya jembatan yang menghubungkan react dengan antarmuka elemen UI asli platform host. React komponen mengembalikan markup yang menjelaskan bagaimana seharusnya tampilannya pada platform host. Jadi react untuk web, ini menerjemahkan langsung ke DOM browser, sementara untuk React native, markup ini diterjemahkan agar sesuai dengan platform host, sehingga `<View>` mungkin menjadi UI View khusus iOS. Proyek React Native inti juga mendukung iOS dan Android. [9]

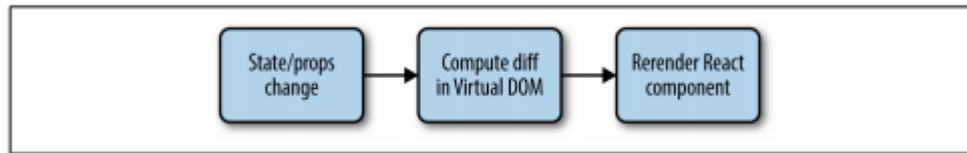
### 2.2.3 Siklus Rendering

Jika sudah terbiasa dengan react, maka akan mudah memahami siklus render yang ada pada react native karena hampir sama, ketika react berjalan di browser, render akan dimulai dengan mount react



Gambar 2.3 Mount React

Setelah react menangani rendering lalu akan merendering ulang komponen yang diperlukan.



Gambar 2.4 render

Untuk React Native, siklus hidupnya sama, tetapi proses renderingnya sedikit berbeda karena react native tergantung pada jembatan. coba lihat pada Gambar 2.2 menerjemahkan panggilan JavaScript dan memanggil platform hostAPI dan elemen UI yang mendasarinya (Objective-C atau Java). Karena React Native tidak berjalan pada satu UI utama.

### 2.3 Expo

Expo merupakan suatu set SDK tools, library, dan services yang digunakan dalam pengembangan aplikasi berbasis react native. Pada dasarnya Expo dan react native itu sendiri sama hal dalam menuliskan kode program akan tetapi pada Expo dapat lebih cepat karena beberapa tools, library, dan services sudah disediakan oleh Expo itu sendiri. [10]

#### 2.3.1 Keunggulan expo

Keunggulan expo adalah sebagai berikut:

1. Expo menggunakan Expo SDK jadi tidak memerlukan Android Studio dan Xcode
2. Dapat memudahkan pengaturan aplikasi, sehingga bisa langsung ditambahkan pada file app.json.
3. Mode debug aplikasi yang dapat langsung di jalankan pada device dengan aplikasi Expo pada Android dan IOS dengan membagikan link / qr code project
4. Beberapa fitur seperti Push Notification dan Facebook login sudah disediakan oleh Expo
5. Beberapa library seperti maps, camera, dan gallery sudah di sediakan Expo sehingga tidak perlu menambahkan library dan dapat berjalan pada Android dan IOS

6. Membuat versi rilis aplikasi lebih mudah karena dilakukan oleh server Expo (Expo akan memberikan link untuk mendownload versi rilis aplikasi yang dibuat)

## 2.4 Javascript

Javascript diperkenalkan pertama kali oleh Netscape pada tahun 1995. Pada awalnya bahasa ini dinamakan “LiveScript” yang berfungsi sebagai bahasa sederhana untuk browser Netscape Navigator 2. Pada masa itu bahasa ini banyak di kritik karena kurang aman, pengembangannya yang terkesan buru buru dan tidak ada pesan kesalahan yang di tampilkan setiap kali kita membuat kesalahan pada saat menyusun suatu program. Kemudian sejalan dengan sedang giatnya kerjasama antara Netscape dan Sun (pengembang bahasa pemrograman “Java” ) pada masa itu, maka Netscape memberikan nama “JavaScript” kepada bahasa tersebut pada tanggal 4 desember 1995. Pada saat yang bersamaan Microsoft sendiri mencoba untuk mengadaptasikan teknologi ini yang mereka sebut sebagai “Jscript” di browser Internet Explorer 3.

Javascript adalah bahasa yang berbentuk kumpulan skrip yang pada fungsinya berjalan pada suatu dokumen HTML, sepanjang sejarah internet bahasa ini adalah bahasa skrip pertama untuk web. Bahasa ini adalah bahasa pemrograman untuk memberikan kemampuan tambahan terhadap bahasa HTML dengan mengijinkan pengekseskuan perintah perintah di sisi user, yang artinya di sisi browser bukan di sisi server web. Javascript bergantung kepada browser(navigator) yang memanggil halaman web yang berisi skrip skrip dari Javascript dan tentu saja terselip di dalam dokumen HTML. Javascript juga tidak memerlukan kompilator atau penterjemah khusus untuk menjalankannya (pada kenyataannya kompilator Javascript sendiri sudah termasuk di dalam browser tersebut). Lain halnya dengan bahasa “Java” (dengan mana JavaScript selalu di banding bandingkan) yang memerlukan kompilator khusus untuk menterjemahkannya di sisi user/klien. [11]

