

## BAB II

### LANDASAN TEORI

#### 2.1 Wearable Device

*Wearable device* merupakan suatu perangkat yang digunakan kepada tubuh serta diklasifikasikan berdasarkan kedalam fungsi, penampilan, atau parameter yang lainnya. Contohnya seperti yang digunakan untuk mendeteksi kebugaran seseorang dimana biasanya teknologi ini disematkan pada *wearable device* seperti *smartband* atau *smartwatch*. Sedangkan dibidang medis, alat yang digunakan untuk memperoleh data fisiologis, data perilaku dan kontekstual untuk diagnosis pengobatan dan pengelolaan penyakit terhadap pasien [7].

Pertama kali *Wearable device* ditemukan oleh Ed Thorp dan Claude Shannon pada tahun 1996 yang saat itu berupa komputer analog memiliki ukuran sebesar kotak rokok yang diperuntukan sebagai menebak roda *roulette*. Seiring dengan perubahan dari keadaan terhadap zaman *wearable device* tersebut bisa untuk melakukan pendeteksian terhadap segala macam aktivitas yang dilakukan oleh manusia, diperlukan adanya peranan dari sensor agar hal ini bisa dilakukan. Beberapa jenis dari sensor disematkan dalam *wearable device* [8] antara lain :

a) Accelerometer

Bisa digunakan sebagai alat ukur dalam beberapa parameter seperti Akselerasi, getaran, rotasi, dan gravitasi.

b) Proximity Ambient

Digunakan untuk melacak keberadaan dan jarak user dengan memanfaatkan pantulan inframerah.

c) Gyroscope

Sebagai alat ukur untuk orientasi yang kemudian penggunaannya dapat dipakai untuk mengukur berat, bentuk serta kecepatan.

d) Fingerptint

Biasanya teknologi ini digunakan untuk keamanan suatu sistem atau perangkat. Cara menggunakannya cukup dengan menempelkan sidik jari

(biasanya jempol atau telunjuk) yang telah terdaftar di sistem kepada sensor sidik jarinya.

e) Heart Rate

Digunakan untuk mengirim detak jantung, biasanya device ditempelkan di dada atau pun di pergelangan tangan yang kemudian saling terhubung dengan *smartphone* melalui konektivitas *bluetooth* agar data detak jantung lebih nyaman untuk dilihat.

## 2.2 Smartband

*Smartband* merupakan perangkat *wearable* yang digunakan pada pergelangan tangan layaknya sebuah arloji, walaupun berbentuk seperti jam tangan dimana *smartband* tersebut bisa terkomputerisasi yang fungsinya lebih daripada sekedar menampilkan waktu [9]. *Smartband* memiliki banyak fitur yang dapat digunakan seperti untuk mendeteksi detak jantung, tekanan darah, kandungan oksigen dalam darah dan juga fitur alarm yang dapat di atur dari *smartphone*. Selain itu *smartband* juga dapat menerima notifikasi pesan dari handphone seperti pesan whatsapp, *short message service* (SMS) ataupun notifikasi dari Aplikasi lainnya.

Terdapat konektivitas *bluetooth* agar *smartband* dapat terhubung dengan *smartphone*. Kemudian agar Aplikasi pada *smartphone* dapat mengambil data dari *smartband* digunakan lah SDK dari *smartband* tersebut (beda merk maka SDK pun akan berbeda). Didalam SDK tersebut terdapat fungsi-fungsi yang digunakan untuk mengambil data dari *smartband* yang telah terhubung dengan *smartphone* melalui konektivitas *bluetooth*.

## 2.3 Jantung

Jantung adalah salah satu dari sekian banyak organ vital manusia yang sangat penting dalam tubuh. Memiliki fungsi untuk memompa darah agar bisa mengalir ke setiap organ melalui pembuluh darah, isi yang terkandung dalam darah tersebut merupakan nutrisi - nutrisi yang dibutuhkan oleh organ - organ yang ada dalam tubuh. Oleh karena itu kesehatan jantung akan mempengaruhi

keadaan tubuh, baik orang yang sehat ataupun orang yang sudah terdapat penyakit menjadi hal penting untuk selalu memperhatikan kondisi jantung [10].

### 2.3.1 Kesehatan Jantung

Kesehatan jantung adalah kondisi dimana organ jantung dalam keadaan yang sehat dan tidak mempunyai suatu masalah tertentu sehingga berfungsi secara baik. Hal tersebut bisa dilakukan oleh setiap orang dengan menjaga dan melakukan pola hidup sehat baik itu dalam perilaku, makanan, dan sebagainya [11]. Meskipun suatu hal yang sangat penting akan tetapi masih banyak orang senantiasa menganggap remeh atau mengabaikan untuk menerapkan pola hidup sehat, padahal dengan pola hidup sehat ini terdapat manfaat yang beragam seperti meningkatkan kesehatan, menghindari atau mencegah penyakit, dan berpartisipasi dalam meningkatkan kualitas kesehatan salah satunya dalam hal kesehatan jantung [12].

### 2.3.2 Denyut Jantung

Denyut jantung adalah cara untuk mengetahui terhadap kondisi jantung, melalui tekanan yang dialirkan oleh darah kedalam bentuk gelombang yang dipompa dari jantung menuju setiap organ dalam tubuh. Intensitas terhadap gelombang yang dialirkan akan beragam tergantung berdasarkan dari faktor usia. Kategori usia dalam keadaan denyut jantung dimana terbagi kedalam empat jenis yaitu bayi (100-140), anak - anak (80-120), remaja (60-100), dan dewasa (60-100) [13]. Lebih jelasnya bisa dilihat pada tabel 2.1 sebagai berikut :

**Tabel 2.1 Denyut Jantung Berdasarkan Usia**

<b>Kategori</b>	<b>BPM</b>			
	<b>Normal</b>	<b>Tinggi</b>	<b>Rendah</b>	<b>Rata - rata</b>
Bayi (0-12 Bulan)	100 – 140	>140	<100	120
Anak – anak (2-10 Tahun)	80 – 120	>120	<80	100
Remaja (11-17 Tahun)	60 – 100	>100	<60	80
Dewasa (>18 Tahun)	60 – 100	>110	<60	80

## 2.4 Aplikasi

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 dapat digolongkan menjadi beberapa kelas, antara lain [14] :

- 1) Perangkat lunak perusahaan (*Enterprise*)
- 2) Perangkat lunak infrastruktur perusahaan
- 3) Perangkat lunak informasi kerja
- 4) Perangkat lunak media dan hiburan
- 5) Perangkat lunak pendidikan
- 6) Perangkat lunak pengembangan media
- 7) Perangkat lunak rekayasa produk

Aplikasi merupakan sebuah program yang dirancang untuk *end user* atau pengguna akhir. Dalam proses membangun aplikasi bisa dilakukan secara proyek pribadi atau menggunakan *source code* yang telah ada. Aplikasi yang dibangun tentu memiliki fungsi untuk membantu dalam tugas – tugas tertentu seperti penerapan atau penggunaan [15]. Perkembangan aplikasi saat ini tersedia di beberapa *platform* yang terdapat dalam tiga jenis *platform* utama yang telah senantiasa mengalami perkembangan terhadap aplikasi antara lain : aplikasi berbasis *mobile*, aplikasi berbasis *web*, dan aplikasi berbasis *desktop*.

