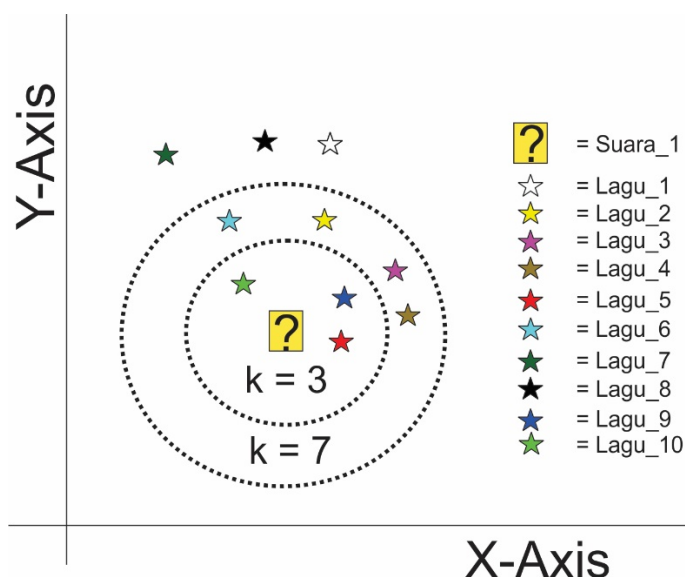


## BAB II TINJAUAN PUSTAKA

### 2.1. *K-Nearest Neighbor*

*K-Nearest Neighbor* adalah suatu metode yang menggunakan algoritma *supervised learning* dimana hasil dari *instance* yang baru diklasifikasikan berdasarkan mayoritas dari kategori  $k$  tetangga terdekat. *K-nearest neighbor* juga dapat dijelaskan sebagai algoritma yang klasifikasi objek berdasarkan data pembelajaran yang jaraknya paling dekat dengan objek yang akan di uji [8]. Dalam menentukan nilai atau kelas  $k$ , sebaiknya digunakan nilai ganjil, jika digunakan nilai genap terdapat kemungkinan kita tidak akan mendapatkan jawaban. Penentuan nilai  $k$  dipertimbangkan berdasarkan banyaknya data yang ada dan ukuran dimensi yang dibentuk oleh data. Semakin banyak data yang ada, angka  $k$  yang dipilih sebaiknya semakin rendah. Namun, semakin besar ukuran dimensi data, angka  $k$  yang dipilih sebaiknya semakin tinggi.



Gambar 2.1. Gambaran KNN

Untuk mencari jarak antar titik pada kelas  $k$  digunakan jarak *Euclidean*. Jarak *Euclidean* merupakan formula untuk mencari jarak antar 2 titik. Berikut merupakan rumus *Euclidean*.

$$\begin{aligned}
 d(p, q) &= d(q, p) \\
 &= \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + \dots + (q_n - p_n)^2} \\
 &= \sqrt{\sum_{i=1}^n (q_i - p_i)^2}
 \end{aligned} \tag{1}$$

Berikut merupakan langkah kerja *KNN*:

1. Menentukan parameter  $K$  (jumlah kelas)
2. Hitung jarak antara data uji dengan data latih yang telah ada menggunakan rumus *Euclidean*
3. Urutkan hasil hitung berdasarkan dari yang terkecil lalu tetapkan persekitaran sesuai dengan nilai  $K$
4. Gunakan kelas dengan jumlah terbanyak lalu tetapkan kelas data uji

Metode *K-Nearest Neighbor* digunakan dalam penelitian ini untuk mencari kedekatan nilai antara suara pengguna sebagai data uji dengan lagu sebagai data latih. Nilai terdekat akan menjadi lagu yang direkomendasikan untuk pengguna.

## 2.2. Lagu

Lagu merupakan seni nada atau suara kombinasi, dan hubungan temporal untuk menghasilkan gubahan musik yang mempunyai kesatuan dan kesinambungan. Lagu sudah ada sejak 180.000 hingga 100.000 tahun yang lalu. Ragam nada atau suara yang berirama disebut juga dengan lagu. Terdapat banyak genre dari sebuah lagu seperti pop, RnB, jazz, hip hop, rock, dan lain-lain.

Dalam sebuah lagu dapat diolah lagi untuk mengetahui melodi dari musik tersebut. Melodi juga dapat diketahui dari cara seseorang bergumam maupun melakukan siulan. Dari melodi tersebut dapat diketahui genre dari sebuah lagu.

Keuntungan dari sebuah identifikasi melodi adalah hanya dibutuhkan sebuah piano untuk dapat menentukan lagu tersebut memiliki genre apa [13].

Lagu dapat dinyanyikan secara solo, berdua (duet), bertiga (trio) atau dalam beramai-ramai (koir). Perkataan dalam lagu biasanya berbentuk puisi berirama, namun ada juga yang bersifat keagamaan ataupun prosa bebas. Lagu dapat dikategorikan pada banyak jenis, bergantung kepada ukuran yang digunakan. Pada saat ini, lagu mengalami perubahan dari segi lirik dan alunan musiknya. Lirik yang dulu bertemakan cinta, persahabatan, dan kehidupan kini dapat berubah menjadi kebebasan, pergaulan bebas, dan bahkan mengandung unsur seks dan konten eksplisit.

Berikut merupakan ciri-ciri lagu:

#### 1. Suara

suara dalam lagu menjelaskan bagaimana suara dinotasikan atau dituliskan. gelombang suara biasanya dibahas dalam bentuk frekuensi. aspek-aspek dasar suara dalam lagu dijelaskan dalam pitch, durasi, intensitas dan timbre.

#### 2. Nada

suara dibagi-bagi kedalam nada yang memiliki tinggi nada atau tala tertentu. Perbedaan tala antara dua nada disebut sebagai interval. Nada dapat diatur dalam tangga nada yang berbeda-beda. Nada dasar suatu lagu menentukan frekuensi tiap nada dalam karya tersebut.

#### 3. Melodi

Dalam arti yang paling harfiah, melodi adalah urutan nada dan jangka waktu nada, sementara, dalam arti lain, istilah tersebut memasukkan suksesi unsur musik lain seperti warna nada. Serangkaian nada dalam waktu. Rangkaian tersebut dapat dibunyikan sendirian

#### 4. Notasi

Notasi musik merupakan penggambaran tertulis atas musik. Dalam notasi balok, tinggi nada digambarkan secara vertikal sedangkan waktu (ritme) digambarkan secara horisontal. Kedua unsur tersebut membentuk paranada.