### **2.4.1 Kelebihan Javascript**

Terdapat sejumlah kelebihan JavaScript yang menjadikan bahasa pemrograman ini lebih unggul jika dibandingkan dengan kompetitornya, terutama di beberapa kasus tertentu. Berikut beberapa kelebihan JavaScript:

1. Tidak membutuhkan compiler karena web browser mampu menginterpretasikannya dengan HTML;
2. Lebih mudah dipelajari jika dibandingkan dengan bahasa pemrograman lainnya;
3. Error atau kesalahan lebih mudah dicari dan ditangani;
4. Dapat dialih tugaskan ke elemen halaman web atau even tertentu, misalnya klik atau mouseover;
5. JS dapat digunakan di berbagai browser, platform, dan lain-lain.
6. Dapat menggunakan JavaScript untuk memvalidasi input dan mengurangi keinginan untuk mengecek data secara manual;
7. Dengan JavaScript, website akan menjadi lebih interaktif
8. Lebih cepat dan ringan jika dibandingkan dengan bahasa pemrograman lainnya.

### **2.4.2 kekurangan javascript**

Setiap bahasa pemrograman pasti memiliki beberapa kekurangan. Berikut adalah beberapa kekurangan JavaScript, yaitu:

1. Berisiko tinggi terhadap eksploitasi;
2. Dapat dimanfaatkan untuk mengaktifkan kode berbahaya di komputer pengguna;
3. Tidak selalu didukung oleh berbagai browser dan perangkat;
4. JS code snippet agak banyak;
5. Dapat di-render secara berbeda pada masing-masing perangkat yang malah mengarah ke inkonsistensi.

## 2.5 Node Js

Javascript merupakan bahasa pemrograman yang lengkap hanya saja selama ini di pakai sebagai bahasa untuk pengembangan aplikasi web yang berjalan pada sisi client atau browser saja. Tetapi sejak ditemukannya Node.js oleh Ryan Dhal pada tahun 2009, Javascript bisa digunakan sebagai bahasa pemrograman di sisi server sekelas dengan PHP, ASP, C#, Ruby dll dengan kata lain Node.js menyediakan platform untuk membuat aplikasi Javascript dapat dijalankan di sisi server. Untuk mengeksekusi Javascript sebagai bahasa server diperlukan engine yang cepat dan mempunyai performansi yang bagus. Engine Javascript dari Google bernama V8 yang dipakai oleh Node.js yang merupakan engine yang sama yang dipakai di browser Google Chrome [12].

### 2.5.1 Keuntungan Node JS

Tak terelakkan bahwa Javascript merupakan bahasa pemrograman yang paling populer. Jika anda sebagai developer pernah mengembangkan aplikasi web maka penggunaan Javascript pasti tidak terhindarkan dan dengan node js sekarang javascript dapat berjalan pada server. Berikut adalah keuntungan pemakaian node js:

1. Pengembang hanya memakai satu bahasa untuk mengembangkan aplikasi lengkap client & server sehingga mengurangi Learning Curve untuk mempelajari bahasa server yang lain.
2. Sharing kode antara client dan server atau istilahnya code reuse.
3. Javascript secara native mendukung JSON yang merupakan standar transfer data yang banyak dipakai saat ini sehingga untuk mengkonsumsi data-data dari pihak ketiga pemrosesan di Node.js akan sangat mudah sekali.
4. Database NoSQL seperti MongoDB dan CouchDB mendukung langsung Javascript sehingga interfacing dengan database ini akan jauh lebih mudah.
5. Node.js memakai V8 yang selalu mengikuti perkembangan standar ECMAScript, jadi tidak perlu ada kekhawatiran bahwa browser tidak akan mendukung fitur-fitur di Node.js.

## 2.6 JSON

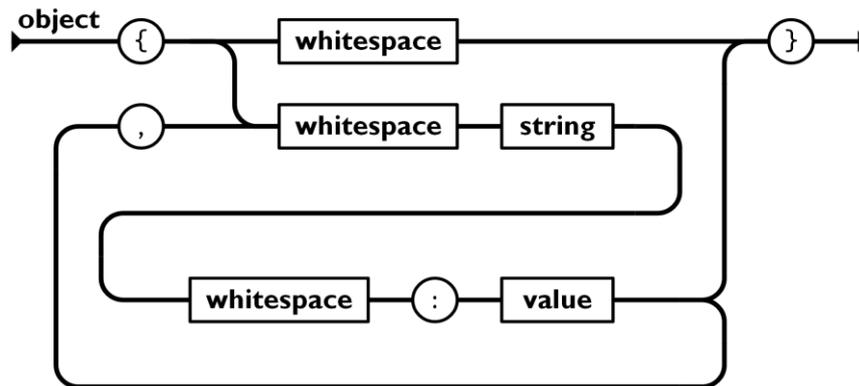
JSON (JavaScript Object Notation) adalah format pertukaran data yang ringan, mudah dibaca dan ditulis oleh manusia, serta mudah diterjemahkan dan dibuat (generate) oleh komputer. Format ini dibuat berdasarkan bagian dari Bahasa Pemrograman JavaScript, Standar ECMA-262 Edisi ke 3 – Desember 1999. JSON 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#, JavaScript, Perl, Python dll. [13] Oleh karena sifatsifat 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), 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 modern mendukung sruktur 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