### 2.4.1 Aplikasi Mobile

Aplikasi *mobile* yang disebut juga sebagai *mobile application* adalah program yang dirancang untuk bisa dioperasikan kedalam perangkat seluler seperti ponsel dan tablet. Pemanfaatan aplikasi *mobile* digunakan sebagai 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 [16].

Awalnya dibangun untuk aplikasi yang bersifat membantu dalam produktivitas seperti email dan kalender akan tetapi seiring perkembangan zaman dan permintaan dari publik terhadap aplikasi menjadikan terjadinya percepatan ke aera yang lain seperti game, layanan berbasis lokasi, tiket pembelian dan yang lainnya. Aplikasi mobile dapat untuk diunduh melalui platform berdasarkan sistem operasi yang digunakan oleh pengguna seperti App Store untuk Apple atau Google Play Store untuk Android, tidak hanya terdapat aplikasi yang bersifat gratis ada beberapa aplikasi yang perlu bayar terlebih dahulu sebelum untuk diunduh [17].

## 2.5 Android

Android adalah sistem operasi *mobile* yang didasarkan pada berbasis dari kernel Linux, dirancang untuk perangkat seluler layar sentuh seperti *smartphone* dan tablet [15]. Dikembangkan oleh *developer* yang dikenal sebagai Open Handset Alliance dimana untuk kontributor utama dan pemasar komersialnya merupakan Google [18]. Dalam sejarah perkembangannya, sistem operasi Android dimulai sejak oktober 2003 disaat Andy Rubin, Rich Miner, Chris White dan Nick Sears mendirikan sebuah perusahaan yang bernama Android, Inc hingga tahun 2006 adanya kehadiran dari google datang untuk mengakuisisi perusahaan tersebut [15]. Google bersama Open Handset Alliance (OHA) merilis Google Android *Software Development Kit* (SDK). Paket SDK yang dirilis untuk mengembangkan aplikasi seperti sistem operasi, *middleware*, dan aplikasi utama untuk perangkat *mobile*. Dengan dirilisnya SDK tersebut membuka kesempatan kepada pengembang untuk mengembangkan sebuah aplikasi berbasis Android. Saat ini, sistem operasi android menjadi pilihan yang menguntungkan bagi banyak vendor *smartphone*, karena memiliki biaya lisensi yang lebih murah dan sifatnya yang semi *open source*. Selain itu, Android tentunya juga akan support dengan berbagai layanan dari Google.

### 2.5.1 Perkembangan Seri Android

Sejak April 2009, versi Android dikembangkan dengan berbagai codename yang diberikan nama berdasarkan dari setiap makanan pencuci mulut atau dessert [19]. Berikut urutan seri Android yang telah ada antara lain :

#### 1) Android 1.0 Alpha

Versi Android 1.0 Alpha ini pertama kali dirilis pada September 2008 dan tidak ditujukan untuk kebutuhan komersial. Walaupun belum dirilis secara komersial, versi Android ini telah dilengkapi fitur dukungan akses web browser, streaming youtube, pemutar media, google maps, dan sinkronisasi dengan aplikasi google lainnya.

#### 2) Android 1.1 Beta

Selang beberapa bulan versi Android 1.1 Beta diluncurkan pada tanggal 9 Februari 2009. Sama seperti pendahulunya yakni Alpha versi Android belum dirilis secara komersial, hanya diperuntukan untuk satu perangkat. Pembaruan ini dilakukan untuk memperbaiki bugs dan meningkatkan beberapa fitur seperti rincian lokasi pada aplikasi maps serta fitur menyembunyikan dan menampilkan tombol panggilan.

#### 3) Android 1.5 Cupcake

Barulah pada tanggal 30 April 2009 android versi android 1.5 Cupcake yang diperkenalkan secara komersial. Versi Cupcake merupakan generasi pertama yang dirilis secara komersial dan berawal dari sinilah Android mulai menggunakan nama makanan manis kepada setiap versi Android yang dirilisnya. Beberapa fitur yang dimiliki Android Cupcake seperti dukungan akan rotasi layar otomatis, widget, dan keyboard virtual.

#### 4) Android 1.6 Donut

Versi pembaruan Android berikutnya adalah versi 1.6 Donut yang dirilis pada tanggal 15 September 2009. Pada versi Donut ini Android menambahkan beberapa fitur seperti persentase daya baterai, dukungan

gestur, fasilitas pencarian di android market atau yang sekarang kita kenal dengan play store.

5) Android 2.0 Éclair

Selang sebulan Android melakukan pembaruan kembali untuk memperbaiki bug dan memperkaya fiturnya tepatnya pada tanggal 26 Oktober 2009. Pembaruan Android versi 2.0 diberi nama Eclair dan pada versi ini terdapat beberapa fitur seperti multi touch, live wallpaper, perubahan tampilan antarmuka dan dukungan browser untuk HTML5.

6) Android 2.2 Froyo

Pada tanggal 20 Mei 2010 Android kembali meluncurkan versi terbarunya yaitu versi 2.2 Froyo. Pada versi Froyo ini Android sudah mulai dikenal luas oleh vendor atau pabrikan ponsel. Pembaruan Android 2.2 membawa beberapa fitur unggulan seperti memperbesar gambar pada galeri dengan gestur, peningkatan fitur USB tethering dan hotspot WIFI serta dukungan animasi GIF pada web browser.

7) Android 2.3 Gingerbread

Pada versi ini tidak bisa diragukan lagi Android telah menjadi sistem operasi mobile yang populer. Kerjasama dengan pabrikan Samsung dalam membuat produk Samsung Galaxy Series semakin menambah kepopuleran Android. Versi 2.3 Gingerbread dirilis pada tanggal 6 Desember 2010 dengan menambah beberapa fitur seperti dukungan NFC, fitur copy atau paste dengan memilih kata melalui layar yang ditekan serta dukungan beberapa sensor lainnya.