#### 5. Ritme

Tanda birama menunjukkan jumlah ketukan dalam birama dan not mana yang dihitung dan dianggap sebagai satu ketukan. nada-nada tertentu dapat diaksentuasi dengan pemberian tekanan

Berikut merupakan manfaat dari adanya lagu:

1. Media menghibur diri
2. sebagai alat untuk menenangkan diri
1. sebagai lintasan hobi atau bakat.

Lagu digunakan dalam penelitian ini karena merupakan hiburan yang dapat didengarkan oleh siapapun dan dimanapun berada. Lagu dapat didengarkan melalui banyak sekali media seperti *smartphone*, *mp3 player*, dan lain-lain.

### 2.3. Suasana Hati

Suasana hati dalam bahasa Inggris yang berarti *mood*, adalah suatu bentuk keadaan emosional yang relatif tahan lama. *Mood* berbeda dari emosi yang sederhana dimana emosi lebih bersifat spesifik, kurang intens, dan kurang mungkin dipicu oleh stimulus atau peristiwa tertentu [14]. suasana hati seseorang dapat berlangsung dalam jangka yang pendek serta hanya beberapa jam saja. Suasana hati dapat dipengaruhi oleh banyak kejadian yang tak terduga .

Suasana hati sangat berbeda dengan emosi karena suasana hati tidak harus disebabkan sesuatu hal. Seseorang dapat merasakan emosi bahagia setelah

menikah atau menerima hadiah, sementara *mood* bahagia merupakan tendensi untuk bereaksi terhadap rangsangan dari luar. Suasana hati cenderung bertahan lebih lama daripada emosi. ciri-ciri kepribadian seperti optimisme dan neurotisisme mempengaruhi jenis suasana hati tertentu. Gangguan *mood* jangka panjang seperti depresi klinis dan gangguan bipolar dianggap sebagai gangguan *mood*. *Mood* terjadi pada keadaan internal dan subyektif tetapi sering dapat disimpulkan dari postur dan perilaku lainnya.

Terdapat beberapa cara untuk mengetahui suasana hati manusia, seperti:

### 1. Isyarat Bahasa Tubuh

Penelitian menunjukkan bahwa kata-kata hanya 15% dari cara kita berkomunikasi sedangkan bahasa tubuh kita 55% dan nada suara 30% mewakili sisanya. penampilan dapat digunakan untuk acuan suasana hati seseorang, cara seseorang berpakaian menjadi salah satu acuannya. postur dan gerakan fisik seseorang saat berbicara terhadap orang lain dapat menjelaskan suasana hati orang tersebut

### 2. Intuisi lawan bicara

Intuisi merupakan perasaan lawan bicara saat terjadi komunikasi. perasaan ini memberikan informasi nonverbal yang lawan bicara terima melalui visual. disisi lain lawan bicara juga dapat memperhatikan empati intuitif seseorang. gejala fisik dan emosi orang lain di tubuh Anda yang merupakan bentuk empati yang kuat.

### 3. Merasakan energi suasana hati

Emosi adalah ekspresi menakjubkan dari energi kita, seperti sebuah getaran yang kita berikan. Beberapa orang merasa senang berada di perkumpulannya. mereka meningkatkan *mood* orang-orang sekitarnya sedangkan terdapat juga yang menurunkan *mood* seseorang. energi ini dapat dirasakan beberapa inci atau kaki dari tubuh, meskipun itu tidak terlihat. Dalam pengobatan Tiongkok, ini disebut *chi* yang merupakan vitalitas yang penting untuk kesehatan. kita berbagi energi emosional melalui kontak fisik seperti arus listrik. saat berjabat tangan orang berkeringat, menandakan kegelisahan

#### 4. Nada suara dan tertawa

Nada dan volume suara kita dapat memberi tahu banyak tentang emosi kita. Frekuensi suara menciptakan getaran. Saat memperhatikan seseorang, perhatikan bagaimana nada suara mereka memengaruhi orang lain. dari nada suara kita dapat mengetahui apakah orang tersebut sedang senang, marah, bahagia atau sedih.

Berikut merupakan penyebab perubahan suasana hati:

1. ketidakseimbangan kimia pada otak
2. kondisi hormon seseorang
3. gangguan mental
2. penyakit tertentu

#### **2.4. *Speech Emotion Recognition***

*Speech emotion recognition* merupakan teknologi untuk mengidentifikasi emosi dari ucapan manusia yang merupakan salah satu tantangan terbaru dalam pemrosesan bicara. Selain ekspresi wajah manusia, saat berbicara juga telah terbukti sebagai salah satu modalitas yang paling menjanjikan untuk pengenalan otomatis emosi manusia. Terutama di bidang sistem keamanan yang semakin berkembang dapat diamati sepanjang tahun lalu. Selain, deteksi kebohongan, *video game* dan bantuan psikiatrik sering diklaim sebagai skenario lebih lanjut untuk pengenalan emosi. Klasifikasi dalam pandangan praktis yang harus dilakukan dianggap bahwa pendekatan teknis hanya bisa mengandalkan keputusan pragmatis tentang jenis, luas dan jumlah emosi sesuai dengan situasi [15].

Terdapat fakta bahwa suara sering mencerminkan emosi yang mendasarinya melalui *tone dan pitch*. Ini juga merupakan fenomena yang digunakan hewan seperti anjing dan kuda untuk bisa memahami emosi manusia. Agar berhasil menerapkan sistem pengenalan emosi wicara, kita perlu mendefinisikan dan

memodelkan emosi dengan cermat. Namun, tidak ada konsensus tentang definisi emosi, dan itu masih merupakan masalah terbuka dalam psikologi. Emosi adalah keadaan psikologis yang berbelit-belit yang terdiri dari beberapa komponen seperti pengalaman pribadi, reaksi fisiologis, perilaku, dan komunikatif. Berdasarkan definisi ini, dua model telah menjadi umum dalam pengenalan emosi bicara yaitu model emosional diskrit, dan model emosional dimensi.

Terdapat tiga point penting agar *speech emotion recognition* dapat berjalan dengan sukses, yaitu:

1. pilihan input suara yang baik,
2. ekstraksi fitur dengan efektif
3. merancang pengklasifikasi yang handal menggunakan algoritma pembelajaran mesin.

Ekstraksi fitur emosional menjadi masalah utama dalam sistem *speech emotion recognition*. Banyak peneliti telah mengusulkan fitur *speech* penting yang berisi informasi emosi, seperti energi, nada, frekuensi formant, Koefisien *Cepstrum Prediksi Linier*, koefisien *cepstrum frekuensi-Mel*, dan fitur spektral modulasi. Dengan demikian, sebagian besar peneliti lebih suka menggunakan set fitur gabungan yang terdiri dari banyak jenis fitur yang mengandung lebih banyak informasi emosional.

