BAB 2
TINJAUAN PUSTAKA

2.1 Sejarah Kopi

Kopi adalah minuman hasil seduhan biji kopi yang telah disangrai dan dihaluskan menjadi bubuk. Kopi merupakan salah satu komoditas di dunia yang dibudidayakan lebih dari 50 negara. Dua varietas pohon kopi yang dikenal secara umum yaitu Kopi Robusta (Coffea canephora) dan Kopi Arabika (Coffea arabica).

Pemrosesan kopi sebelum dapat diminum melalui proses panjang yaitu dari pemanenan biji kopi yang telah matang baik dengan cara mesin maupun dengan tangan kemudian dilakukan pemrosesan biji kopi dan pengeringan sebelum menjadi kopi gelondong. Proses selanjutnya yaitu penyangraian dengan tingkat derajat yang bervariasi. Setelah penyangraian biji kopi digiling atau dihaluskan menjadi bubuk kopi sebelum kopi dapat diminum.

Sejarah mencatat bahwa penemuan kopi sebagai minuman berkhasiat dan berenergi pertama kali ditemukan oleh Bangsa Etiopia di benua Afrika sekitar 3000 tahun (1000 SM) yang lalu. Kopi kemudian terus berkembang hingga saat ini menjadi salah satu minuman paling populer di dunia yang dikonsumsi oleh berbagai kalangan masyarakat. Indonesia sendiri telah mampu memproduksi lebih dari 400 ribu ton kopi per tahunnya. Di samping rasa dan aromanya yang menarik, kopi juga dapat menurunkan risiko terkena penyakit kanker, diabetes, batu empedu, dan berbagai penyakit jantung (kardiovaskuler).

2.1.1 Jenis-Jenis Kopi

Minuman kopi yang ada saat ini sangatlah beragam jenisnya.Masing-masing jenis kopi yang ada memiliki proses penyajian dan pengolahan yang unik. Berikut ini adalah beberapa contoh minuman kopi yang umum dijumpai:

1. Espresso, merupakan kopi yang dibuat dengan mengekstraksi biji kopi menggunakan uap panas pada tekanan tinggi.

2. Latte (*coffee latte*), merupakan sejenis kopi espresso yang ditambahkan susu dengan rasio antara susu dan kopi 3:1.
3. *Café au lait*, serupa dengan *caffe latte* tetapi menggunakan campuran kopi hitam.
4. Caffè macchiato, merupakan kopi espresso yang ditambahkan susu dengan rasio antara kopi dan susu 4:1.
5. Cappuccino, merupakan kopi dengan penambahan susu, krim, dan serpihan cokelat.
6. *Frappé*, merupakan espresso yang disajikan dingin.
7. Kopi instan, berasal dari biji kopi yang dikeringkan dan digranulasi.
8. Kopi Irlandia (*irish coffee*), merupakan kopi yang dicampur dengan wiski.
9. Kopi tubruk, kopi asli Indonesia yang dibuat dengan memasak biji kopi bersama dengan gula.
10. Kopi moka, serupa dengan cappuccino dan latte, tetapi dengan penambahan sirup cokelat.
11. Kopi hitam, merupakan hasil ektraksi langsung dari perebusan biji kopi yang disajikan tanpa penambahan perisa apapun.
12. Oleng, kopi khas Thailand yang dimasak dengan jagung, kacang kedelai, dan wijen.

2.1.2 Pengertian Kafein

Kafein adalah zat alkaloid yang di temukan pada berbagai jenis tanaman terutama tanaman kopi, cola, teh dan lain sebagainya. Kafein berfungsi sebagai zat stimulan untuk sistem syaraf pusat, zat perangsang serta dapat menangkal katuk dan mengembalikan kewaspadaan. Tetapi efek sampingnya, kafein memiliki potensi menyebabkan **penyimpangan kromoson** yang *bersifat mutagenik*.

