BAB 2
TINJAUAN PUSTAKA

Bagian ini berisi tentang beberapa teori-teori yang mendukung dalam proses analisis dan implementasi berdasarkan masalah yang diangkat dalam pembangunan Aplikasi Pinjamin Untuk Penyewaan Barang Secara Aman Berbasis On Demand Menggunakan Graph API Facebook Dan Sensor GPS Smartphone Android.

2.1 Pengertian Sewa – Menyewa

Menurut pasal 1548 KUHPerdata, Sewa Menyewa ialah suatu perjanjian, dengan mana pihak yang satu mengikatkan dirinya untuk memberikan kepada pihak yang lainnya kenikmatan dari sesuatu barang, selama suatu waktu tertentu dengan pembayaran sesuatu harga, yang oleh pihak tersebut belakangan itu disanggupi pembayarannya[3].

Dari ketentuan Pasal 1548 KUHPerdata tersebut, nampak bahwa dalam Perjanjian Sewa Menyewa terdapat beberapa unsur penting yaitu:

1. Perjanjian

Dalam Pasal 1313 KUHPerdata, secara tegas diatur bahwa Perjanjian adalah suatu perbuatan dengan mana satu orang atau lebih mengikatkan dirinya terhadap satu orang lain atau lebih. Rumusan dalam Pasal 1313 KUHPerdata tersebut mengatur bahwa Perjanjian itu harus memenuhi unsur-unsur:

a. Suatu Perbuatan : Perbuatan harus diartikan sebagai perbuatan hukum, dimana perbuatan tersebut menimbulkan akibat hukum
b. Antara sekurang-kurangnya 2 (dua) orang
c. Perbuatan tersebut melahirkan perikatan di antara pihak-pihak yang berjanji tersebut

Perjanjian dapat dikatakan sah dan mengikat harus memenuhi 4 (empat) syarat mutlak, hal ini sebagaimana ditentukan dalam Pasal 1320 KUHPerdata, yaitu:

a. Kesepakatan para pihak,

b. Kecakapan untuk membuat suatu Perjanjian,
c. Suatu hal tertentu,
d. Suatu sebab yang halal.

Dan sebagaimana ketentuan dalam Pasal 1338 KUHPerdata, setiap Perjanjian yang dibuat secara sah akan berlaku sebagai Undang-undang bagi mereka yang membuatnya.

2. Kewajiban Para Pihak

Dimana pihak yang menyewakan (pemilik) memiliki kewajiban untuk menyerahkan barangnya untuk dinikmati oleh pihak penyewa, sedangkan pihak penyewa memiliki kewajiban untuk melakukan pembayaran harga sewa. Jadi barang yang disewakan tidak untuk dimiliki seperti halnya dalam jual beli, melainkan hanya untuk dipakai dan dinikmati kegunaannya.

3. Jangka Waktu

Berdasarkan rumusan dari Pasal 1548 KUHPerdata dikatakan bahwa sewa menyewa itu berlangsung selama waktu tertentu, yang berarti bahwa dalam Perjanjian Sewa Menyewa harus selalu ditentukan jangka waktu tertentu, tetapi dalam Perjanjian Sewa Menyewa itu dapat juga tidak ditetapkan suatu jangka waktu tertentu asalkan sudah disetujui harga sewa satu bulan dan lain-lain. Penetuan jangka waktu ini dimaksudkan untuk membatasi pemberian kenikmatan kepada Penyewa.

2.2 Landasan Teori

Landasan teori menjelaskan beberapa definisi dan teori yang berkaitan dengan pembangunan aplikasi sebagai dasar pemahaman dalam sebuah sistem serta metode yang digunakan untuk kegiatan pembangunan aplikasi tersebut.

2.2.1 Global Positioning System (GPS)

GPS (*Global Positioning System*) adalah sistem navigasi yang menggunakan satelit yang didesain agar dapat menyediakan posisi secara instan, kecepatan dan informasi waktu di hampir semua tempat di muka bumi, setiap saat dan dalam kondisi cuaca apapun[4].

Sedangkan alat untuk menerima sinyal satelit yang dapat digunakan oleh pengguna secara umum dinamakan *GPS Tracker* atau *GPS Tracking*, dengan menggunakan alat ini maka dimungkinkan user dapat melacak posisi kendaraan, armada ataupun mobil dalam keadaan *Real-Time*[4].

2.2.1.1 Cara Kerja GPS

Sistem kerja GPS adalah dengan menstransmisikan sinyal dari satelit ke perangkat GPS (portable GPS murni, ataupun smartphone yang sudah memiliki fitur GPS).GPS membutuhkan transmisi dari 3 satelit untuk mendapatkan informasi dua dimensi (lintang dan bujur), dan 4 satelit untuk tiga dimensi (lintang, bujur dan ketinggian). Karena GPS bekerja mengandalkan satelit, maka penggunaannya disarankan di tempat terbuka. Penggunaan di dalam ruangan, atau di tempat yang menghalangi arah satelit (di angkasa), maka GPS tidak akan bekerja secara akurat dan maksimal[4].

Teknologi GPS pertama kali digunakan oleh United States Departement of Defense (DOD) untuk kebutuhan militer. Sistem GPS mulai digunakan sejak tahun 1980, namun pemakaian secara umum oleh publik baru sekira tahun 1990-an[4].

2.2.2 Location Based Service (LBS)

Location Based Service (LBS) atau layanan berbasis lokasi adalah sebuah layanan informasi yang dapat diakses dengan perangkat bergerak melalui jaringan dan mampu menampilkan posisi secara geografis keberadaan perangkat bergerak tersebut. *Location Based Service* dapat berfungsi sebagai layanan untuk mengidentifikasi lokasi dari seseorang atau suatu objek tertentu, seperti menemukan lokasi tempat yang diinginkan terdekat atau lokasi lainnya[7].