Selain wajah, suara adalah modalitas penting untuk ekspresi emosional. Kemampuan bicara seseorang merupakan saluran komunikasi yang relevan yang diperkaya dengan emosi. suara dalam ucapan tidak hanya menyampaikan pesan semantik tetapi juga informasi tentang keadaan emosi pembicara. Beberapa vektor fitur suara penting yang telah dipilih untuk penelitian seperti frekuensi dasar, koefisien cepstral mel-frekuensi, koefisien cepstral prediksi, dll.

Sinyal ucapan seseorang berisi sejumlah besar parameter yang mencerminkan karakteristik emosional. Salah satu poin penting dalam pengenalan emosi adalah fitur apa yang harus digunakan. Dalam penelitian terbaru, banyak

fitur umum diekstraksi, seperti energi, pitch, formant, dan beberapa fitur spektrum seperti koefisien prediksi linier (LPC), koefisien cepstrum frekuensi-mel (MFCC) frekuensi-mel, dan fitur spektral modulasi.

berbicara terdiri dari tiga bagian yaitu cara berbicara dengan bersuara, berbicara tanpa bersuara, dan diam. Berbicara dengan suara dihasilkan oleh getaran lipatan vokal yang menciptakan eksitasi periodik ke saluran vokal selama pelafalan fonem yang merupakan unit suara yang berbeda secara perseptual yang membedakan satu kata dengan kata yang lain seperti tas, ras, mas. Di sisi lain, berbicara tanpa suara adalah hasil dari udara yang melewati penyempitan pada saluran vokal, menghasilkan suara sementara dan turbulen yang merupakan rangsangan aperiodik pada saluran vokal. Karena sifatnya yang periodik, berbicara yang disuarakan dapat diidentifikasi dan diekstraksi.

*Speech emotion recognition* pada penelitian ini digunakan untuk mengetahui suasana hati daripada subjek. Seperti yang sudah dijelaskan pada paragraf sebelumnya bahwa suara manusia juga efektif untuk diketahui suasana hatinya.

## **2.5. API**

*API* atau *Application Programming Interface* merupakan antarmuka pemrograman aplikasi. *API* digunakan untuk komunikasi antara klien dengan server yang ada. Tujuan dari *API* adalah untuk mempercepat pembuatan suatu aplikasi karena programmer tidak perlu menulis kode dari nol. *API* juga bisa diartikan sebagai sekumpulan perintah maupun fungsi yang dapat digunakan oleh programmer saat membangun perangkat lunak. Salah satu kelebihan dari penggunaan *API* adalah dapat digunakan lintas aplikasi.

Ketika dua sistem menghubungkan melalui *API* dalam sebuah integrasi memiliki dua sisi, masing-masing dengan nama khusus. Satu sisi berbicara tentang server. Ini adalah sisi yang benar-benar melibatkan *API*. Ini membantu untuk mengingat bahwa *API* ini tidak hanya satu program saja yang berjalan pada server. Sebuah protokol komputer adalah seperangkat aturan yang mengatur bagaimana dua komputer dapat berkomunikasi satu sama lain. Dibandingkan



dengan standar Bahasa manusia, bagaimanapun, protokol komputer sangat kaku. Pikirkan sejenak dari dua kalimat "Warna favorit saya adalah biru" dan "Biru adalah warna favorit saya." Orang-orang dapat memecah setiap kalimat dan melihat bahwa mereka berarti hal yang sama, meskipun kata-kata yang dalam urutan yang berbeda. Sayangnya, komputer tidak begitu pintar [16].

Untuk membuat permintaan yang valid, klien perlu menyertakan empat hal:

1. URL

URL merupakan alamat unik untuk hal. URL dapat dibuat untuk halaman WEB, gambar, dan video.

2. Metode

Metode digunakan untuk memberitahu server apa tindakan yang klien inginkan. Terdapat empat metode yang paling sering digunakan dalam *API* yaitu:

1. *GET*

2. *POST*

3. *PUT*

4. *DELETE*

3. *Header*

*Header* sendiri menyediakan meta-informasi tentang permintaan. Klien menggunakan header untuk memberitahu server bahwa jenis perangkat apa yang klien gunakan.

4. Tubuh

Pada tubuh sendiri berisi tentang perintah-perintah yang klien inginkan.

*API* dapat digunakan pada penelitian ini untuk mengintegrasikan sistem pada aplikasi yang akan dibuat dengan aplikasi lain yang telah tersedia

## 2.6. Vokaturi *API*

Vokaturi diciptakan pada tahun 2016 di Amsterdam. Para developer Vokaturi *API* merancang *libraries* dalam bentuk C dan Python. Vokaturi *API*

dapat mencerminkan keadaan seni dalam pengenalan emosi dari suara manusia. Algoritme telah dirancang, dan terus ditingkatkan, oleh Paul Boersma, profesor Ilmu Fonetik di Universitas Amsterdam, yang merupakan peneliti utama analisis perangkat lunak praat bicara terkemuka di dunia. Vokaturi dapat mengukur langsung dari suara seseorang apakah seseorang itu sedang bahagia, sedih, takut, marah, atau memiliki suasana hati yang netral [6].

Untuk deteksi prosodik versi OpenVokaturi 2.2b digunakan, untuk mengekstrak dimensi utama pitch, intensitas, dan kemiringan spektralsaat mendeteksi emosi. Sembilan fitur yang diekstraksi dibandingkan ke dua database rekaman audio untuk memprediksi lima emosi yaitu kebahagiaan, kesedihan, kemarahan, ketakutan, dan netralitas. Prediksi dikembalikan sebagai nilai yang mewakili probabilitas setiap emosi. Nilai tertinggi di antara semua prediksi, merupakan suasana hati dominan dari rekaman audio. Deteksi emosi menggunakan teknologi ini telah divalidasi hingga mencapai 66,5%

Saat ini versi *open-source* dari perangkat lunak ini memiliki lima emosi dengan akurasi tinggi, bahkan jika saat mendengar pembicara untuk pertama kalinya. Pengenalan emosi Vokaturi dapat dengan mudah diintegrasikan ke dalam aplikasi perangkat lunak yang ada. Perangkat lunak telah divalidasi dengan basis data emosi yang ada dan berfungsi secara independen bahasa. Perangkat lunak Vokaturi tersedia sebagai pustaka *drop-in* untuk sebagian besar platform modern, termasuk iOS, Android, Windows, MacOS, dan Linux.

Vokaturi sendiri memberikan 3 pilihan libraries yang dapat digunakan oleh developer. Untuk libraries yang disediakan dapat dilihat pada tabel 2.1

**Tabel 2.1 Libraries Vokaturi**

NO	Name	Accuracy	Pricing
1	OpenVokaturi	66.5	Free
2	VokaturiPlus	76.1	Paid
3	VokaturiPro	76.1	Paid

Vokaturi *API* digunakan dalam penelitian ini sebagai *libraries* untuk mengidentifikasi suara seseorang yang dijadikan kedalam nilai emosi yang ada.