8) Android 3.0 Honeycomb

Versi Android berikutnya adalah versi 3.0 Honeycomb yang diperkenalkan pada tanggal 22 Februari 2011. Versi Android ini dikhususkan untuk perangkat tablet PC, dengan membawa beberapa fitur yang ditujukan untuk komputasi bisnis pada tablet. Fitur yang dimiliki

Android Honeycomb seperti dukungan prosesor multi core, dukungan obrolan video dengan Google Talk dan percepatan saat berpindah aplikasi yang sedang berjalan dengan fitur multitasking recent apps.

9) Android 4.0 Ice Cream Sandwich

Tepat pada tanggal; 19 Oktober 2011 Android kembali meluncurkan versi terbarunya 4.0 Ice Cream Sandwich. Versi ini membawa fitur yang dimiliki oleh Honeycomb untuk bisa berjalan pada smartphone yang sebelumnya hanya ditujukan untuk tablet PC. Selain fitur yang sebelumnya ada di Honeycomb, versi 4.0 juga menambahkan beberapa fitur lain seperti perbaikan antarmuka dan kostumisasiwidget.

10) Android 4.1 Jelly Bean

Android 4.1 Jelly Bean dirilis pada tanggal 27 Juni 2012. Pengembangan versi Jelly Bean lebih berfokus terhadap peningkatan performa tampilan antarmuka. Fitur terbaru yang disematkan pada versi Jelly bean adalah keyboard yang bisa dikostumisasi oleh pengguna dan dukungan gestur pada keyboard, UI yang lebihsmooth, dukungan tampilan nirkabel, widget yang bisa diatur dan disesuaikan ukurannya.

11) Android 4.4 KitKat

Nama KitKat dipilih Android untuk digunakan pada versi terbarunya yaitu versi 4.4 yang diresmikan pada tanggal 31 Oktober 2013. Pada versi ini Android meningkatkan optimalisasi dengan memberikan fitur yang lebih baik. Beberapa fitur yang dibawa android KitKat seperti WebViews yang berbasis Chromium, pengoptimalan kinerja terhadap perangkat dengan spesifikasi rendah, dukungan sensorbatching dan step detector.

12) Android 5.0 Lollipop

Versi Android selanjutnya adalah versi 5.0 Lollipop yang dirilis pada tanggal 25 Juni 2014. Pada versi ini Android tidak hanya menjadi sistem operasi pada perangkat smartphone, namun juga telah berjalan pada

perangkat mobile lainnya seperti Android TV dan juga Google Fit. Beberapa fitur yang ditambahkan pada versi ini adalah pengendara interface yang mengikuti desain Google yaitu material design dan fitur factory reset protection untuk menjaga smartphone agar tidak di reset apabila hilang.

13) Android 6.0 Marshmallow

Android 6.0 Marshmallow di perkenalkan pada tanggal 5 Mei 2015. Fitur yang dibawa oleh versi Android Marshmallow adalah dukungan sensor sidik jari untuk mengakses smartphone, fasilitas menjalankan beberapa aplikasi pada tata letak layar dengan dukungan multi window, dukungan platform virtual reality, dan kemampuan dalam mengurangi pemakaian bandwidth pada mode data saver.

14) Android 7.0 Nougat

Diperkenalkan pada tanggal 19 Oktober 2016 Android 7.0 Nougat berfokus pada peningkatan performapengendara interface sehingga lebih intuitif dan penggunaan aplikasi secara bersamaan lebih banyak pada fitur multi window. Selain peningkatan fitur tadi, Android Nougat juga menambahkan beberapa fitur lain seperti dukungan cahaya malam atau mode malam, keyboard default yang dapat mengirim animasi GIF langsung dan dukungan panggilan multi-endpoint.

15) Android 8.0 Oreo

Nama Oreo dipilih Android untuk digunakan pada versi Android 8.0 yang diluncurkan pada bulan Agustus 2017. Pengendara interface pada Android Oreo lebih simpel agar memudahkan dalam mengakses aplikasi. Pembaruan pada versi Oreo membawa beberapa fitur seperti fitur Autofill yang memberikan kemudahan dalam mengisi formulir misal, dukungan gambar dalam gambar dan pengoptimalan booting agar lebih cepat.

#### 16) Android 9.0 Pie

Android 9.0 Pie merupakan versi Android terbaru yang dirilis pada bulan Agustus 2018. Fitur unggulan yang dimiliki oleh versi 9.0 Pie ini adalah kemampuan AI atau kecerdasan buatan. Dengan fitur AI smartphone kamu akan menganalisa dan mempelajari pola pemakaian kamu menggunakan smartphone secara otomatis. Selain itu fitur lainnya yang diusung seperti Adaptive Brightness yang akan menyesuaikan kecerahan layar secara otomatis dan dukungan pada ponsel bezel less.

#### 17) Android 10

Nama Android berikutnya yang paling terbaru tidak diberi nama seperti pendahulunya yang berkaitan dengan makanan manis. Versi Android terbaru diberi nama Android 10 untuk memperingati bahwa Android telah mencapai 1 dekade secara komersial. Versi Android 10 lebih berfokus pada penyempurnaan mode malam atau gelap serta peningkatan fitur sound amplifier untuk mengatur kualitas audio.

### 2.5.2 Android Software Development Kit

*Android Software Development Kit* atau SDK merupakan suatu alat yang berupa *Application Programming Interface* atau API yang berfungsi untuk melakukan suatu pengembangan aplikasi pada sistem operasi Android menggunakan bahasa pemrograman Java oleh para *developer* [20]. Terdapat berbagai macam *tools* yang telah disediakan pada SDK untuk bisa digunakan saat dalam proses pengembangan aplikasi seperti debugger, software libraries, emulator, dokumentasi, sample code, dan tutorial.

SDK pun bisa diartikan sebagai alat yang memiliki kegunaan untuk mengembangkan perangkat lunak yang juga memungkinkan dalam pembuatan aplikasi tertentu, kerangka kerja software, platform hardware, sistem komputer, video game console, sistem operasi layaknya suatu platform. Salah satu dari contoh terhadap Android SDK adalah Java SE Development kit yang menjadi bahasa pemrograman sering digunakan untuk mengembangkan aplikasi

Android, beberapa bahasa lainnya seperti C++, Go, dan kotlin termasuk bahasa yang ditetapkan oleh Google pada tahun 2017.

