

## **BAB 2**

### **TINJAUAN PUSTAKA**

#### **2.1 Landasan Teori**

Dalam penelitian ini ada beberapa landasan teori yang menjadi bahan dalam melakukan penelitian yang akan di jelaskan sebagai berikut :

##### **2.1.1 Wearable Device**

*Wearable Device* adalah bentuk dari perangkat mutakhir saat ini, didalamnya tertanam perangkat – perangkat tambahan yang canggih dengan teknologi cerdas yang memiliki bentuk sangat kecil, dan bobotnya ringan sehingga dapat dikenakan di tubuh seseorang sehingga biasanya perangkat ini dapat berupa aksesoris seperti jam tangan, kacamata, gelang pintar, ataupun baju.

Untuk dapat berinteraksi dengan perangkat ini biasanya menggunakan *smartphone* sebagai media kontrol dan sistem untuk mengatur penggunaanya yang terkoneksi melalui *bluetooth*. Di dalam *Wearable Device* ini memiliki sensor – sensor untuk mendeteksi tubuh seseorang, tujuannya yaitu untuk memonitoring kesehatan, aktivitas, ataupun kegunaan lainnya yang dapat di implementasikan. Salah satu contoh seperti sensor detak jantung yang biasanya ada pada *Wearable Device* jenis jam tangan pintar atau gelang pintar untuk memonitoring detak jantung per sekian waktu yang informasinya dapat di akses langsung dari wearable tersebut atau melalui *smartphone* yang dimana dapat dilihat secara rinci karna telah di olah oleh sistem tertentu.

*Wearable Device* memiliki antarmuka keluaran dimana tempat informasi disajikan kepada pemakai. Seperti getaran (sentuhan) telah digunakan, contohnya adalah fungsi getar pada telepon seluler, di mana pengguna secara diam-diam diberi tahu jika ada panggilan masuk. Banyak perangkat portabel menggunakan antarmuka audio. Dalam kedua kasus, jumlah informasi yang diberikan cukup kecil. Antarmuka keluaran lainnya adalah *visual* antarmuka. Ini termasuk,

misalnya, tampilan tujuh segmen atau dot matrix, LCD, OLEDs dan PLEDs, dan display serat optik (FODs) [9].

Pada penelitian ini wearable yang digunakan berjenis gelang pintar yang dapat dilihat pada Gambar 2.1.



Sumber : gearbest.com  
**Gambar 2.1 Xiaomi Mi Band 3**

Gelang pintar ini memiliki sensor accelerometer, dan detak jantung. Melalui dua sensor tersebut maka akan dilakukan percobaan untuk mendeteksi *Mood* dan aktivitas seseorang yang kemudian di olah dan di klasifikasikan berdasarkan *Mood* dan aktivitas tertentu untuk memberikan rekomendasi lagu.

*Wearable Device* yang digunakan memiliki spesifikasi yang dapat dilihat pada Tabel 2.1 sebagai berikut :

**Tabel 2.1 Spesifikasi Mi band 3**

<b>Nama</b>	<b>Xiaomi Mi Band 3</b>
Dimensi	46
9 x 17	9 x 12 mm
Berat	20 gram
Material	Plastik dan alloy
Jenis Layar	OLED
Ukuran	0,78 inci
Resolusi	128×80 piksel
OS	Proprietary OS

Nama	Xiaomi Mi Band 3
Konektivitas	Bluetooth versi 4.2
Fitur Ketahanan	IP67, tahan air 50 meter
Sensor	ADI ultra low-power acceleration sensor + photoelectric heart rate sensor
Fitur	call and message notifications , call and message notifications, caller ID, NFC, weather notifications, alarm, distance and calorie counter, sedentary reminder, Unlock phone without password, sleep monitoring, and social app notifications
Kompatibilitas	Android and iOS

Sumber : <http://www.mi.co.id/id/miband/>

Spesifikasi tersebut sudah cukup memenuhi untuk penelitian yang dilakukan sensor-sensor yang diperlukan sudah terdapat pada mi band 3 seperti *bluetooth*, *accelerometer*, dan sensor *heart rate*.

### 2.1.2 API Spotify

Spotify adalah layanan *streaming* musik interaktif yang didirikan di Swedia, sesuai dengan keinginannya situs web, saat ini beroperasi di 28 negara, dengan lebih dari 24 juta pengguna, di antaranya 6 juta dilaporkan membayar pelanggan [10]. Spotify menawarkan akses ke pendengar musik melalui layanan *streaming* yang dapat di akses melalui aplikasi mobile atau melalui situs resminya di <https://www.spotify.com> .Spotify memiliki beberapa layanan di antaranya berbayar atau premium yang mewajibkan pengguna untuk berlangganan, dan juga layanan gratis. Untuk layanan gratis biasanya akan diselingi iklan disaat kita sedang melakukan *streaming* musik, layanan berlangganan premium bebas iklan dengan peningkatan fungsi dan beberapa fitur tambahan yang tidak dibatasi.

Untuk menawarkan layanan ini secara sah tanpa melanggar hak cipta, Spotify harus mendapatkan izin dari pemegang hak yang relevan mereka adalah pemilik hak yang ada dalam rekaman suara, secara tradisional perusahaan rekaman, dan komposer atau penerbit mereka, Sebagai imbalan untuk penggunaan ini, pemegang hak menerima pembayaran keuangan [10].

Selain itu spotify juga menyediakan API yang dapat digunakan oleh developer untuk membangun aplikasi dengan menggunakan data dari API Spotify. Data tersebut berupa katalog lagu, artis/musisi, *Playlist* lagu, dsb. *Playlist* lagu pada API Spotify menyediakan berbagai macam *Playlist* yang dapat disesuaikan seperti *Playlist* berdasarkan *Mood* dan juga aktivitas yang akan di gunakan pada penelitian ini. Untuk dokumentasi dari API spotify dapat di akses melalui <https://developer.spotify.com> . Sebelum menggunakan API spotify diwajibkan untuk memiliki akun sebagai akses untuk mendapatkan *client id* yang diperlukan untuk melakukan request data ke API spotify. API spotify menyediakan beberapa dokumentasi berdasarkan *platform* yang digunakan untuk membangun aplikasi mulai dari Web API, Android SDK, IOS SDK.

Untuk menggunakan API Spotify, terdapat *Authorization Scopes* yang harus di sertakan dalam pemanggilan API. *Scopes* memberi tahu pengguna Spotify untuk menggunakan aplikasi pihak ketiga yang dipercaya bahwa hanya informasi yang mereka pilih yang akan dibagikan, dan tidak lebih.

Berikut beberapa *scopes* tersebut dapat dilihat pada Tabel 2.2.

**Tabel 2.2 Scopes API Spotify**

	<i>Scopes</i>	<b>Deskripsi</b>
<i>Listening History</i>	user-read-recently-played	Membaca akses ke lagu terakhir yang diputar pengguna.
	User-top-read	Membaca akses ke artis dan lagu top dari pengguna.
<i>Library</i>	user-library-modify	Menghapus atau memperbaharui <i>library</i> pengguna.
	User-library-read	Membaca akses <i>library</i> pengguna.
<i>Playlist</i>	<i>Playlist-read-private</i>	Membaca akses ke <i>Playlist</i> tertutup dari pengguna.
	<i>Playlist-modify-public</i>	Membaca akses ke <i>Playlist</i> publik dari pengguna.
	<i>Playlist-modify-private</i>	Menghapus atau memperbaharui <i>Playlist</i> tertutup dari pengguna.
	<i>Playlist-read-collaborative</i>	Sertakan daftar putar kolaboratif saat meminta daftar putar pengguna.
<i>Users</i>	user-read-email	Membaca akses terhadap email pengguna.

	<b>Scopes</b>	<b>Deskripsi</b>
	User-read-birthdate	Membaca akses terhadap tanggal lahir pengguna.
	User-read-private	Membaca akses tipe akun dari pengguna.
<i>Spotify Connect</i>	user-read-playback-state	Membaca akses ke pemutar lagu pengguna.
	User-modify-playback-state	Menulis akses ke pemutar lagu pengguna.
	User-read-currently-playing	Membaca akses terhadap lagu yang sedang diputar pengguna.
<i>Playback</i>	app-remote-control	Melakukan remote terhadap aplikasi spotify.
	Streaming	Melakukan remote terhadap aplikasi spotify dengan akun premium.
<i>Follow</i>	user-follow-read	Membaca <i>followers</i> pengguna.
	User-follow-modify	Memodifikasi <i>followers</i> pengguna.