Hal paling penting dari *Location Based Service* (LBS) dapat bekerja sesuai yang diinginkan oleh pengembang aplikasi Android. Android pun memungkinkan pengembang menentukan metode pencarian lokasi yang dibutuhkan dan juga dapat mengatur kebutuhan daya, biaya dan akurasi berdasarkan spesifik yang akan dibuat untuk aplikasi tersebut [7].

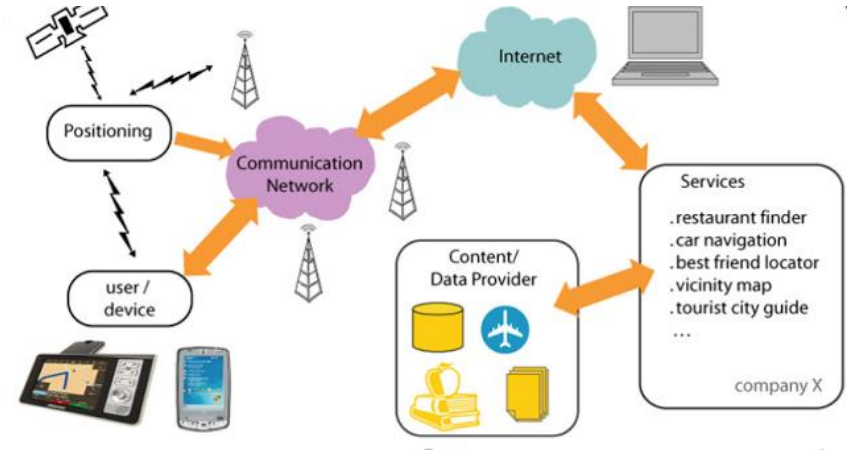
2.2.2.1 Cara Kerja Location Based Service

Untuk menggambarkan cara kerja LBS, aplikasi LBS akan mencarikan informasi mengenai lokasi di rute pengguna[7].

1. Fungsi pencarian telah diaktifkan, posisi pengguna sebenarnya dari perangkat mobile diperoleh dari *Positioning Service*. Hal ini dapat dilakukan baik oleh perangkat menggunakan *Global Positioning System* (GPS) sendiri atau layanan posisi jaringan yang berasal dari provider (Cell Tower). Setelah itu perangkat mobile pengguna mengirimkan permintaan informasi, yang berisi tujuan untuk mencari dan mengirimkan posisi melalui jaringan komunikasi ke gateway telekomunikasi.
2. Gateway memiliki tugas untuk bertukar pesan di antara jaringan komunikasi selular dan internet. Oleh karena itu dia mengetahui alamat web dari beberapa aplikasi server dan rute permintaan ke spesifik server tertentu. Gateway akan menyimpan juga informasi tentang perangkat mobile yang telah meminta informasi
3. Aplikasi server membaca permintaan dan mengaktifkan layanan yang terkait.
4. Kemudian, service menganalisis lagi pesan dan memutuskan mana informasi dan posisi pengguna diperlukan untuk menjawab permintaan pengguna.
5. Selanjutnya service akan menemukan informasi lokasi yang diinginkan.
6. Setelah sekarang semua informasi service akan melakukan buffer spasial dan query routing untuk mendapatkan beberapa lokasi terdekat. Setelah

itu hasil dikirim kembali ke pengguna melalui internet, gateway dan jaringan mobile.

7. Kemudian, informasi mengenai lokasi akan disampaikan kepada pengguna baik dalam bentuk peta digital.

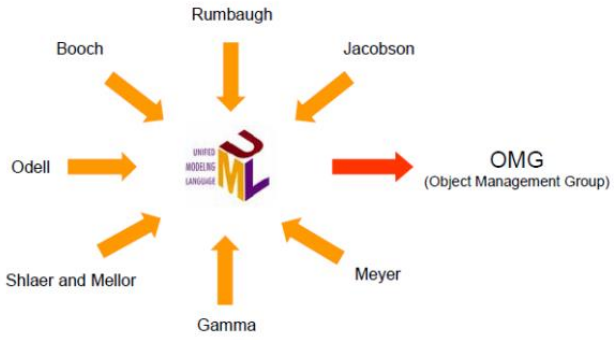


Gambar 2.1 Cara Kerja LBS

(sumber : T. Taryono, "Aplikasi Peta ATM Menggunakan Aplikasi GPS pada Handphone Android)[7]

2.2.3 Unified Modeling Language

Unified Modeling Language (UML) adalah termasuk kedalam rumpun jenis permodelan notasi grafis yang didukung oleh model – model tunggal. Permodelan ini berguna untuk membantu dalam menjelaskan dan merancang perangkat lunak yang dibangun dengan *object – oriented* (OO). UML merupakan standar terbuka yang dikelola oleh Open Management Group (OMG) yang berada dibawah naungan perusahaan – perusahaan konsorsium terbuka. UML merupakan suatu bahasa permodelan yang terdiri banyak model [5].