## 2.6 Java

Java merupakan bahasa pemrograman yang bisa dijalankan untuk berbagai perangkat komputer atau perangkat lainnya seperti dalam *smartphone*, bahasa ini dibuat oleh James Gosling pada saat masih bergabung di Sun Microsystems yang saat ini bagian dari Oracle. Bahasa pemrograman ini termasuk yang bersifat umum atau non-spesifik (*general purpose*), dan secara khusus agar bisa untuk memanfaatkan dependensi implementasi. Java banyak mengadopsi sintaksis yang berasal dari bahasa C dan C++ akan tetapi dengan sintaksis model objek lebih sederhana. Dengan fungsionalitas yang memungkinkan untuk berjalan dalam beberapa sistem operasi yang berbeda, java dikenal memiliki slogan, “*Tulis sekali, jalankan dimanapun*” [21].

Aplikasi berbasis java akan di *compile* kedalam *bytecode* dan dijalankan pada berbagai *Java Virtual Machine (JVM)*. JVM adalah virtual mesin yang terdapat pada perangkat keras untuk eksekusi *bytecode* java sehingga dapat berjalan diberbagai platform. Program java dieksekusi dengan Java Runtime Environment (JRE) untuk menjalankan aplikasi java di berbagai versi sistem operasi android.

Java merupakan bahasa pemrograman paling populer, dikarenakan memiliki berbagai kelebihan yang dirinci kedalam setiap fiturnya. Berikut ini fitur - fitur dari java antara lain: berorientasi objek, *multithreaded*, bersifat platform *independent*, bersifat *architectural-neutral*, sederhana, aman, kuat dan *powerful*, portabel, terinterpestasi, terdistribusi, dan dinamis. Saat ini java merupakan bahasa pemrograman yang sering digunakan dan secara luas dimanfaatkan dalam pengembangan berbagai jenis perangkat lunak aplikasi ataupun aplikasi.

## 2.7 Extensible Markup Language

*Extensible markup language* atau XML adalah bahasa yang berbasis teks dimana markup sebagai keperluan umum yang disarankan untuk keperluan

pertukaran data antar sistem yang beraneka ragam. XML merupakan kelanjutan *Hyper Text Markup Language* atau HTML yang merupakan bahasa standar. Teknologi XML adalah teknologi keturunan dari *Standard Generalized Markup Language* atau SGML yang dikembangkan pada tahun 1980-an. XML ini suatu bahasa yang berisikan kode berupa setiap tanda tertentu dengan aturan tertentu untuk memformat dokumen teks dengan tag sendiri agar dapat mudah dimengerti. Adanya dari XML berfungsi untuk mengidentifikasi dan menandai data terstruktur, dilengkapi mampu agar menyimpan data secara ringkas dan mudah diatur [13].

## 2.8 Android Studio

Android studio merupakan *Integrated Development Environment* atau IDE yang resmi digunakan sebagai pengembangan dalam aplikasi Android serta memiliki sifat *open source* atau gratis. Android Studio ini pertama kali diumumkan oleh Google pada 16 Mei 2013 dalam *event Google I/O Conference* di tahun 2013. Berawal dari *event* yang telah dilakukan tersebut, Android Studio menggantikan Eclipse sebagai IDE yang resmi untuk mengembangkan aplikasi *native* Android [22]. Setiap proyek di Android Studio berisi satu atau beberapa modul dengan file kode sumber dan file sumber daya. Jenis – jenis modul tersebut seperti modul aplikasi android, modul perpustakaan, dan modul google *app* engine. Beragam fitur yang terdapat dalam Android Studio bisa dimanfaatkan oleh setiap dari pengguna, antara lain :

- 1) Sistem versi berbasis Gradle yang fleksibel,
- 2) Emulator yang cepat dan kaya fitur,
- 3) Lingkungan yang menyatu untuk pengembangan bagi semua perangkat android,
- 4) Instant Run untuk mendorong perubahan ke aplikasi yang berjalan tanpa membuat APK baru,
- 5) Template kode dan integrasi GitHub untuk membuat fitur aplikasi yang sama dan mengimpor kode contoh,
- 6) Alat pengujian dan kerangka kerja yang ekstensif,

- 7) Alat Lint untuk meningkatkan kinerja, kegunaan, kompatibilitas versi, dan masalah yang lain,
- 8) Dukungan C++ dan NDK,
- 9) Dukungan bawaan untuk Google Cloud Platform, mempermudah pengintegrasian Google Cloud Messaging dan App Engine.

## 2.9 Xiaomi Mi Band

Xiaomi mi band merupakan produk *smartband* atau *smartwatch* yang diproduksi oleh Xiaomi yang memiliki berbagai macam fitur diantaranya terdapat fitur olahraga seperti *Treadmill*, latihan, lari *outdoor*, bersepeda, jalan, berenang di kolam, menghitung langkah, jarak, dan kalori yang terbakar. Kemudian selain fitur dalam hal olahraga terdapat juga fitur yang berfungsi memantau kesehatan diantaranya mengsetel target olahraga, notifikasi saat mencapai target olahraga, memantau ketika sedang tidur, memantau detak jantung, dan peringatan ketika diam yang terlalu lama [23].

Mi Band dilengkapi dengan adanya sensor detak jantung, *accelerometer*, dan *bluetooth* yang memungkinkan pengguna untuk bisa melacak setiap aktivitas yang telah dilakukan oleh pengguna. Mi Band keluaran terbaru saat ini yaitu Mi Band versi 6, pengguna banyak memilih dari produk yang dikeluarkan oleh Xiaomi dikarenakan harganya yang relatif lebih murah dibandingkan dengan produk *smartband* atau *smartwatch* yang lain [24]. Keunggulan dalam Xiaomi Mi Band seperti berikut :

- 1) Hadirnya fitur *near field communication* atau NFC yang akan memudahkan dalam perihal bertransaksi secara digital,
- 2) Layar amoled cukup luas sebagai penunjang layar yang mumpuni,
- 3) Terdapat pilihan enam jenis macam olahraga.

## 2.10 Sensor Detak Jantung

Sensor detak jantung atau *Sensor Heart Rate Monitor* (HRM) adalah sensor elektroda yang berfungsi untuk mengukur detak jantung dan menampilkan hasil

pengukuran tersebut secara kontinyu. Pada prinsipnya, cara kerja sensor ini adalah dengan mengukur perubahan volume darah yang melewati suatu organ tubuh, dimana perubahan volume darah akan menyebabkan perubahan pula pada intensitas cahaya yang melewati organ tersebut [25].