## 2.7. *VOiP*

*Voice over Internet Protocol (VOiP)* adalah Teknologi yang menjadikan media internet untuk bisa melakukan komunikasi suara jarak jauh secara langsung. Sinyal suara analog, seperti yang anda dengar ketika berkomunikasi di telepon diubah menjadi data digital dan dikirimkan melalui jaringan berupa paket-paket data secara *real-time*. Prinsip kerja *VoIP* adalah mengubah suara analog yang didapatkan dari speaker pada Komputer menjadi paket data digital, kemudian dari *PC* diteruskan melalui Hub/ Router/ ADSL Modem dikirimkan melalui jaringan internet dan diterima oleh tempat tujuan melalui media yang sama. Atau bisa juga melalui media telepon diteruskan ke *phone adapter* yang disambungkan ke internet dan bisa diterima oleh telepon tujuan [17].

*Voice over IP (VOiP)* diimplementasikan dalam berbagai macam jalan menggunakan hak milik dan standar serta protocol terbuka. Berikut merupakan contoh protocol jaringan untuk mengimplementasikan *VoIP*:

1. *Real-time Transport Protocol (RTP)*
2. *Session Initiation Protocol (SIP)*
3. *Inter-Asterisk eXchange (IAX)*
4. H.323
5. *Media Gateway Control Protocol (MGCP)*
6. *Session Description Protocol (SDP)*

Kualitas yang dihasilkan dari suara *VoIP* bergantung pada beberapa parameter yaitu kapasitas *bandwidth*, jika terjadi gangguan pada jaringan maka paket yang dikirim pun akan hilang. Tingkat hilang paket merupakan parameter yang menyatakan besarnya laju kesalahan yang terjadi sepanjang jalur pengiriman data paket dari pengirim ke penerima. Waktu tunda merupakan parameter yang menyatakan rentang waktu yang diperlukan untuk mengirim paket dari pengirim ke penerima.

*VoIP* digunakan dalam penelitian ini karena untuk melakukan panggilan telepon secara daring digunakan *VoIP*.

## 2.8. **Sinch**

Sinch adalah produk yang dapat membuat panggilan suara dan pesan instan ke android aplikasi seluler dengan mudah. Sinch ini memberikan akses untuk dapat membuat telepon secara daring antara satu handphone dengan handphone yang lain. Sinch selain mudah digunakan juga memberikan akses dan konektivitas yang beragam. Panggilan suara yang dihasilkan pun jernih dan tidak kalah bagus dengan aplikasi telepon daring lainnya. [4].

Berikut merupakan beberapa fitur yang dapat digunakan pada Sinch:

1. Mendukung *VoiceCall*
2. Mendukung *Video Calls*
3. Mendukung *SMS*

Sinch digunakan pada penelitian ini karena merupakan protocol jaringan untuk pengimplementasian *VoIP*. Disisi lain *client* ini gratis dan bersifat *open source* menjadikannya mudah dimodifikasi

## 2.9. **Android**

*Android* merupakan istilah dalam Bahasa Inggris yang berarti “Robot yang menyerupai Manusia”. Logo *Android* sendiri, dicerminkan seperti sebuah robot berwarna hijau, yang mengacu kepada arti kata *Android*. *Android* adalah sebuah sistem operasi untuk smartphone dan tablet. Sistem operasi dapat diilustrasikan sebagai “jembatan” antara peranti dan penggunanya, sehingga pengguna dapat berinteraksi dengan *device*-nya dan menjalankan aplikasi aplikasi yang tersedia pada *device* [18].

*Android* merupakan sistem operasi berbasis linux yang dirancang untuk sebuah perangkat layer sentuh berukuran kecil. Untuk mengembangkan *Android* dibentuklah *Open Handset Alliance* yang merupakan gabungan dari 34

perusahaan peranti keras, peranti lunak dan telekomunikasi termasuk Google, HTC, Intel, Motorola, Qualcomm, TMobile, dan NVidia. Android sendiri didirikan di California pada bulan Oktober 2003 oleh Andy Rubin, Rich Miner, Nick Sears dan Chris White.pada tanggal 17 Agustus 2005 Google yang merupakan salah satu perusahaan teknologi terbesar mengakuisi *Android* inc dan menjadikannya sebagai anak perusahaan yang sepenuhnya dimiliki oleh google [19].

Spesifikasi minimal sebuah perangkat untuk menjalankan sebuah sistem operasi *Android* adalah sebagai berikut:

1. 512 MB RAM
2. Prosesor ARMv7 32-bit
3. arsitektur MIPS
4. OpenGL ES 2.0

Hingga saat ini *Android* telah mengeluarkan begitu banyak versi. Berikut merupakan versi beserta *API level* dari *Android*:

- |                                  |                       |
|----------------------------------|-----------------------|
| 2. <i>Android</i> 1.0            | / <i>API level</i> 1  |
| 3. <i>Android</i> 1.1            | / <i>API level</i> 2  |
| 4. <i>Android</i> 1.5            | / <i>API level</i> 3  |
| 5. <i>Android</i> 1.6            | / <i>API level</i> 4  |
| 6. <i>Android</i> 2.0            | / <i>API level</i> 5  |
| 7. <i>Android</i> 2.0.1          | / <i>API level</i> 6  |
| 8. <i>Android</i> 2.1.x          | / <i>API level</i> 7  |
| 9. <i>Android</i> 2.2.x          | / <i>API level</i> 8  |
| 10. <i>Android</i> 2.3 – 2.3.2   | / <i>API level</i> 9  |
| 11. <i>Android</i> 2.3.3 – 2.3.4 | / <i>API level</i> 10 |
| 12. <i>Android</i> 3.0.x         | / <i>API level</i> 11 |
| 13. <i>Android</i> 3.1.x         | / <i>API level</i> 12 |

14. <i>Android 3.2</i>	/ <i>API level 13</i>
15. <i>Android 4.0 – 4.0.2</i>	/ <i>API level 14</i>
16. <i>Android 4.0.3 - 4.0.4</i>	/ <i>API level 15</i>
17. <i>Android 4.1 – 4.1.1</i>	/ <i>API level 16</i>
18. <i>Android 4.2 – 4.2.2</i>	/ <i>API level 17</i>
19. <i>Android 4.3</i>	/ <i>API level 18</i>
20. <i>Android 4.4</i>	/ <i>API level 19</i>
21. <i>Android 4.4W</i>	/ <i>API level 20</i>
22. <i>Android 5.0</i>	/ <i>API level 21</i>
23. <i>Android 5.1</i>	/ <i>API level 22</i>
24. <i>Android 6.0</i>	/ <i>API level 23</i>
25. <i>Android 7.0</i>	/ <i>API level 24</i>
26. <i>Android 7.1 – 7.1.1</i>	/ <i>API level 25</i>
27. <i>Android 8.0</i>	/ <i>API level 26</i>
28. <i>Android 8.1</i>	/ <i>API level 27</i>
29. <i>Android 9</i>	/ <i>API level 28</i>
30. <i>Android 10.0</i>	/ <i>API level 29</i>

*Android* dipilih dalam penelitian ini karena berdasarkan perangkat yang digunakan oleh perusahaan pada penelitian ini merupakan tablet berbasis *Android*. Disisi lain sensor yang begitu ragam pada *Android* dapat mendukung penelitian ini.