Gambar 2.2 Contoh Uml

(sumber : M. Fowler , *A brief guide to the standard object modeling language)*[5]

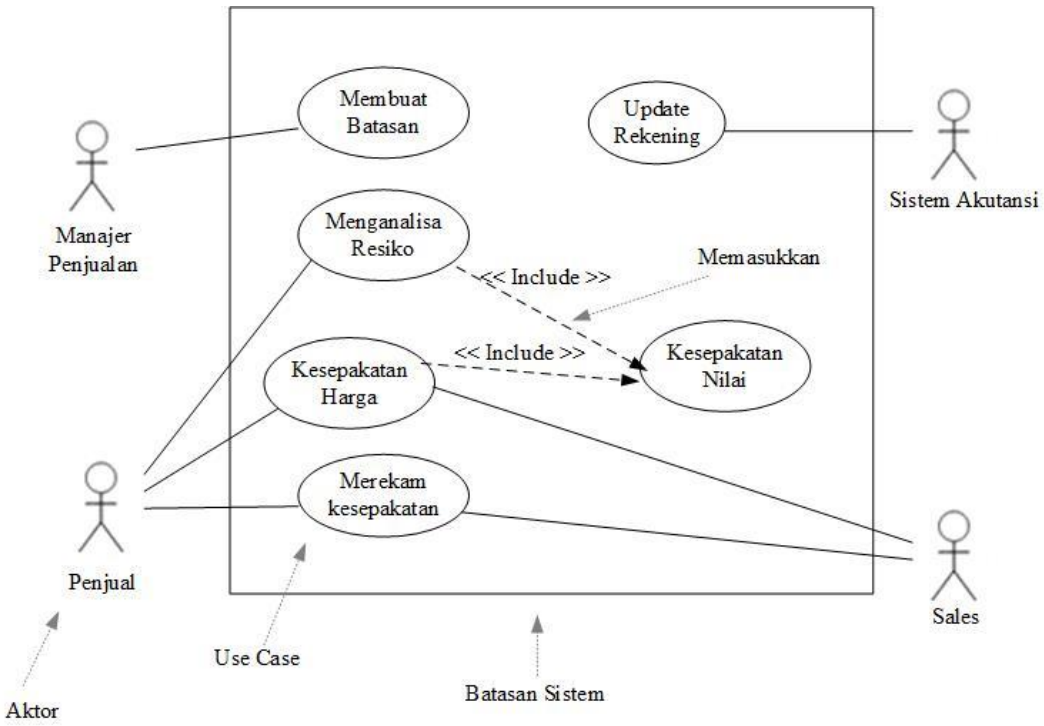
2.2.3.1 Use Case Diagram

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 [5].

Use case diagram menampilkan aktor, *use case*, dan hubungan antara mereka [5]:

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* :

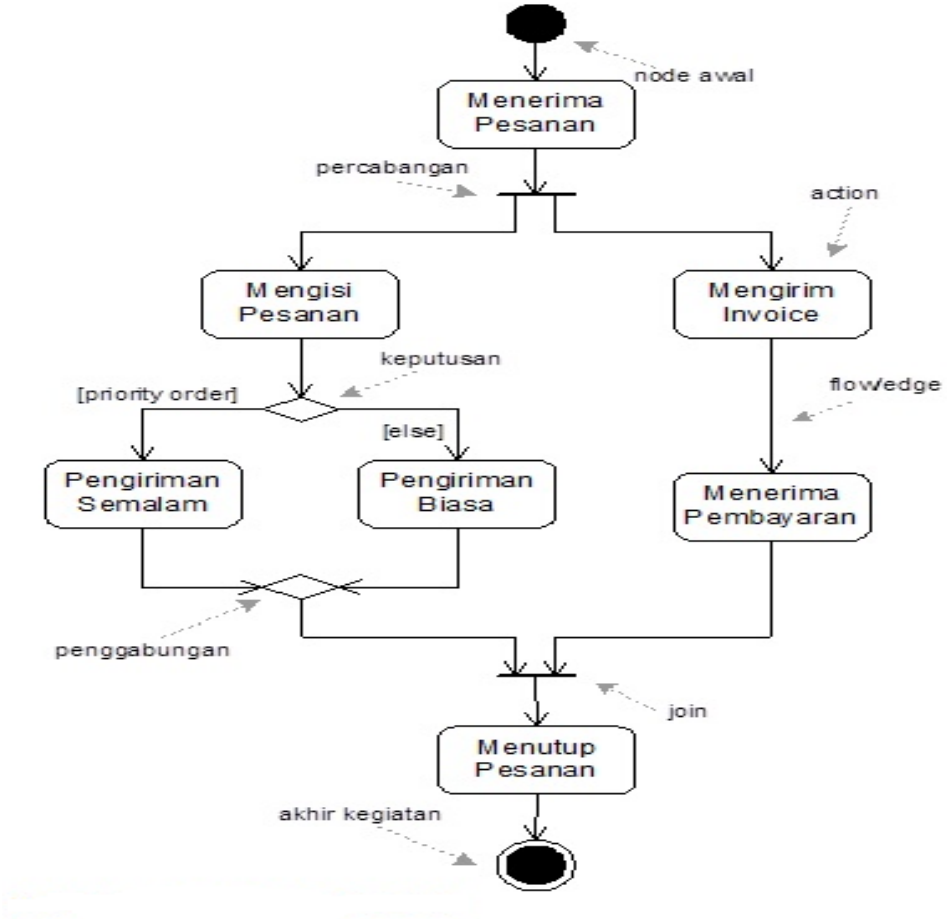


Gambar 2.3 Contoh diagram Use Case

[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 [5]