Pada awalnya, kafein di gunakan dalam bidang pengobatan, baik obat manusia maupun sebagai pestisida alami beberapa jenis serangga. Kafein termasuk dalam golongan zat psikoaktif karena dapat merangsang sistem syaraf pusat. Kemampuan kafein sebagai obat perangsang dan stimulan sering kali di gunakan sebagai obat

penahan kantuk sementara yang sangat laku di dunia. Kafein biasanya di tambahkan dalam berbagai jenis minuman berkarbonasi (minuman ringan) dan minuman energi. Meski tergolong dalam zat yang memiliki sifat psikoaktif, kafein dapat di konsumsi secara legal dan penggunaanya tidak di batasi oleh hukum.

Kafein telah di kenal sejak zaman batu. Di Cina zat ini pertama kali di temukan dalam daun teh. Karena pada masa itu, kopi masih belum menyebar ke seluruh pelosok dunia seperti sekarang. Kopi sebagai sumber terbesar zat kafein merupakan tanaman khas daerah Ethiopia yang kemudian populer di Persia hingga kemudian tersohor di seantero dunia. Orang mengkonsumsi kopi pada awalnya adalah agar tetap bisa terjaga sepanjang waktu. Tanpa mengalami disorientasi. Untuk beberapa waktu mungkin kopi dapat membantu. Tetapi otak juga butuh istirahat, terlalu sering tidak tidur akan menyebabkan otak lelah sehingga kinerjanya tidak lagi efektif.

Penelitian modern menyatakan bahwa meski kafein dapat menyebabkan kecanduan, namun zat ini memiliki manfaat dan khasiat yang sangat baik untuk kesehatan. Manfaat dan khasiat kafein diantaranya adalah:

1. Mencegah kantuk
2. Merangsang sistem syaraf agar selalu awas
3. Mengurangi sakit kepala seperti vertigo
4. Mempertajam daya ingat
5. Melancarkan buang air kecil (bersifat diuretik)
6. Mencegah alzeimer dan parkinson (dalam taraf penelitian), dll.

Selain manfaat di atas, banyak orang masih khawatir tentang efek samping dari kafein. Karena beberapa penelitian melaporkan ahli mengatakan beberapa efek buruk dari kafein pada tubuh diantaranya adalah:

1. Penyakit Jantung
2. Hipertensi dan Darah Gangguan Kapal
3. Kehamilan

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 (*Global Positioning System*) adalah sistem navigasi yang berbasiskan satelit yang saling berhubungan yang berada di orbitnya. Satelit-satelit itu milik Departemen Pertahanan (*Departemen of Defense*) Amerika Serikat yang pertama kali diperkenalkan mulai tahun 1978 dan pada tahun 1994 sudah memakai 24 satelit. Untuk dapat mengetahui posisi seseorang maka diperlukan alat yang diberinama GPS *reciever* yang berfungsi untuk menerima sinyal yang dikirim dari satelit GPS. Posisi di ubah menjadi titik yang dikenal dengan nama *Way-point* nantinya akan berupa titik-titik koordinat lintang dan bujur dari posisi seseorang atau suatu lokasi kemudian di layar pada peta elektronik [4].

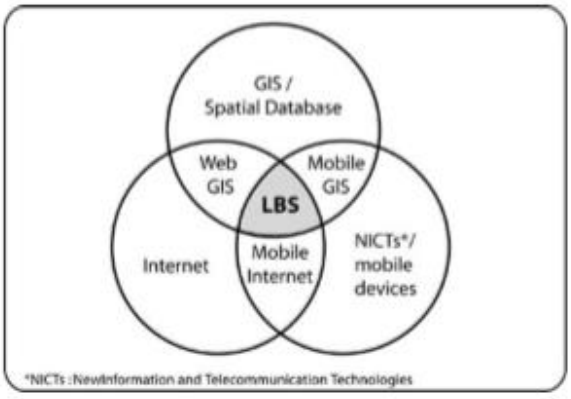
Sejak tahun 1980, layanan GPS yang dulunya hanya untuk leperluan militer mulai terbuka untuk publik. Uniknya, walau satelit-satelit tersebut berharga ratusan juta dolar, namun setiap orang dapat menggunakannya dengan gratis. Satelit-satelit ini mengorbit pada ketinggian sekitar 12.000 mil dari permukaan bumi. Posisi ini sangat ideal karena satelit dapat menjangkau area coverage yang lebih luas. Satelit-satelit ini akan selalu berada posisi yang bisa menjangkau semua area di atas permukaan bumi sehingga dapat meminimalkan terjadinya blank spot (area yang tidak terjangkau oleh satelit) [4].