### **2.10. *Smartphone***

Ponsel cerdas atau *smartphone* merupakan telepon genggam yang memiliki fitur serta kemampuan yang menyerupai komputer. Kebanyakan alat yang dikategorikan sebagai ponsel cerdas menggunakan sistem operasi yang berbeda. Biasanya ponsel cerdas dilengkapi dengan layar berukuran besar, kamera, pemutar musik, internet, pemutar video, layanan internet serta dukungan aplikasi yang terdapat pada komputer. Hampir semua orang di dunia menggunakan ponsel cerdas untuk meringankan segala bentuk pekerjaan yang ada.

Ponsel tidak lagi menjadi barang mewah sehingga kebutuhan ponsel cerdas semakin tinggi. Perputaran pelanggan ponsel di Indonesia diprediksi mencapai 8,6 persen dalam sebulan. Angka perputaran pelanggan di India mencapai 4 persen per bulan. Hal ini menunjukkan kebutuhan terhadap telekomunikasi multi fungsi kian berkembang, termasuk tuntutan kebutuhan *smartphone*. Ragam produk *smartphone*, termasuk maraknya *smartphone made in China* dan semakin murah nya tarif, maka konsumen memiliki banyak alternatif pilihan. Aspek lain yang mendukung terhadap permintaan *smartphone* yaitu fakta mobilitas individu yang didukung fenomena *online communities* di perkotaan [20].

*Smartphone* pada penelitian ini menjadi dasar dari perangkat yang digunakan dikarenakan memiliki begitu banyak keunggulan serta sensor yang dapat dimanfaatkan.

### **2.11. Sensor Microphone**

Mikrofon merupakan suatu jenis transduser atau sensor yang mengubah suara analog menjadi sinyal listrik. Mikrofon merupakan salah satu alat untuk membantu komunikasi manusia. Mikrofon dipakai pada banyak alat seperti telepon, alat perekam, alat bantu dengar, dan pengudaraan radio serta televisi.

Berikut merupakan cara kerja *microphone*:

1. Saat berbicara, suara akan membentuk gelombang suara dan menuju ke *microphone*.
2. Gelombang suara akan menabrak diafragma yang terdiri dari membran plastik yang sangat tipis. Diafragma akan bergetar sesuai dengan gelombang suara yang diterimanya.
3. Sebuah kumparan kawat kecil yang terdapat di bagian belakang diafragma akan ikut bergetar sesuai dengan getaran diafragma.
4. Magnet kecil yang dikelilingi oleh Coil atau Kumparan tersebut akan menciptakan medan magnet seiring dengan gerakan Coil.
5. Pergerakan Coil di medan magnet ini akan menimbulkan sinyal listrik.

6. Sinyal listrik yang dihasilkan tersebut kemudian mengalir ke alat perekam suara.

Sensor *microphone* dimanfaatkan dalam penelitian ini untuk merubah suara seseorang yang tadinya gelombang analog menjadi gelombang elektrik.

## 2.12. *Android Studio*

*Android Studio* adalah *Integrated Development Enviroment (IDE)* untuk sistem operasi *Android*, yang dibangun diatas perangkat lunak JetBrains IntelliJ *IDEA* dan didesain khusus untuk pengembangan *Android*. *IDE* ini merupakan pengganti dari *Eclipse Android Development Tools* yang sebelumnya merupakan *IDE* utama untuk pengembangan aplikasi *android*. *Android studio* menawarkan alat bantu, diantaranya adalah pengeditan *debugging*, kode yang lengkap , dan pengujian.

Untuk membuat aplikasi berbasis *Android*, banyak *Integrated Development Environment (IDE)* yang bisa digunakan. *Android Studio* dengan bahasa pemrograman *Java* merupakan *IDE* dan bahasa pemrograman *native* yang didukung secara resmi oleh Google. Sebagai *IDE* yang didukung oleh Google, AS menjadi *IDE* terbaik yang bisa digunakan untuk membangun aplikasi. Namun demikian banyak pemrogram pemula yang merasa kesulitan untuk memulai terutama yang baru memulai konsep pemrograman untuk perangkat *mobile*. *Android Studio* juga dikenal sebagai *IDE* yang membutuhkan *requirement* komputer yang handal serta memiliki fitur yang sangat kompleks [21].

*Android studio* dalam penelitian ini dipilih karena memiliki keunggulan yang lebih baik dari pada *IDE* lain. Selain itu juga *Android studio* merupakan *IDE* resmi dari *android* sehingga banyak panduan yang telah dikeluarkan oleh google.



### 2.13. MySQL

MySQL merupakan sistem manajemen basis data SQL atau DBMS yang multialur, multipengguna, dengan sekitar 6 juta instalasi di seluruh dunia. MySQL tersedia sebagai perangkat lunak gratis di bawah lisensi GNU *General Public License* (GPL), tetapi mereka juga menjual di bawah lisensi komersial untuk kasus-kasus di mana penggunaannya tidak cocok dengan penggunaan GPL. SQL merupakan kependekan dari kata "*Structured Query Language*". SQL merupakan suatu bahasa permintaan yang terstruktur yang melekat pada satu *database* atau SDBD tertentu, sedangkan MySQL merupakan *database*-nya. Dengan kata lain, *MySQL* merupakan SDBD-nya dan *SQL* adalah perintah atau bahasa yang melekat di dalam SDBD tersebut [22].

*MySQL* sendiri ditulis dalam C dan C++. Agar lebih mudah dipahami, negara-negara di Amerika Selatan sebagai contohnya. Secara geografis, negara-negara tersebut tidaklah sama, bahkan sejarahnya pun berbeda. Namun, masyarakat di semua negara di Amerika Selatan menggunakan bahasa Spanyol untuk berkomunikasi. API yang tersedia memungkinkan aplikasi-aplikasi komputer yang ditulis dalam berbagai bahasa pemrograman untuk dapat mengakses basis data *MySQL* antara lain bahasa pemrograman C, C++, C#, bahasa pemrograman *Eiffel*, bahasa pemrograman *Smalltalk*, bahasa pemrograman *Java*, bahasa pemrograman Lisp, Perl, PHP, bahasa pemrograman *Python*, *Ruby*, *REALbasic* dan Tcl. Sebuah antarmuka ODBC memanggil MyODBC yang memungkinkan setiap bahasa pemrograman yang mendukung ODBC untuk berkomunikasi dengan basis data *MySQL*.