Sumber : <https://developer.spotify.com/documentation/>

Scopes tersebut diperlukan untuk dapat mengakses beberapa dari *endpoint* yang tersedia pada API Spotify. Referensi API *endpoint* yang terdapat pada Tabel 2.3.

**Tabel 2.3 Endpoint API Spotify**

Method	Endpoint	Returns
GET	/v1/albums/{id}	album
GET	/v1/albums/{id}/tracks	tracks
GET	/v1/albums	albums
GET	/v1/artists/{id}	artist
GET	/v1/artists/{id}/albums	albums
GET	/v1/artists/{id}/top-tracks	tracks
GET	/v1/artists/{id}/related-artists	artists
GET	/v1/artists	artists
GET	/v1/browse/categories/{category_id}	category
GET	/v1/browse/categories/{category_id}/Playlists	Playlists
GET	/v1/browse/categories	categories
GET	/v1/browse/featured-Playlists	Playlists
GET	/v1/browse/new-releases	albums
GET	/v1/recommendations	Object
GET	/v1/me/following/contains	true/false
GET	/v1/Playlists/{Playlist_id}/followers/contains	true/false
PUT	/v1/me/following	-
PUT	/v1/Playlists/{Playlist_id}/followers	-

Method	Endpoint	Returns
GET	/v1/me/following?type=artist	artists
DELETE	/v1/me/following	-
DELETE	/v1/Playlists/{Playlist_id}/followers	-
GET	/v1/me/albums/contains	true/false
GET	/v1/me/tracks/contains	true/false
GET	/v1/me/albums	albums
GET	/v1/me/tracks	tracks
DELETE	/v1/me/albums?ids={ids}	-
DELETE	/v1/me/tracks	-
PUT	/v1/me/albums?ids={ids}	-
PUT	/v1/me/tracks	-
GET	/v1/me/top/{type}	artists or tracks
GET	/v1/me/player/devices	devices
GET	/v1/me/player	context
GET	/v1/me/player/recently-played	object
GET	/v1/me/player/currently-playing	object
PUT	/v1/me/player/pause	-
PUT	/v1/me/player/seek	-
PUT	/v1/me/player/repeat	-
PUT	/v1/me/player/volume	-
POST	/v1/me/player/next	-
POST	/v1/me/player/previous	-
PUT	/v1/me/player/play	-
PUT	/v1/me/player/shuffle	-
PUT	/v1/me/player	-
POST	/v1/Playlists/{Playlist_id}/tracks	-
PUT	/v1/Playlists/{Playlist_id}	-
POST	/v1/users/{user_id}/Playlists	-
GET	/v1/me/Playlists	
GET	/v1/users/{user_id}/Playlists	Playlists
GET	/v1/Playlists/{Playlist_id}/images	objects
GET	/v1/Playlists/{Playlist_id}	Playlist
GET	/v1/Playlists/{Playlist_id}/tracks	tracks
DELETE	/v1/Playlists/{Playlist_id}/tracks Remove	-
PUT	/v1/Playlists/{Playlist_id}/tracks	-
PUT	/v1/Playlists/{Playlist_id}/tracks	-
PUT	/v1/Playlists/{Playlist_id}/images	-
GET	/v1/audio-analysis/{id}	object
GET	/v1/audio-features/{id}	audio features
GET	/v1/audio-features	audio features
GET	/v1/tracks	tracks
GET	/v1/tracks/{id}	track
GET	/v1/users/{user_id}	user

Sumber : <https://developer.spotify.com/documentation/web-API/reference/>

### 2.1.3 Lagu

Lagu merupakan seni nada atau suara dalam urutan, kombinasi, dan hubungan temporal dan biasanya di iringi dengan alar musik untuk menghasilkan irama yang mempunyai kesatuan dan kesinambungan. Lagu dapat dinyanyikan secara solo, duet, ataupun grup vokal. Lagu memiliki beragam jenis, mulai dari *genre* seperti rock, pop, metal, dan juga banyak lagi. Setiap *genre* memiliki ciri khas biasanya dari irama dan juga alat musik yang di pakai untuk mengiringi lagu tersebut, seperti lagu ber *genre* rock yang memiliki irama yang keras dan tinggi selain itu alat musik yang digunakan biasanya terdiri dari gitar elektrik, drum, dan bass.

Pada jaman ini lagu dapat dinikmati dengan berbagai cara karna rekaman lagu sudah berbentuk digital yang memudahkan untuk di putar melalui *smartphone*, komputer, atau barang elektronik lainnya yang dapat memutar lagu berbentuk digital. Berbeda pada jaman dahulu rekaman lagu masih menggunakan piringan hitam, atau pada kaset sehingga untuk memutarnya kita memerlukan alat khusus sesuai dengan media tempat rekaman lagu tersebut tersimpan.

Para peneliti dari McGill University di Montreal, Amerika Serikat, mengatakan bahwa mendengarkan musik membuat *Mood* kita terpengaruh karena musik membuat tubuh menghasilkan dopamin. Meningkatnya dopamin di dalam otak yang merespon musik membuktikan bahwa manusia mendapatkan kesenangan setelah mendengarkan musik [11]. Ada beberapa *genre* lagu yang dapat mempengaruhi *Mood* sebagai berikut [12]:

1. Blues

Musik blues dapat membantu melepaskan lelah. Ritme musik blues dapat sedikit memperlambat detak jantung dan membuat perasaan tenang.

2. Rock & punk

Musik ini enerjik dan bisa mendorong tingkat *Mood*. Menikmati musik rock dan punk, adrenalin dan dorongan energi akan jauh meningkat.

3. Reggae

Genre atau jenis musik satu ini cocok bagi yang emosinya sering meledak-ledak. Reggae adalah salah satu genre musik yang bisa menenangkan. Beberapa orang bilang genre reggae memiliki ritme musik yang menyehatkan untuk jantung manusia, khususnya ketika mendengarkannya saat beristirahat.

4. Pop/rock mainstream

Mendengarkan musik yang mainstream di pasaran, seperti di TV, radio, internet, baik itu pop atau rock bisa membantu mendapat semangat. Tempo yang menenangkan dan ritme yang mudah dicerna membuat detak jantung normal dan nafas konsisten.

5. Dance

Lagu bergenre dance akan membuat kita merasa lebih baik dengan cepat. Tempo yang cepat dan tingginya tingkat energi yang dimiliki akan membuat tubuh bergoyang.

6. Musik klasik

Genre musik ini bisa meningkatkan *Mood* seseorang sekaligus menenangkan. Musik jenis ini juga disebut bisa mempengaruhi tingkat kemampuan otak, oleh karena itulah musik klasik sering digunakan di sekolah dan juga di rumah sakit.

7. Rap & hip-hop J

Jenis musik satu ini bisa mengubah *Mood* dalam beberapa cara tergantung dari lirik lagunya, tempo musiknya, dan keseluruhan pesan yang terkandung dalam lagunya sendiri. Lagu rap sering membawa pesan tentang kemarahan, keangkuhan, dan kekerasan.

8. Heavy metal

Lirik dan melodi yang agresif bisa mempengaruhi dengan efek tertentu pada *Mood*, bisa berupa pemberontakan, bisa juga semangat.

9. Country

Musik country berhubungan dengan depresi dan bahkan bunuh diri. Jika sedang depresi musik ini tidak direkomendasikan untuk didengarkan karena akan membuat merasa makin kesepian.

10. New Age

*New Age* lebih banyak diisi dengan instrumen dan berbagai macam suara. Menggunakan genre musik ini membawa kepada ketenangan dan tepat untuk meditasi dan relaksasi.

#### 11. Broadway

Mendengarkan lagu-lagu dari *soundtrack* teater dan musikal bisa sangat memotivasi. Musik ini memberikan inspirasi dan kepekaan soal kreativitas.

#### 12. Latino Musik

latin bersifat mendorong energi. Musik latin bisa mempengaruhi jadi lebih semangat, merasa sexy atau macho, sekaligus menghilangkan gejala-gejala kelelahan.

#### 13. Meditasi

Yang termasuk dalam genre musik satu ini adalah lagu-lagu yang bernuansa menenangkan, seperti lagu-lagu akustik atau folk. Dengan mendengarkan lagu ini bisa mengurangi tingkat stres dan kegelisahan dan meningkatkan rasa tenang dan relaksasi.