Sensor detak jantung pada dasarnya menerapkan teknik *photoplethysmography*, yaitu teknik optik *non-invasive* untuk mengukur perubahan volume darah berdasarkan variasi intensitas cahaya yang lewat atau dipantulkan oleh organ tubuh manusia. *Photoplethysmography* menggunakan sumber cahaya dari LED (*Light Emitting Diode*) dan PD (*Photodetector*) untuk mendeteksi adanya perubahan volume darah dalam pembuluh darah. Cahaya dapat ditransmisikan melalui jaringan kapiler (*capillary bed*). Saat pulsasi arteri (*arterial pulsations*) mengisi jaringan kapiler, perubahan volume pembuluh mengubah penyerapan (*absorption*), pantulan (*reflection*), dan hamburan (*scattering*) cahaya [26].

## 2.11 Javascript Object Notation

*Javascript object notation* atau JSON merupakan format pertukaran data yang ringan, mudah dibaca dan ditulis oleh manusia, serta mudah diterjemahkan dan dibuat (*generate*) oleh komputer. Format ini dibuat berdasarkan bagian dari bahasa pemrograman JavaScript, Standar ECMA-262 Edisi ke-3 – Desember 1999 [27].

JSON adalah format teks yang tidak bergantung pada bahasa pemrograman apapun karena menggunakan gaya bahasa yang umum digunakan oleh *programmer* keluarga C termasuk C, C++, C#, Java, JavaScript, Perl, Python, dan lain – lain. Oleh karena sifatnya tersebut, menjadikan JSON ideal sebagai bahasa pertukaran data. JSON terbuat dari dua struktur [27] :

- 1) Kumpulan pasangan nama/nilai. Pada beberapa bahasa, hal ini dinyatakan sebagai objek (*object*), rekaman (*record*), struktur (*struct*), kamus (*dictionary*), tabel hash (*hash table*), daftar berkunci (*keyed list*), atau *associative array*.

- 2) Daftar nilai terurutkan (an ordered list of values). Pada kebanyakan bahasa, hal ini dinyatakan sebagai larik (array), vektor (vector), daftar (list), atau urutan (sequence)

## 2.12 Application Programming Interface

*Application programming interface* atau API merupakan seperangkat definisi subrutin, protokol, dan alat yang digunakan untuk membangun perangkat lunak aplikasi. API dikatakan bagus disaat memudahkan pengembangan *program* komputer dengan menyediakan semua blok bangunan yang kemudian disatukan oleh *programmer* [28]. Isi yang terdapat dalam API merupakan barisan - barisan perintah yang tersimpan dalam bentuk *library*, dalam API memiliki beragam fungsi atau perintah - perintah untuk menggantikan bahasa yang digunakan pada *system calls* dengan bahasa yang lebih terstruktur dan mudah dimengerti oleh *programmer*. Keuntungan dalam penggunaan API seperti berikut :

- 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.

### 2.12.1 Google Fit

Platform google fit dirancang memudahkan pengguna *smartphone* atau *wearable device* agar mensinkronisasikan data aktifitas latihan fisik seperti berjalan, berlari, bersepeda, *push-up*, *sit-up*, *pull-up*, *squat* ke manajemen *cloud* kesehatan yang disediakan oleh google. Data yang disimpan bisa dijadikan referensi untuk pencapaian dan perkembangan *program* olahraga yang dilakukan, pengguna google fit memungkinkan untuk membuat target – target tertentu baik itu program harian, mingguan, atau bulanan [29]. Supaya google fit bisa diintegrasikan kedalam aplikasi android, dibutuhkan pustaka google play

*services* dalam *android software development kit* (SDK). Beberapa fitur dari google fit antara lain :

1) Sensor

Sensor akan mengakses ke aliran data sensor mentah dari sensor yang tersedia di perangkat android dan dari sensor yang tersedia di perangkat pendamping, seperti *wearable device*.

2) Recording API

Recording API menyediakan penyimpanan otomatis data. Google fit menyimpan data tersebut dari jenis yang ditentukan di *background*.

3) History

History akan mengakses kepada riwayat dan memungkinkan aplikasi melakukan operasi masal, seperti menyisipkan, menghapus, dan membaca data. Aplikasi juga dapat mengimpor data kumpulan ke google fit.

### 2.13 Firebase

Firebase adalah layanan yang disediakan oleh *google* untuk penyimpanan dan penyelarasan data kedalam aplikasi Android, iOS, atau web. Firebase ini merupakan solusi yang ditawarkan oleh *Google* agar mempermudah dan mempercepat pekerjaan *Developer*. Adanya Firebase, *developer* bisa fokus dalam hal mengembangkan aplikasi tanpa harus memberikan *effort* yang besar untuk urusan *backend* [30].

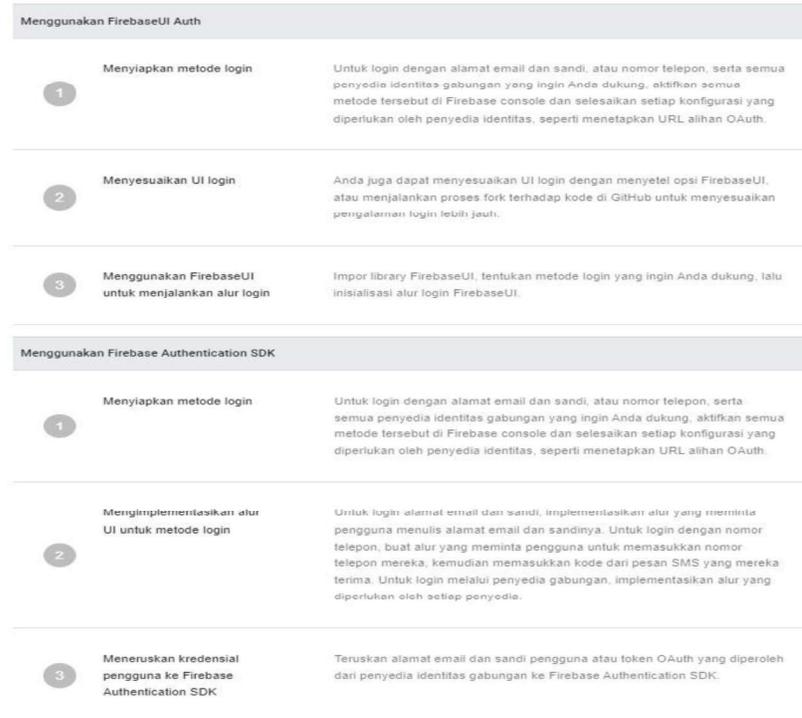
Firebase didirikan oleh Andrew Lee dan James Tamplin pada tahun 2011 dan *Google* mengakuisisi pada oktober 2014 [31]. Salah satu fitur yang dimiliki oleh Firebase merupakan *Real Time Database*, fitur ini yang memberikan sebuah NoSQL *database* yang bisa diakses secara realtime oleh pengguna aplikasi. Fitur pada firebase bukan hanya realtime *database*, jauh lebih dari itu firebase memiliki banyak fitur seperti *authentication*, *storage*, *hosting*, pemberitahuan dan lain - lain. Pada penelitian ini fitur Firebase yang akan digunakan adalah *Authentication*, *Firestore Realtime Database* dan *Cloud Messaging*.