Berikut merupakan beberapa fitur pada *MySQL*:

1. *Multi-User*, dapat digunakan oleh beberapa user secara bersamaan
2. *Command* dan *function*, *MySQL* memiliki fungsi dan operator secara penuh yang mendukung perintah *select* dan *where* dalam *query*
3. Mendukung kalimat *SQL GROUP BY* dan *ORDER BY* serta mendukung fungsi seperti *AVG()*, *STD()*, dan lain-lain.
4. Mendukung *field* yang dijadikan *index*
5. Memiliki banyak tipe data

6. Mendukung *foreign key*
7. Keamanan yang baik
8. Mendukung *replication*

*MySQL* digunakan dalam penelitian ini karena dapat memudahkan peneliti dalam menyimpan serta mengolah data yang ada.

## **2.14. Bahasa Pemrograman**

Bahasa pemrograman, adalah bahasa komputer atau bahasa pemrograman komputer, yang merupakan instruksi standar untuk memerintah komputer. Bahasa pemrograman ini merupakan suatu himpunan dari aturan sintaks dan semantik yang dipakai untuk mendefinisikan program komputer. Bahasa ini memungkinkan seorang programmer dapat menentukan secara persis data mana yang akan diolah oleh komputer, bagaimana data ini akan disimpan/diteruskan, dan jenis langkah apa yang akan diambil dalam berbagai situasi secara persis. Jumlah bahasa pemrograman ini sangat banyak.

### **2.15.1 Java**

*Java* adalah bahasa pemrograman yang dapat dijalankan di berbagai komputer termasuk telepon genggam. Bahasa ini banyak mengadopsi pada bahasa 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 berbagai Mesin *Virtual Java* (JVM). *Java* merupakan bahasa pemrograman yang bersifat umum/non-spesifik.

Terdapat tiga *IDE* populer yang dapat digunakan seperti *NetBeans*, *Eclipse* dan *IntelliJDEA*. *Java* memiliki sangat banyak tipe data yang dasar dan kompleks. Tipe data yang kompleks dapat ditemukan seperti *ArrayList*, *HashMap*, *HashTable*, *Vector*, *Array*, dan lain-lain.

*Java* dikeluarkan oleh sebuah perusahaan yang bernama *Sun Microsystem* dalam tiga edisi yaitu:

1. *Standard Edition* (J2SE): Didesain untuk dijalankan pada personal computer.
2. *Enterprise Edition* (J2EE): Dengan fitur built in untuk Servlet, JSP, and XML, edisi ini didesain untuk aplikasi berbasis server.
3. *Micro Edition* (J2ME): Didesain untuk *device* yang memiliki keterbatasan memori, tampilan dan proses seperti *handphone* atau *PDA (Personal Digital Assistant)* [23].

Bahasa pemrograman *Java* dalam penelitian ini digunakan karena *java* merupakan bahasa dasar dari *android*.

### 2.15.2 PHP

PHP adalah bahasa pemrograman yang berfungsi untuk membuat website dinamis maupun aplikasi web. Berbeda dengan HTML yang hanya bisa menampilkan konten statis, PHP bisa berinteraksi dengan *database*, *file* dan *folder*, sehingga membuat PHP bisa menampilkan konten yang dinamis dari sebuah website. Blog, Toko *Online*, CMS, Forum, dan *Website Social Networking* adalah contoh aplikasi web yang bisa dibuat oleh PHP. PHP adalah bahasa *scripting*, bukan bahasa *tag-based* seperti HTML. PHP termasuk bahasa yang *cross-platform*, ini artinya PHP bisa berjalan pada sistem operasi yang berbeda-beda.

PHP khususnya PHP 5 sudah mendukung beberapa konsep OOP. Akan tetapi PHP 5 tidak mendukung konsep Multiple-inheritance dikarenakan konsep Multiple-inheritance terdapat di bahasa pemrograman bahasa C [24]. Ratusan fungsi yang disediakan oleh PHP serta ribuan lainnya yang tersedia melalui berbagai ekstensi tambahan.

Berikut merupakan tipe data yang didukung oleh PHP:

1. *Boolean*

2. *Integer*
3. *Float/ Double*
4. *String*
5. *Array*
6. *Object*
7. *Resource*
8. *NULL*

PHP dalam penelitian ini digunakan sebagai bahasa pemrograman untuk *API* yang nantinya akan menyambungkan antara aplikasi dengan database.

### **2.15. UML**

*UML* atau *Unified Modeling Language* adalah pengganti dari metode analisis berorientasi *object* dan *design* berorientasi *object*. *UML* sendiri diciptakan oleh *Object Management Group* yang diawali dengan versi 1.0 pada Januari 1997. *UML* digunakan untuk memodelkan suatu sistem yang menggunakan konsep berorientasi *object*. Dan juga untuk menciptakan suatu bahasa pemodelan yang dapat digunakan baik oleh manusia maupun mesin.