Setiap satelit mampu mengelilingi bumi hanya dalam waktu 12 jam. Sangat cepat, sehingga mereka selalu bisa menjangkau dimana pun posisi Anda di atas permukaan bumi. GPS reciever sendiri berisi beberapa integrated circuit (IC) sehingga murah dan teknologinya mudah untuk di gunakan oleh semua orang. GPS dapat digunakan utnuk berbagai kepentingan, misalnya mobil, kapal, pesawat terbang, pertanian dan di integrasikan dengan komputer maupun laptop [4].

2.2.2 Location Based Service

Location Based Service adalah aplikasi yang bergantung pada lokasi tertentu dan didefinisikan pula sebagai layanan informasi dengan memanfaatkan teknologi yang mengetahui posisi sesuatu. Layanan berbasis lokasi menggunakan teknologi Positioning System, teknologi ini memungkinkan para pengguna dapat memperoleh informasi lokasi sesuai dengan kebutuhannya [5].

LBS termasuk dalam kategori teknologi yang sama dengan geographic information system (GIS), dan aplikasi global positioning system (GPS), yaitu dikenal dengan teknologi geospatial. Teknologi ini terdiri atas perangkat untuk mengumpulkan, menyimpan, menganalisa dan mendistribusikan data yang sesuai dengan kebutuhan pengguna tehadap sistem koordinat bumi. Layanan ini menjadi sangat penting bagi penggunanya karena mampu menghubungkan antara lokasi geographic informasi terhadap lokasi penggunanya, hal ini sangat mendukung era mobilitas seperti pada masa ini. Keberadaan aplikasi LBS merupakan hasil penggabungan dari tiga buah teknologi yaitu New Information and Communication Technologies (NICTS), internet dan GIS dengan menggunakan database spasial. Teknologi LBS ini terdiri atas perangkat-perangkat yang digunakan untuk mengumpulkan, menyimpan, menganalisa dan mendistribusikan data dan informasi pada berdasarkan sistem koordinat geographic bumi secara real-time. Identifikasi kordinat pengguna memungkinkan aplikasi LBS untuk menyediakan layanan bagi pengguna perangkat mobile [5].



Gambar 2.1 Location Based Service Sebagai Simpang Tiga Teknologi

Secara garis besar jenis layanan berbasis lokasi juga dapat dibagi menjadi dua, yaitu:

1. Pull Service, Layanan diberikan berdasarkan permintaan dari pelanggan akan kebutuhan suatu informasi. Jenis layanan ini dapat dianalogikan seperti mengakes suatu web pada jaringan internet [5].
2. Push Service, Layanan ini diberikan langsung oleh service provider tanpa menunggu permintaan dari pelanggan, tentu saja informasi yang diberikan tetap berkaitan dengan kebutuhan pelanggan [5].

2.2.3 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].

2.2.4 RESTful Web Service

EST (*REpresentational State Transfer*) merupakan standar arsitektur komunikasi berbasis web yang sering diterapkan dalam pengembangan layanan berbasis web. Umumnya menggunakan HTTP (*Hypertext Transfer Protocol*) sebagai protocol untuk komunikasi data. REST pertama kali diperkenalkan oleh Roy Fielding pada tahun 2000.

Pada arsitektur REST, REST server menyediakan *resources* (sumber daya/data) dan REST client mengakses dan menampilkan *resource* tersebut untuk penggunaan selanjutnya. Setiap *resource* diidentifikasi oleh URIs (*Universal*

*Resource Identifier*s) atau global ID. *Resource* tersebut direpresentasikan dalam bentuk format teks, JSON atau XML. Pada umumnya formatnya menggunakan JSON dan XML. Berikut adalah keuntungan dari REST :