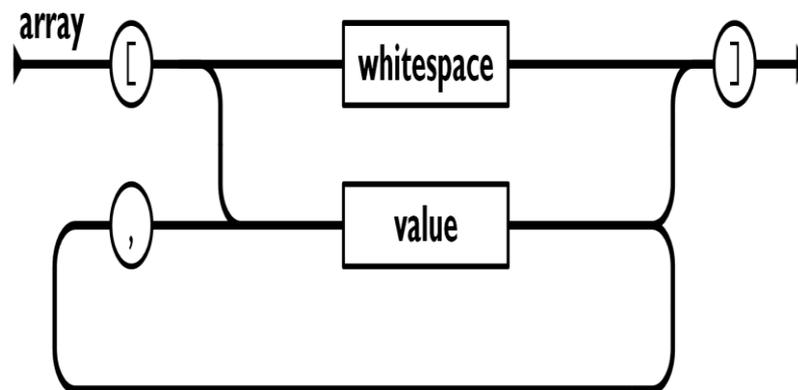
Objek adalah sepasang nama / nilai yang tidak terurutkan. Seperti yang diilustrasikan pada Gambar 2.5 objek dimulai dengan { (kurung kurawal buka) dan diakhiri dengan } (kurung kurawal tutup). Setiap nama diikuti dengan : (titik dua) dan setiap pasangan nama/nilai dipisahkan oleh , (koma). Objek biasanya digunakan untuk menyimpan data tunggal dalam bentuk JSON



Gambar 2.5 Object

## 2. Larik

Larik adalah kumpulan nilai yang terurutkan. Larik dimulai dengan [ (kurung kotak buka) dan diakhiri dengan ] (kurung kotak tutup). Setiap nilai dipisahkan oleh , (koma) seperti yang diilustrasikan pada Gambar 2.6. Larik dalam JSON dapat digunakan sebagai value dari JSON object hal ini dapat berguna jika JSON menyimpan data bertingkat.

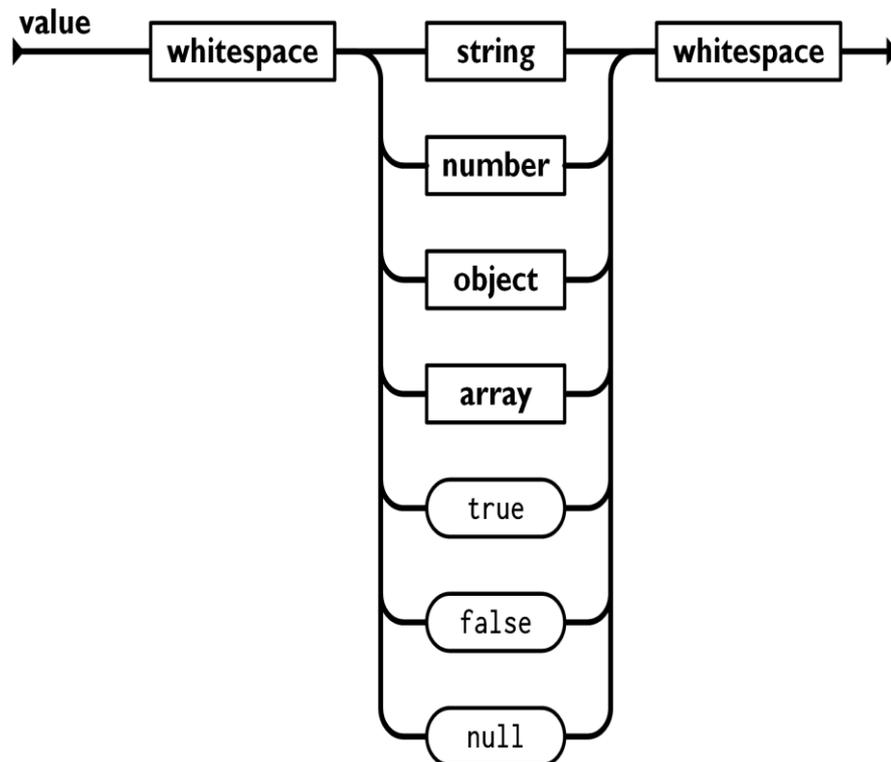


Gambar 2.6 Larik

Bentuk data JSON objek dan larik dapat saling dikombinasikan untuk mendukung struktur data yang lebih kompleks. JSON mendukung beberapa tipe data untuk menjadi value seperti Angka, String, Boolean dan Nilai NULL .