### 2.1.4 Mood

Emosi (*Emotions*) adalah perasaan intens yang diarahkan pada seseorang atau sesuatu. *Mood (Moods)* adalah perasaan yang kurang intens dibandingkan emosi, yang tidak diarahkan, dan terkadang muncul tanpa ada kejadian yang menstimuli [13]. keadaan emosional yang bersifat sementara yang artinya *Mood* dapat berubah – ubah seiring waktu. *Mood* terbagi menjadi *Mood* positif dan juga *Mood* negatif dan masing-masing memiliki jenisnya. *Mood* sangat mempengaruhi keadaan seseorang terutama dalam menjalani kegiatan sehari – hari, untuk menjalani kegiatan berat tentu saja seseorang harus dalam keadaan *Mood* yang sangat bagus agar bersemangat dalam menyelesaikan kegiatan tersebut terutama kegiatan yang mengandalkan otak dan menuntut untuk berpikir keras seperti sedang belajar atau melakukan pekerjaan analisis agar kegiatan tersebut dapat terselesaikan dengan baik.

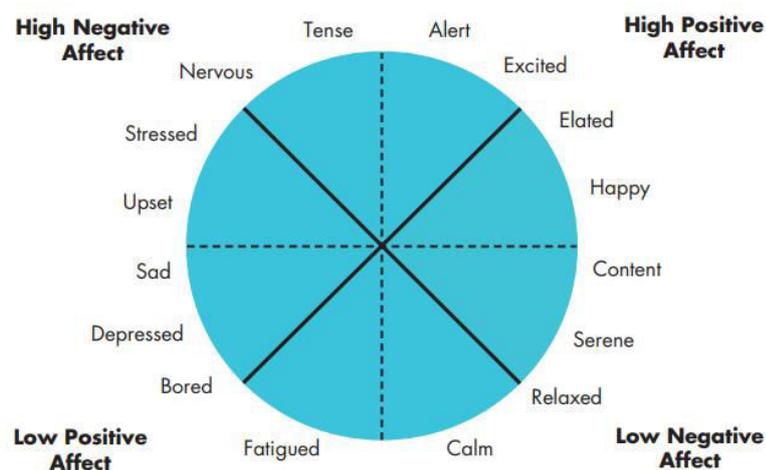
Emosi muncul dan hilang dalam waktu lebih sekejap dibandingkan *Mood*. Emosi dapat mempengaruhi dan merubah *Mood*, begitu pula *Mood* dapat membuat individu lebih emosional saat menanggapi kejadian. Perasaan pada *Mood* tidak dapat dikaitkan pada satu kejadian manapun. Seseorang yang mengalami suatu *Mood*, berada dalam kondisi bukan dirinya yang biasanya.

#### 1. Basic Emotions

Penamaan terhadap berbagai emosi memiliki jumlah yang sangat banyak. Rene Descartes mengidentifikasi enam gairah sederhana *wonder*, *love*, *hatred*, *desire*, *joy*, dan *sadness*, dan berpendapat emosi lainnya tersusun atas kombinasi keenamnya atau sejenis dengan keenamnya.

#### 2. Basic Moods

Emosi dapat diklasifikasikan secara positif atau negatif, namun tidak termasuk secara netral. Dengan klasifikasi ini, maka berbagai emosi dapat disatukan menjadi suatu konsep umum, yaitu posisi *Mood* (*Mood state*). Posisi *Mood* terdiri atas 2 dimensi. Pertama rasa positif (*positive affect*) adalah dimensi *Mood* yang terdiri atas emosi positif, seperti kegembiraan (*joy*) atau syukur (*gratitude*). Kedua rasa negatif (*negative affect*) adalah dimensi *Mood* yang terdiri atas emosi negatif, seperti marah (*anger*) atau rasa bersalah (*guilt*). Klasifikasi *Mood* untuk lebih jelasnya dapat dilihat pada Gambar 2.2.



Sumber : e-book - "organizational behaviour" [13].

### Gambar 2.2 Klasifikasi *Mood*

Emosi negatif cenderung berubah menjadi *Mood* negatif. Hal ini disebabkan beberapa hal:

- a. Pertama individu mengingat sebuah kejadian negatif lima kali lebih sering dibanding kejadian positif
- b. Kedua kejadian negatif lebih unik dan jarang ditemui.
- c. Ketiga *Mood* positif lebih sering dirasakan seseorang. Menurut konsep *positivity offset*, pada kondisi biasa dan tak terjadi apa-apa, seseorang akan mengalami *Mood* yang sedikit positif.

### 3. Fungsi Emosi

Rasionalitas dan emosi saling bertentangan satu sama lain, dengan menunjukkan emosi, kita akan cenderung berpikir secara irasional.

### 4. Sumber Emosi dan *Mood*

Emosi dan *Mood* muncul karena pengaruh yang datang dari berbagai sumber sebagai berikut :

- a. Kepribadian
- b. Hari dan Periode Waktu
- c. Cuaca
- d. Stres
- e. Aktivitas Sosial
- f. Istirahat
- g. Kebugaran
- h. Umur
- i. Jenis Kelamin

### 2.1.5 Aktivitas

Aktivitas dalam KBBI memiliki arti kegiatan atau kegiatan kerja yang dilaksanakan dalam tiap bagian tertentu pada sebuah organisasi yang dapat dilakukan oleh individu atau kelompok [12]. Sedangkan menurut Tjokroamudjo mengemukakan bahwa "Aktivitas adalah usaha-usaha yang dikemukakan untuk melaksanakan semua rencana dan kebijaksanaan yang telah

ditetapkan untuk melengkapi segala kebutuhan alat-alat yang diperlukan, siapa yang akan melaksanakan, ditempat mana pula dan berakhir, dan bagaimana cara yang harus dilaksanakan”.

Pada penelitian ini aktivitas yang dimaksud lebih menuju kepada aktivitas psikis. Secara psikologi, aktivitas adalah sebuah konsep yang mengandung arti fungsi individu dalam interaksinya dengan sekitarnya. Aktivitas psikis merupakan hubungan khusus dari benda hidup dan lingkungan hidup. Aktivitas psikis ini didorong oleh kebutuhan yang diarahkan pada objek yang dapat memenuhi kebutuhan tersebut [14].

Aktivitas psikis manusia terbagi menjadi dua yaitu aktivitas psikis eksternal dan aktivitas internal. Aktivitas psikis eksternal terdiri dari operasi manusia yang spesifik dengan objek-objek disekitarnya artinya terdapat interaksi dengan lingkungan sekitarnya yang melibatkan objek sekitar seperti contoh berolahraga, belajar, makan dan lain-lain. Sedangkan aktivitas psikis internal timbul atas aktivitas eksternal, dan merealisasikannya terhadap aktivitas eksternal seperti contoh aktivitas berpikir.

### **2.1.6 Android**

Android merupakan suatu sistem operasi mobile yang berbasis pada sistem operasi Linux. Android pertama kali dikembangkan oleh perusahaan startup di California bernama Android, Inc., yang digawangi oleh Andy Rubin, Rich Miner, Nick Sears, dan Chris White. Pada 2005, Google membeli Android dan mengambil alih proses pengembangannya hingga saat ini. Google merilis versi beta Android SDK (System Development Kit) pada November 2007 Android menawarkan pendekatan yang menyeluruh dalam pengembangan aplikasi. Artinya, satu aplikasi Android yang dibangun dapat berjalan di berbagai perangkat yang menggunakan sistem operasi Android baik itu smartphone, smartwatch, tablet, dan perangkat lainnya. Perkembangan teknologi Android yang begitu pesat juga tidak dapat dilepas dari peranan AOSP (Android Open Source

Project) yang bertanggung jawab dalam pengembangan sistem operasi Android dan dipimpin langsung oleh Google [15].

#### 1. Versi – versi Android

Dalam perkembangnya android di tahun 2019 ini sudah memiliki banyak versi.

Versi – versi Android tersebut dapat dilihat pada Tabel 2.4 dan Tabel 2.5.

**Tabel 2.4 Versi – versi Android**

Versi Android	API Level	Tanggal Rilis	Codename
1.0	1	23 September 2008	-
1.1	2	9 Februari 2009	-
1.5	3	30 April 2009	Cupcake
1.6	4	15 September 2009	Donut
2.0/2.0.1/2.1	5/6/7	26 Oktober 2009	Eclair
2.2 - 2.2.3	8	20 Mei 2010	Froyo
2.3 – 2.3.2/2.3.3-2.3.7	9/10	6 Desember 2010	Gingerbread
3.0/.3.1/3.2-3.2.6	11/12/13	22 Februari 2011	Honeycomb
4.0-4.0.2/4.0.3-4.0.4	14/15	19 Oktober 2011	Ice Cream Sandwaich
4.1-4.1.2/4.2-4.2.2/4.3-4.3.1	16/17/18	27 Juni 2012	Jelly Bean
4.4-4.4.4/4.4w	19/20	31 Oktober 2013	Kitkat
5.0-5.0.2/5.1-5.1.1	21/22	12 November 2014	Lollipop
6.0-6.0.1	23	5 Oktober 2015	Marshmallow

Sumber Data : Buku - “Pemrograman Android dengan Android Studio IDE” [15].