1. Bahasa dan platform agnostic
2. Lebih sederhana/simpel untuk dikembangkan ketimbang SOAP
3. Mudah dipelajari, tidak bergantung pada tools
4. Ringkas, tidak membutuhkan layer pertukaran pesan (messaging) tambahan
5. Secara desain dan filosofi lebih dekat dengan web

Sedangkan untuk kelemahan dari REST adalah sebaga berikut ini :

1. Mengasumsi model point-to-point komunikasi - tidak dapat digunakan untuk lingkungan komputasi terdistribusi di mana pesan akan melalui satu atau lebih perantara
2. Kurangnya dukungan standar untuk keamanan, kebijakan, keandalan pesan, dll, sehingga layanan yang mempunyai persyaratan lebih canggih lebih sulit untuk dikembangkan ("dipecahkan sendiri")
3. Berkaitan dengan model transport HTTP

Berikut metode HTTP yang umum digunakan dalam arsitektur berbasis REST:

1. GET, menyediakan hanya akses baca pada *resource*
2. PUT, digunakan untuk menciptakan *resource* baru
3. DELETE**,** digunakan untuk menghapus *resource*
4. POST**,** digunakan untuk memperbarui *resource* yang ada atau membuat *resource* baru
5. OPTIONS**,** digunakan untuk mendapatkan operasi yang disupport pada *resource*

Web service adalah standar yang digunakan untuk melakukan pertukaran data antar aplikasi atau sistem, karena aplikasi yang melakukan pertukaran data bisa ditulis dengan bahasa pemrograman yang berbeda atau berjalan pada platform yang

berbeda. Contoh implementasi dari web service antara lain adalah SOAP dan REST. *Web service* yang berbasis arsitektur REST kemudian dikenal sebagai RESTful web services. Layanan web ini menggunakan metode HTTP untuk menerapkan konsep arsitektur REST.

2.2.4.1 Cara Kerja RESTful Web Service

Sebuah client mengirimkan sebuah data atau request melalui HTTP Request dan kemudian server merespon melalui HTTP Response. Komponen dari http request :

1. Verb, HTTP method yang digunakan misalnya GET, POST, DELETE, PUT dll.
2. *Uniform Resource Identifier* (URI) untuk mengidentifikasikan lokasi resource pada server.
3. HTTP Version, menunjukkan versi dari HTTP yang digunakan, contoh HTTP v1.1.
4. Request Header, berisi metadata untuk HTTP Request. Contoh, type client/browser, format yang didukung oleh client, format dari body pesan, seting cache dll.
5. Request Body, konten dari data.

Sedangkan komponen dari http respon**se** :

1. Status/Response Code, mengindikasikan status server terhadap resource yang direquest. misal : 404, artinya resource tidak ditemukan dan 200 response OK.
2. HTTP Version, menunjukkan versi dari HTTP yang digunakan, contoh HTTP v1.1.
3. Response Header, berisi metadata untuk HTTP Response. Contoh, type server, panjang content, tipe content, waktu response, dll
4. Response Body, konten dari data yang diberikan.

2.2.5 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 [7]. Oleh karena sifat-sifat tersebut, menjadikan JSON ideal sebagai bahasa pertukaran-data. JSON terbuat dari dua struktur:

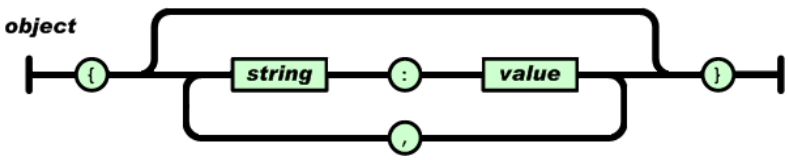
1. Kumpulan pasangan nama/nilai. Pada beberapa bahasa, hal ini dinyatakan sebagai objek (object), rekaman (record), struktur (struct), kamus (dictionary), tabel hash (hash table), daftar berkunci (keyed list), atauassociative array [7].
2. Daftar nilai terurutkan (an ordered list of values). Pada kebanyakan bahasa, hal ini dinyatakan sebagai larik (array), vektor (vector), daftar (list), atau urutan (sequence) [7].

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 [7].