### 3. Nilai

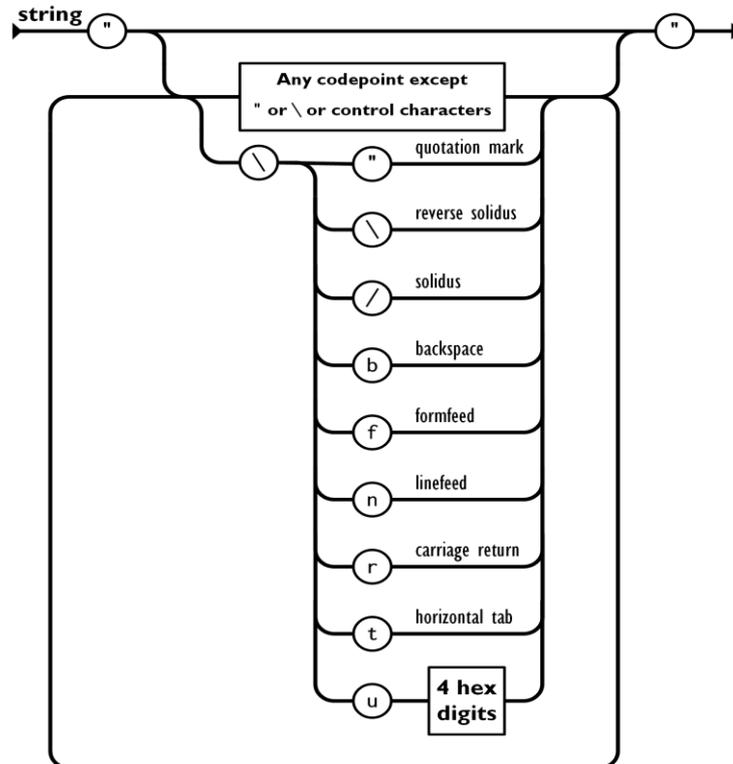
Nilai (value) dapat berupa sebuah string dalam tanda kutip ganda, atau angka, atau true atau false atau null, atau sebuah objek atau sebuah larik. Strukturstruktur tersebut dapat disusun bertingkat .



**Gambar** Error! No text of specified style in document..1 Nilai

### 4. String

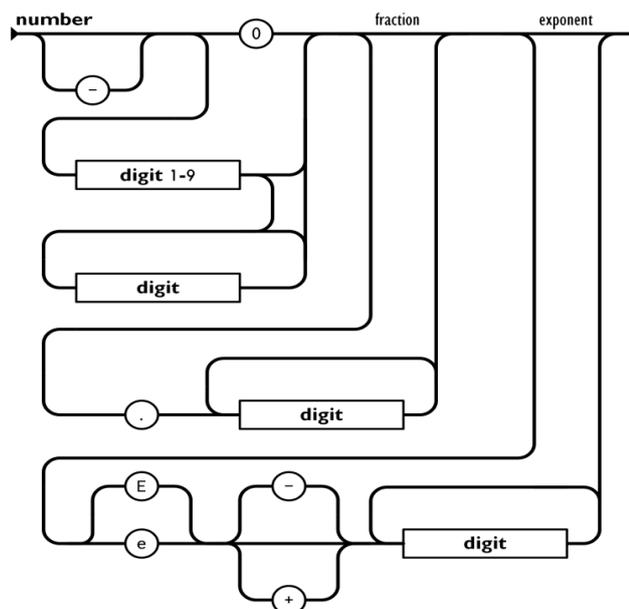
String adalah kumpulan dari nol atau lebih karakter Unicode, yang dibungkus dengan tanda kutip ganda. Di dalam string dapat digunakan backslash escapes "\" untuk membentuk karakter khusus. Sebuah karakter mewakili karakter tunggal pada string. String sangat mirip dengan string C atau Java .



Gambar 2.8 String

## 5. Angka

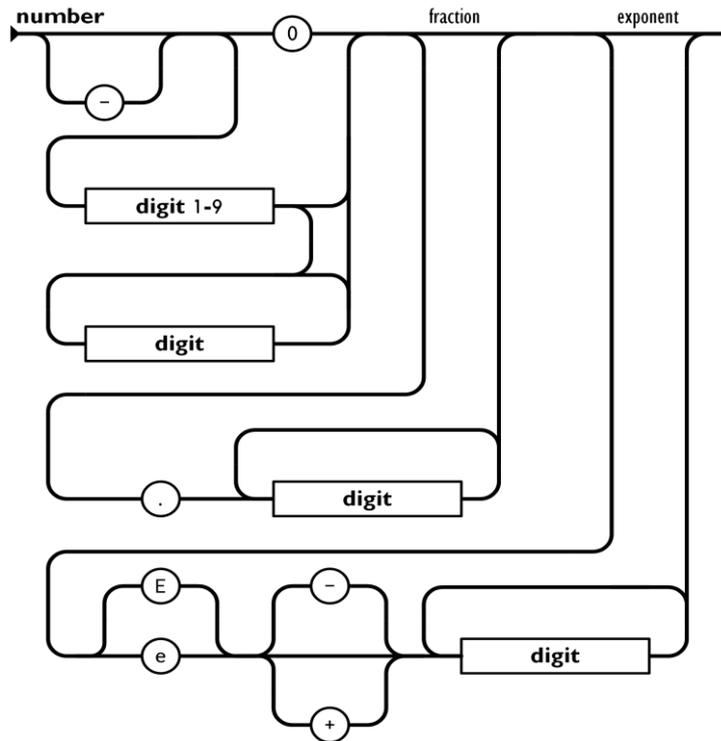
Angka, Format oktal dan heksadesimal tidak digunakan .



