

## **BAB 2**

### **TINJAUAN PUSTAKA**

#### **2.1 Air**

Air merupakan salah satu komponen penting dalam kehidupan[1] manusia karena 60% dari tubuh pria dewasa adalah air. Air memiliki banyak fungsi untuk tubuh seperti membantu dalam pembuatan ludah, membantu menyerap nutrisi, meningkatkan sirkulasi oksigen yang dibawa oleh darah dan untuk mengatur suhu di dalam tubuh dengan cara mengeluarkan keringat. Air bagi kesehatan guna untuk mlancarkan sirkulasi darah, membantu mengeluarkan racun, membantu distribusi gizi dan mencegah penyakit berbahaya. Penyakit berbahaya dari air diantaranya kekurangan air, kelebihan air ataupun keracunan air.

Banyak faktor yang menentukan berapa banyak air yang dibutuhkan untuk tubuh agar tidak kurang ataupun lebih. Salah satu faktor itu ada pada tubuh manusia sendiri salah satunya berat badan. Faktor lainnya dapat berupa cuaca saat beraktivitas ataupun kondisi badan.

##### **2.1.1 Dehidrasi**

Dehidrasi merupakan suatu kondisi dimana tubuh kehilangan lebih banyak cairan daripada yang didapatkan, sehingga menyebabkan keseimbangan zat gula dan garam menjadi terganggu. Dehidrasi juga dapat diklasifikasikan menjadi tiga yaitu ringan, sedang dan berat tergantung dengan jumlah cairan yang hialng dari tubuh. Dehidrasi sering dianggap permasalahan pada tubuh yang tidak perlu ditangani secara serius dan kebanyakan anak-anak dan remaja menganggapnya sebagai haus biasa. Derhidrasi memiliki tanda-tanda awal berikut adalah gejala-gejalanya :

1. Merasa haus
2. Pusing.

3. Mulut kering.
4. Kulit kering dan dingin.
5. Kelelahan
6. Jarang buang air kecil

Untuk gejala jika dehidrasi sudah cukup buruk yaitu :

1. Tidak buang air kecil atau urine berwarna kuning gelap.
2. Kulit sangat kering.
3. Merasa mual.
4. Detak jantung cepat.
5. Bernapas cepat.
6. Mengantuk, merasa tidak berenergi atau ada perasaan lekas marah.
7. Pingsan.

### **2.1.2 Overhidrasi**

Overhidrasi merupakan kondisi yang terbalik dari Dehidrasi. Overhidrasi terjadi ketika tubuh mendapatkan cairan lebih banyak dibandingkan dengan yang hilang. Overhidrasi dapat terjadi ketika minum lebih banyak dibandingkan dengan jumlah yang dibutuhkan tubuh. Athlete yang minum lebih banyak air demi menghindari dehidrasi dapat terkena overhidrasi. Orang-orang yang memiliki gangguan kejiwaan dengan nama *psychiatric disorder* juga dapat minum kebanyakan air. Kebanyakan air dan kekurangan sodium akan mengakibatkan Overhidrasi. Overhidrasi memiliki gejala-gejala sebagai berikut :

1. Mual.
2. Muntah.
3. Sakit Kepala.
4. Perubahan keadaan mental seperti kebingungan ataupun disorientasi.

Jika kondisi overhidrasi cukup parah gejala yang dapat timbul diantaranya:

1. Peningkatan tekanan darah
2. Penglihatan berganda
3. Kesulitan bernafas
4. Kejang-kejang
5. Koma
6. Pingsan.

### **2.1.3 Hiperhidrasi**

Hiperhidrasi atau juga dikenal dengan keracunan air. Keracunan air dapat terjadi ketika elektrolit dalam tubuh tidak seimbang dikarenakan asupan air yang cepat. Kondisi ini dapat menyebabkan efek yang sangat serius terhadap fungsi otak dan dapat menyebabkan kematian jika pengawasan medis tidak tersedia. Gejala pertama dapat berupa :

1. Pusing ringan.
2. Mual.
3. Muntah.
4. Sakit kepala.

Jika kondisi elektrolit dalam tubuh terus memburuk dapat menyebabkan kejang kejang, koma dan kematian. Hiperhidrasi biasanya terpengaruh dengan jumlah sodium dalam tubuh dan olahragawan yang sudah kehilangan sodium melalui berkeringan tetapi hanya minum air dalam jumlah yang banyak dapat mengalami Hiperhidrasi.