2.2.5.1 Bentuk JSON

1. Objek

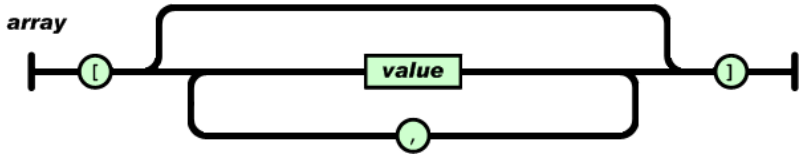
Objek adalah sepasang nama/nilai yang tidak terurutkan. Objek dimulai dengan { (kurung kurawal buka) dan diakhiri dengan } (kurung kurawal tutup). Setiap nama diikuti dengan : (titik dua) dan setiap pasangan nama/nilai dipisahkan oleh , (koma) [7].



Gambar 2.2 Objek JSON

2. Larik

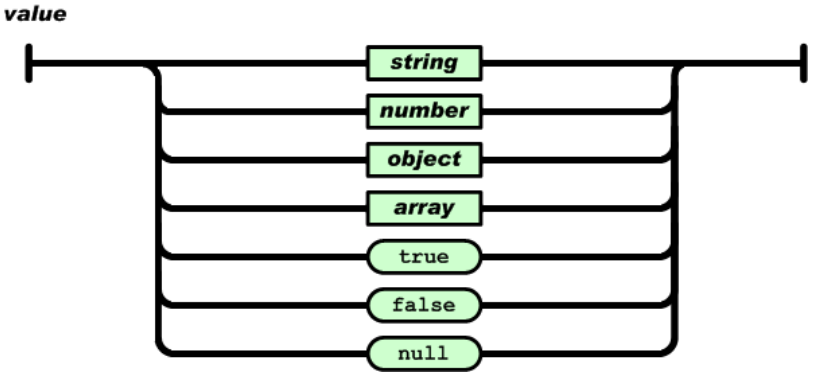
Larik adalah kumpulan nilai yang terurutkan. Larik dimulai dengan [(kurung kotak buka) dan diakhiri dengan] (kurung kotak tutup). Setiap nilai dipisahkan oleh , (koma) [7].



Gambar 2.3 Larik JSON

3. Value

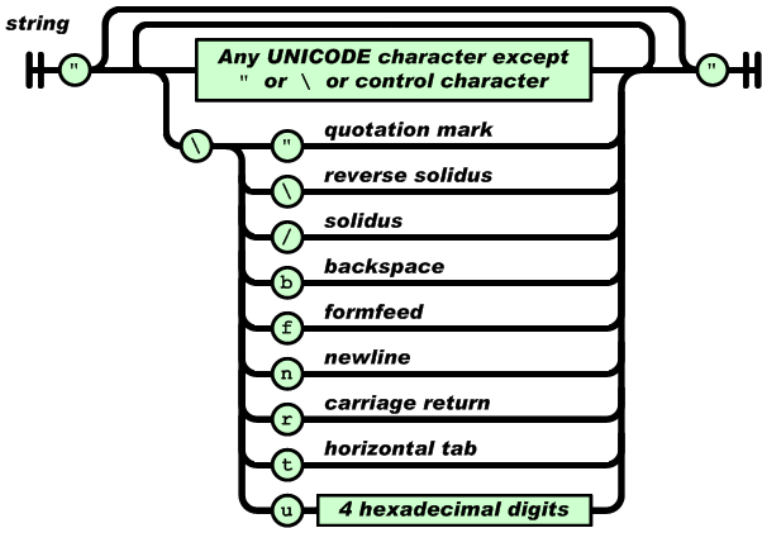
Nilai (*value*) dapat berupa sebuah string dalam tanda kutip ganda, atau angka, atau true atau false atau null, atau sebuah objek atau sebuah larik. Struktur-struktur tersebut dapat disusun bertingkat [7].