Gambar 2.9 Angka

## 6. Spasi

Spasi dapat disisipkan di antara setiap token



**Gambar 2.10 Spasi**

## 2.7 API

API adalah sekumpulan perintah, fungsi, dan protokol yang dapat digunakan saat membangun perangkat lunak untuk sistem operasi tertentu. API memungkinkan programmer untuk menggunakan fungsi standar untuk berinteraksi dengan system operasi. API atau Application Programming Interface juga merupakan suatu dokumentasi yang terdiri dari antar muka, fungsi, kelas, struktur untuk membangun sebuah perangkat lunak. Dengan adanya API, maka memudahkan seorang programmer untuk membongkar suatu software untuk kemudian dapat dikembangkan atau diintegrasikan dengan perangkat lunak yang lain. API dapat dikatakan sebagai penghubung suatu aplikasi dengan aplikasi lainnya. Suatu rutin standar yang memungkinkan developer menggunakan system function. Proses ini dikelola melalui operating system. Keunggulan dari API ini

adalah memungkinkan suatu aplikasi dengan aplikasi lainnya untuk saling berinteraksi . Keuntungan dengan menggunakan API adalah sebagai berikut:

1. Portabilitas.

Developer yang menggunakan API dapat menjalankan programnya dalam sistem operasi mana saja asalkan sudah terinstal API tersebut.

2. Lebih Mudah Dimengerti

API menggunakan bahasa yang lebih terstruktur dan mudah dimengerti daripada bahasa system call. Hal ini sangat penting dalam hal editing dan pengembangan. System call interface ini berfungsi sebagai penghubung antara API dan system call yang dimengerti oleh sistem operasi. System call interface ini akan menerjemahkan perintah dalam API dan kemudian akan memanggil system calls yang diperlukan. Untuk membuka suatu file tersebut user menggunakan program yang telah dibuat dengan menggunakan bantuan API, maka perintah dari user tersebut diterjemahkan dulu oleh program menjadi perintah `open()`.

Perintah `open()` ini merupakan perintah dari API dan bukan perintah yang langsung dimengerti oleh kernel sistem operasi. Oleh karena itu, agar keinginan pengguna dapat dimengerti oleh sistem operasi, maka perintah `open()` tadi diterjemahkan ke dalam bentuk system call oleh system call interface. Implementasi perintah `open()` tadi bisa bermacam-macam tergantung dari sistem operasi yang digunakan .

Cara menggunakan API :

1. Dilakukan dengan mengimpor package/kelas
2. Ada beberapa kelas bernama sama dipackage yang berbeda, yaitu :
  - a. Import salah satu dan gunakan nama lengkap untuk yang lain
  - b. Gunakan nama lengkap semua kelas

Kebanyakan Sistem Operasi seperti Windows, menyediakan fasilitas API sehingga programmer dapat melakukan aktivitas programming dengan lebih konsisten. Meskipun API didesain untuk programmer, namun API juga baik untuk user karena setidaknya dapat menjamin bahwa program tersebut memiliki interface yang sama, sehingga lebih mudah untuk dipelajari.[8]

## 2.8 Clarifai

Clarifai adalah perusahaan kecerdasan buatan yang unggul dalam pengenalan visual (Visual Recognition), Clarifai didirikan pada tahun 2013 oleh Matthew Zeiler, seorang ahli terkemuka dalam Machine Learning, Clarifai telah menjadi pemimpin pasar sejak memenangkan lima tempat teratas dalam klasifikasi citra pada kompetisi ImageNet 2013. Clarifai adalah alat yang dapat mengidentifikasi atau mengenali gambar atau video yang dimasukkan sebagai input yang dapat memberikan hasil berupa prediksi tentang apa yang ada di dalam gambar atau video berupa besaran probabilitas kemungkinannya. [14]

API Clarifai dibangun dengan konsep ide dasar yang sederhana dimana anda memasukkan input berupa gambar atau video dan kemudian anda akan mendapatkan hasil dari prediksi yang diminta. Setiap tipe dari prediksi adalah berdasar dari model apa yang dipilih untuk menjalankan prediksi dari input gambar atau video. Sebagai contoh apabila anda memilih model makanan untuk gambar masukkan anda maka Clarifai akan memprediksi gambar berdasarkan model makanan yang telah diketahui dan begitu seterusnya.