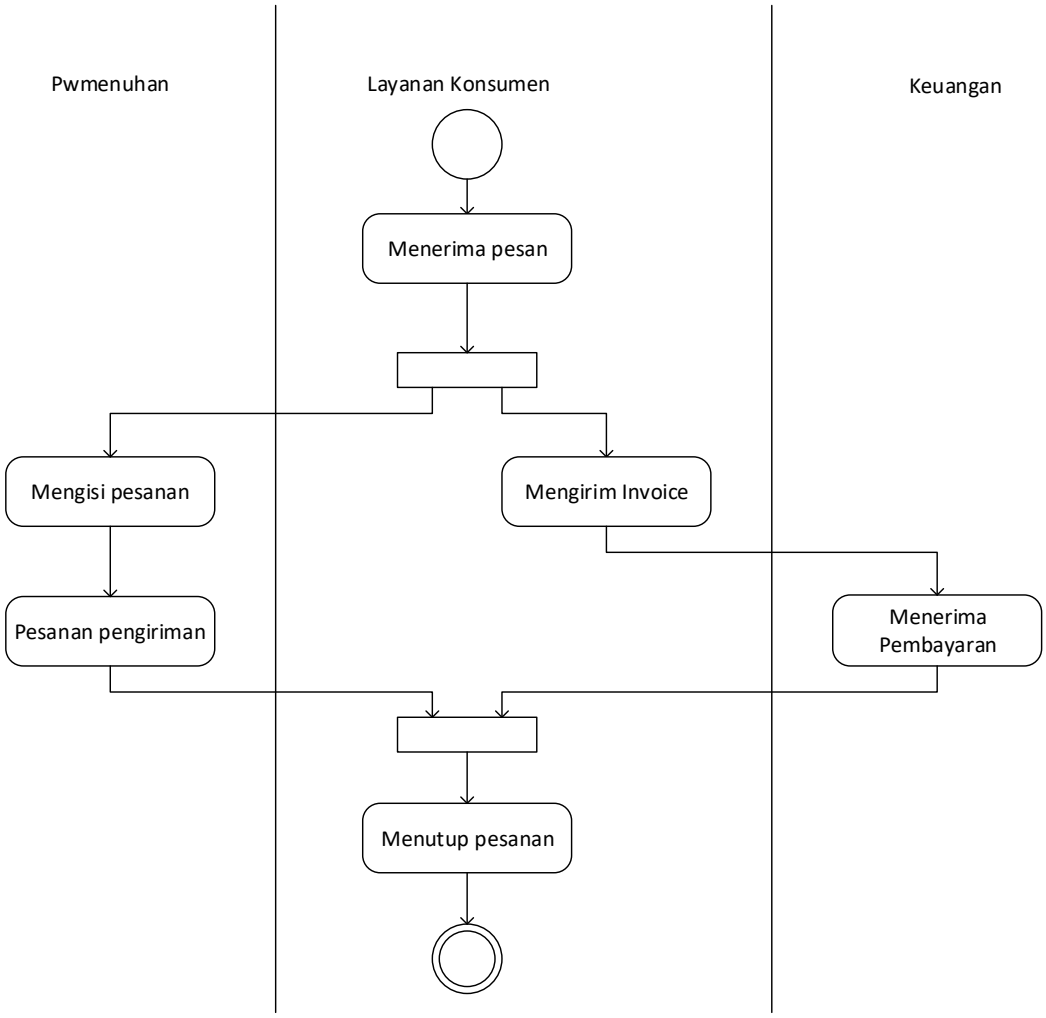
2.2.3.2 Activity Diagram

Diagram *activity* adalah teknik untuk menggambarkan logika prosedural, proses bisnis, dan jalur kerja. [5].



Gambar 2.4 Contoh Diagram Aktivitas

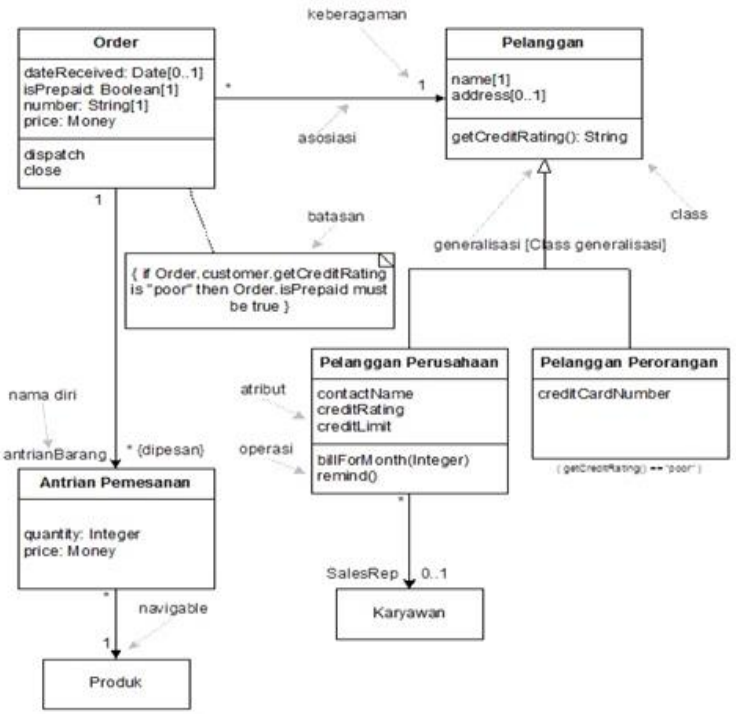
Activity diagram memberi tahu anda tentang apa yang terjadi, tetapi diagram ini tidak memberi tahu anda tentang siapa yang melakukan apa. Jika anda ingin menunjukkan siapa melakukan apa, anda dapat memisahkan activity diagram ke dalam partisi-pasrtisi yang menampilkan action mana yang dilakukan oleh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