**Tabel 2.5 Versi – Versi Android**

Versi Android	API Level	Tanggal Rilis	Codename
7.0-7.1	24/25	22 Agustus 2016	Nougat
8.0-8.1	26/27	21 Agustus 2017	Oreo
9.0	28	Agustus 2018	Pie

Sumber Data : Website - <https://developer.Android.com/studio/releases/platforms>.

#### 2. Fitur – fitur Android

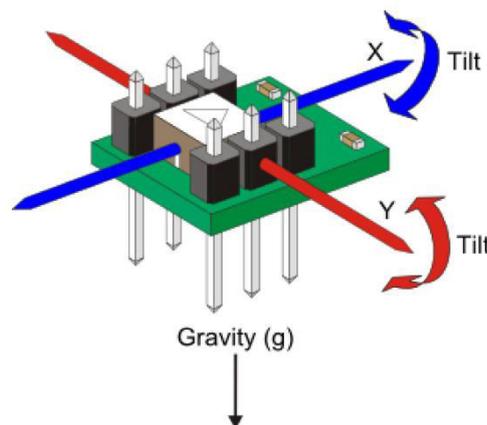
Android juga memiliki beberapa fitur perangkat keras di dalamnya yang dapat dimanfaatkan developer dalam mengembangkan aplikasi untuk Android seperti beberapa contoh berikut yang sering terdapat pada perangkat Android [16]:

- a. Touchscreen
- b. GPS
- c. Accelerometer
- d. SD Card
- e. Camera

### 2.1.7 Accelerometer

*Accelerometer* adalah sensor untuk mengukur percepatan suatu objek. Percepatan yang dapat di ukur bisa berupa percepatan dinamis pada objek bergerak dan percepatan statis terhadap gravitasi bumi. Cara kerja accelerometer ini adalah percepatan dari sebuah per yang memiliki beban dan di lepaskan dengan percepatan tertentu sampai kondisi berhenti. Untuk pendeteksian 3 dimensi maka diperlukan 3 pasang plat yang saling tegak lurus.

Fungsi dari *Accelerometer* antara lain seperti pada kamera untuk menstabilkan tangkapan gambar agar meminimalisir blur dari hasil tangkapan. Selain itu pada *Wearable Device* memiliki fungsi sebagai *pedometer* untuk menghitung langkah atau mendeteksi pergerakan tubuh yang terjadi. Gambar sensor *accelerometer* dapat dilihat pada gambar Gambar 2.3.



Sumber : e-journal – “VLC-Based Positioning System for an Indoor Environment Using an Image Sensor and an Accelerometer Sensor” [17].

**Gambar 2.3 Sensor Accelerometer**

Accelerometer memungkinkan pengukuran akselerasi dalam tiga sumbu tegak lurus, untuk merasakan kemiringan, gerakan, guncangan, dan getaran yang biasanya terdapat pada ponsel pintar, tablet, komputer perifer, antarmuka manusia-mesin, perangkat augmented reality, jaringan sensor di mana-mana, dll . mengukur kemiringan maju dan mundur pada sumbu x, dan mengukur sisi kemiringan ke sisi pada sumbu y [17].

### **2.1.8 Activity Recognition API**

Perangkat seluler telah menjadi bagian kehidupan sehari-hari bagi banyak orang bahkan mereka membawanya bersama sepanjang hari saat mereka mengemudi, berjalan, berolahraga, bekerja, dan bermain [18].

Memahami apa yang dilakukan pengguna di dunia fisik memungkinkan aplikasi menjadi lebih pintar tentang cara berinteraksi dengan pengguna. Misalnya, aplikasi dapat mulai melacak detak jantung pengguna ketika ia mulai berjalan, aplikasi lain dapat beralih ke mode berkendara ketika mendeteksi bahwa pengguna telah mulai mengemudi.

*Activity Recognition* API dibangun di atas sensor yang tersedia di perangkat. Sensor perangkat memberikan wawasan tentang apa yang sedang dilakukan pengguna. Namun, dengan lusinan sinyal dari berbagai sensor dan sedikit variasi dalam cara orang melakukan sesuatu, mendeteksi apa yang dilakukan pengguna tidaklah mudah.

*Activity Recognition* API secara otomatis mendeteksi aktivitas dengan secara berkala membaca ledakan singkat data sensor dan memprosesnya menggunakan model pembelajaran mesin. Untuk mengoptimalkan sumber daya, API dapat menghentikan pelaporan aktivitas jika perangkat diam selama beberapa saat, dan menggunakan sensor berdaya rendah untuk melanjutkan pelaporan saat mendeteksi pergerakan.

### 2.1.9 Bluetooth

Bluetooth adalah spesifikasi industri untuk jaringan kawasan pribadi (personal area networks atau PAN) tanpa kabel. Bluetooth menghubungkan dan dapat dipakai untuk melakukan tukar-menukar informasi di antara peralatan-peralatan. Spesifikasi dari peralatan Bluetooth ini dikembangkan dan didistribusikan oleh kelompok Bluetooth Special Interest Group. Bluetooth beroperasi dalam pita frekuensi 2,4 GHz dengan menggunakan sebuah frequency hopping traceiver yang mampu menyediakan layanan komunikasi data dan suara secara real time antara host-host bluetooth dengan jarak terbatas. Kelemahan teknologi ini adalah jangkauannya yang pendek dan kemampuan transfer data yang rendah.

Teknologi nirkabel Bluetooth akan berfungsi terutama sebagai pengganti inter koneksi kabel antara berbagai perangkat pribadi, termasuk komputer notebook, telepon seluler, personal digital assistant (PDA), kamera digital, dll [19]. Teknologi nirkabel Bluetooth bertujuan untuk melayani sebagai biaya rendah universal, ramah pengguna, yang akan menggantikan kebanyakan kabel eksklusif yang perlu dibawa dan digunakan orang untuk menghubungkan perangkat pribadi mereka. Fungsi *bluetooth* dapat dilihat pada Gambar 2.4

	audio streaming	data transfer	location services	device networks
solution				
topology	 point-to-point	 point-to-point	 broadcast	 mesh
radio	Bluetooth BR/EDR	Bluetooth Low Energy		

Sumber : <https://www.bluetooth.com>

### **Gambar 2.4 Fungsi Bluetooth**

Bluetooth berada di dalam kecepatan transfer antara 1 Mbps dan bluetooth versi 2.0 mempunyai kecepatan yang lebih baik lagi yaitu mencapai 3 Mbps. Ketika jaringan antar perangkat bluetooth bertemu, maka akan membentuk personal area network sehingga terjadi transaksi atau pertukaran data antar perangkat tersebut, Tentunya dengan syarat persetujuan antar kedua perangkat Bluetooth.

#### **2.1.10 Sensor Heart Rate**

Sensor detak jantung optik mengukur denyut nadi, yang merupakan perubahan volume pembuluh darah yang terjadi ketika jantung memompa darah. Denyut nadi terdeteksi dengan mengukur perubahan volume menggunakan sensor optik dan LED hijau. Mengadopsi filter optik yang dioptimalkan untuk deteksi denyut nadi di blok sensor meminimalkan efek cahaya sekitar seperti sinar merah dan inframerah. Ini memungkinkan sinyal pulsa berkualitas tinggi diperoleh, bahkan di luar ruangan.

Selain itu, meningkatkan teknologi sensor optik yang dikembangkan selama bertahun-tahun memungkinkan ROHM untuk secara signifikan meningkatkan sensitivitas blok sensor. Dukungan untuk kecerahan rendah LED VF rendah memungkinkan untuk mencapai sistem pemantauan detak jantung optik daya rendah tanpa perlu sirkuit eksternal (mis. Sirkuit boost). Ini berkontribusi pada waktu pengoperasian yang lebih lama pada perangkat yang dapat dikenakan dengan kapasitas baterai yang terbatas [20].