**Gambar 2.11 Model Clarifai**

### 2.8.1 Model Clarifai

Gambar atau video yang menjadi data masukkan yang dijalankan ditandai oleh model dimana sebuah model adalah classifier yang terlatih dan dapat mengenali apa yang terdapat di dalam gambar atau video sesuai dengan apa yang telah diketahui. Model satu dan yang lainnya digunakan untuk mengetahui hal-hal yang berbeda pula. Menjalankan model yang berbeda untuk mengenali suatu

gambar atau video dapat memberikan hasil yang berbeda secara drastis. Berikut adalah beberapa contoh dari model pada Clarifai :

1. Face model

Face model memberikan hasil prediksi berupa besaran probabilitas yang terdeteksi berdasar model wajah yang diketahui dari gambar yang Anda masukkan

2. General model

General model adalah model dimana berisi tag yang lebih umum dan berisi berbagai topik yang berbeda dan akan menghasilkan prediksi yang umum yang terdapat di gambar.

3. Not Safe For Work (NSFW) model

NSFW model analisis berupa gambar atau video berdasar pada besaran skor probabilitas hasilpengembalian kemungkinan bahwa gambar bermuatan pornografi.

3. Food model

Model Makanan menganalisa gambar dan video dan skor probabilitas pengembalian kemungkinan bahwa gambar berisi bahan makanan diakui dan hidangan. Model saat ini dirancang untuk mengidentifikasi item makanan tertentu dan bahan-bahan yang terlihat

4. Travel model

Model Travel menganalisa gambar dan skor probabilitas pengembalian kemungkinan bahwa gambar berisi kategori perjalanan terkait diakui. Model saat ini dirancang untuk mengidentifikasi fitur khusus dari perumahan, hotel dan properti perjalanan terkait.

Selain yang telah dipaparkan di atas masih ada model-model yang lain seperti color, wedding, celebrity, demographic, logo, apparel, focus, moderation yang masing-masing memiliki fungsi dan fokus yang berbeda.

### **1.8.2 Authentication**

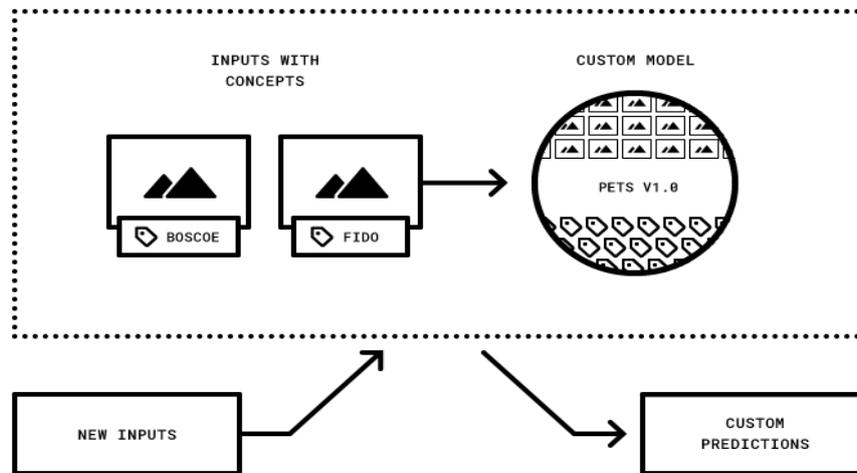
Otentikasi ke API ditangani melalui API key. Anda dapat membatasi cakupan API key, yang memungkinkan kunci untuk melakukan operasi yang sangat spesifik pada aplikasi yang diberikan, agar aplikasi Anda tetap aman.

### **1.8.3 Predict**

Predict, melakukan prediksi dan analisa gambar yang Anda masukkan dan memberi tahu Anda apa isi yang terkandung di gambar. API akan mengembalikan daftar konsep dengan probabilitas yang sesuai dari seberapa besar kemungkinan konsep-konsep ini terkandung di dalam gambar. Bila Anda membuat prediksi melalui API, Anda memberi tahu model apa yang akan digunakan. Sebuah model berisi sekelompok konsep. Model hanya akan melihat konsep yang dikandungnya. Anda dapat menggunakan model yang berbeda untuk menganalisis gambar dengan berbagai cara

### **1.8.4 Train**

Train atau pelatihan memungkinkan Anda membuat model sendiri menggunakan konsep kustom Anda sendiri. Anda mulai dengan menambahkan masukan gambar yang sudah Anda ketahui berisi konsep yang Anda minati. Anda tidak memerlukan banyak gambar untuk memulai. Sebaiknya mulai dengan 10 dan tambahkan lebih banyak sesuai kebutuhan. Kemudian membuat model dan menentukan konsep apa yang dikandungnya. Setelah membuat model, Anda melatihnya berdasarkan gambar dan konsep yang Anda berikan. Operasi pelatihan ini bersifat asinkron. Mungkin perlu beberapa detik agar model Anda dilatih dan siap sepenuhnya. Setelah selesai latihan, Anda bisa menggunakan model itu untuk memprediksi konsep tersebut pada gambar baru



**Gambar 2.12 Model Train**

### 1.9 Firebase

Firebase merupakan salah satu layanan dari Google yang memudahkan para app developer dalam mengembangkan aplikasi. Firebase sendiri termasuk kedalam kategori Baas (Backend as a Service).[12]

Untuk mendapatkan layanan firebase pertama harus mengakses halaman konsol Firebase dengan alamat <https://console.firebase.google.com>. Firebase sendiri memungkinkan untuk digunakan pada platform iOS, Android dan Web. Karena penelitian ini menggunakan aplikasi hybrid berbasis web, maka implementasi dilakukan pada platform berbasis web.