## **2.2 Kebutuhan Air Minum**

Kebutuhan air untuk setiap orang berbeda contohnya untuk laki laki yang sehat direkomendasikan untuk minum sebanyak 3L dan untuk Perempuan yang sehat sebanyak 2,5L[1], [14]. Jumlah air yang direkomendasikan bukan hanya air yang diminum tetapi juga termasuk air yang diserap dari makanan. Faktor yang akan berpengaruh terhadap jumlah air yang dibutuhkan juga datang dari kondisi kesehatan dari tubuh orang tersebut. Sebagai contoh untuk orang yang obesitas (Body Mass Index > 30) kebutuhan air minum pun ikut berubah berdasarkan

penelitian Asher et al[15] terjadinya peningkatan konsumsi sebesar 0,8%. Selain bertambahnya kebutuhan air minum ada juga kondisi tubuh yang membutuhkan konsumsi yang kurang dari rekomendasi untuk orang sehat dalam kondisi seperti ini disarankan mengikuti rekomendasi oleh dokter karena dapat menyebabkan kematian.

### **2.3 Rekomendasi Air Minum**

Rekomendasi air yang pasti tidak dapat dijawab dengan jawaban yang efektif dan tidak adanya konsensus universal. Kompleksitas dinamis dari jaringan pengaturan air dan perbedaan antar-individu adalah alasan utama mengapa konsensus mengenai kebutuhan air harian belum tercapai hingga saat ini[3], [5], [16], [17]. Walaupun kebutuhan air harian belum memiliki jawaban yang pasti bukan berarti tidak memiliki jawaban dalam penelitian dalam panduan WHO setidaknya estimasi masih dapat diberikan berdasarkan rata-rata dari tahun tahun sebelumnya[1]. Dalam penelitian ini penulis menggunakan panduan yang diberikan oleh American Council on Exercise[18]. Dalam panduan yang diberikan disarankan untuk minum air sebelum, sesaat dan sesudah dari berolahraga. Berikut merupakan panduan yang diberikan :

- Minum air sebanyak 500 – 600ml sebelum memulai olahraga.
- Minum air sebanyak 250ml saat 20 – 30 menit memulai berolahraga atau saat pemanasan.
- Minum air sebanyak 200 – 300ml saat berada di 10 – 20 menit ketika berolahraga.
- Minum air sebanyak 250ml air tidak lebih dari 30 menit setelah berolahraga.

Athlete mungkin butuh mengukur berapa banyak cairan yang hilang selama berolahraga dengan pengukuran yang lebih spesifik berdasarkan dengan jumlah air yang butuh diminum dengan hitungan 450 – 700ml air untuk setiap 0,5 kg berat badan yang hilang. Jika olahraga yang dilakukan melebihi 1 jam dan menghasilkan banyak keringat disarankan juga untuk meminum minuman selain air putih seperti jus ataupun minuman olahraga lainnya. Hal yang dicari dari minuman tersebut adalah karbohidrat, garam dan juga gula. Air yang diminum saat olahraga

disarankan untuk dilakukan secara berkala dikarenakan maksimal kapasitas ekskresi ginjal sekitar 0,7 hingga 1L/jam [14].

## 2.4 Aplikasi Mobile

Aplikasi berasal dari kata *application* yang berarti suatu bagian dari perangkat lunak komputer yang dibuat untuk mendukung suatu pekerjaan atau aktivitas yang dilakukan manusia. Aplikasi menurut Kamus Besar Bahasa Indonesia (KBBI) adalah penerapan dari rancangan sistem untuk mengolah data yang menggunakan ketentuan bahasa pemrograman tertentu dan juga suatu program komputer yang dibuat untuk mengerjakan dan melaksanakan tugas khusus dari pengguna. Aplikasi adalah kumpulan perintah yang ada untuk dieksekusi komputer.

Mobile adalah kata sifat yang memiliki arti dapat bergerak atau dapat digerakan dengan bebas atau mudah. Mobile juga dapat pula diartikan sebagai benda yang sudah berteknologi tinggi dan dapat bergerak tanpa menggunakan kabel. Aplikasi mobile dapat diartikan sebagai perangkat lunak yang berjalan di *smartphone*. Pemanfaatan aplikasi mobile untuk hiburan paling banyak digemari oleh 70% pengguna *smartphone* karena adanya fitur *game*, *music player*, sampai *video player* yang mempermudah untuk menikmati hiburan kapan saja dan dimana saja [19].

## 2.5 Android

Android adalah sistem operasi berbasis linux yang dirancang untuk perangkat *mobile* yang mencakup sistem operasi, *middleware* atau aplikasi. Android dikembangkan oleh Android, Inc., yang kemudian dibeli oleh Google pada tahun 2005. Android dirilis secara resmi pada tahun 2007 dan bersamaan dengan didirikannya Open Handset Alliance yang bertujuan untuk memajukan standar terbuka untuk perangkat seluler [20]. Google juga merilis kode-kode dibawah lisensi apache, sebuah lisensi untuk perangkat lunak yang *open source*.