Gambar 2.5 Partisi Pada Activity Diagram

2.2.3.3 Diagram Class

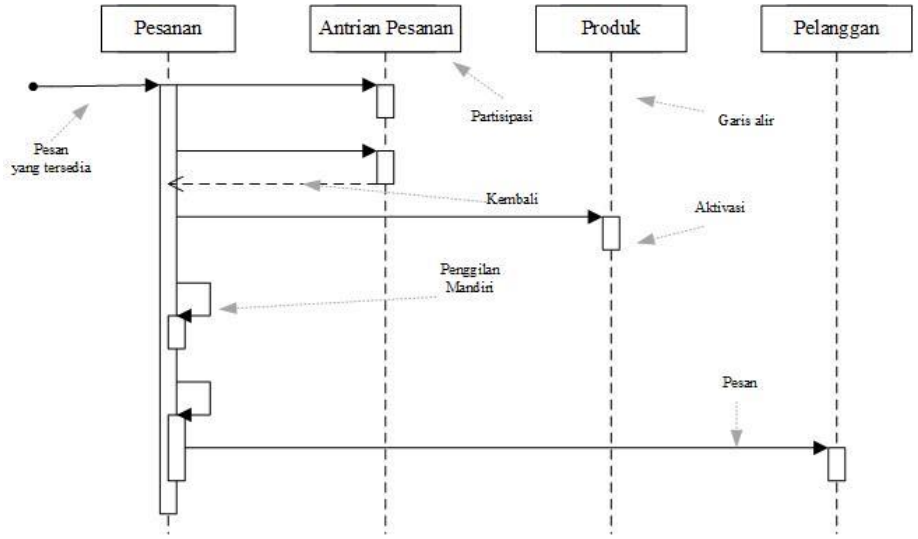
Diagram *class* mendeskripsikan jenis-jenis objek dalam sistem dan berbagai macam hubungan statis yang terdapat diantara mereka. Diagram *class* juga menunjukan properti dan operasi sebuah *class* dan batasan-batasan yang terdapat pada sebuah hubungan-hubungan objek. UML menggunakan istilah fitur sebagai istilah umum yang meliputi properti dan operasi sebuah *class.* Didalam class dibagi menjadi tiga bagian. Nama *class*, atributnya, dan operasinya. [5].



Gambar 2.6 Contoh Diagram Class

2.2.3.4 Sequence Diagram

Diagram *sequence* secara khusus menjabarkan *behavior* sebuah skenario tunggal. Diagram tersebut menunjukan 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 [5].



Gambar 2.7 Contoh Diagram Sequence

2.2.4 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 Pemprograman JavaScript, Standar ECMA-262 Edisi ke-3 - Desember 1999. JSON merupakan format teks yang tidak bergantung pada bahasa pemprograman apapun karena menggunakan gaya bahasa yang umum digunakan oleh programmer keluarga C termasuk C, C++, C#, Java, JavaScript, Perl, Python dll[8]. 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), atauassociative array[8].

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)[8].

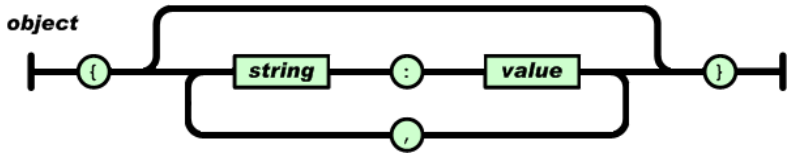
Struktur-struktur data ini disebut sebagai struktur data universal. Pada dasarnya, semua bahasa pemprograman moderen mendukung struktur data ini dalam bentuk yang sama maupun berlainan. Hal ini pantas disebut demikian karena format data mudah dipertukarkan dengan bahasa-bahasa pemprograman yang juga berdasarkan pada struktur data ini[8].

2.2.4.1 Bentuk JSON

JSON menggunakan bentuk sebagai berikut:

1. Objek

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) [8].

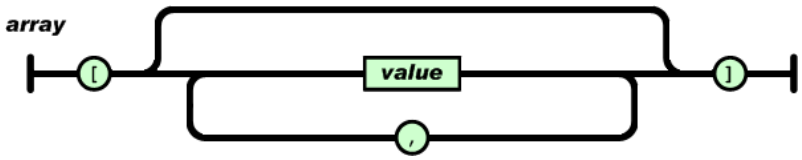


Gambar 2.8 Objek JSON

(sumber : G. Parikshit and H. Bhushan. JSON org)[8]

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) [8].

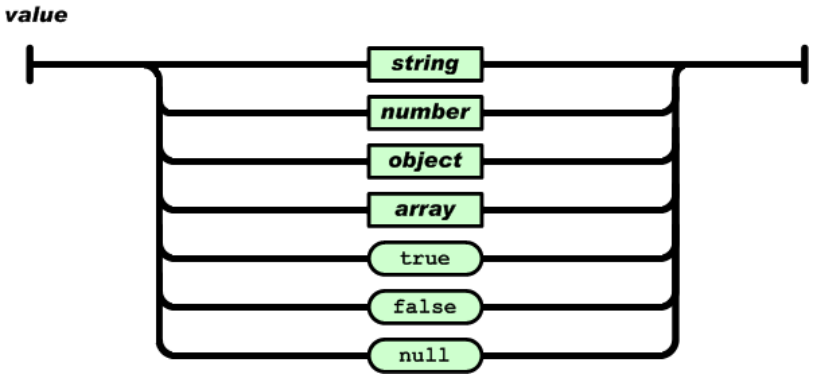


Gambar 2.9 Larik JSON

(sumber : G. Parikshit and H. Bhushan. JSON org)[8]

3. *Value*

Nilai (*value*) dapat berupa sebuah string dalam tanda kutip ganda, atau angka, atau true atau false atau null, atau sebuah objek atau sebuah larik. Struktur-struktur tersebut dapat disusun bertingkat [8].