**Gambar 2.13 Memulai Google Firebase**

Gambar di atas adalah tahapan awal memulai Google Firebase untuk aplikasi hybrid, yaitu memilih “Tambahkan Firebase ke aplikasi web Anda”. Untuk

menambahkan Firebase ke aplikasi berbasis web, yang perlu dilakukan adalah menambahkan cuplikan kode seperti di bawah ini ke HTML.

```
// TODO: Replace with your project's customized code snippet
<script src="https://www.gstatic.com/firebasejs/3.4.8/firebase.js"></script>
<script>
  // Initialize Firebase
  var config = {
    apiKey: '<your-api-key>',
    authDomain: '<your-auth-domain>',
    databaseURL: '<your-database-url>',
    storageBucket: '<your-storage-bucket>'
  };
  firebase.initializeApp(config);
</script>
```

**Gambar 2.14 Kode Inisialisasi Firebase**

Cuplikasi kode di atas berisi informasi inisialisasi untuk mengkonfigurasi Firebase JavaScript SDK agar dapat menggunakan Autentikasi, Storage dan Realtime Database.

### 2.9.1 Firebase Authentication

Sebagian besar aplikasi tentu membutuhkan autentikasi untuk mengetahui identitas pengguna. Mengetahui identitas pengguna memungkinkan aplikasi untuk secara aman menyimpan data pengguna di cloud dan memberikan pengalaman pribadi yang sama di semua perangkat pengguna. Firebase menyediakan layanan backend, SDK yang mudah digunakan dan pustaka siap pakai untuk mengautentikasi pengguna ke aplikasi. Saat ini Firebase mendukung autentikasi menggunakan sandi, sosial media populer seperti Google, Facebook, Twitter dan lainnya. Ujicoba autentikasi yang dilakukan pada penelitian ini menggunakan email dan kata sandi.

#### 1. Registrasi

Ketika pengguna pertama sekali mengakses aplikasi, maka akan ditampilkan formulir registrasi sederhana dengan meminta pengguna tersebut memasukkan email dan kata sandi. Secara otomatis setelah proses registrasi selesai, maka pengguna tersebut masuk ke aplikasi.

## 2. Login

Pengguna yang telah berhasil melakukan registrasi, maka ia bisa login ke aplikasi menggunakan email dan kata sandi tersebut.

## 3. Logout

Logout atau keluar dari aplikasi.

Selain fitur di atas tersebut, Google Firebase Authentication juga menyediakan fitur untuk mengelola pengguna seperti ubah kata sandi, ubah email dan verifikasi email.

### **2.9.2 Firebase Realtime Database**

Firebase Realtime Database merupakan cloud database. Data disimpan dalam format JSON dan disinkronkan secara realtime ke setiap klien yang terhubung. Ketika membangun aplikasi hybrid lintas platform, seperti Android dan iOS maka semua klien berbagi satu instance Realtime Database dan secara otomatis menerima pembaruan dengan data tertentu. Firebase Realtime Database adalah basis data NoSQL dan karena itu memiliki optimalisasi dan fungsionalitas yang berbeda dibandingkan dengan basis data relasional. Membuat database Firebase bisa melalui import file JSON ke konsol Firebase, atau dapat juga dibuat langsung melalui halaman konsol Realtime Database secara manual.

### **2.9.3 Firebase Storage**

Firebase Storage digunakan untuk menyimpan data seperti gambar, audio dan video. Sebagian aplikasi pasti menggunakan file gambar, audio atau video. Adanya Firebase Storage sangat memudahkan proses unggahan dan unduhan untuk aplikasi. Beberapa kelebihan utama dari Firebase Storage adalah sebagai berikut:

#### a. Strong

Firebase Storage dapat melakukan unggahan dan unduhan file di semua kualitas jaringan internet. Ia dapat berhenti atau melanjutkan, sehingga menghemat waktu dan bandwidth pengguna.

b. Secure

Firestore terintegrasi dengan Firebase Authentication untuk menyediakan autentikasi yang mudah dan intuitif untuk pengembang. Model keamanan dapat diatur berdasarkan nama file, ukuran, tipe konten dan metadata lainnya.

c. Scalable

Firestore didukung oleh Google Cloud Storage hingga skala petabyte. Ini sangat penting, ketika aplikasi yang dari prototipe kemudian berkembang viral dengan jutaan pengguna.