Monitor detak jantung pada jaman modern umumnya menggunakan salah satu dari dua metode berbeda untuk mendeteksi denyut jantung. Kedua metode ini dapat memberikan data detak jantung dasar yang sama. Teknologi asli didasarkan pada sensor listrik dan ini masih merupakan standar yang digunakan untuk perangkat medis. Teknologi yang lebih baru didasarkan pada sensor optik.

- a. Sensor EKG (Elektrokardiografi) mengukur potensi bio yang dihasilkan oleh sinyal listrik yang mengontrol ekspansi dan kontraksi bilik jantung.

- b. Sensor PPG (Photoplethysmography) menggunakan teknologi berbasis cahaya untuk merasakan laju aliran darah yang dikendalikan oleh aksi pemompaan jantung.

### 2.1.11 Android Studio

Android studio adalah sebuah software tool Integrated Development Enviroment (IDE) untuk platform Android. Android studio ini diluncurkan pada tanggal 16 Mei 2013 pada Konferensi Google I/O oleh Produk Manajer Google, Ellie Powers [16]. Logo Android Studio dapat dilihat pada Gambar 2.5.



Sumber Gambar : <https://developer.android.com>

**Gambar 2.5 Android Studio**

Android studio ini bersifat free dibawah Apache License 2.0. Android Studio awalnya dimulai dengan versi 0.1 pada bulan mei 2013, dan dibuat versi beta 0.8 yang di rilis pada bulan juni 2014. Android Studio ini didesain khusus untuk android development dan berbasiskan JetBrains' IntelliJ IDEA.

Android studio memiliki fitur yang sangat membantu developer untuk membangun aplikasi android berikut beberapa fitur Android Studio yang di ambil dari situs resmi di <https://developer.android.com/studio> .

#### 1. Visual layout editor

Fitur ini memungkinkan untuk membuat tata letak yang rumit dengan ConstraintLayout dengan menambahkan batasan dari setiap tampilan ke tampilan lainnya. Kemudian pratinjau tata letak dapat disesuaikan pada ukuran

layar apa pun dengan memilih salah satu dari berbagai konfigurasi perangkat atau hanya dengan mengubah ukuran jendela pratinjau.

## 2. APK Analyzer

APK Analyzer memungkinkan developer memeriksa konten file APK, bahkan jika APK tersebut tidak dibuat dengan Android Studio. Seperti memeriksa file manifest, resources, dan file DEX.

## 3. Emulator

Android studio memiliki fitur emulator bawaan sehingga kita dapat langsung melakukan uji coba aplikasi yang telah dibuat langsung dari perangkat komputer tanpa perlu perangkat fisik Android.

## 4. Flexible Build System

Didukung oleh Gradle, sistem build Android Studio memungkinkan untuk menyesuaikan build untuk menghasilkan beberapa varian build untuk perangkat yang berbeda dari satu proyek Aplikasi Android.

## 5. Realtime Profilers

Fitur ini menyediakan statistik waktu nyata untuk CPU, memori, dan aktivitas jaringan aplikasi. Identifikasi kinerja dengan merekam jejak metode, memeriksa tumpukan dan alokasi, dan melihat muatan jaringan internet yang masuk dan keluar.

### **2.1.12 Java**

Java sebagai salah satu bahasa pemrograman yang sudah berumur dari era 1990-an, kian berkembang dan melebarkan dominasinya di berbagai bidang. Salah satu penggunaan terbesar Java adalah dalam pembuatan aplikasi native untuk Android [21].

Java merupakan bahasa pemrograman yang dapat berjalan di berbagai platform terutama komputer berbasis web atau dekstop dan juga mobile pada Android. Bahasa java dibuat pada awalnya oleh James Gosling. Bahasa ini banyak mengadopsi sintak yang terdapat pada bahasa C dan C++. Selain itu java

merupakan bahasa pemrograman yang berbasis Object Oriented Programming (OOP). Logo java dapat dilihat pada Gambar 2.6.



Sumber Gambar : <https://www.codepolitan.com/>

**Gambar 2.6 Logo Java**

## 1. Kelebihan Java

Bahasa java memiliki kelebihan sebagai berikut :

- a. Multiplatform
- b. OOP
- c. Kelas library yang lengkap
- d. Pengembangan mudah dilakukan
- e. Penyusunan script sangat mudah
- f. Dinamis

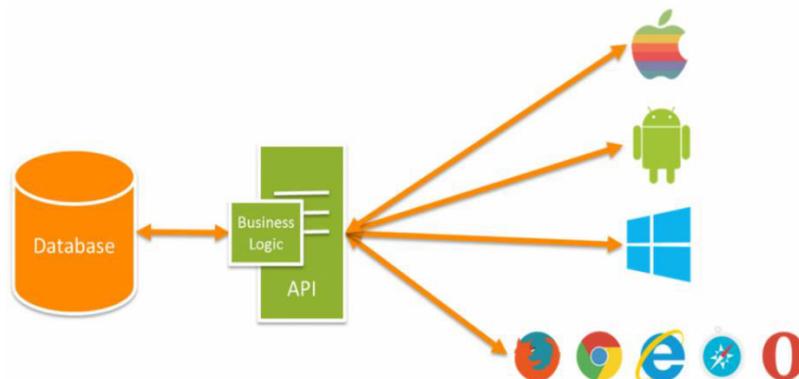
## 2. Kekurangan Java

Selain itu bahasa java juga memiliki kekurangan sebagai berikut :

- a. Mudah di dekompile
- b. Penggunaan memori yang banyak
- c. Butuh di compile terlebih dahulu

### 2.1.13 API

API (*Application Programming Interface*) merupakan bagian besar dari web. Di tahun 2013 ada lebih dari 10.000 API yang diterbitkan oleh perusahaan yang di buka untuk publik [22]. Fungsi dari API ini pada umumnya adalah sebagai sumber data yang bisa digunakan untuk kebutuhan sistem atau aplikasi tertentu, API memungkinkan untuk dapat bertukar informasi data dari satu aplikasi ke aplikasi lainnya. Untuk lebih jelas gambaran dari API bisa dilihat pada Gambar 2.7.



Sumber gambar : <https://www.codepolitan.com/>

### Gambar 2.7 Skema API

Selain sebagai sumber data API juga mempercepat proses *development* dengan menyediakan fungsi secara terpisah sehingga *developer* tidak perlu membuat fungsi yang sama. API sendiri terdiri dari beberapa elemen seperti *functions*, *protocols*, dan *tools* untuk membantu *developer* membuat aplikasi. Apabila membangun aplikasi yang sangat kompleks penggunaan API ini akan sangat terasa sangat membantu sekali contoh dalam pembuatan aplikasi yang membutuhkan integrasi *payment gateway* peran API disini sangat di utamakan.

Untuk melakukan *request* kepada API *client* harus memenuhi beberapa syarat tertentu sebagai berikut :

1. URL

Dalam HTTP (*Hyper Text Transfer Protocol*) URL adalah alamat unik dari suatu web atau sistem tertentu dan akan di arahkan sesuai proses bisnis yang ada pada server. URL dapat dibuat berupa halaman sebuah website, gambar, video, atau data tertentu.

2. Method

*Request method* yang di lakukan oleh *client* memberitahu server bahwa aksi apa yang perlu di lakukan untuk memenuhi permintaan *client*. Ada 4 method yang biasanya digunakan yaitu :

- a. GET – meminta server untuk memberikan respon sumber.
- b. POST – meminta server untuk membuat sumber baru.
- c. PUT – meminta server untuk melakukan edit/update pada sumber yang sudah ada
- d. DELETE – meminta server untuk melakukan penghapusan sumber.

3. Headers

Hedaers biasanya berisi meta-informasi seperti waktu permintaan, format data yang diinginkan, atau informasi perangkat yang digunakan oleh *client* sehingga server dapat megirimkan data sesuai dengan perangkat yang digunakan.

4. Body

Body berisi data yang ingin dikirimkan client ke server, sehingga server dapat melakukan respon aksi berdasarkan data yang dikirimkan client ke server.

#### **2.1.14 MySQL**

MySQL merupakan software RDBMS (Relational Database Management System atau server database) yang dapat mengelola database dengan sangat cepat, dapat menampung data dalam jumlah sangat besar, dapat diakses oleh banyak user (multi-user), dan dapat melakukan suatu proses secara sinkron atau berbarengan (multi-threaded). MySQL merupakan software yang bersifat open source. Database sendiri merupakan sekumpulan tabel-tabel yang saling berhubungan. Sedangkan open source sendiri menyatakan bahwa software ini dilengkapi dengan source code (kode yang dipakai untuk membuat MySQL). Selain tentu saja bentuk executable-nya atau kode yang dapat dijalankan secara langsung dalam sistem operasi. Beberapa alasan memilih MySQL sebagai server database untuk aplikasi yang dikembangkan.