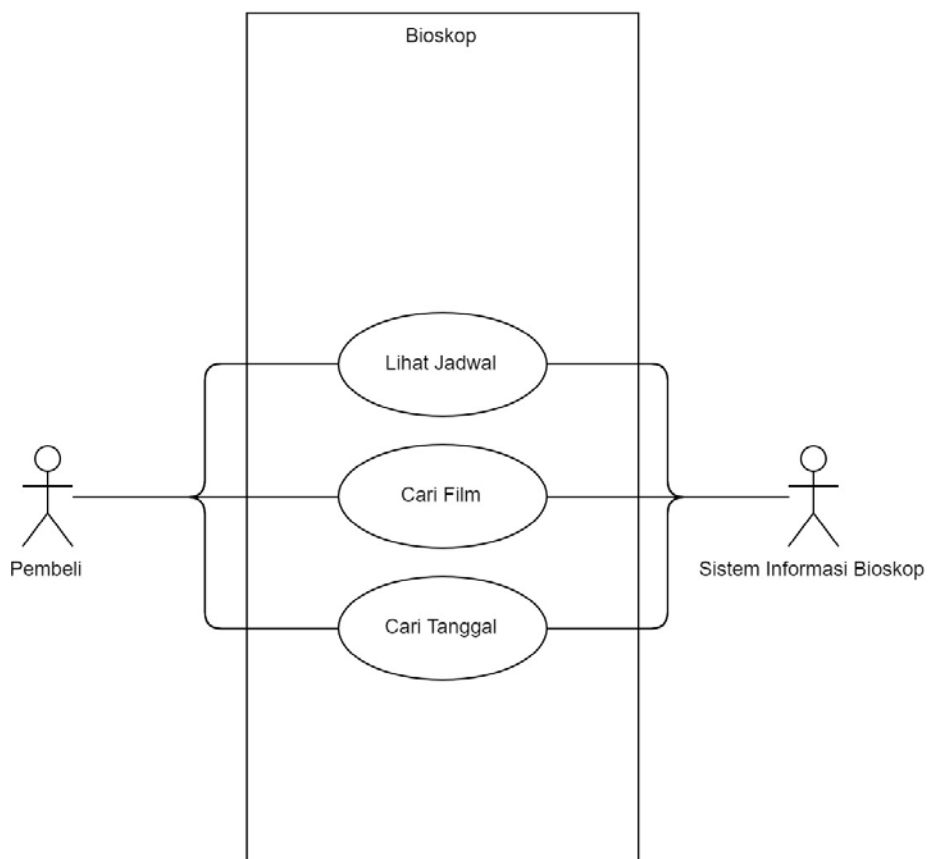
Model *UML* dalam pengembangan sistem memudahkan proses perancangan sistem yang dibuat sehingga dapat menjadi lebih menyesuaikan dengan keinginan user. Pada penelitian yang akan datang diharapkan pengembangan sistem informasi dapat menyesuaikan dengan kebutuhan saat ini, sehingga dapat sesuai dengan keinginan *user* [25].

*UML* dalam penelitian ini digunakan supaya mempermudah peneliti dalam penggambaran serta perancangan sistem yang akan dibangun. Disisi lain *UML* memodelkan suatu sistem yang menggunakan konsep berorientasi *object* sehingga cocok dengan bahasa pemrograman *Java*.

### **2.16. Use Case Diagram**

*Use Case Diagram* adalah gambaran grafis dari beberapa atau semua *actor*, *use case*, dan interaksi diantaranya yang memperkenalkan suatu sistem. Di dalam *use case* ini akan diketahui fungsi-fungsi apa saja yang berada pada sistem yang dibuat. *Use case* diagram tidak menjelaskan secara detail tentang penggunaan *use case*, tetapi hanya memberi gambaran singkat hubungan antara *use case*, aktor, dan sistem.

Setelah prioritas dan risiko ditetapkan pada *use case*, dapat membantu mengelola masalah yang terdapat dalam pekerjaan. *Use case* dapat ditugaskan ke tim atau individu untuk diimplementasikan dan dapat terus dipantau jalannya progres, karena merupakan salah satu kelebihan *use case*. Jika terdapat suatu masalah pada *project* maka *use case* bisa ditunda sehingga masalah tidak bertambah besar [26].



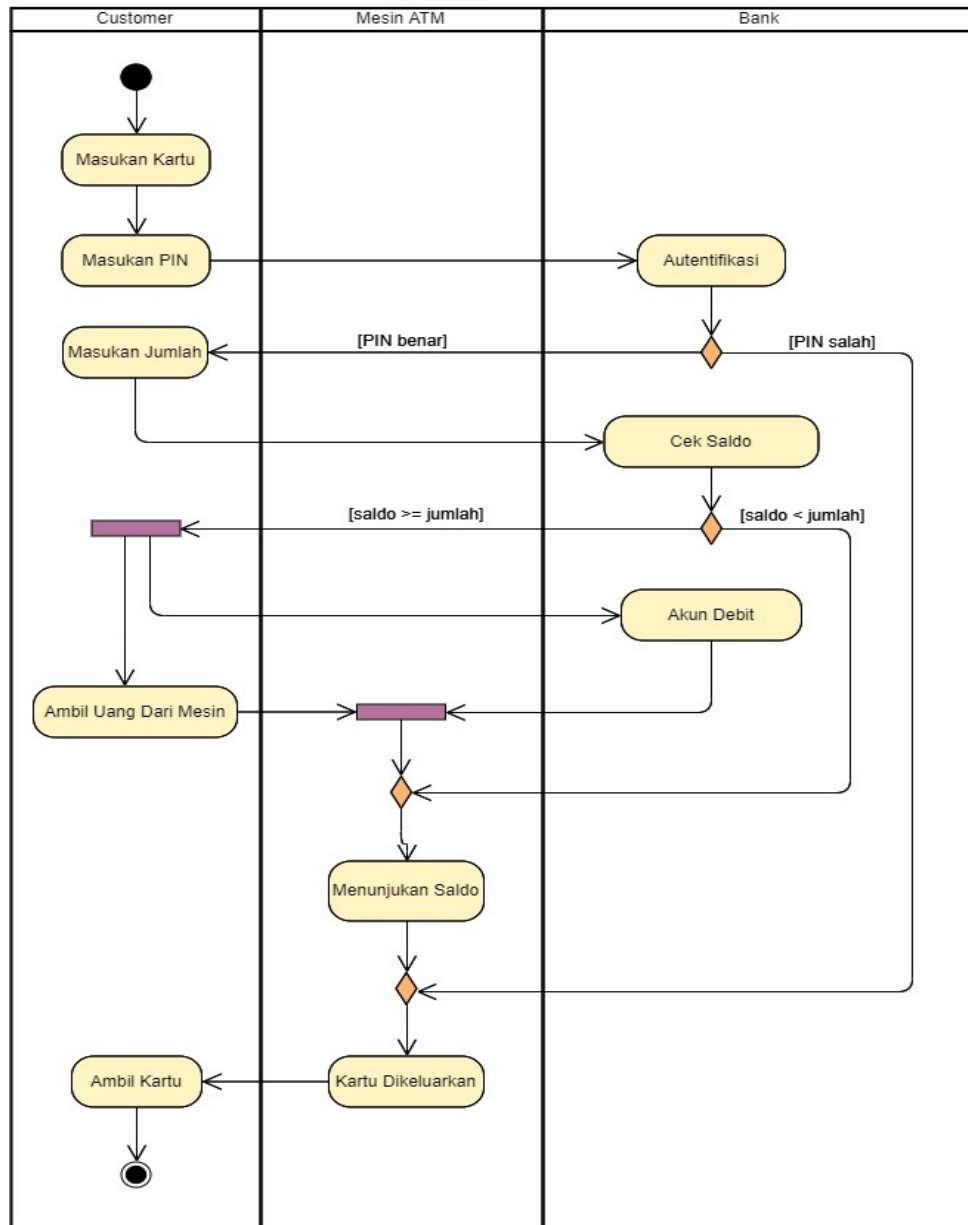
**Gambar 2.2. Contoh Use Case Diagram**

*Use Case* diagram digunakan dalam penelitian ini karena membantu peneliti dalam penggambaran pekerjaan yang dilakukan oleh *actor* pada sebuah sistem yang akan dibangun.