Gambar 2.4 Value JSON

4. String

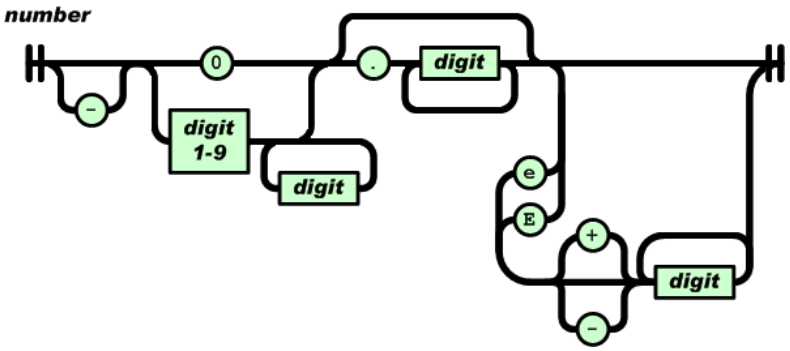
String adalah kumpulan dari nol atau lebih karakter Unicode, yang dibungkus dengan tanda kutip ganda. Di dalam string dapat digunakan backslash escapes "\" untuk membentuk karakter khusus. Sebuah karakter mewakili karakter tunggal pada string. String sangat mirip dengan string C atau Java [7].



Gambar 2.5 String JSON

5. Angka

Angka adalah sangat mirip dengan angka di C atau Java, kecuali format oktal dan heksadesimal tidak digunakan [7].



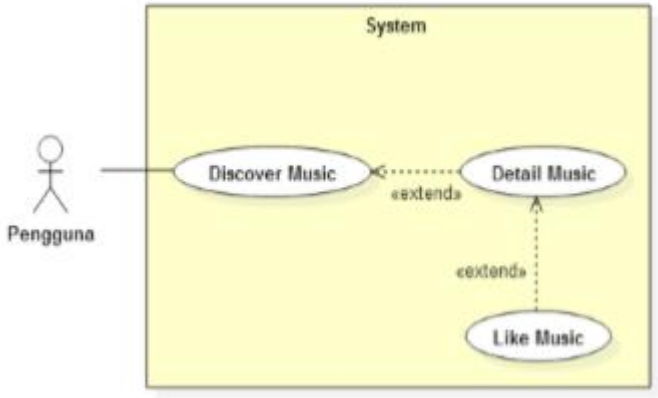
Gambar 2.6 Angka dalam JSON

2.2.6 Unified Modeling Language

Unified Modelling Language (UML) adalah sebuah bahasa yg telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak. UML menggunakan class dan operation object dalam konsep dasarnya, maka ia lebih cocok untuk penulisan piranti lunak dalam bahasa bahasa berorientasi objek [8].

2.2.6.1 Use Case Diagram

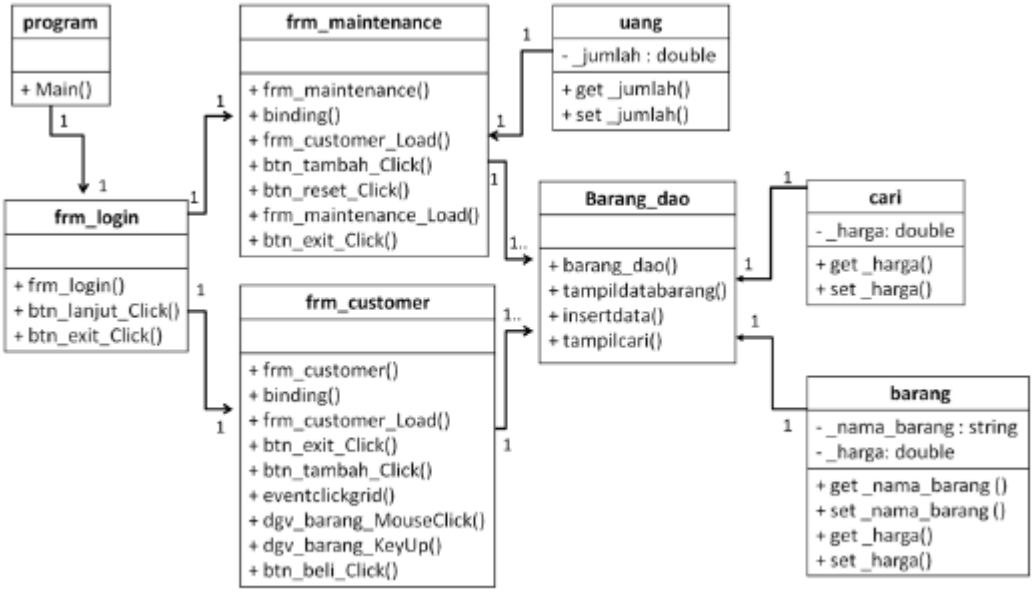
Use Case Diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”. Sebuah Use case merepresentasikan sebuah interaksi antara aktor dengan sistem. Sebuah use case dapat meng-include fungsionalitas use case lain sebagai bagian dari proses dalam dirinya. Secara umum diasumsikan bahwa use case yang di-include akan dipanggil setiap kali use case yang meng-include dieksekusi secara normal. Sebuah use case dapat di- include oleh lebih dari satu use case lain, sehingga duplikasi fungsionalitas dapat dihindari dengan cara menarik keluar fungsionalitas yang serupa. Sebuah use case juga dapat meng-extend usecase lain dengan behaviour-nya sendiri. Sementara hubungan generalisasi antar use case menunjukkan bahwa use case yang satu merupakan spesialisasi dari yang lain. Dasar menentukan sebuah use case adalah use case merupakan sesuatu yang menyediakan beberapa hasil terukur kepada pengguna atau sistem eksternal. Use case harus memiliki sangat jelas kriteria lulus / gagal. Pengembang, tester, penulis teknis, dan pengguna harus secara eksplisit tahu apakah sistem memenuhi kasus penggunaan atau tidak. Setiap bagian dari use case yang memenuhi tes sederhana ini mungkin menjadi kandidat yang baik untuk use case [9].