### 2.13.1 Firebase Authentication

Firestore *authentication* adalah layanan siap pakai yang dimiliki oleh Firestore SDK. Firestore *authentication* memungkinkan aplikasi untuk melakukan autentikasi secara aman, sekaligus meningkatkan pengalaman *login* dan pengalaman aktivasi bagi *end-user*. Metode autentikasi yang digunakan meliputi *email* dan *password-based authentication*, *federated identity provider* integration (*authentication* yang menggunakan akun Google, Facebook, Twitter atau Github), *custom authentication system integration* hingga *anonymous authentication*.

Firestore *authentication* ini bekerja dengan cara mengirimkan server response dari firestore server berdasarkan *credential* yang dikirimkan oleh *client* ke firestore *server*. *Credential* tersebut dapat berupa alamat *email* dan *password* ataupun sebuah token OAuth dari sebuah *federated identity provider*. Melalui server response yang diterima dari firestore server, aplikasi dapat mengakses informasi dasar profil pengguna dan mengontrol akses pengguna terhadap produk atau layanan firestore yang terdapat pada aplikasi.

Firestore *authentication* dibangun untuk memberikan API yang mudah kepada pengembang yang digunakan untuk proses *sign-in* dari *federated providers* dengan skema *email* dan *password*, atau yang sudah terintegrasi dengan autentikasi yang sudah ada. Alur implementasi dari Firestore Authentication bisa dilihat pada gambar 2.1 sebagai berikut :

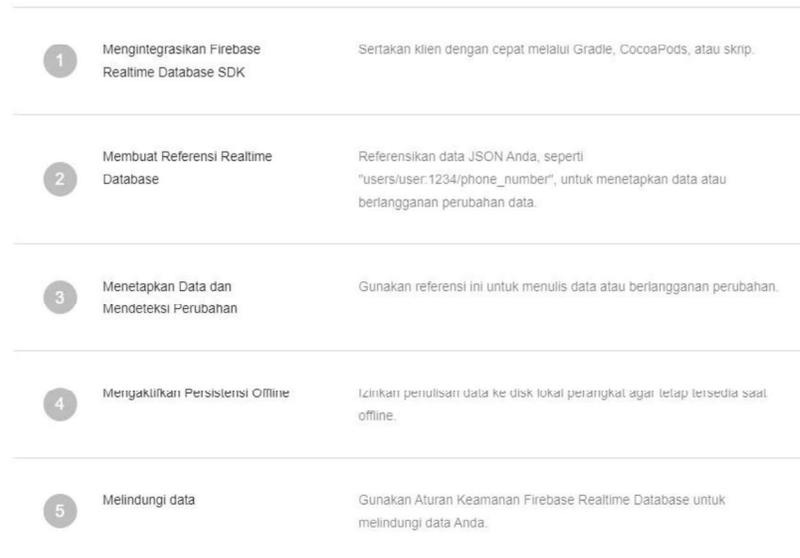


**Gambar 2.1 Alur Implementasi Firebase Authentication**

### 2.13.2 Firebase Realtime Database

Firestore realtime *database* merupakan database NoSQL yang di-host di *cloud*. Data disimpan sebagai JSON dan disinkronkan secara realtime terhadap setiap *client* yang terhubung. Ketika developer membuat aplikasi lintas *platform* dengan SDK Android, iOS, dan JavaScript, semua *client* akan berbagi sebuah *instance* Realtime *database* dan menerima *update* data terbaru secara otomatis [32]. Firestore realtime *database* memungkinkan *developer* untuk membuat aplikasi kolaboratif dan kaya fitur dengan menyediakan akses yang aman ke *database*.

Terdapat menyediakan bahasa aturan berbasis ekspresi yang fleksibel, atau disebut juga aturan keamanan firestore realtime *database* agar menentukan metode strukturisasi data dan kapan data dapat dibaca atau ditulis. Ketika diintegrasikan dengan firestore *authentication*, *developer* dapat menentukan siapa yang memiliki akses ke data tertentu dan bagaimana mereka dapat mengaksesnya [32]. Alur implementasi dari Firestore Realtime Database bisa dilihat pada gambar 2.2 sebagai berikut :

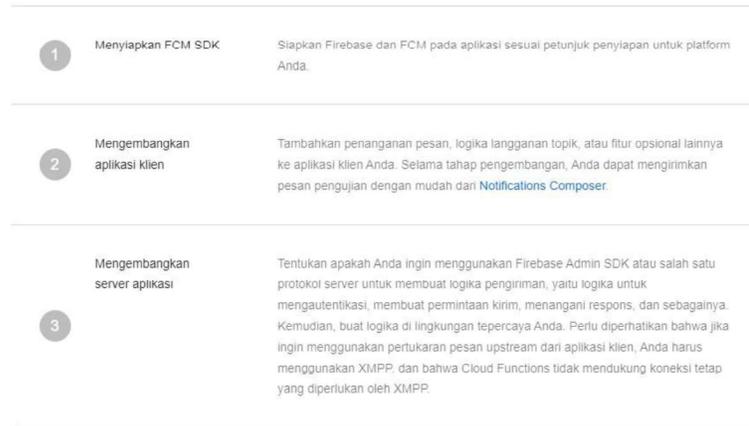


**Gambar 2.2 Alur Implementasi Firebase Realtime Database**

### 2.13.3 Firebase Cloud Mesagging

Firestore *cloud messaging* (FCM) adalah solusi dalam pengiriman pesan lintas *platform* untuk mengirim pesan dan layanan pemberitahuan yang disediakan oleh Google secara gratis. Layanan FCM juga menyediakan fungsi untuk melakukan *push notification*, yaitu notifikasi yang muncul dibagian atas layar *smartphone* dan dapat diseret ke bawah, untuk mengakses pesan lengkapnya pengguna cukup menekan pesan yang tampil pada notifikasinya.