Android merupakan salah satu sistem operasi yang paling banyak digunakan di *smartphone* karena memiliki beberapa kelebihan pada sistemnya seperti :

1. Google Play Store Android memiliki Google Play Store untuk dapat mengunduh aplikasi-aplikasi apa saja yang diinginkan selama tersedia di Google Play Store.

2. Android selalu berkembang dari waktu ke waktu ini dibuktikan dengan banyaknya versi android pada perangkat seluler. Pada akhir tahun 2019 kemarin android merilis versi terbarunya yang bernama Android version 10 dulu sempat dikenal dengan Android Q.
3. Android memiliki dukungan Google yang berarti layanan-layanan google dapat digunakan dengan mudah karena rata-rata sudah terintegrasi dari awal seperti Google Maps, Gmail, Drive dan lainnya.

## 2.6 Accelerometer

Menurut Riyadi dkk [21] accelerometer adalah sensor yang digunakan untuk mengukur percepatan, mendeteksi dan mengukur getaran, dan mengukur percepatan akibat gravitasi (inklinasi). Sensor accelerometer juga dapat mengukur getaran pada mobil dan bangunan. Accelerometer mengukur percepatan dengan menggunakan nilai dari tiga sumbu x,y,z lalu berdasarkan dari tiga sumbu tersebut akan dihitung untuk mengetahui percepatan atau perlambatan yang terjadi.

Sensor accelerometer pada *smartphone* digunakan untuk mengubah tampilan pada layar dari potrait ke landscape maupun sebaliknya dengan memiringkan badan dari *smartphone* maka akan terjadi perubahan nilai pada tiga sumbu dari *smartphone*. Prinsip dari accelerometer adalah sebuah per dengan beban dan dilepaskan, beban bergerak dengan suatu percepatan sampai kondisi tertentu lalu berhenti. Bila ada sesuatu yang mengguncangkannya maka beban berayun kembali. Accelerometer dapat digunakan untuk mengukur getaran pada mobil, mesin, bangunan, dan instalasi pengamananan. Sensor accelerometer juga dapat diaplikasikan pada pengukuran aktivitas gempa, pedometer atau penghitung langkah.

## 2.7 *Global Positioning System (GPS)*

*Global Positioning System (GPS)* adalah suatu sistem navigasi atau penentu posisi berbasis satelit. Sistem ini didesain untuk memberikan posisi dan informasi mengenai waktu, secara kontinyu di seluruh dunia tanpa tergantung waktu dan cuaca. Penentuan posisi GPS digambarkan dengan menggunakan nilai koordinat X dan Y atau garis bujur dan garis lintang (longitude atau latitude) [22].

### 1. Titik Kordinat

Titik koordinat adalah titik yang berpedoman pada garis latitude dan longitude suatu daerah. Kaitannya dengan latitude dan longitude adalah, kedua garis lintang dan bujur inilah (latitude = garis lintang, longitude = garis bujur) yang menentukan di perolehnya suatu nilai derajat dari suatu titik yang diukur. Koordinat pada umumnya dituliskan dengan format DD (*Degree Decimal*), terkadang juga dituliskan dengan format DMS (*Degree-Minute-Second*).

### 2. Konversi Derajat Bujur dan Lintang

Perlu diketahui pula konversi derajat menit (,) dan detik (") pada letak astronomis suatu tempat dalam satuan jarak (km dan m). Dalam pemakaian GPS sering didapatkan titik koordinat suatu posisi lokasi dalam bentuk *Latitude* (Lintang), *Longitude* (Bujur) atau *Degree* (Derajat). Dalam hal ini ada suatu formula yang digunakan untuk mengkonversi bilangan tersebut dari bentuk *Degree Minute Second* (DMS) menjadi *Degree Decimal* (DD), atausebaliknya konversi *Degree Decimal* (DD) menjadi *Degree Minute Second* (DMS).

## 2.8 *Application Programming Interface (API)*

*Application Programming Interface (API)* adalah *interface* dari sebuah aplikasi perangkat lunak yang mempresentasikan dirinya ke aplikasi yang lain, developer, ataupun pengguna. API merupakan blok pembangun yang mengijinkan interoperabilitas dari platform bisnis besar di web[23]. Isi dari API merupakan barisan barisan perintah yang tersimpan dalam bentuk *library*. Hal ini dapat dicontohkan dengan kita memiliki buku yang dapat menghasilkan apapun yang dituliskan didalamnya. Dalam hal ini kita sebagai penulis tidak perlu mengetahui bagaimana benda dapat muncul, dengan hanya kita tulis di dalam buku tersebut. Dari analaogi tersebut benda yang keluar dari buku merupakan perangkat lunak yang akan dibuat, dan buku adalah API yang bekerja dalam bagian tertentu dari

perangkat lunak itu, tanpa harus mengetahui prosedur dalam pengerjaan pekerjaan tersebut.