Gambar 2.7 Contoh Use Case Diagram

2.2.6.2 Class Diagram

Class diagram merupakan diagram yang selalu ada di permodelan sistem berorientasi objek. Class diagram menunjukan hubungan antar class dalam sistem yang sedang dibangun dan bagaimana mereka saling berkolaborasi untuk mencapai suatu tujuan. Kelas pada kelas diagram terdiri dari 3 bagian utama yaitu nama kelas, isi property dari kelas beserta metode yang ada pada kelas tersebut. Kelas juga memiliki jenis - jenis hubungan seperti asosiatif, dependensi, agregasi, komposisi, spesifikasi dan generalisasi. Hubungan ini digunakan untuk menggambarkan bagaimana hubungan dan interaksi yang terjadi antar kelas.

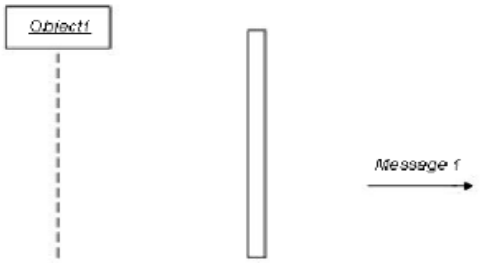


Gambar 2.8 Contoh Class Diagram

2.2.6.3 Sequence Diagram

Sequence diagram adalah suatu diagram interaksi yang menekankan pada pengaturan waktu dari pesan-pesan. Diagram ini menampilkan sekumpulan peran dan pesan-pesan yang dikirim dan diterima oleh instansi yang memegang peranan tersebut. Sequence diagram menangkap objek dan class yang terlibat dalam scenario dan urut-urutan pesan yang ditukar antara objek diperlukan untuk

melaksanakan fungsionalitas skenario. Sequence diagram berasosiasi dengan use case selama proses pengembangan. Dalam Unified Model Language (UML), objek dalam sequence diagram di dengan segiempat yang berisi nama objek yang diberi garis bawah. Objek dapat diberi nama dengan tiga cara : (nama objek), (nama objek dan class) atau (hanya nama class (anonymous object)).



Gambar 2.9 Contoh Sequence Diagram

2.2.7 Object Oriented Analisys and Design

Konsep OOAD mencakup Analisis dan Desain sebuah sistem dengan pendekatan objek, yaitu analisis berorientasi objek (OOA) dan desain berorientasi objek (OOD). OOA adalah metode analisis yang memeriksa *requirement* (syarat / keperluan) yang harus dipenuhi sebuah sistem dari sudut pandang kelas kelas dan objek – objek yang ditemui dalam ruang lingkup sistem. Sedangkan OOD adalah metode untuk mengarahkan arsitektur software yang didasarkan pada manipulasi objek – objek sistem atau subsistem.