Penggunaan fitur *push notification* dengan FCM sangat membantu karena FCM akan mengirimkan notifikasi secara realtime [33]. Fitur – fitur yang diberikan oleh FCM sebenarnya tidak terlalu jauh berbeda dengan Google *Cloud Messaging*. Dengan FCM kita bisa memberikan pemberitahuan dan membuat komunikasi dua arah antara perangkat. Alur implementasi dari FCM bisa dilihat pada gambar 2.3 sebagai berikut ;



**Gambar 2.3 Alur Implementasi Firebase Cloud Messaging**

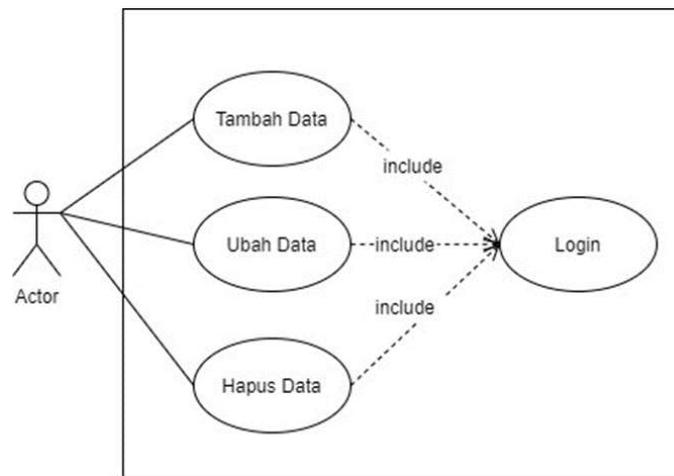
## 2.14 Unified Modelling Language

*Unified modeling language* atau UML merupakan termasuk dalam bahasa pemodelan spesifikasi standar sistem digunakan sebagai mendokumentasikan keadaan dari perangkat lunak yang akan dibangun [34]. UML digunakan untuk menjadi standar dalam visualisasi, perancangan dan juga pendokumentasian system, berbagai fungsi yang terdapat dalam UML agar pengembang bisa 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 yang dapat mudah dimengerti oleh setiap orang.

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 tersebut. Ada diagram – diagram yang perlu diketahui terhadap dari penggunaan UML dimana sebagai alat bantu dalam proses untuk pemodelan diantaranya adalah *use case diagram*, *activity diagram*, *sequence diagram*, dan *class diagram* [22].

### 2.14.1 Use Case Diagram

*Use case diagram* merupakan diagram yang menggambarkan kelakuan terhadap sistem yang dibuat, adanya sebuah interaksi antara satu atau lebih aktor. Sistem yang dibangun digunakan untuk memenuhi satu atau lebih dari kebutuhan pengguna [34]. Secara sederhana, *use case diagram* digunakan untuk memahami fungsi apa saja yang ada di dalam sebuah sistem dan siapa saja yang dapat menggunakan fungsi-fungsi tersebut. Contoh dari *use case diagram* bisa dilihat pada gambar 2.4 sebagai berikut :



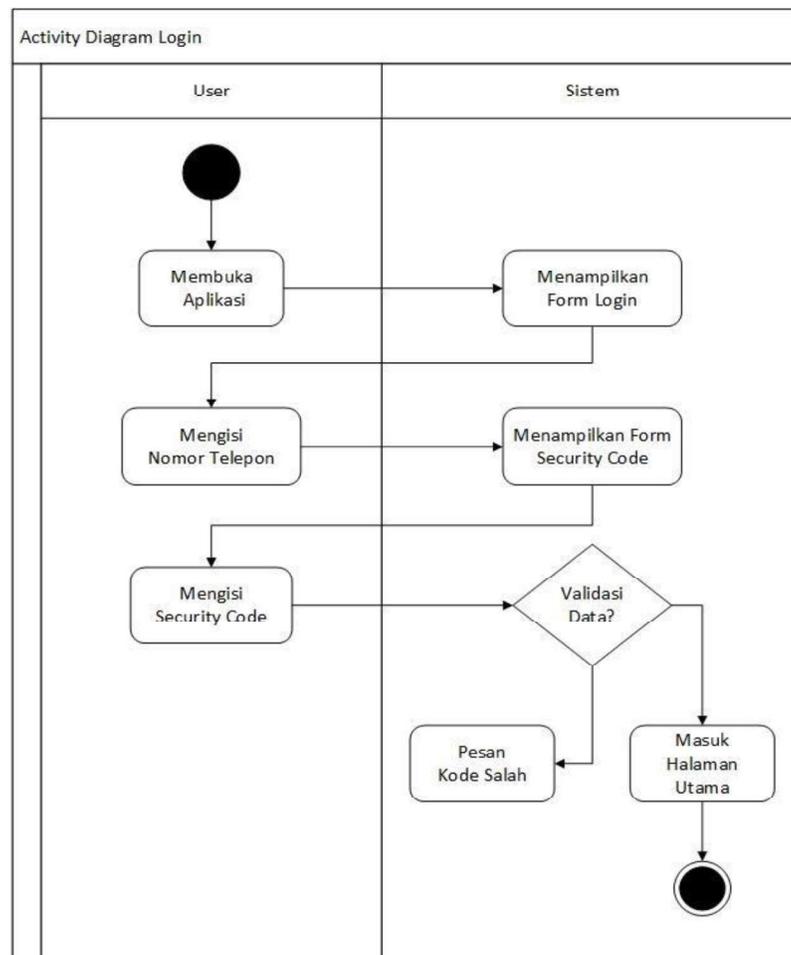
Gambar 2.4 Contoh Use Case Diagram

Menggambarkan interaksi antara sistem dan lingkungannya, terdapat beberapa simbol yang digunakan pada *use case diagram*. Use case yang merepresentasikan tugas tertentu yang melibatkan interaksi dengan lingkungannya dilambangkan dengan elips. Sementara aktor yang terlibat dalam *use case* tersebut dilambangkan dengan sosok tongkat. Terdapat beberapa jenis relasi yang terjadi, baik antar *use case*, antar aktor, maupun antara *use case* dan aktor. Relasi yang terjadi antar *use case*, terdapat istilah *extend*, *include* dan generalisasi. Relasi *<<extend>>* digunakan saat sebuah *use case* secara kondisional menambahkan tahapan lain dari *use case* pusat (*base use case*). *Use case* pusat ini masih dapat berdiri sendiri tanpa *use case* tambahan tersebut (*extend use case*). Generalisasi juga dapat terjadi antar aktor. Selain bentuk relasi yang telah dijelaskan sebelumnya, terdapat pula relasi asosiasi yang terjadi antara *use case* dan aktor yang menandakan bahwa terjadi interaksi atau

komunikasi antara aktor dan *use case*. Sebuah *use case* memiliki paling tidak satu relasi asosiasi kepada satu aktor dan satu aktor memiliki paling tidak satu relasi asosiasi kepada satu *use case*.