Dalam API terdapat fungsi-fungsi atau perintah-perintah untuk menggantikan bahasa yang digunakan dalam *system calls* dengan bahasa yang lebih terstruktur dan mudah dimengerti oleh programmer. Keuntungan menggunakan API diantaranya:

1. Probabilitas, API dapat digunakan untuk bahasa pemrograman ataupun untuk sistem operasi mana saja asalkan paket-paket API sudah terpasang.
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.
3. Mudah dikembangkan, dengan adanya API memudahkan programmer untuk mengembangkan suatu sistem.

## **2.9 Google Maps API**

Google Maps API merupakan API gratis yang disediakan oleh Google dan sangat populer bagi developer yang membutuhkan untuk menampilkan peta di aplikasinya. Google Maps API merupakan peta dunia yang dapat digunakan untuk melihat suatu daerah, selain hanya menampilkan, developer dapat menambahkan fitur-fitur lainnya ke google maps yang digunakan[24]. Fitur-fitur tersebut dapat merupakan fitur yang gratis ataupun berbayar. Contoh dari fitur-fitur tersebut seperti Direction API yang dapat memberikan response berupa rute dari titik awal ke titik tujuan, Distance Matrix API untuk memberikan jarak dan waktu perjalanan dan masih banyak yang lainnya. Google Maps API juga tidak terpaku dalam 1 platform tapi tersedia untuk banyak platform seperti android, iOS, web dan juga web service.

Dalam perkembangannya google maps API diberikan kemampuan untuk mengambil peta statis. Melakukan geocoding dan memberikan penuntun arah. Kekurangan pada google maps API yaitu jika ingin melakukan akses harus terdapat layanan internet pada perangkat yang digunakan, sedangkan kelebihan yang dimiliki yaitu :



1. Dukungan penuh yang dilakukan google sehingga terjamin dan fitur yang bervariasi pada google maps API.
2. Banyak pengembang yang menggunakan google maps API sehingga mudah dalam mencari referensi dalam pengembangan aplikasi.

## **2.10 Activity Recognition API**

*Activity recognition* API merupakan salah satu source closed API yang disediakan oleh google untuk mengecek aktivitas dari pengguna. *Activity recognition* API diperkenalkan saat acara *100 Days of Google Dev* didalam acara tersebut *activity recognition* API diperkenalkan sebagai API yang dapat membantu developer agar mendapatkan konteks dari aktivitas yang dilakukan oleh pengguna, dari konteks yang didapatkan developer dapat meningkatkan “*experience*” untuk pengguna saat menggunakan aplikasi yang dikembangkan.

*Activity recognition* API menggunakan *machine learning model* untuk mengenali aktivitas pengguna berdasarkan data dari sensor yang diterima. *Activity recognition* API membaca data dari sensor secara periodik dan memprosesnya menggunakan *machine learning model*. *Activity recognition* API dapat mengenali 5 jenis aktivitas yaitu diam ditempat, berjalan, berlari, bersepeda dan berkendara. *Activity recognition* API juga sudah dikembangkan untuk mengurangi penggunaan memori dan mengurangi penggunaan baterai.

## **2.11 Android Studio**

Android studio merupakan sebuah *Integrated Development Environment* (IDE) yang dikembangkan oleh Google dan JetBrains. Android studio diluncurkan pertama kali pada tanggal 16 Mei 2013 di *Google I/O Conference*. Android Studio adalah IDE resmi untuk sistem operasi Android.

Android Studio dibangun berdasarkan IntelliJ IDEA [25]. Selain editor kode dan fitur developer IntelliJ yang andal, Android Studio menyediakan banyak fitur yang dapat meningkatkan produktivitas dalam pembuatan aplikasi android seperti :

1. Dukungan pembangunan berbasis gradle
2. Dukungan untuk membangun aplikasi Android Wear

3. Instant Run untuk menjalankan perubahan ke aplikasi yang berjalan tanpa membuat APK baru
4. Editor kode yang pintar, Android Studio dapat menganalisis dan menyediakan saran *syntax* yang dapat digunakan dengan sistem *auto-complete*
5. Adanya layout editor GUI yang mempermudah developer dalam melihat dan merancang aplikasi
6. Fitur Lint yang dapat menemukan struktur kode yang buruk atau yang dapat memengaruhi keandalan dan efisiensi dari aplikasi
7. Penggunaan *library* yang mudah dengan hanya menambahkan 1 baris kode dalam file *gradle*
8. Integrasi GitHub yang membantu dalam membuat *versioning*