2.2.8 Perangkat Lunak Pendukung

Berikut adalah beberapa perangkat lunak pendukung untuk membangun aplikasi yang akan dibuat:

2.2.8.1 Eclipse

Eclipse adalah sebuah IDE (*Integrated Development Environment*) untuk mengembangkan perangkat lunak dan dapat dijalankan di semua platform (platform-independent). Berikut ini adalah sifat dari Eclipse:

1. Multi-platform: Target sistem operasi Eclipse adalah Microsoft Windows, Linux, Solaris, AIX, HP-UX dan Mac OS X.
2. Mulit-language: Eclipse dikembangkan dengan bahasa pemrograman Java, akan tetapi Eclipse mendukung pengembangan aplikasi berbasis bahasa pemrograman lainnya, seperti C/C++, Cobol, Python, Perl, PHP, dan lain sebagainya.
3. Multi-role: Selain sebagai IDE untuk pengembangan aplikasi, Eclipse pun bisa digunakan untuk aktivitas dalam siklus pengembangan perangkat lunak, seperti dokumentasi, test perangkat lunak, pengembangan web, dan lain sebagainya.

2.2.8.2 Java

Java adalah bahasa pemrograman yang dapat dijalankan di berbagai computer termasuk telepon genggam. Bahasa ini awalnya dibuat oleh James Gosling saat masih bergabung di Sun Microsystems saat ini merupakan bagian dari Oracle dan dirilis tahun 1995. Bahasa ini banyak mengadopsi sintaksis yang terdapat pada C dan C++ namun dengan sintaksis model objek yang lebih sederhana serta dukungan rutin-rutin aras bawah yang minimal. Aplikasi-aplikasi berbasis Java umumnya dikompilasi ke dalam p-code (bytecode) dan dapat dijalankan pada berbagi mesin virtual java (JVM). Java merupakan bahasa pemrograman yang bersifat umum/non-spesifik (general purpose), dan secara khusus didisain untuk memanfaatkan dependensi implementasi seminimal mungkin.

Java merupakan bahasa berorientasi objek (OOP) yaitu cara ampuh dalam pengorganisasian dan pengembangan perangkat lunak. Pada OOP, program computer sebagai kelompok objek yang saling berinteraksi. Deskripsi OOP secara ringkas adalah mengorganisasikan program sebagai kumpulan komponen yang disebut objek. Objek-objek ini ada secara independen, mempunyai aturan-aturan berkomunikasi dengan objek lain dan untuk memerintahkan objek lain guna meminta informasi tertentu atau meminta objek lain mengerjakan sesuatu.

2.2.8.3 MySQL

MySQL adalah sebuah aplikasi *Relational Database Management Server* (RDBMS) yang sangat cepat dan kokoh. Dengan menggunakan MySQL Server maka data dapat diakses oleh banyak pemakai secara bersamaan sekaligus dapat membatasi akses para pemakai berdasarkan *privilege* (hak akses) yang diberikan.

MySQL menggunakan bahasa SQL (*Structured Query Language*) yaitu bahasa standar yang digunakan untuk pemrograman *database*. *Statement* atau *command* yang termasuk bahasa SQL secara umum dibagi menjadi dua kategori utam atau *data sublanguage* yaitu DDL dan DML. Setiap *sublanguage* mengkhususkan diri terhadap aspek tertentu dari bahasa tersebut. DDL menyertakan *statement* yang mendukung definisi atau kreasi dari objek data base, seperti tabel, *indeks*, *sequence*, dan tampilan. Beberapa statemen*t* adalah *CREATE, ALTER, dan DROP*. DML menyertakan *statement* yang memperbolehkan proses atau manipulasi objek database. Beberapa *staetment* DML yang sangat umum digunakan diantaranya *SELECT, INSERT, DELETE*, dan UPDATE.