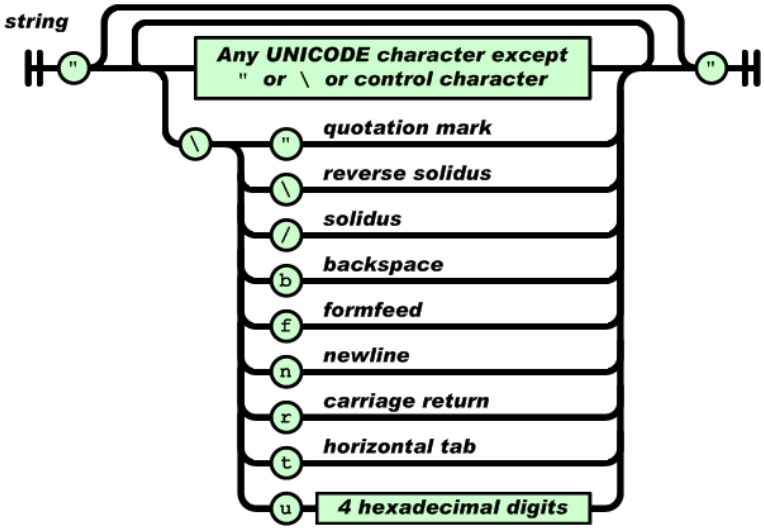


Gambar 2.10 Value JSON

(sumber : G. Parikshit and H. Bhushan. JSON org)[8]

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 [8].

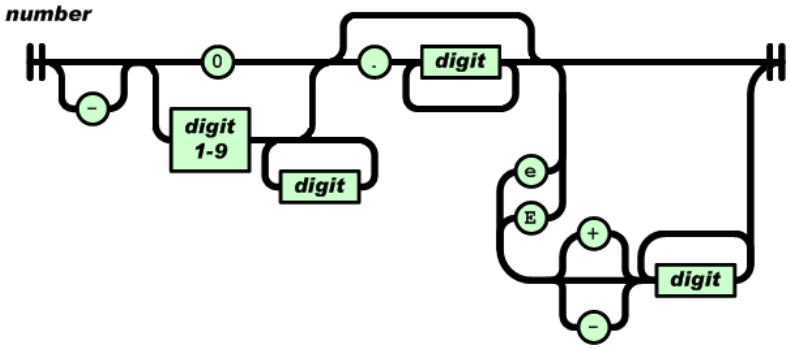


Gambar 2.11 String JSON

(sumber : G. Parikshit and H. Bhushan. JSON org)[8]

5. Angka

Angka adalah sangat mirip dengan angka di C atau Java, kecuali format oktal dan heksadesimal tidak digunakan [8].



Gambar 2.12 Angka dalam JSON

(sumber : G. Parikshit and H. Bhushan. JSON org)[8]

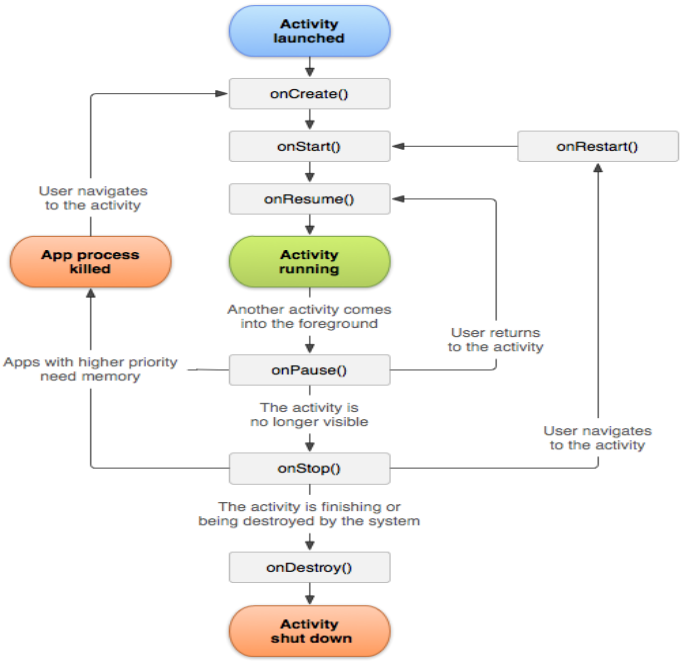
2.2.5 Android

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 [9].

2.2.6.1 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 [10].



Gambar 2.13 Android Life Cycle

(Sumber : https://developer.android.com/guide/components/images/activity_lifecycle.png)[10]

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[10]:

1. Resumed

Resumed terjadi ketika aplikasi berjalan setelah state *paused* . State ini akan menjalankan perintah program yang ditulis pada method onResume() [10].

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 [10].

3. Stopped

Dalam keadaan ini, aplikasi benar-benar tidak ditampilkan dan tidak terlihat oleh pengguna tetapi masih meninggalkan service dibackground [10].

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(), krmudian diikuti oleh onResume().

2.2.6 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) [11].

Ada tiga platform Java yang masing-masing diarahkan untuk tujuan tertentu dan untuk lingkungan komputasi yang berbeda-beda [11]:

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.

Micro Edition (J2ME) : Didesain untuk piranti dengan memori terbatas, layar display terbatas dan power pemrosesan yang juga terbatas.

2.2.7 Web Service

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 [12].

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 detil pemrograman yang terdapat di dalamnya [12].