## 2.12 Java

Java adalah bahasa pemrograman yang berorientasi objek dan dibuat oleh James Gosling yang berada di Sun Microsystem pada tahun 1995. Java awalnya bernama OAK, tetapi karena OAK merupakan nama dari bahasa pemrograman yang sudah ada maka diubah menjadi Java. Java memiliki slogan *Write Once, Run Anywhere* (WORA) yang berarti developer mengembangkan kode java dalam satu sistem operasi dan dapat dijalankan di sistem operasi yang lainnya [26]. Java memiliki beberapa keunggulan dibandingkan bahasa pemrograman lainnya yaitu :

1. Berorientasi Objek
2. Multiplatform
3. Multithread
4. Dapat didistribusikan dengan mudah
5. Dinamis

## 2.13 JSON

JSON (*JavaScript Object Notation*) merupakan format pertukaran data, seperti XML atau CSV. Walaupun JSON memiliki arti *Javascript* bukan berarti saat ingin menggunakan JSON wajib mengetahui *Javascript*. *Javascript* disini ada karena format pertukaran data JSON merupakan subset dari *Javascript*. JSON juga

merepresentasikan datanya dengan mudah untuk konsep pemrograman universal[27].

JSON merupakan salah satu format pertukaran data yang terkenal untuk mengirimkan request atau response API. Hal ini dikarenakan JSON berbasis teks dan terbaca dengan mudah oleh manusia, begitu juga dengan komputer. Komputer dapat menerjemahkan dan juga menghasilkan JSON lebih cepat dibandingkan XML[28].

#### 2.14 PHP

PHP dikenal secara umum dikenal sebagai bahasa pemrograman script-script yang membuat dokumen HTML secara langsung dan cepat yang dieksekusi di server web, dokumen yang dihasilkan dari suatu aplikasi bukan dokumen HTML yang dibuat dengan menggunakan editor teks atau editor HTML, dikenal juga sebagai bahasa pemrograman server side [29]. PHP diciptakan oleh Rasmus Lerdorf pertama kali tahun 1994. Pada awalnya PHP adalah singkatan dari "*Personal Home Page Tools*". Selanjutnya diganti menjadi FI (*Forms Interpreter*). Sejak versi 3.0, nama bahasa ini diubah menjadi "*PHP: Hypertext Preprocessor*" dengan singkatannya "PHP". Dapat dilihat pada Gambar 2.1 merupakan logo dari PHP.



Gambar 2.1 Logo PHP

Sumber Gambar : <https://www.php.net/download-logos.php>

#### 2.15 MySQL

MySQL adalah salah satu aplikasi DBMS (*Database Management System*) yang sudah sangat banyak digunakan oleh para pemrograman aplikasi web. Kelebihan dari MySQL ini adalah gratis, handal, selalu di-update dan banyak forum yang memfasilitasi para pengguna jika memiliki kendala [30]. MySQL masuk kedalam RDBMS (*Relational Database Management System*) yang dapat

menangani data yang bervolume besar. Meskipun begitu, tidak menuntut resource yang besar.

MySQL adalah program database yang mampu mengirim dan menerima data dengan sangat cepat dan memiliki fitur multi user. MySQL memiliki dua bentuk lisensi yaitu free software dan shareware. MySQL yang tersedia sebagai software gratis berada di bawah lisensi GNU/GPL (*General Public License*). Selain dibawah lisensi GNU/GPL ada juga lisensi komersial untuk kasus-kasus yang bersifat khusus. MySQL bukan merupakan program yang ditopang oleh sebuah komunitas umum, tetapi oleh sebuah perusahaan yang profesional yaitu MySQL AB di Swedia.

### **2.16 Unified Modelling Language (UML)**

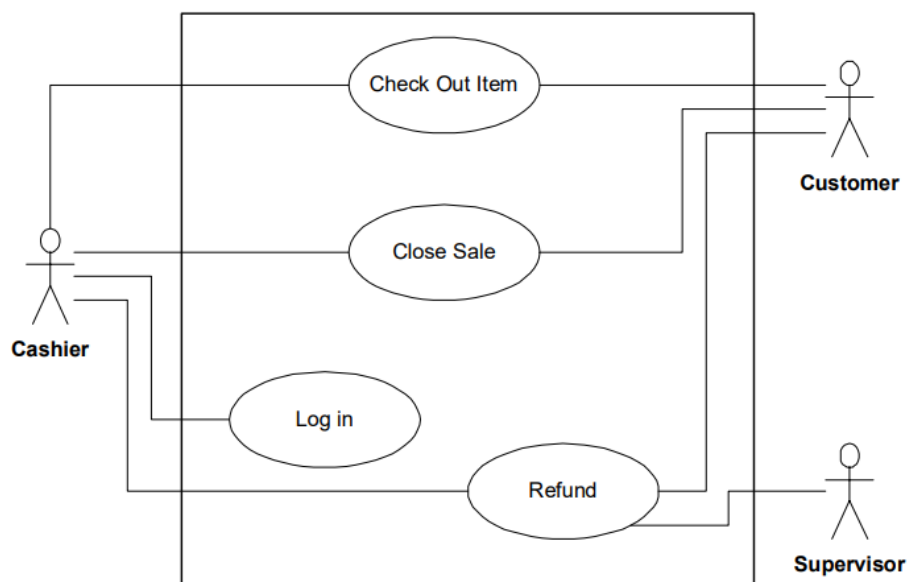
*Unified Modelling Language (UML)* merupakan sebuah bahasa pemodelan yang umum digunakan dalam pengembangan perangkat lunak berorientasi objek[31]. UML digunakan untuk menjadi standar dalam visualisasi, perancangan dan juga pendokumentasian sistem. UML memiliki fungsi agar pengembang dapat memahami, mendesain, mengkonfigurasi, memelihara dan mengontrol informasi mengenai sistem. Hal ini bertujuan untuk digabungkan dengan metode-metode pengembangan dan bagian dari siklus hidup perangkat lunak.