1. Fleksibel MySQL dapat digunakan untuk mengembangkan aplikasi dekstop maupun aplikasi web dengan menggunakan teknologi yang bervariasi.
2. Performa Tinggi MySQL memiliki mesin query dengan performa tinggi, dengan demikian proses transaksional dapat dilakukan dengan sangat cepat. Hal ini terbukti dengan digunakannya MySQL sebagai database dari beberapa aplikasi web yang memiliki traffic (lalu lintas) sangat tinggi.
3. Lintas Platform MySQL dapat digunakan pada platform atau lingkungan (dalam hal ini sistem operasi) yang beragam, bisa Microsoft Windows, Linux, atau UNIX. Ini menyebabkan proses migrasi data antar sistem operasi dapat dilakukan secara lebih mudah. Misal pada saat ingin mengganti sistem operasi pada mesin server.
4. Gratis MySQL dapat digunakan secara gratis. Meskipun demikian, ada juga software MySQL yang bersifat komersial. Biasanya yang sudah ditambahi dengan kemampuan spesifik dan mendapat pelayanan dari technical support.

5. Proteksi Data yang Andal Perlindungan terhadap keamanan data merupakan hal nomor satu yang dilakukan oleh para profesional di bidang database. MySQL menyediakan mekanisme yang powerful untuk menangani hal tersebut yaitu dengan menyediakan fasilitas manajemen user, enkripsi data dan lain sebagainya
6. Komunitas Luas Dikarenakan pengguna MySQL banyak sehingga MySQL memiliki komunitas yang luas. Hal ini sangat berguna apabila sedang menemui suatu permasalahan dalam proses pengolahan data menggunakan MySQL, sehingga dapat mendiskusikan langsung permasalahan tersebut melalui sebuah forum agar mendapatkan solusi.

a. DDL

Merupakan kumpulan perintah SQL yang digunakan untuk membuat, mengubah dan menghapus struktur dan definisi metadata dari objek-objek database [23].

Objek – objek yang dimaksud adalah sebagai berikut :

1. Database
2. Table
3. View
4. Index
5. Procedure (Stored Procedure)
6. Function
7. Trigger

Perintah – perintah DDL pada MySQL dapat dilihat pada Tabel 2.6.

**Tabel 2.6 Tabel Matrix Perintah DDL pada MySQL**

<i>OBJECT</i>	<i>CREATE</i>	<i>ALTER</i>	<i>DROP</i>	<i>RENAME</i>
<i>DATABASE</i>	Ya	Ya	Ya	
<i>FUNCTION</i>	Ya	Ya	Ya	
<i>INDEX</i>	Ya		Ya	
<i>PROCEDURE</i>	Ya	Ya	Ya	
<i>TABLE</i>	Ya	Ya	Ya	Ya
<i>TRIGGER</i>	Ya		Ya	
<i>VIEW</i>	Ya	Ya	Ya	

Sumber data : <http://mysql.phi-integration.com>

b. DML

Merupakan kumpulan perintah SQL yang digunakan untuk proses pengolahan isi data di dalam tabel seperti memasukkan, merubah dan menghapus isi data - dan tidak terkait dengan perubahan struktur dan definisi tipe data dari objek database.

Beberapa perintah DML adalah sebagai berikut :

1. *Call*
2. *Delete*
3. *Do*
4. *Handler*
5. *Insert*
6. *Load data infile*
7. *Replace*
8. *Select*
9. *Truncate*
10. *Update*

### 2.1.15 UML

*Unified Modeling Language* (UML) adalah sketsa yang biasa digunakan developer untuk membantu menyampaikan beberapa aspek dari sebuah sistem [24]. Sama seperti *blueprint* sketsa dapat digunakan dalam sebuah *forward engineering* yang artinya diargam UML ini dibuat sebelum menuliskan kode atau *reverse engineering* yaitu membuat diagram berdasarkan kode yang sudah ada untuk membantu memahami kode tersebut.

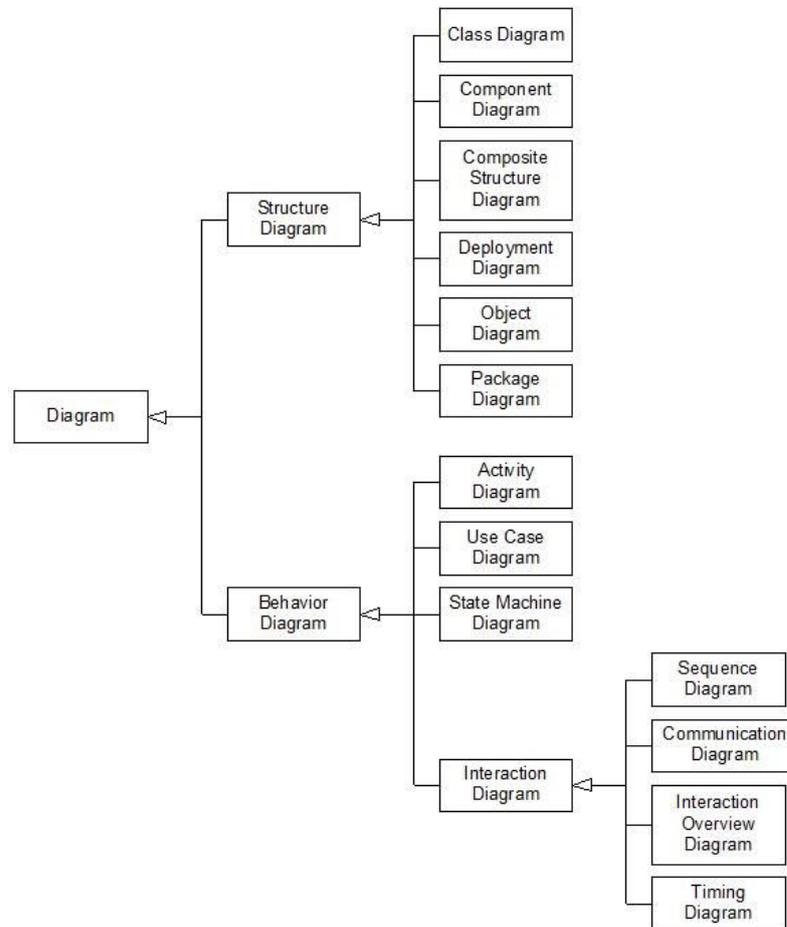
Diagram UML terdiri dari 13 jenis diagram resmi dan memiliki klasifikasi tersendiri, jenis diagram tersebut dapat dilihat pada Tabel 2.7 dan klasifikasi jenis diagram uml dapat dilihat pada Gambar 2.8 Klasifikasi jenis – jenis diagram UML.

**Tabel 2.7 Jenis – jenis Diagram resmi UML**

Diagram	Kegunaan	Turunan
---------	----------	---------

Diagram	Kegunaan	Turunan
Activity	Behavior prosedural dan paralel	Di UML 1
Class	Class, fitur, dan hubungan-hubungan	Di UML 1
Communication	Interaksi antar objek, penekanan pada jalur	Diagram kolaborasi UML 1
Component	Struktur dan koneksi komponen	Di UML 1
Composite Structure	Dekomposisi runtime sebuah class	Baru di UML 2
Deployment	Pemindahan artifak ke node	Di UML 1
Interaction overview	Campuran sequence dan activity diagram	Baru di UML 2
Object	Cintih konfigurasi dari contoh-contoh	Tidak resmi di UML 1
Package	Struktur hirarki compile-time	Tidak resmi di UML 1
Sequence	Interaksi antar objek; penekanan pada sequence	Di UML 1
State machine	Bagaimana even mengubah objek selama aktif	Di UML 1
Timing	Interaksi antar objek; penekanan pada timing	Baru di UML 2
Use case	Bagaimana pengguna berinteraksi dengan sebuah sistem	Di UML 1

Sumber : Buku – “UML Distilled” [15].