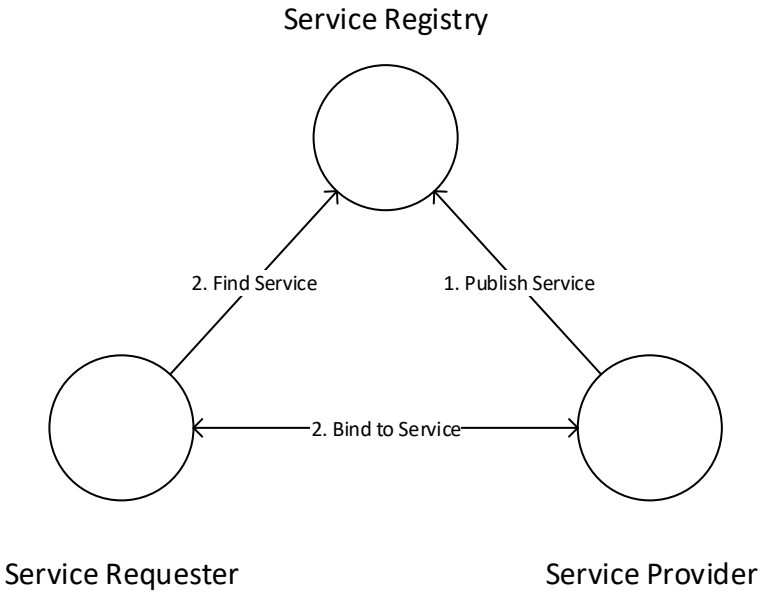
Beberapa alasan mengapa digunakannya web service adalah sebagai berikut:[12]

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 [12].
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 [12].
3. Web service berjalan di port 80 yang merupakan protokol standar HTTP, dengan demikian web service tidak memerlukan konfigurasi khusus di sisi firewall [12].

2.1.8.1. Arsitektur Web Service

Web service memiliki tiga entitas dalam arsitekturnya, yaitu: [12]

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



Gambar 2.14 Arsitektur Web Service

a. Service Provider berfungsi untuk menyediakan layanan/service dan mengolah sebuah registry agar layanan-layanan tersebut dapat tersedia [12].
b. Service Registry berfungsi sebagai lokasi central yang mendeskripsikan semua layanan/service yang telah di-register [12].
c. Service Requestor meminta layanan yang mencari dan menemukan layanan yang dibutuhkan serta menggunakan layanan tersebut [12].

2.2.8 Processor Hypertext Protocol (PHP)

PHP merupakan salah satu pemograman server-side di antara beberapa pemograman 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 pemograman ini cepat di kenal di kalangan luas [13].

Tanpa mengurangi kemampuan-kemampuan yang sudah ada, PHP 5 hadir dengan fitur lebih kompleks , dan merupakan rilis terbaru yang di keluarkan 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 [13].

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 [13].

2.2.9 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 [14].

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 [14].

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 [14].

2.2.10 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 [15].

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 [15].

2.2.11 Uclassify API

uClassify adalah alat yang terbuka untuk memungkinkan pengguna untuk membuat dan berbagi kelompok. TODO uClassify ini menyediakan tiga api (URL, XML dan Server lokal), berikut ini adalah ikhtisar pendek dan perbandingan dari mereka[16].

2.2.11.1. URL API

URL API sederhana untuk menggunakan, hanya mengirim permintaan GET dengan URL untuk mendapatkan respon.

Misalnya membuat klasifikasi sentimen (mendaftar untuk kunci api Baca gratis) [16]:
http://uclassify.com/browse/uClassify/Sentiment/ClassifyText?readkey=YOUR_READ_API_KEY_HERE&text=I+Love+School&Version=1.01

Tanggapan akan dalam XML atau alternatif JSON. Anda dapat mengelompokkan teks atau membiarkan uClassify download sebuah URL untuk Anda dan mengklasifikasikan konten (opsional dengan atau tanpa HTML).

Ketika menggunakan URL API [16]:

1. Ingin bangun dan berjalan cepat
2. Rendah volume klasifikasi (bisa lambat pada banyak permintaan sekaligus)
3. Setiap teks hanya dijalankan melalui satu classifier (jika tidak Anda harus batch dengan XML API)
4. Jika Anda hanya tertarik pada mengklasifikasikan dan tidak pelatihan atau fitur mewah lainnya.

2.2.11.2. XML API

Dengan XML API dapat melakukan semuanya. Anda dapat membuat kelompok, melatih mereka, mengklasifikasikan, ekstrak kata kunci dan banyak lagi. Fitur penting lainnya adalah bahwa Anda dapat batch panggilan. Ini berarti bahwa Anda bisa mendapatkan ribuan, mengatakan, tweet atau posting blog, diklasifikasikan dalam satu satu permintaan selama sebagai string permintaan kurang dari 1 MB[16].