UML dikembangkan pada tahun 1994-1996 oleh Grady Booch, Ivar Jacobson dan James Rumbaugh yang bekerja di Rational Software. Pada tahun 1997 UML diadopsi sebagai standar oleh *Object Management Group (OMG)* dan dikelola oleh organisasi ini sejak saat itu. *Organization for Standardization (ISO)* juga menerbitkan UML sebagai standar ISO yang disetujui. UML menyediakan beberapa diagram untuk memvisualisasikan pemodelan berorientasi objek yaitu, diagram kelas, diagram objek, diagram aktivitas, diagram urutan dan use case diagram.

### **2.17 Use Case Diagram**

*Use Case Diagram* adalah salah satu diagram yang memodelkan perilaku dari sistem. *Use Case Diagram* merupakan interaksi antara aktor luar dengan sistem yang digunakan gunanya untuk mendeskripsikan respon balik dari sistem. *Use Case*

Diagram juga menyediakan deskripsi pandangan eksternal dengan sistem dan interaksi – interaksinya dengan luar sistem. *Use Case Diagram* memiliki beberapa elemen yang dapat digunakan yaitu aktor, *use case* dan hubungan antar aktor dan *use case*[32].

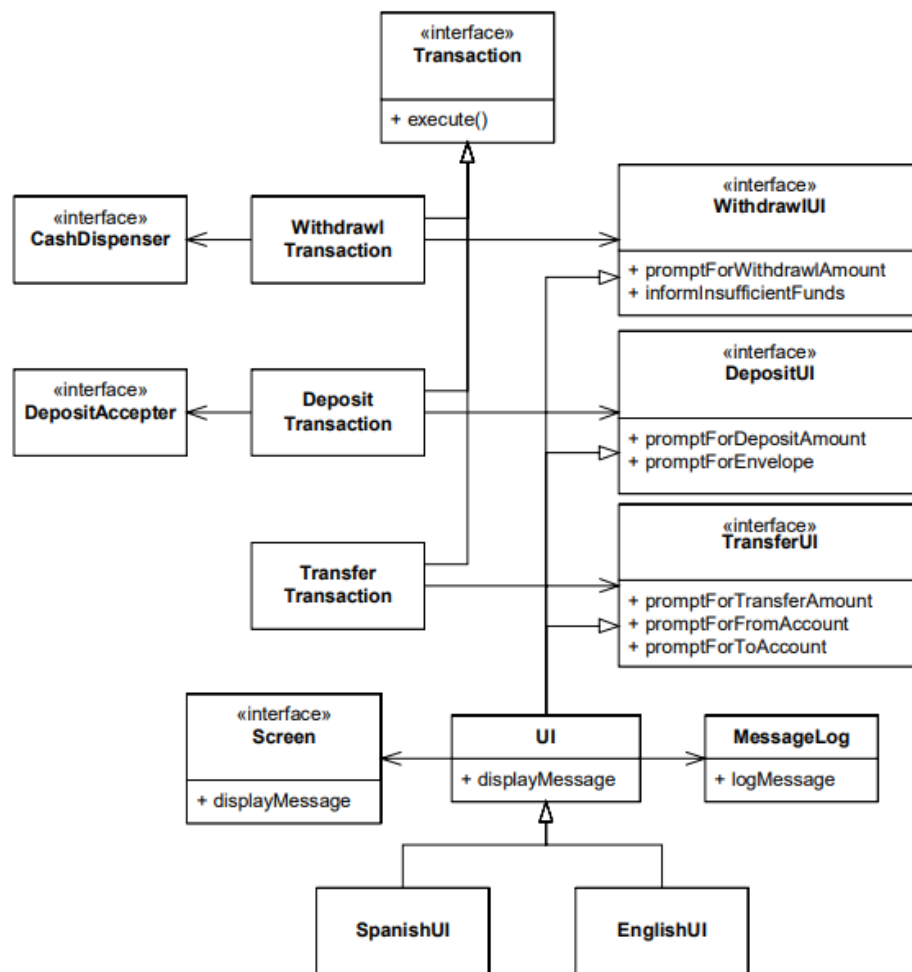


Gambar 2.2 Contoh Use Case Diagram  
Sumber : *UML for Java Beginners* [33]