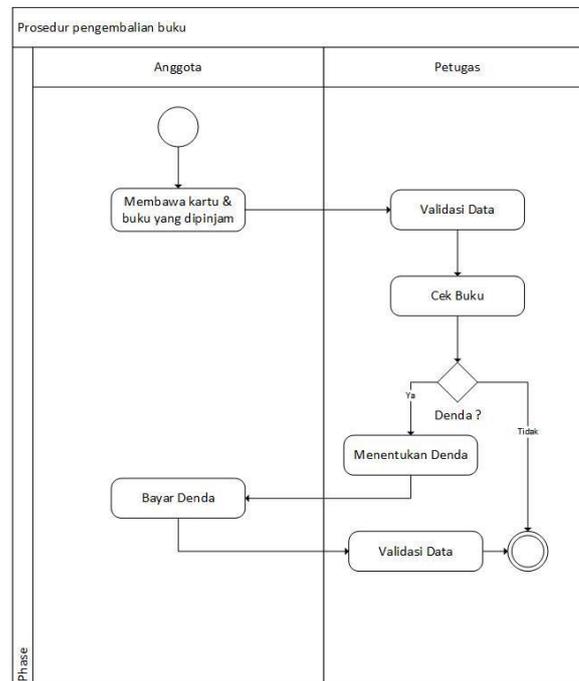


**Gambar 2.8** Klasifikasi jenis – jenis diagram UML

Berikut penjelasan dari setiap jenis – jenis diagram UML :

#### 1. Activity Diagram

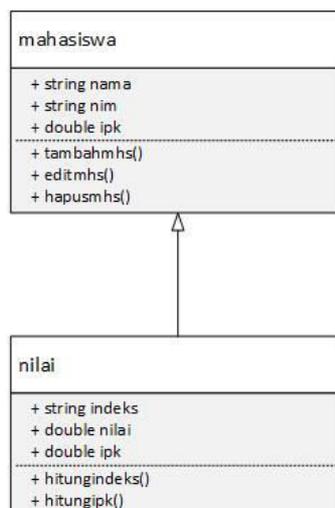
Activity diagram merupakan teknik dalam menggambarkan logika, prosedural, proses bisnis, dan alur kerja. Activity diagram ini telah mengalami beberapa perubahan dari perkembangan versi – versi UML. Dalam UML 1, activity diagram merupakan kasus khusus state diagram. Sementara pada UML 2 ikatan tersebut sudah dihilangkan. Contoh activity diagram dapat dilihat pada Gambar 2.9.



**Gambar 2.9 Contoh Activity Diagram**

## 2. Class

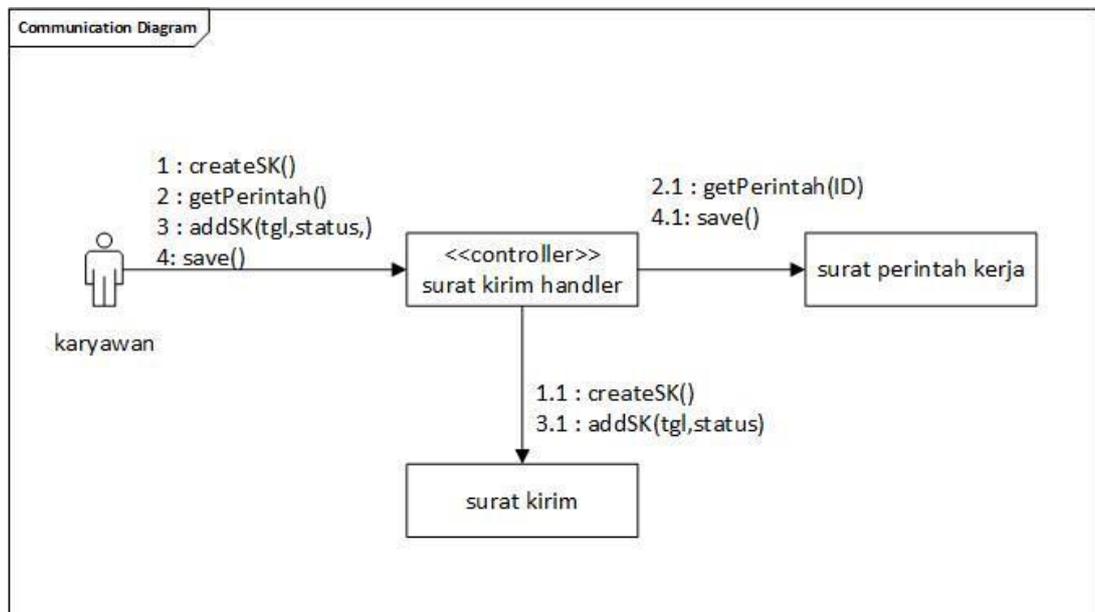
Class diagram merupakan pendeskripsian jenis-jenis dari objek di dalam sistem dengan berbagai macam hubungan statis yang terdapat mereka. Properti dan operasi pada sebuah class juga batasan – batasan yang ada dalam hubungan objek-objek dapat ditunjukkan oleh class diagram juga. Contoh class diagram dapat dilihat pada Gambar 2.10.



**Gambar 2.10 Contoh Class Diagram**

### 3. Communication

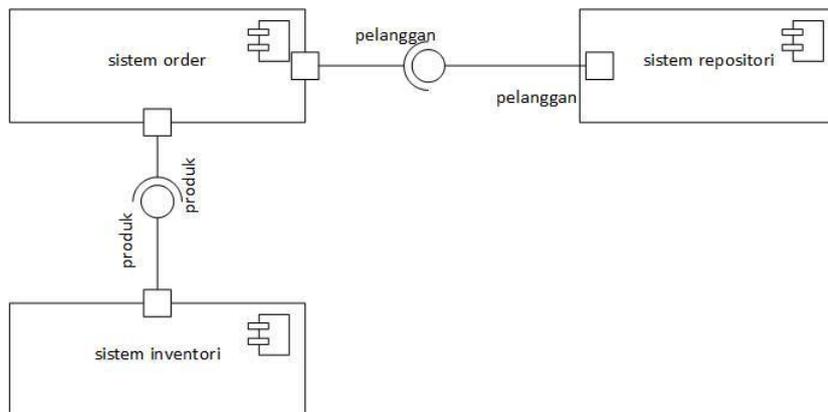
*Communication* diagram tidak menggambarkan setiap partisipan sebagai sebuah garis alir yang menunjuka pesan-pesan tertentu dengan arahnya yang vertikal seperti pada sequence diagram, diagram ini memungkinkan agar dapat menempatkan partisipan-partisipan secara bebas. Contoh *communication* diagram dapat dilihat pada Gambar 2.11.



**Gambar 2.11 Contoh Communication Diagram**

### 4. Component

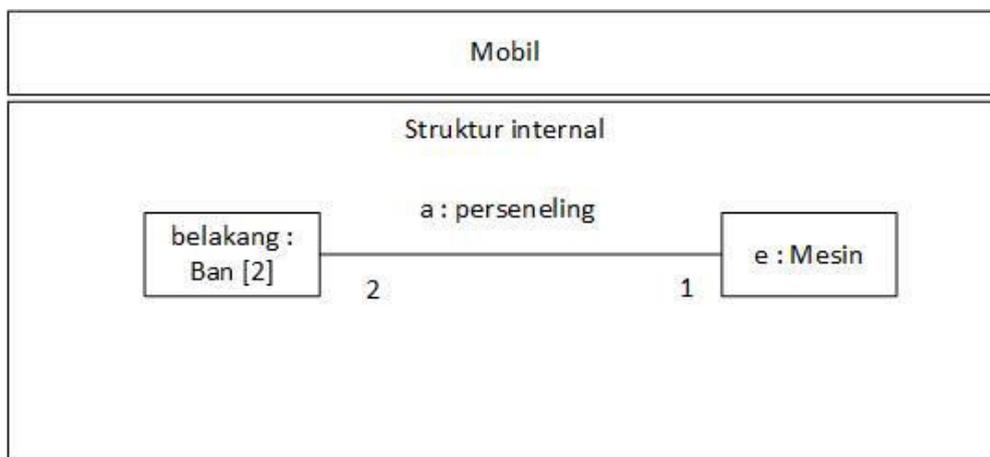
Pada versi-versi awal UML, komponen digunakan untuk mewakili struktur fisik seperti DLL. Komponen merupakan bagian – bagian yang bisa dibeli secara mandiri juga dapat di upgrade. Component diagram dapat dilihat pada Gambar 2.12.