Untuk membuat permintaan XML, Anda hanya membangun string XML dan posting untuk:
http://api.uclassify.com/ pada port 80.
Berikut adalah bagaimana Anda batch dua panggilan ke satu permintaan:

<? Versi XML = "1.0" pengkodean = "utf-8"? >
<uclassify xmlns="http://api.uclassify.com/1/RequestSchema" version="1.01">
<texts>
<textBase64 id="UnknownFantasyText1">
SXcgYklRdHFEYXEgYklsZW5namFq</textBase64>
<textBase64 id="UnknownFantasyText2">
UGFkb2wgcmFpZCwgYXRoYW4gaGVuZGFk</textBase64>
</texts>
<readCalls readAPIKey="YOUR_READ_API_KEY_HERE">
<classify id="Classify1" classifierName="FantasyLanguage"
textId="UnknownFantasyText1"/>
<classify id="Classify2" classifierName="FantasyLanguage"
textId="UnknownFantasyText2"/>
</readCalls>
</uclassify>

Ketika menggunakan XML API

1. Mendapatkan kinerja dengan batching beberapa panggilan ke satu permintaan

2. Membutuhkan akses ke API penuh (classifier penciptaan, pelatihan, kata kunci dll)
3. Ini adalah cara yang direkomendasikan

2.2.11.3. Local Server

Sebagai pilihan terakhir Anda dapat menjalankan server lokal klasifikasi. Ini adalah untuk pengguna yang memiliki sejumlah besar data untuk memproses. API ini cukup banyak yang sama seperti XML API kecuali bahwa otentikasi tidak diperlukan[16].

Ketika menggunakan server lokal klasifikasi

1. Anda memiliki sejumlah besar data untuk memproses
2. Kinerja CPU sangat penting (Anda tidak perlu berbagi siklus CPU dengan pengguna lain uClassify)
3. Menginstal server di pusat data Anda sendiri, untuk menghindari setiap lag jaringan uclassify.com
4. Memiliki kontrol penuh atas server

2.2.12 Metode Trust-Based Cooperative System

Metode *trust-based cooperative system* adalah metode penilaian reputasi yang memanfaatkan aktifitas public penggunaan sosial media yang nantinya dihitung dengan menggunakan *Reputation Algorithm* yang terdapat dalam metode ini[2].

2.2.12.1. Algoritma Reputasi

Algoritma Reputasi adalah fitur utama pada metode ini, Reputation Algorthm adalah algoritma yang digunakan untuk menghitung aktifitas pengguna disosial dan aktifitas selama menggunakan aplikasi yang nantinya dihitung dan digunakan sebagai penilaian reputasi untuk kepercayaan dalam menggunakan aplikasi. Dan untuk maksimal dan minimum penilaian reputasi ini adalah 0 dan 1. Berikut ini adalah cara menghitung penilaian reputasi menggunakan metode trust-based cooperative system[2].

1. Menghitung Derajat Pertemanan.

Derajat Pertemanan adalah tingkat keakraban sesama pengguna disosial media. Berikut adalah cara menghitung derajat pertemanan [2] :

$$df_{ij} = \begin{cases} \frac{\sum(FA_{ij} \cdot wA_{ij})}{MFC_i}, & e_{ij} \in E, \\ 0, & otherwise, \end{cases}$$

Dimana :

- FA_{ij} disebut tindakan persahabatan, dan mungkin komentar, seperti, atau tindakan lain dari pengguna i pada pengguna j pada jaringan sosial.
- wA_{ij} adalah berat FA_{ij} tindakan dan mewakili kepentingan atas tindakan lain. Nilai adalah antara 0 dan 1. Sebagai contoh, Facebook, komentar dianggap lebih penting daripada seperti dalam proposal. Khususnya, di dalam proposal, berat komentar adalah 0.727, sementara berat seperti like 0.273, menurut nilai-nilai yang Diperoleh dari data yang ditampilkan dalam survei. Sistem dapat mendeteksi komentar positif, negatif dan netral yang menggunakan mesin belajar algoritma.
- MFC_i disebut koefisien persahabatan maksimal, dan mewakili jumlah maksimum tertimbang tindakan dilakukan oleh pengguna i. Nilainya adalah : $Max_j(\sum(FAij \, . \, wAij))$.

2. Menghitung Reputasi.

Berikut ini adalah tatacara untuk menghitung reputasi seseorang, yang datanya didapat dari aktivitas selama menggunakan sosial media dan aktivitas selama menggunakan aplikasi seperti pemberian ratting[2].

$$TR_{ij} = \begin{cases} DT_{ij}, & \text{if users } i \text{ and } j \text{ are direct friends }, \\ IT_{ij}, & \text{if some friendship chain exists from user } i \text{ to user } j, \\ NT_{ij}, & \text{if no friendship chain exists from user } i \text{ to user } j, \end{cases}$$

Dimana :

- Jika pengguna i dan j adalah langsung teman, langsung percaya (DT) adalah diperoleh dengan menggunakan persamaan.

$Dtij = dfij \cdot wF + ari \cdot wR.$

- Jika beberapa jaringan teman-teman (“Bukan Teman Langsung”) dari pengguna i untuk pengguna j ada, tidak langsung percaya (TI) yang diberikan oleh persamaan

$IT_{ij} = F\ (i \rightarrow j) \cdot wF + ar_i \cdot wR.$

- Jika tidak ada jaringan persahabatan dari user i ke pengguna j, persamaan diterapkan.

$NT_{ij} = ar_i \cdot wR,$

Dimana :

- wF disebut berat persahabatan, dan nilainya (0.625) diperoleh dengan kebahagiaan rata-rata orang dengan teman-teman mereka dalam jaringan sosial Facebook ;
- wR adalah berat Rating, yang nilainya adalah diperoleh dengan menggunakan 1 − wF, sehingga 0.375;
- F(i→j) memberikan tingkat persahabatan dari pengguna i untuk pengguna j diperoleh melalui jaringan persahabatan {saya, i1, i2,..., ic, j} antara keduanya. Nilai ini diperoleh mempertimbangkan kemungkinan peristiwa independen. $F\ (i \rightarrow j) = dfii1 \cdot dfi1i2 \cdots dficj$. Nilai ini selalu antara 0 dan 1, dan menurunkan bersama dengan panjang rantai persahabatan.