### **2.17. Activity Diagram**

*Activity diagram* merupakan salah satu diagram UML yang paling mudah diakses karena menggunakan simbol yang mirip dengan notasi *flowchart* yang dikenal luas. Oleh karena itu, *activity* diagram berguna untuk menggambarkan proses kepada khalayak luas. Faktanya, diagram aktivitas memiliki dasar dalam diagram *flowcharts*, serta diagram status UML, data *flow* diagram, dan Petri Nets [26].

*Activity Diagram* menggambarkan bagaimana kegiatan dikoordinasikan untuk menyediakan layanan yang berbeda pada tingkat abstraksi. Biasanya, suatu peristiwa perlu dicapai oleh beberapa operasi, terutama dimana operasi dimaksudkan untuk mencapai sejumlah hal berbeda yang memerlukan koordinasi, atau bagaimana peristiwa dalam satu kasus penggunaan berhubungan satu sama lain, khususnya, kasus penggunaan di mana kegiatan mungkin tumpang tindih dan membutuhkan koordinasi. Ini juga cocok untuk memodelkan bagaimana kumpulan *use case* mengoordinasikan untuk mewakili *business workflows*.



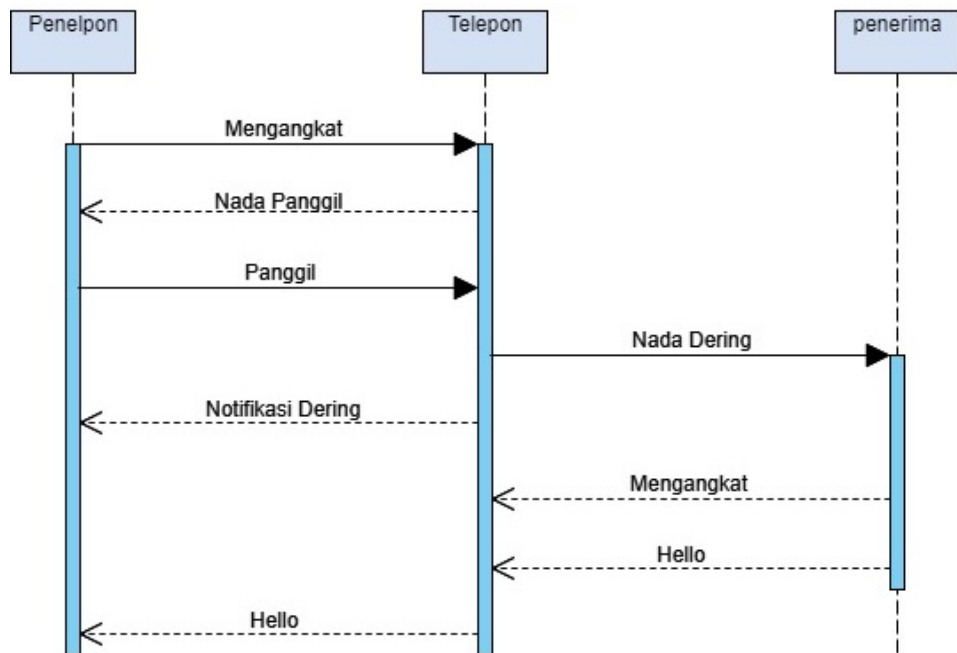
**Gambar 2.3. Contoh Activity Diagram**

*Activity diagram* pada penelitian ini digunakan untuk menggambarkan proses aktivitas yang dapat dilakukan oleh sistem.

### 2.18. Sequence Diagram

*Sequence diagram* merupakan sebuah urutan perintah antara bagian pada sistem. Dengan menggunakan *sequence diagram* dapat menggambarkan interaksi

yang akan terjadi dan dipicu ketika *use case* tertentu dilaksanakan. *Sequence diagram* menjelaskan informasi tentang interaksi antara satu sistem dengan sistem yang lain. *Sequence diagram* sangat penting dalam UML dan merupakan diagram yang paling terkenal [26].



**Gambar 2.4. Contoh *Sequence Diagram***

*Sequence Diagram* pada penelitian ini digunakan karena dibutuhkan gambaran *input* serta *output* yang didapat dari sebuah aktivitas untuk memperjelas proses yang dilakukan oleh sistem.

### 2.19. *Class Diagram*

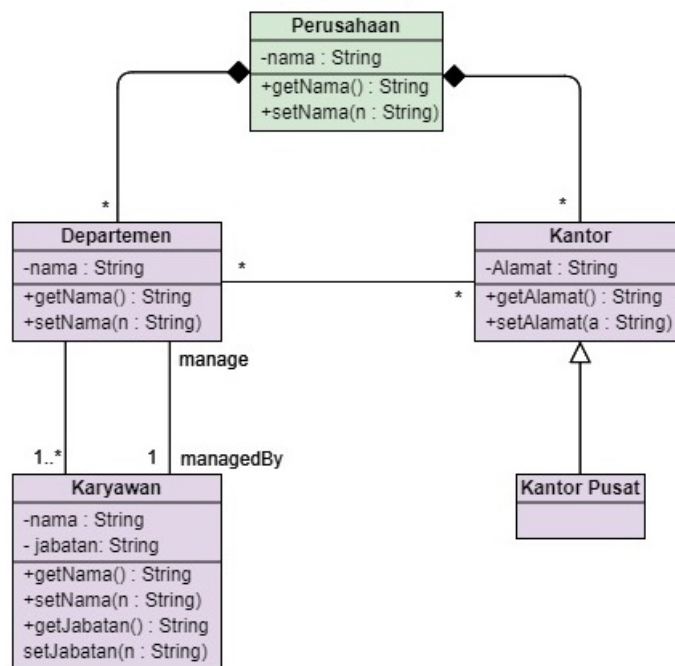
*Class diagram* adalah sekumpulan class dari sebuah sistem. *Class* dapat memiliki relasi satu dengan yang lainnya. Sebuah *class* bisa menjadi penyederhanaan dari *class* lain atau bisa juga berisi objek-objek dari kelas lain



tergantung pada seberapa kuat hubungan antara sebuah *class* dengan class lainnya [26].

Dalam suatu *class* diagram dapat memiliki salah satu sifat berikut:

1. *Public*
2. *Private*
3. *Protected*



**Gambar 2.5. Contoh Class Diagram**

*Class diagram* pada penelitian ini digunakan untuk penggambaran hubungan antara *class* pada sistem.