**Gambar 2.12 Contoh Component Diagram**

### 5. Composite Structure

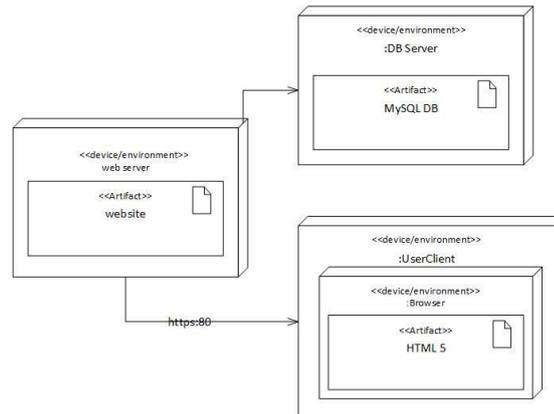
Composite structure merupakan salah satu fitur baru pada UML 2. Untuk membedakan antara paket-paket dengan kelompok yang sedang berjalan dan composite structure adalah cara terbaik karna disesuaikan untuk menunjukan komponen juga bagaimana dipecah menjadi bagian-bagian. Contoh composite diagram dapat dilihat pada Gambar 2.13.



**Gambar 2.13 Contoh Composite Diagram**

## 6. Deployment

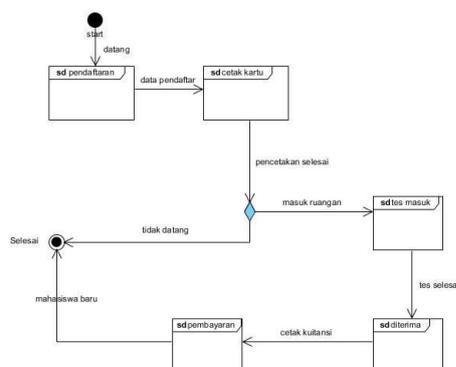
Diagram ini menunjukkan susunan fisik dari sebuah sistem, menunjukkan bagian perangkat lunak yang mana yang berjalan pada bagian perangkat keras. Deployment diagram ini sangat sederhana. Deployment diagram dapat dilihat pada Gambar 2.14.



**Gambar 2.14 Contoh Deployment Diagram**

## 6. Interaction Overview

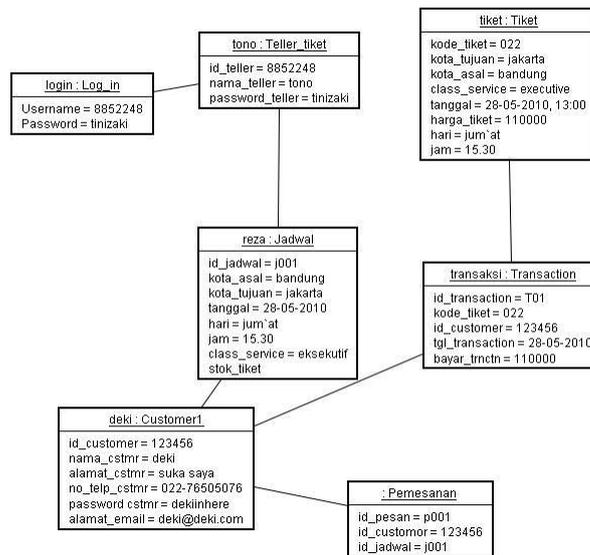
Ini merupakan hasil pencakokan dari activity diagram dan sequence diagram. *Interaction overview* juga bisa dianggap baik sebagai diagram activity dengan activity yang diganti oleh sequence diagram. Contoh diagram *interaction overview* dapat dilihat pada Gambar 2.15.



**Gambar 2.15 Contoh Interaction Overview**

## 7. Object

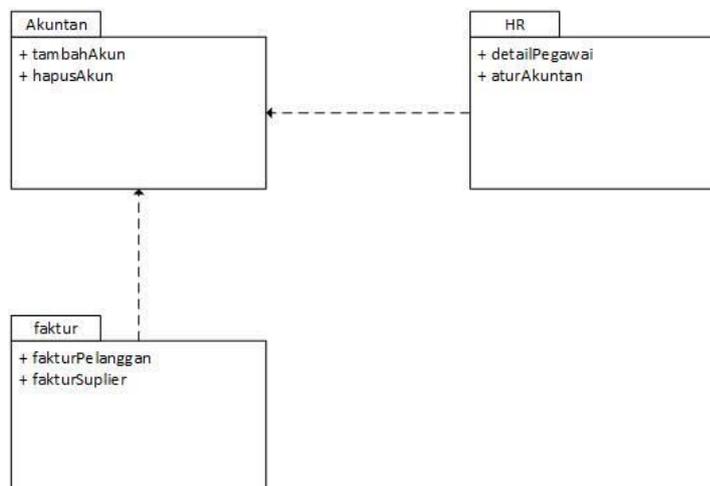
*Object* diagram digunakan untuk menunjukkan sebuah konfigurasi contoh dari objek yang saling berhubungan. *Object* diagram dapat dilihat pada Gambar 2.16.



**Gambar 2.16** Contoh Object Diagram

## 8. Package

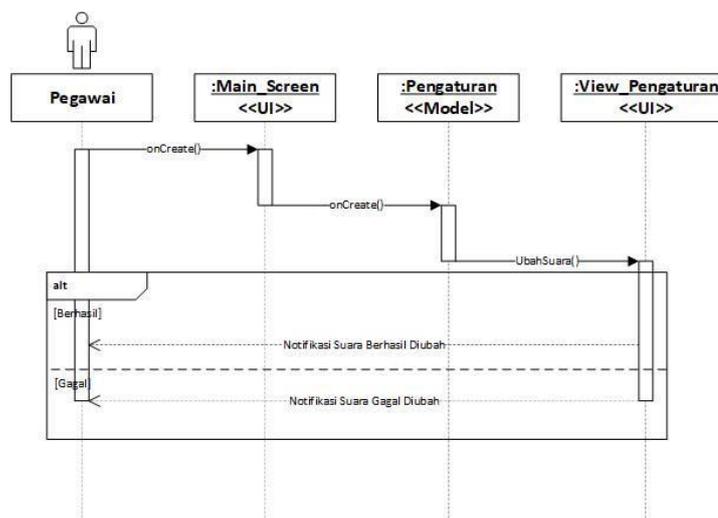
*Package* adalah sebuah bentuk pengelompokan yang memungkinkan untuk mengambil setiap bentuk di UML. Yang paling umum adalah untuk mengelompokkan *class*. *Package* diagram dapat dilihat pada Gambar 2.17.



**Gambar 2.17 Contoh Package Diagram**

9. Sequence

*Sequence* diagram secara khusus menjabarkan *behavior* sebuah skenario tunggal. Diagram tersebut menunjukkan sejumlah objek contoh dan juga pesan-pesan yang melewati objek di dalam *use case*. *Sequence* diagram dapat dilihat pada Gambar 2.18.



**Gambar 2.18 Contoh Sequence Diagram**

10. State Machine

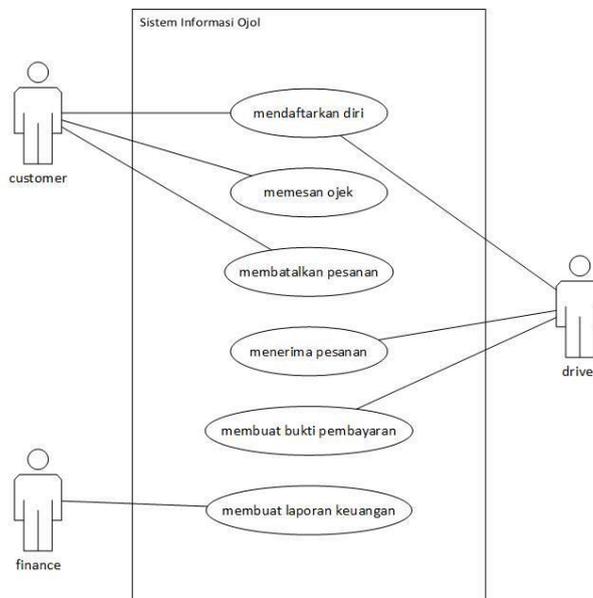
*State machine* diagram biasa digunakan untuk menggambarkan *behavior* sebuah sistem. *State machine* diagram dapat dilihat pada Gambar 2.19.



**Gambar 2.19 Contoh State Machine Diagram**

## 11. Use Case

*Use case* digunakan untuk merekam persyaratan fungsional sebuah sistem. *Use case* juga mendeskripsikan interaksi yang tipikal antara pengguna sistem dengan sistem itu sendiri dengan cara memberikan sebuah narasi bagaimana sistem tersebut digunakan. *Use case* diagram dapat dilihat pada Gambar 2.20.



**Gambar 2.20 Contoh Use Case Diagram**