## 2.18 Class Diagram

*Class Diagram* merupakan diagram yang menggambarkan struktur dan menunjukkan aspek statis dalam sistem terutama untuk mendukung kebutuhan fungsional sistem[34]. Kebutuhan fungsional merupakan fungsi-fungsi atau layanan-layanan yang disediakan oleh sistem ke pengguna. *Class Diagram* menggambarkan hubungan antar *class* dalam hubungan itu digambarkan apa yang terjadi.

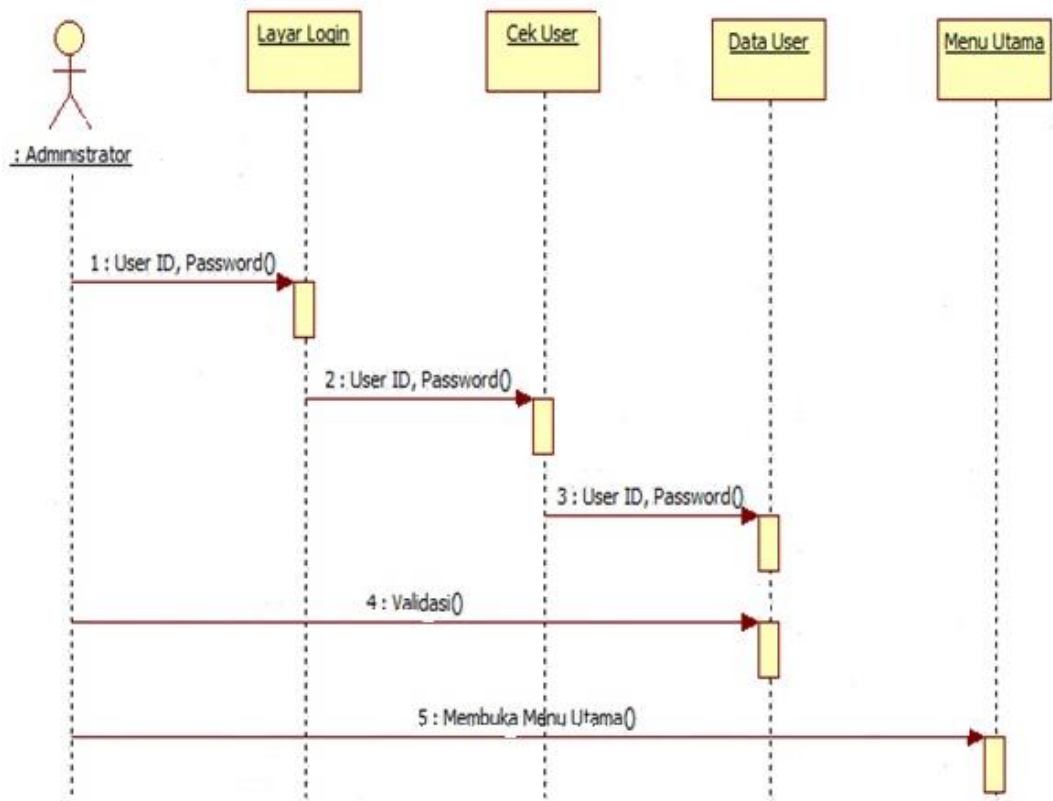
Di dalam *class diagram* terdapat *class*, *class* memiliki nama *class*. Nama dari setiap *class* harus unik karena menjadi pengenalan di dalam program. *Class* dapat terdiri dari atribut dan *method*. *Class* memiliki beberapa jenis hubungan dengan *class* lain contohnya *Inheritance*, *Aggregation*, *Dependency* dan *Composition*.



Gambar 2.3 Contoh Class Diagram  
 Sumber : *UML for Java Beginners* [33]

## 2.19 Sequence Diagram

*Sequence* Diagram adalah diagram yang menggambarkan interaksi antara *object* dengan *object* yang lainnya[35]. Di dalam *sequence* diagram garis vertikal menunjukkan *object* dan garis horizontal menunjukkan tiap kejadian dari *object* pengirim ke *object* yang menerima. Waktu di dalam *sequence* diagram berlalu dari atas ke bawah dengan lama dari waktu tidak berhubungan dengan jarak dari atas kebawah.

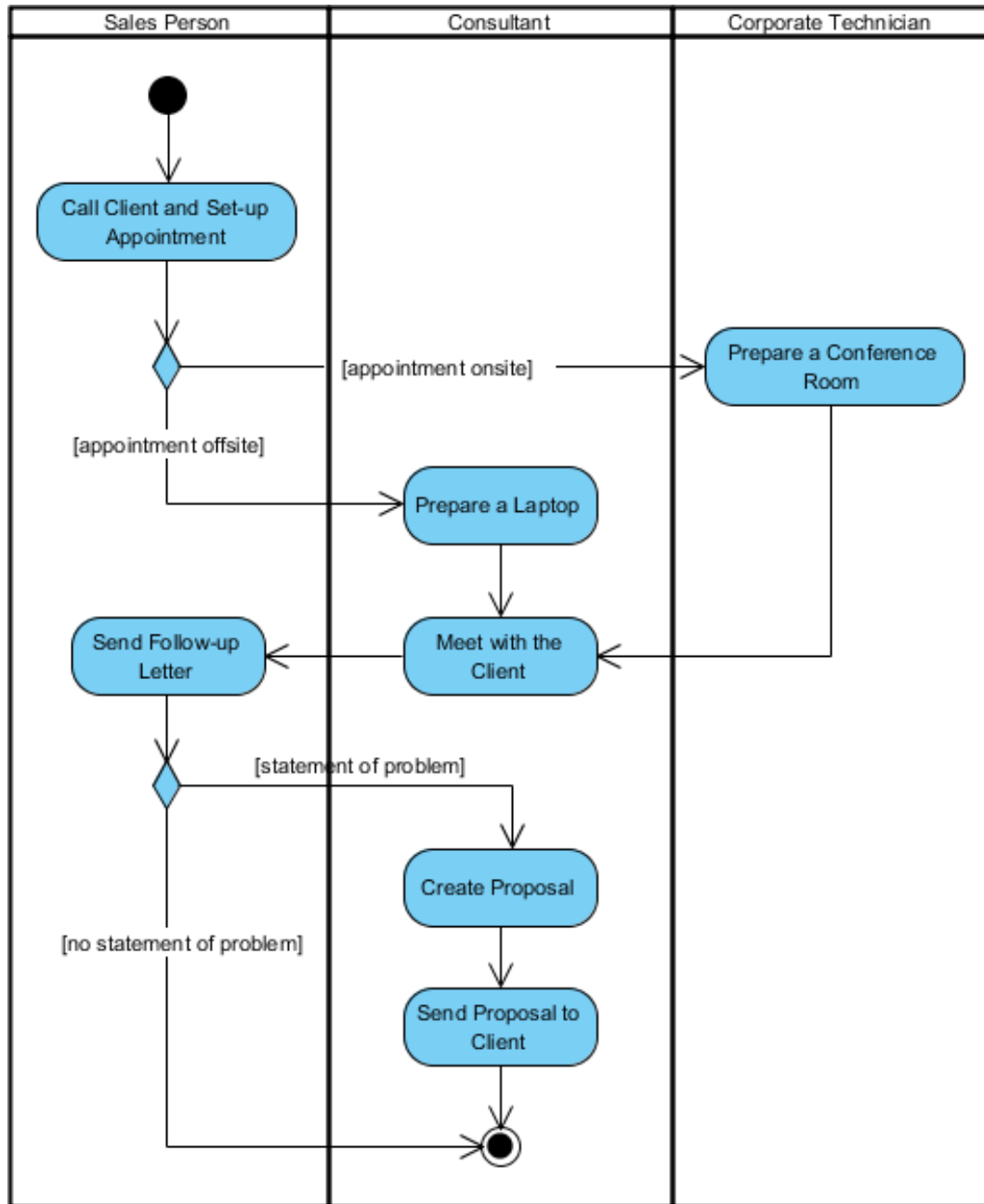


Gambar 2.4 Contoh Sequence Diagram

Sumber Gambar : <https://www.codepolitan.com/belajar-uml-sequence-diagram-57fdb1a5ba777-17044>

## 2.20 Activity Diagram

*Activity Diagram* memiliki kemiripan dengan *flowchart* tetapi lebih diperluas dengan menunjukkan aliran kendali dari aktivitas satu ke aktivitas yang lain. *Activity Diagram* menggambarkan *work flow* antar aktivitas dalam sebuah sistem atau proses bisnis yang ada pada perangkat lunak[34]. *Activity Diagram* berfokus pada aktivitas-aktivitas yang terjadi dan saling berhubungan dalam sebuah proses. *Activity Diagram* memiliki beberapa elemen yaitu state awal, aksi atau state aktivitas, alur aksi atau alur kontrol, node keputusan, *fork*, *join*, *swimlane* dan state akhir.



Gambar 2.5 Contoh Activity Diagram

Sumber Gambar : <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-activity-diagram/>