### 2.14.2 Activity Diagram

Diagram aktivitas adalah diagram yang menggambarkan memodelkan aliran kerja atau *workflow* dari urutan aktivitas dalam suatu proses yang mengacu pada *use case diagram* yang ada [35]. Contoh dari *activity diagram* bisa dilihat pada gambar 2.5 sebagai berikut :



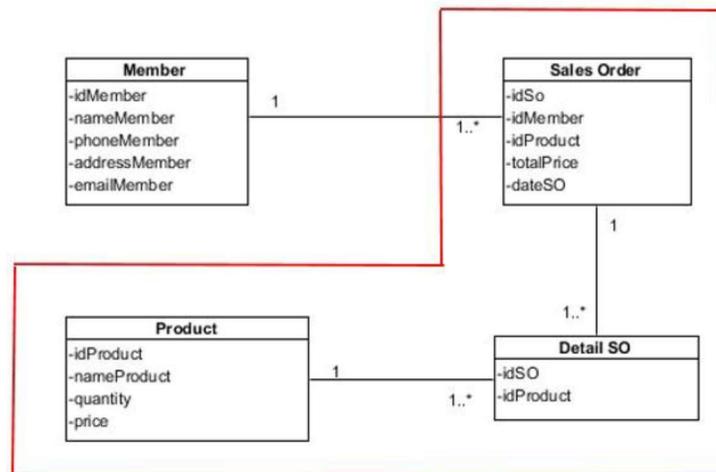
Gambar 2.5 Contoh Activity Diagram

*Activity Diagram* juga digunakan sebagai mendefinisikan atau mengelompokkan aliran tampilan dari sistem dengan mendeskripsikan aktor, tindakan yang perlu dilakukan, dan kapan tindakan tersebut terjadi. Diagram

aktivitas memiliki komponen dalam bentuk tertentu yang dihubungkan dengan tanda panah, dan tanda panah tersebut mengarah pada urutan aktivitas sistem dari awal hingga akhir.

### 2.14.3 Class Diagram

Diagram kelas adalah diagram yang menggambarkan relasi atau hubungan yang dimiliki oleh antar kelas yang di dalamnya terdapat atribut dan fungsi dari suatu objek. Semua jenis objek yang berbeda dalam sistem berinteraksi satu sama lain dengan adanya penggunaan dari *class diagram* [34]. Contoh dari class diagram bisa dilihat pada gambar 2.7 sebagai berikut :



**Gambar 2.6 Contoh Class Diagram**

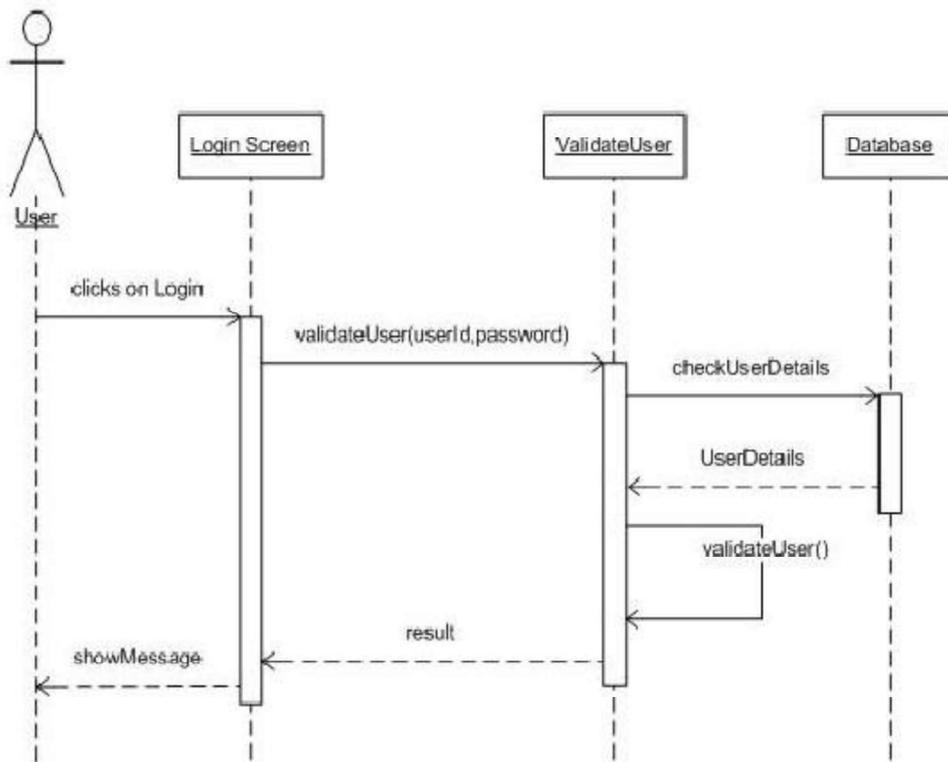
Diagram kelas sendiri merupakan istilah yang mendeskripsikan sekelompok objek yang semuanya memiliki peran serupa dalam sistem. Sekelompok objek ini terdiri atas fitur struktural yang mendefinisikan apa yang diketahui *class* dan fitur operasional yang mendefinisikan apa yang bisa dilakukan oleh *class*. Meski demikian, terdapat beberapa fungsi lainnya dari *class diagram* sebagai berikut ini :

- 1) Menunjukkan struktur statis pengklasifikasi dalam suatu sistem.
- 2) Memberikan notasi dasar untuk diagram struktur lain yang ditentukan oleh UML.

- 3) Dapat digunakan *business analyst* untuk membuat model sistem dari perspektif bisnis.

#### 2.14.4 Sequence Diagram

*Sequence diagram* merupakan diagram yang menggambarkan pengaplikasian dari masing - masing skenario *use case* yang ada pada sistem. Biasanya menjelaskan interaksi mana yang akan dipicu ketika kasus penggunaan tertentu dijalankan dan dalam urutan apa interaksi tersebut akan terjadi [34]. Contoh dari *sequence diagram* bisa dilihat pada gambar 2.6 sebagai berikut :



**Gambar 2.7 Contoh Sequence Diagram**

*Sequence Diagram* menggambarkan bagaimana entitas dalam sistem berinteraksi, termasuk pesan yang digunakan selama interaksi. Semua pesan dijelaskan dalam urutan pelaksanaannya.