Proses unggahan ke storage dapat dilakukan melalui aplikasi pengguna menggunakan AngularJS. Ini lebih mudah diimplementasikan karena tidak perlu menggunakan pemrograman berbasis server seperti PHP atau ASP. Ketika file telah berhasil diunggah ke storage, maka akan mengembalikan link unduhan (download link) yang dapat disimpan ke Firestore Database.

#### **2.9.4 Firebase Cloud Messaging**

Firebase Cloud Messaging merupakan solusi perpesanan lintas platform yang memungkinkan aplikasi mengirimkan pesan dan pemberitahuan ke aplikasi pengguna dengan tanpa biaya (gratis). Ada Firebase Cloud Messaging, aplikasi bisa memberitahu aplikasi pengguna ketika ada email baru atau pemberitahuan dan notifikasi. Ini cukup bagus untuk meningkatkan retensi pengguna dan mendorong pengguna aplikasi agar kembali mengakses aplikasi. FCM juga menyediakan konsol notifikasi dengan alamat: <https://firebase.google.com/docs/notifications> yang dapat digunakan untuk mengirimkan pesan ke aplikasi pengguna. Gambar di bawah ini menunjukkan bagaimana arsitektur dari FCM ini.



**Gambar 2.15 Firebase Cloud Messaging**

Pesan notifikasi bisa dikirimkan ke pengguna melalui Notifications Console GUI dan HTTP request seperti terlihat pada gambar di atas.

## 2.10UML

Unified Modeling Language (UML) adalah bahasa pemodelan visual yang digunakan untuk menspesifikasikan, memvisualisasikan, membangun, dan mendokumentasikan rancangan dari suatu sistem perangkat lunak. Pemodelan memberikan gambaran yang jelas mengenai sistem yang akan dibangun baik dari sisi struktural ataupun fungsional. UML dapat diterapkan pada semua metode pengembangan, tingkatan siklus sistem, dan berbagai macam domain aplikasi. Dalam UML terdapat konsep semantik, notasi, dan panduan masing-masing diagram. UML juga memiliki bagian statis, dinamis, ruang lingkup, dan organisasional. UML bertujuan menyatukan teknik-teknik pemodelan berorientasi objek menjadi terstandarisasi. [7]

### **2.10.1 Class Diagram**

Diagram kelas atau class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi.

1. Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas
2. Operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas.

Kelas-kelas yang ada pada struktur sistem harus dapat melakukan fungsi-fungsi sesuai dengan kebutuhan sistem sehingga pembuat perangkat lunak dapat membuat kelas-kelas di dalam program perangkat lunak sesuai dengan perancangan diagram kelas. Susunan struktur kelas yang baik pada diagram kelas sebaiknya memiliki jenis-jenis kelas sebagai berikut:

1. Kelas main yaitu kelas yang memiliki fungsi awal dieksekusi ketika sistem dijalankan.
2. Kelas yang menangani tampilan sistem (view) yaitu kelas yang mendefinisikan dan mengatur tampilan ke pemakai.
3. Controller yaitu kelas yang menangani fungsi-fungsi yang harus ada diambil dari pendefinisian use case, kelas ini biasa disebut kelas proses.
4. Model yaitu kelas yang digunakan untuk memegang atau membungkus data menjadi sebuah kesatuan yang diambil maupun akan disimpan ke basis data.

### **2.10.2 Object Diagram**

Diagram objek menggambarkan struktur sistem dari segi penamaan objek dan jalannya objek dalam sistem. Pada diagram objek harus dipastikan semua kelas sudah didefinisikan pada diagram kelas harus dipakai objeknya, karena jika tidak, pendefinisian kelas itu tidak dapat dipertanggungjawabkan.

### **2.10.3 Use Case Diagram**

Diagram use case merupakan pemodelan untuk kelakuan (behavior) sistem informasi yang akan dibuat. Use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, use case

digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu.

Ada dua hal utama pada use case yaitu pendefinisian apa yang disebut aktor dan use case.

1. Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu orang.
2. Use case merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.

#### **2.10.4 Activity Diagram**

Diagram aktivitas atau activity diagram menggambarkan workflow (aliran kerja) atau aktifitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram aktifitas menggambarkan aktifitas bukan apa yang dilakukan aktor.

Diagram aktifitas juga banyak digunakan untuk mendefinisikan hal-hak berikut:

1. Rancangan proses bisnis dimana setiap urutan aktifitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan.
2. Urutan atau pengelompokan tampilan dari sistem / user interface dimana setiap aktifitas dianggap memiliki sebuah rancangan antarmuka tampilan.
3. Rancangan pengujian dimana setiap aktifitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya
4. Rancangan menu yang ditampilkan pada diagram aktifitas

#### **2.10.5 Sequence Diagram**

Sequence diagram menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup objek dan pesan yang dikirim dan diterima antar objek. Oleh karena itu untuk menggambarkan diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah use case beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu