

BAB 2

TINJAUAN PUSTAKA

2.1 Stres

Stres adalah masalah kesehatan yang dapat berdampak negatif pada kesehatan fisik dan mental, mempengaruhi kinerja dan keseimbangan emosional seseorang [16]. Stres merupakan elemen yang tak bisa dihindari dan jika tidak diobati dapat menyebabkan penurunan kesehatan, kesejahteraan individu dan situasi sosial-ekonomi.

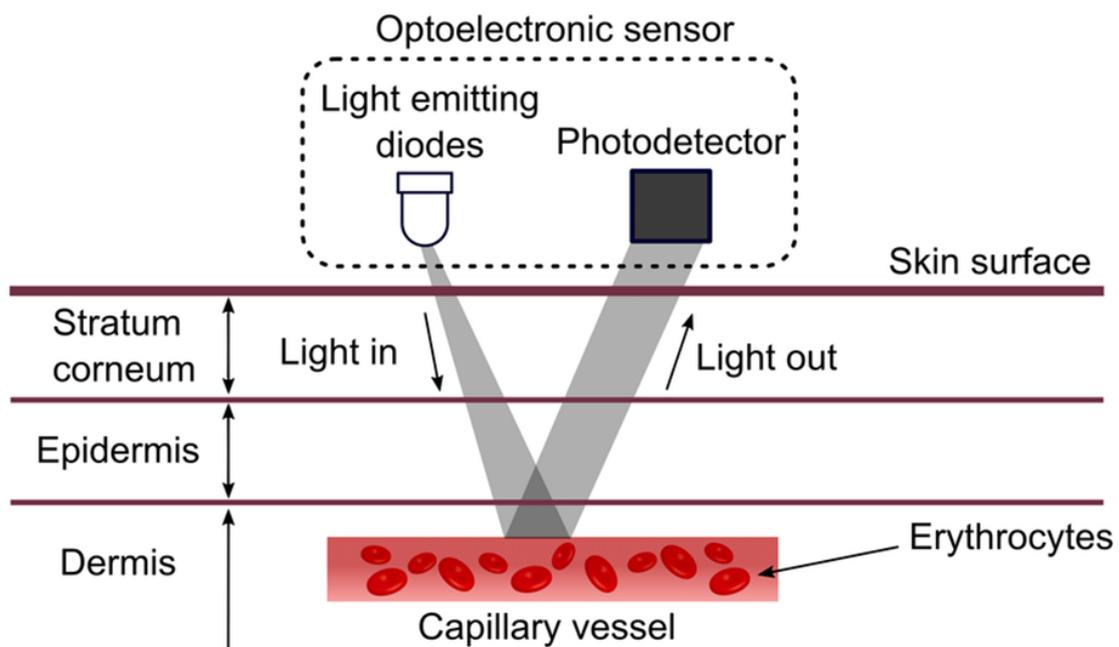
Stres itu muncul dari suatu reaksi fisik dan psikis terhadap setiap tuntutan yang menyebabkan ketegangan dan gangguan stabilitas kehidupan sehari-hari. Stres berkaitan dengan kenyataan yang tidak sesuai dengan harapan atau situasi yang menekan. Kondisi stres ini dapat mengakibatkan gejala-gejala seperti sulit tidur, sedih, sering menangis, takut, cemas, terlihat mudah marah [2]. Stres juga dapat memunculkan perasaan seperti cemas, marah, dan frustrasi [17].

2.2 Tingkat Stres

Tingkat stres adalah kondisi psikologis dan fisiologis yang terjadi sebagai respons terhadap tekanan atau tantangan yang dialami oleh individu. Beberapa teori psikologi memberikan wawasan mendalam tentang faktor-faktor yang mempengaruhi tingkat stres seseorang, memungkinkan pemahaman yang lebih baik tentang dinamika stres dalam kehidupan sehari-hari [18]. Tingkat stres ini bisa dihitung menggunakan analisa *Heart Rate Variability (HRV) domain frequency* berupa *low frequency (LF)* yang meningkat pada stres berat dan nilai *high frequency (HF)* [7], dimana pada kasus ini terdapat beberapa kondisi atau gejala yang menentukan tingkat stres. Hasil dari analisa ini nanti akan dibagi menjadi 3 level yaitu stress ringan, sedang, dan tinggi [7].

Cara pengambilan *Heart Rate Variability* tersebut menggunakan sensor yang terpasang pada *Smartwatch* yang akan digunakan pada penelitian ini, yaitu sensor *Photoplethysmography (PPG)*. Sensor ini bekerja dengan cara menembakkan 2 sinar cahaya sensor berwarna hijau dan merah. Masing-masing dari warna cahaya tersebut

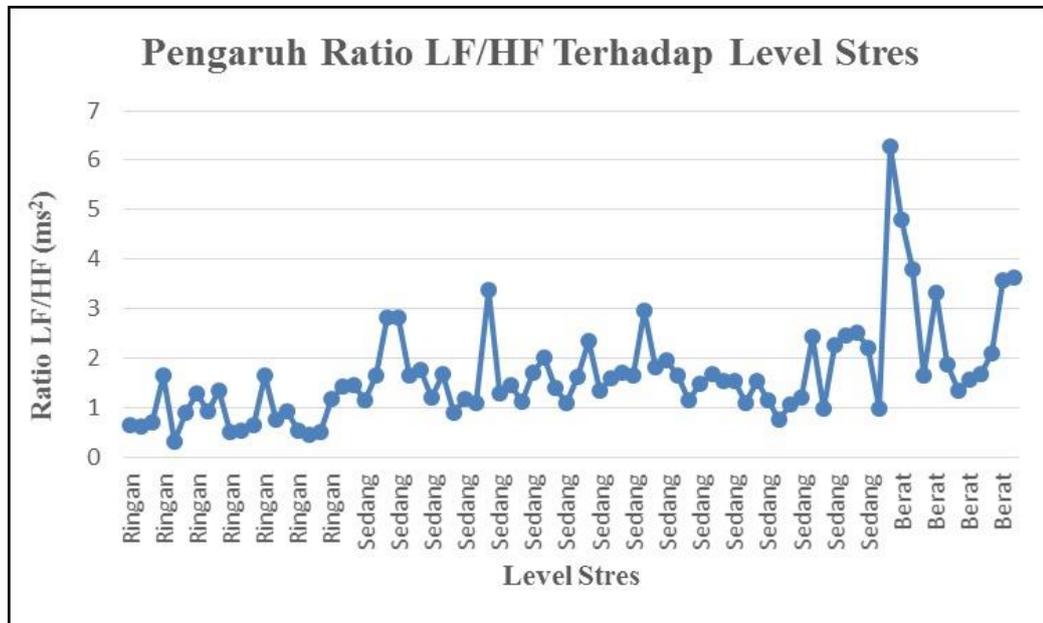
memiliki kegunaannya tersendiri, yaitu warna hijau untuk mendeteksi detak jantung dan oksigen sedangkan warna merah untuk mendeteksi HRV itu sendiri melalui pergerakan darah pada pembuluh darah kapiler. Cahaya tersebut memiliki kelebihan dan kekurangan tersendiri, yaitu cahaya warna hijau bisa terus mendeteksi pembuluh darah kapiler meskipun tangan bergerak dengan cepat namun tidak akurat, dan sebaliknya pada cahaya warna merah itu sulit mendeteksi pembuluh darah kapiler jika terlalu banyak bergerak namun dengan hasil yang jauh lebih akurat dari pada cahaya hijau. Pada penelitian ini cahaya yang akan digunakan sebagai alat penelitian adalah cahaya merah untuk mendeteksi *Heart Rate Variability* sebagai data yang akan dikalkulasikan oleh smartwatch menjadi tingkat stres. Berikut adalah penggambaran cara kerja cahaya sensor *Photoplethysmography* yang dapat dilihat pada Gambar 2. 1 berikut.



Gambar 2. 1 Cara Kerja Sensor Photoplethysmograph

Berikut adalah sumber *Low Frequency* pada jurnal yang dijadikan sebagai referensi Metode Analisa *Heart Rate Variability* dapat dilihat pada Gambar 2. 2 berikut.

Berikut adalah sumber perbandingan *Low Frequency* dengan *High Frequency* pada jurnal yang dijadikan sebagai referensi Metode Analisa *Heart Rate Variability* dapat dilihat pada Gambar 2. 4 berikut.

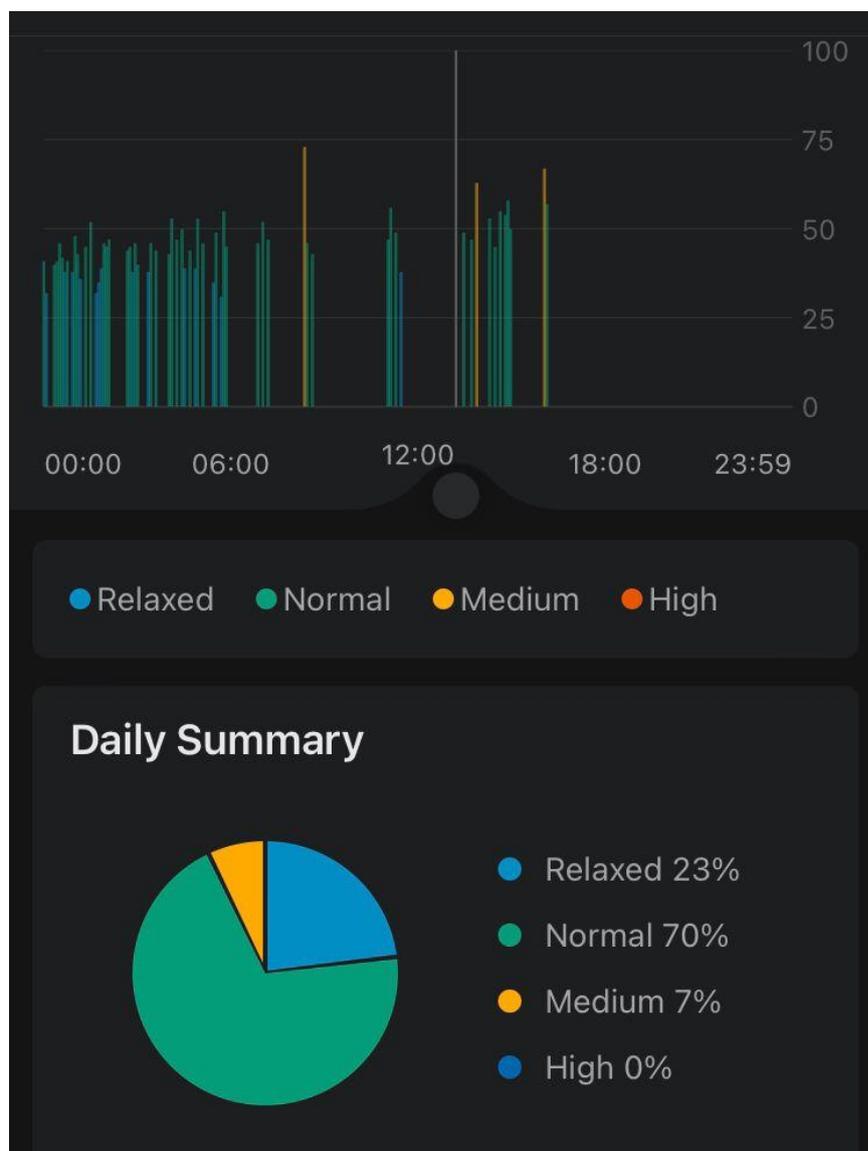


Gambar 2. 4 Pengaruh Rasio LF dengan HF terhadap level stress [7]

Stress biasanya muncul karena disebabkan oleh berbagai tuntutan, seperti adanya tekanan, sedang menghadapi tantangan, atau yang sedang berusaha menyelesaikan suatu permasalahan atau tuntutan [2]. Teori stres umum yang dilakukan oleh penguji menjelaskan konsep bahwa respons tubuh terhadap stresor. Tahap alarm melibatkan aktivasi respons "*fight or flight*" sebagai reaksi awal terhadap tekanan. Fase resistensi melibatkan upaya adaptasi berkelanjutan untuk mengatasi stresor. Apabila stresor terus berlanjut tanpa pemulihan yang memadai, individu dapat mencapai tahap kelelahan dan juga trauma mental. Gejala yang didapatkan diantaranya seperti sedih, sering menangis, takut, sering merasa cemas [2], depresi, perubahan suasana hati, insomnia, atensi, gangguan hiperaktif, dan kemarahan [18]. Teori ini membantu menjelaskan bagaimana tubuh manusia beradaptasi terhadap stres dalam jangka waktu tertentu, dan bagaimana terlalu lama berada dalam kondisi stres dapat berdampak negatif pada kesehatan.

Untuk tingkat stres sendiri pada penelitian ini yang akan digunakan itu berdasarkan dua sumber yaitu Kesimpulan dari jurnal yang membahas Analisa *Heart*

Rate Variability [7], dan juga berdasarkan aplikasi dari *wearable device smartband* yang digunakan yaitu *Amazfit 5 Bip*, lalu diklasifikasikan menjadi 4 klasifikasi yaitu *relaxed* (santai) mulai dari angka 0-40 yang ditandai dengan warna biru, *normal* (ringan) mulai dari angka 41-60 yang ditandai dengan warna hijau, *medium* (sedang) mulai dari angka 61-80 yang ditandai dengan warna kuning, dan *high* (tinggi) mulai dari angka 81-100 yang ditandai dengan warna oranye seperti pada Gambar 2. 5 berikut.



Gambar 2. 5 Gambar Tingkat Stress Pada Aplikasi Amazfit

2.3 Wearable Device Smartwatch

Wearable device smartwatch adalah perangkat teknologi yang dikenakan di pergelangan tangan dan memiliki sensor dengan berbagai fungsi, seperti pemantauan aktivitas fisik, pengukuran detak jantung, *monitoring* tidur [19]. *Smartwatch* mengadopsi teknologi sensor non-invasif, yang memungkinkan pengguna untuk memantau kesehatan mereka tanpa perlunya intervensi fisik yang signifikan. Penggunaan teknologi non-invasif adalah salah satu aspek kunci dalam kenyamanan dan popularitas smartband sebagai alat pemantauan kesehatan yang praktis. Data penting yang akan diambil dari *smartwatch* pada penelitian ini adalah tingkat stres yang sudah diolah sedemikian rupa oleh algoritma pada *smartwatch* tersebut sampai menjadi angka pasti, lalu diintegrasikan dengan aplikasi *smartphone* untuk penggunaan lebih lanjut. *Smartwatch Amazfit 5 Bip* ini juga memiliki fitur untuk mendeteksi tingkat stres pengguna yang juga sudah dilengkapi dengan ukuran tingkat stressnya yang dibagi menjadi 4 yaitu *Relaxed* mulai dari angka 0-40, *Low* mulai dari angka 41-60, *Medium* mulai dari angka 61-80, dan *High* mulai dari angka 81-100. *Wearable device* yang akan digunakan pada penelitian ini adalah *Smartband Amazfit Bip 5* yang dapat dilihat pada Gambar 2. 6 berikut.



Gambar 2. 6 Wearable Device Smartwatch Amazfit Bip 5

2.4 Mental Stres

Mental stres adalah suatu kondisi medis yang memengaruhi pemikiran, perasaan, atau perilaku sehingga menyebabkan kesulitan dalam kehidupan sehari-hari. Beberapa tanda dan gejala mental stres adalah perubahan signifikan dalam kepribadian, ketidakmampuan untuk menyelesaikan masalah dan tugas sehari-hari [4], dan gangguan kecemasan berlebih yang merupakan gejala paling sering terjadi sehingga menimbulkan mental stres [1]. Sedangkan mental stres ini jika tidak cepat disembuhkan dalam jangka panjang, dapat menimbulkan penyakit yang lebih serius seperti depresi, *cardiovascular disease*, *cancer*, dan *respiratory symptoms*.

2.5 Manajemen Kesehatan Mental Stres

Manajemen kesehatan mental stres merupakan cara untuk membantu penderita stres untuk memajemen tingkat stressnya, sehingga penderita stres bisa menurunkan tingkat stres dengan melakukan metode-metode yang sudah disarankan oleh jurnal yang membahas kesehatan mental dan juga modul kesehatan. Beberapa metode yang disarankan untuk menurunkan tingkat stress seperti teknik pernafasan zhen qi [20], aktivitas ringan hingga olahraga berat [21], serta aktivitas yang

disarankan oleh Kementerian Kesehatan Republik Indonesia untuk mengatasi stress dan mencapai jiwa yang sehat [22].

2.6 Bluetooth

Bluetooth adalah teknologi *wireless* dengan daya rendah dari *Bluetooth Special Industrial Group* (SIG) yang memungkinkan perangkat elektronik untuk berkomunikasi dan bertukar data dengan kecepatan kurang lebih 3 megabit per detik dan jangkauan 10 meter [23]. Bluetooth merupakan salah satu teknologi nirkabel yang paling umum digunakan saat ini. Sebagai teknologi nirkabel, Bluetooth memungkinkan perangkat untuk berkomunikasi tanpa menggunakan kabel fisik. Hal ini memfasilitasi konektivitas antar perangkat dengan cara yang lebih mudah dan praktis.

Bluetooth beroperasi di spektrum frekuensi 2,4 GHz yang juga digunakan oleh banyak perangkat lain seperti Wi-Fi [23]. Teknologi ini menggunakan teknik spread spectrum untuk mengurangi interferensi dengan perangkat lain dalam spektrum yang sama. Bluetooth menggunakan teknik modulasi frequency-hopping spread spectrum (FHSS) untuk mengirimkan data secara aman dan andal. FHSS memungkinkan perangkat Bluetooth untuk beralih secara dinamis antara beberapa saluran frekuensi, mengurangi kemungkinan gangguan atau intersepsi data. Ikon *Bluetooth* dapat dilihat pada Gambar 2. 7 berikut.



Gambar 2. 7 Bluetooth

2.7 Android

Android adalah sistem operasi berbasis Linux yang dikembangkan oleh Google, dirancang terutama untuk perangkat mobile seperti *smartphone* dan *tablet*. Android menyediakan platform *open source* yang memungkinkan *developer* untuk

menciptakan aplikasi yang bisa dijalankan di berbagai perangkat dari berbagai *developer* [24]. Sistem operasi ini sangatlah fleksibel, sebagai *software open source*, ekosistem aplikasinya yang sangat luas, dan didistribusikan melalui *Google Play Store*.

Android mencakup berbagai komponen seperti *framework* aplikasi, *runtime*, dan *driver* perangkat keras. Antarmuka pengguna Android sangat intuitif dan dapat dikustomisasi, memungkinkan pengguna untuk mengubah tampilan dan fungsionalitas perangkat sesuai preferensi masing-masing. Android juga mendukung integrasi layanan *Google* seperti *Gmail*, *Google Maps*, dan *Google Assistant*, serta kompatibilitas dengan berbagai aplikasi pihak ketiga yang mendukung berbagai kebutuhan dan minat pengguna. Karena sifatnya yang terbuka dan dukungan perangkat lunaknya luas bagi komunitas *developer*, Android telah menjadi sistem operasi mobile paling populer di dunia. Sejak perilisan pertamanya hingga sekarang terdapat beberapa versi dari android itu sendiri, detil dari versi android dapat dilihat pada Tabel 2. 1 berikut.

Tabel 2. 1 Rilis Versi Android

| Tahun Rilis | Nama Kode | Versi | API Level |
|----------------|-------------|---------------------|-------------------------|
| September 2008 | Alpa | 1.0 | API Level 1 |
| Februari 2009 | Beta | 1.1 | API Level 2 |
| April 2009 | Cupcake | 1.5 | API Level 3, NDK 1 |
| September 2009 | Donut | 1.6 | API Level 4, NDK 2 |
| - | Éclair | 2.0 – 2.0.1 – 2.1 | API Level 5, 6, 7 |
| - | Froyo | 2.2 | API Level 8, NDK 8 |
| Desember 2010 | Gingerbread | 2.3 – 2.3.2 – 2.3.3 | API Level 9, 10 |
| Februari 2011 | Honeycomb | 3.0 – 3.1 – 3.2 | API Level 11, 12, 13 |
| Oktober 2011 | Sandwich | 4.0.1 – 4.0.4 | API Level 14, 15 |

| | | | |
|---------------------|-------------------|-----------------|----------------------|
| Juni 2012 | Jelly Bean | 4.1 – 4.2 – 4.3 | API Level 16, 17, 18 |
| September 2013 | KitKat | 4.4 | API Level 19 |
| Juni 2014 | Lollipop | 5.0 – 5.1 | API Level 21, 22 |
| Agustus 2015 | Marshmallow | 6.0 | API Level 23 |
| Agustus 2016 | Nougat | 7.0 – 7.1 | API Level 24, 25 |
| Agustus 2017 | Oreo | 8.0 – 8.1 | API Level 26, 27 |
| Agustus 2018 | Pie | 9 | API Level 28 |
| September 2019 | Android Q | 10 | API Level 29 |
| September 2020 | Red Velvet | 11 | API Level 30 |
| Oktober 2021 | Snow Cone | 12 | API Level 31, 32 |
| Februari 2022 | Tiramisu | 13 | API Level 33 |
| Oktober 2023 | Upside-Down-Cake | 14 | API Level 34 |
| Agustus 2024 (Beta) | Vanilla Ice Cream | 15 (Beta) | API Level 35+ (Beta) |

Aplikasi yang akan dibangun pada penelitian ini akan menggunakan target minimal SDK API Level 26 yaitu *Oreo*, dan target API Level 35 yaitu *Vanilla Ice Cream*.

2.8 Android Studio

Android Studio adalah kode editor IDE (*Integrated Development Environment*) yang digunakan oleh para pengembang untuk membuat aplikasi Android yang gratis dan *open source* [25]. Android Studio dibangun pada platform Android Development Kit (SDK) yang menyediakan berbagai alat seperti emulator, perpustakaan, dan sumber daya untuk mengembangkan aplikasi Android. SDK Android menyediakan segala sesuatu yang diperlukan untuk merancang, mengembangkan, menguji, dan menerapkan aplikasi Android.

Android Studio dilengkapi dengan editor kode yang kuat yang mendukung bahasa pemrograman *Kotlin* dan *Java*, yang merupakan bahasa resmi untuk pengembangan aplikasi Android. Editor menyediakan fitur seperti penyorotan

sintaks, penyelesaian kode, navigasi cepat, dan integrasi kontrol versi untuk meningkatkan produktivitas pengembang. Ikon dan logo Android Studio dapat dilihat pada Gambar 2. 8 berikut.



Gambar 2. 8 Android Studio

Android Studio terintegrasi dengan layanan *Google* seperti *Firebase*, *Google Cloud Platform*, dan *Google Play Services*. Integrasi ini memungkinkan pengembang aplikasi untuk mengakses berbagai fitur dan layanan *Google* yang berguna, termasuk analitik, basis data, otentikasi, dan distribusi aplikasi.

2.9 Visual Studio Code

Visual Studio Code adalah sebuah IDE ringan yang dapat digunakan untuk mengembangkan berbagai jenis aplikasi, termasuk aplikasi web, aplikasi desktop, dan pengembangan berbasis cloud. *VS Code* menyediakan berbagai fitur yang kuat untuk mendukung pengembangan perangkat lunak [26].

Seperti fitur debugging yang terintegrasi untuk berbagai bahasa pemrograman. Pengembang aplikasi juga dapat menginspeksi variabel, dan melacak kode dengan mudah menggunakan alat debugging yang disediakan. Ikon dan Logo *Visual Studio Code* dapat dilihat pada Gambar 2. 9 berikut.



Gambar 2. 9 Visual Studio Code

2.10 Flutter

Flutter framework perangkat lunak open-source yang dikembangkan oleh *Google* [27]. Flutter menggunakan bahasa pemrograman Dart sebagai bahasa utamanya [27]. *Dart* adalah bahasa yang efisien, modern, dan dikembangkan oleh *Google* [27]. *Dart* dirancang untuk membuat pengembangan aplikasi lintas platform lebih mudah dengan fitur-fitur seperti sintaks yang mudah, manajemen memori otomatis, dan performa yang cepat [27]. Ikon dan Logo *Flutter* dapat dilihat pada Gambar 2. 10 berikut.



Gambar 2. 10 Flutter

Salah satu fitur yang paling diminati dari *Flutter* adalah *hot reload*, yang memungkinkan untuk melihat perubahan yang dilakukan pada kode secara langsung di emulator atau perangkat fisik. *Hot reload* mempercepat siklus pengembangan dan memungkinkan untuk mengembangkan aplikasi lebih cepat dan efisien.

Keunggulan Flutter lainnya yang akan digunakan untuk membangun aplikasi pada penelitian ini adalah *flutter material*, *flutter local notifiicator*, *flutter background*, *flutter storage*, dan *flutter chart*.

2.11 Dart

Dart adalah bahasa pemrograman modern yang dikembangkan oleh *Google* [27], teroptimisasi dan dapat digunakan di berbagai platform terutama untuk pengembangan aplikasi *web* dan *mobile* [27]. *Dart* merupakan bahasa pemrograman yang bersifat statis dan dinamis. *Dart* memiliki fitur tipe data statis yang memungkinkan pengembang untuk mendeteksi kesalahan dalam kode saat kompilasi [27]. Namun, *Dart* juga mendukung tipe data dinamis yang memungkinkan untuk menulis kode yang lebih fleksibel. Fitur *dart* yang akan digunakan untuk membangun aplikasi pada penelitian ini adalah *dart:async*, *dart:http*, *dart:intl*, dan *dart:convert*. Ikon dan Logo *Dart* dapat dilihat pada Gambar 2. 11 berikut.



Gambar 2. 11 Dart

2.12 Firebase

Firebase adalah platform pengembangan aplikasi yang menyediakan berbagai API yang terintegrasi dengan layanan Google untuk mendukung pengembangan aplikasi mobile dan web [28]. Salah satu fitur yang terkenal adalah *Firebase Realtime Database* yang memungkinkan pengembang untuk mengakses dan mengelola data secara waktu nyata [28]. Database ini menyediakan API yang sederhana untuk menyimpan, mengambil, dan menyinkronkan data antar perangkat. Penggunaan API ini memungkinkan pengembang untuk membangun aplikasi yang responsif dan dinamis [28]. *Firebase Database Realtime* merupakan salah satu sifat *database* yang sangat disukai oleh para *developer*, dikarenakan sifat penyimpanan datanya yang berbentuk dokumen bukan tabel dan juga cara pengolahan datanya yang berbentuk *No SQL (structured query language)* sehingga *developer* tidak perlu kesulitan untuk membangun perancangan basis data pada aplikasinya. Fitur *firebase* yang akan digunakan untuk membangun aplikasi pada penelitian ini adalah *Firebase*

Database Realtime sebagai penyimpanan database, *Firestore Auth* sebagai otentikasi dan penyimpanan akun yang didaftarkan, *Firestore Storage* sebagai penyimpanan foto profil yang akan diupload oleh user pada. Ikon dan Logo *Firestore* dapat dilihat pada Gambar 2. 12 berikut.



Gambar 2. 12 Firebase

2.13 Application Programming Interface (API)

Application Programming Interface, adalah kumpulan aturan dan protokol yang memungkinkan perangkat lunak untuk berkomunikasi dengan perangkat lunak lainnya [29].

Fungsi utama API adalah menyediakan cara untuk program atau aplikasi menggunakan fungsionalitas atau layanan yang disediakan oleh perangkat lunak lainnya. Dengan menggunakan API, pengembang dapat mengintegrasikan fitur atau data dari satu aplikasi ke dalam aplikasi lainnya.

2.10.1 Zepp OS API

Zepp OS adalah sistem operasi yang dikembangkan oleh Zepp Health, yang sebelumnya dikenal sebagai Huami, dan digunakan dalam perangkat *wearable* seperti *smartwatch Amazfit*. Zepp OS menyediakan berbagai API yang memungkinkan *developer* untuk mengakses dan memanfaatkan fitur-fitur *device* secara maksimal. Salah satu fitur utama dari Zepp OS API adalah akses ke data sensor stres, yang memungkinkan *developer* untuk membuat aplikasi untuk mengirimkan data kesehatan yang informatif dan akurat.

Zepp OS API menyediakan *interface* pemrograman yang sederhana untuk mengelola data pengguna. Penggunaan API ini memungkinkan *developer* untuk membangun aplikasi yang interaktif dan responsif, memaksimalkan data yang dibutuhkan oleh *developer*. Fitur Zepp OS API yang akan digunakan dalam penelitian

ini meliputi akses ke data sensor stres untuk dikirimkan ke *database* sebagai data yang akan dikelola oleh aplikasi yang akan dibangun pada penelitian ini.

Zepp OS juga memiliki beberapa versi yaitu Zepp OS 1.0, Zepp OS 2.0, Zepp OS 3.0, dan Zepp. Perbedaan yang dimiliki dari varian OS tersebut cukup signifikan seperti contohnya adalah hanya Zepp OS 3.0 keatas yang bisa menggunakan fitur *Running App Background*. Sedangkan pada penelitian ini sendiri, Zepp OS yang akan digunakan oleh *smartwatch Amazfit 5 Bip* adalah varian Zepp OS 2.1. Ikon dan logo Zepp OS dapat dilihat pada Gambar 2. 13 berikut.



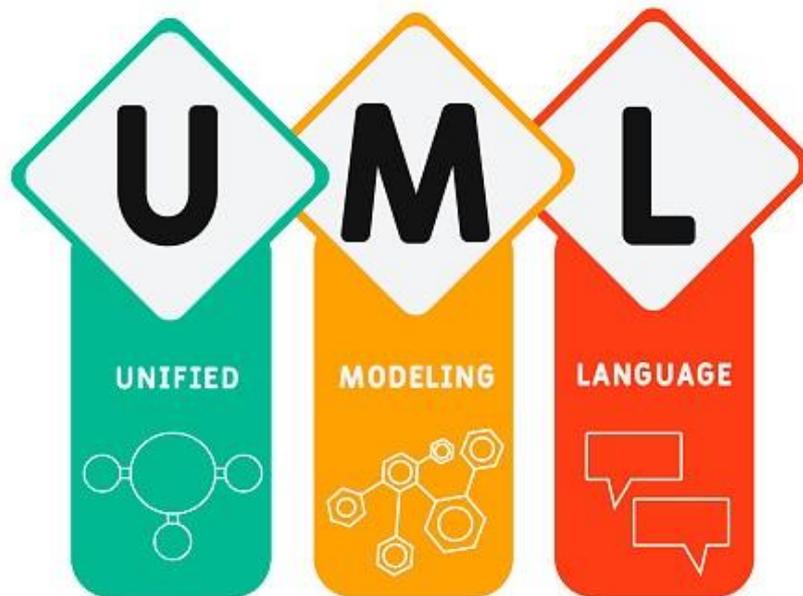
Gambar 2. 13 Zepp OS

2.14 Unified Modeling Language (UML)

Unified Modeling Language (UML) adalah sebuah bahasa pemodelan yang digunakan dalam rekayasa perangkat lunak untuk menggambarkan, merencanakan, dan mendokumentasikan desain sistem perangkat lunak [30]. UML memberikan cara yang standar dan terstruktur untuk menyusun dan mengkomunikasikan informasi mengenai sistem perangkat lunak, baik kepada pengembang perangkat lunak maupun pemangku kepentingan lainnya dengan orientasi objek [31]. Dengan menggunakan UML, dalam bidang rekayasa perangkat lunak dapat menggambarkan berbagai aspek sistem secara visual, termasuk struktur, fungsi, interaksi antar komponen, dan perilaku sistem, yang membantu memahami kompleksitas sistem secara lebih baik .

UML terdiri dari berbagai jenis diagram, seperti *Use Case Diagram*, *Activity Diagram*, *Sequence Diagram*, dan *Class Diagram*, yang masing-masing memiliki tujuan dan kegunaan spesifik dalam pemodelan sistem [30]. Melalui penggunaan UML, pengembang perangkat lunak dapat memvisualisasikan konsep desain, menganalisis arsitektur sistem, dan mengidentifikasi kebutuhan serta interaksi antara

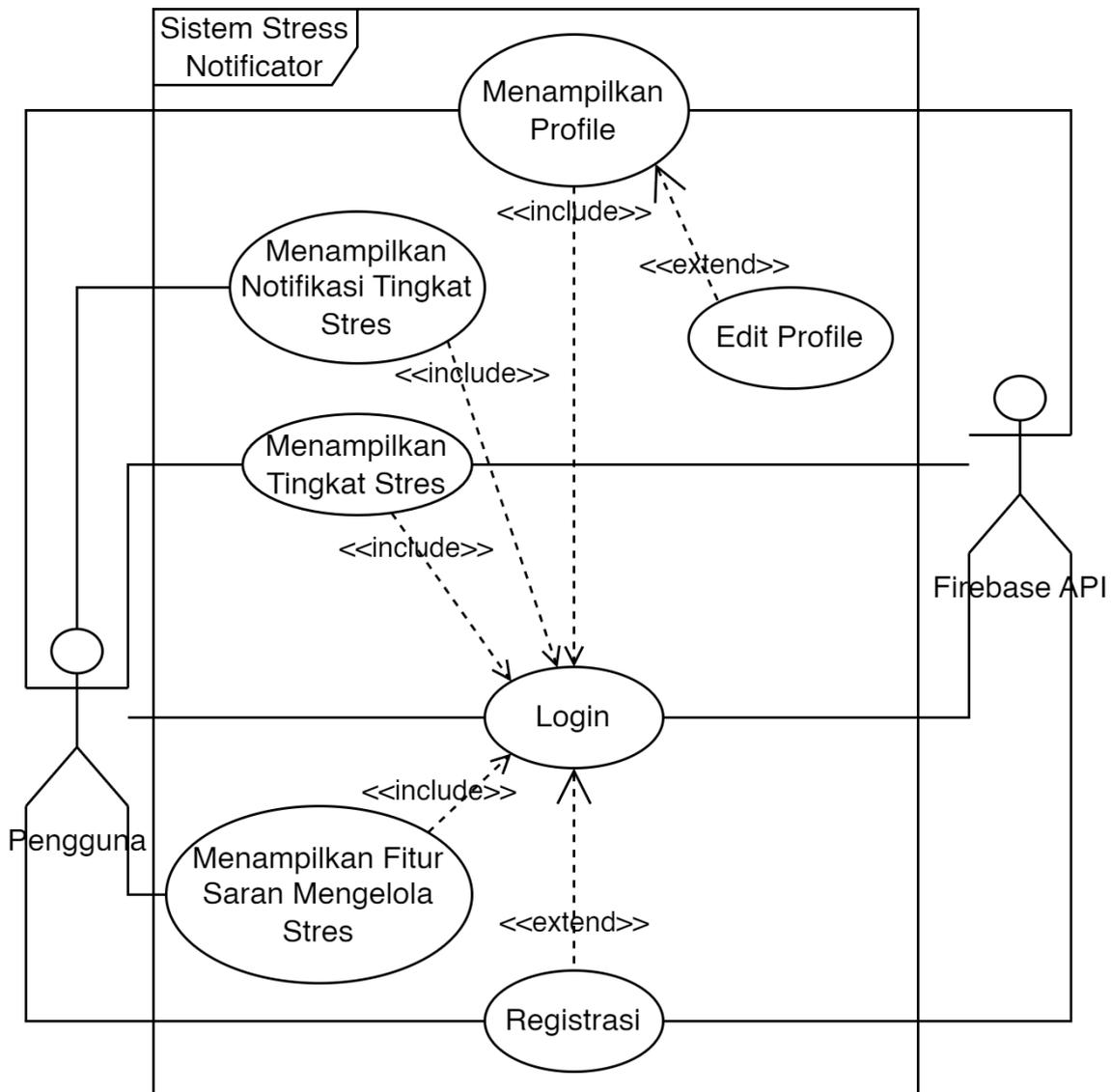
komponen sistem secara lebih efektif. Ikon dan Logo UML dapat dilihat pada Gambar 2. 14 berikut.



Gambar 2. 14 Unified Modeling Language (UML)

2.14.1 Use Case Diagram

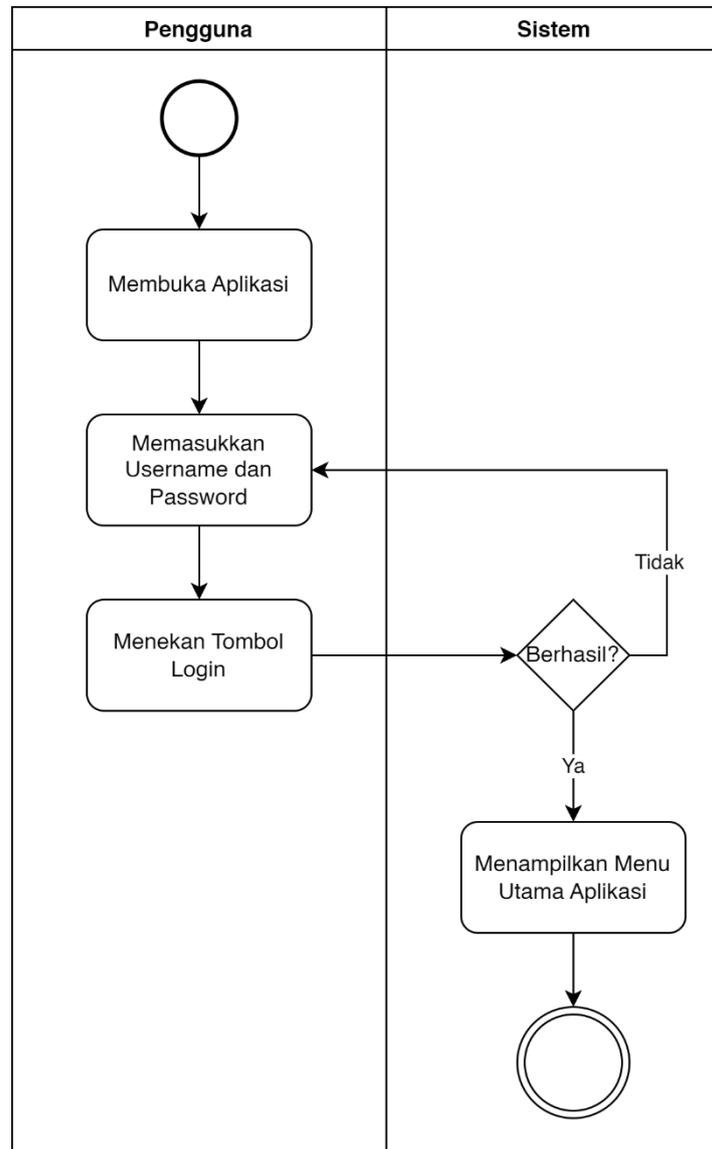
Use Case Diagram (Diagram Kasus Penggunaan) adalah salah satu jenis diagram dalam *Unified Modeling Language* (UML) yang digunakan untuk menggambarkan interaksi dengan sistem yang sedang dikembangkan [30]. Diagram ini memberikan pandangan tentang fungsionalitas sistem dari perspektif pengguna yang berinteraksi dengan sistem. Dalam use case diagram, actor atau pengguna digambarkan sebagai bentuk lingkaran atau persegi panjang di sekitar sistem, sementara use case diagram digambarkan sebagai elips atau oval pada sistem. Contoh dari *Use Case Diagram* dapat dilihat pada Gambar 2. 15 berikut.



Gambar 2. 15 Use Case Diagram

2.14.2 Activity Diagram

Activity Diagram (Aktivitas Diagram) adalah salah satu jenis diagram dalam *Unified Modeling Language* (UML) yang digunakan untuk menggambarkan aliran kerja atau aktivitas yang terjadi dalam suatu proses [30]. Diagram ini menggambarkan urutan langkah-langkah atau aktivitas yang dilakukan dalam suatu proses, serta hubungan antara aktivitas tersebut. Setiap aktivitas direpresentasikan oleh simbol elips, sedangkan aliran antara aktivitas direpresentasikan oleh panah. Contoh dari *Activity Diagram* dapat dilihat dari Gambar 2. 16 berikut.

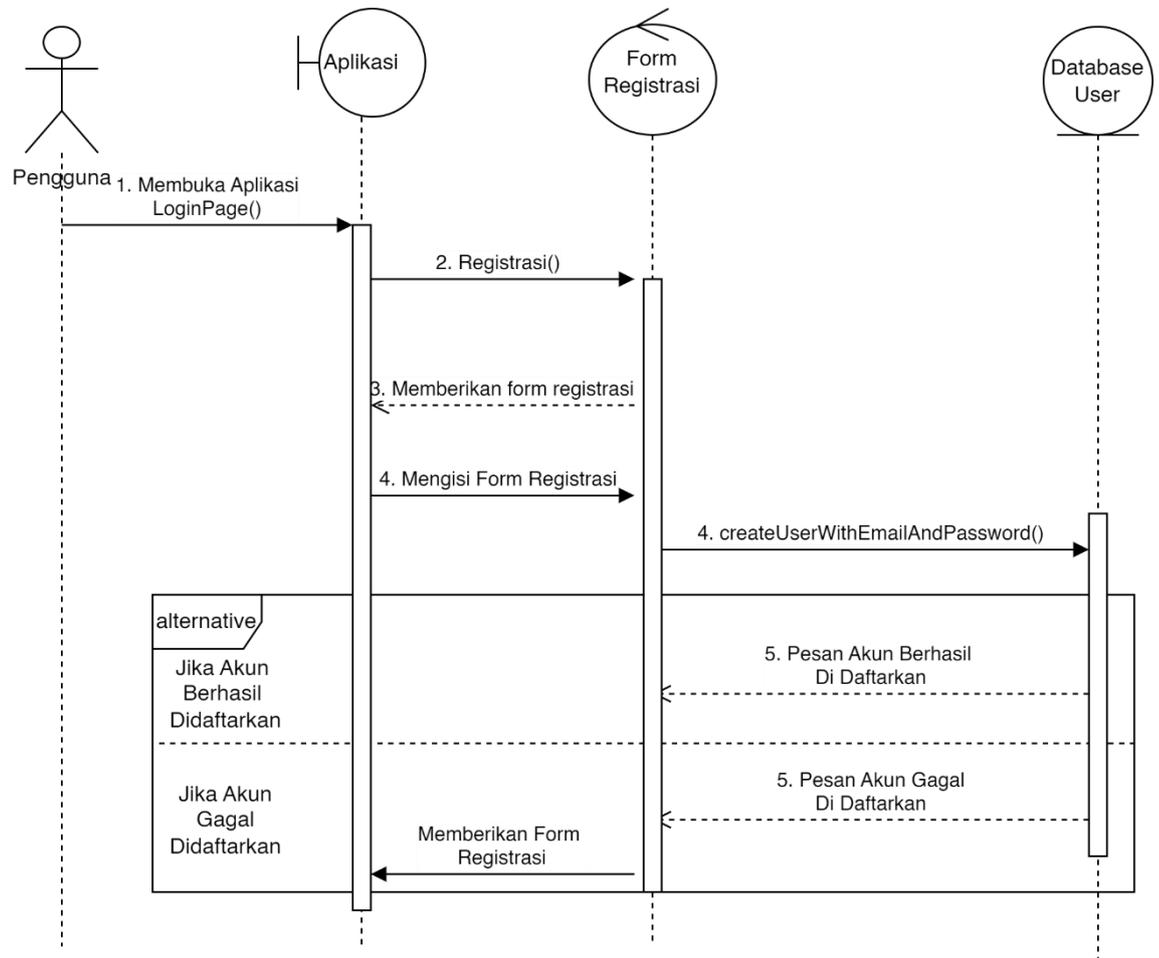


Gambar 2. 16 Activity Diagram

2.14.3 Sequence Diagram

Sequence Diagram (Diagram Urutan) adalah jenis diagram dalam *Unified Modeling Language* (UML) yang digunakan untuk menggambarkan interaksi antara objek dalam suatu sistem secara urutan waktu [30]. Diagram ini menggambarkan bagaimana objek-objek saling berinteraksi dalam skenario tertentu dengan menunjukkan urutan pesan yang dikirim antara objek-objek tersebut. Objek direpresentasikan oleh kotak dengan nama di bagian atas, dan urutan pesan antar objek direpresentasikan oleh panah yang bergerak dari objek pengirim ke objek

penerima. Pesan dapat berupa panggilan metode, pemanggilan prosedur, atau pertukaran pesan lainnya antara objek-objek. Contoh dari *Sequence Diagram* dapat dilihat pada Gambar 2. 17 berikut.



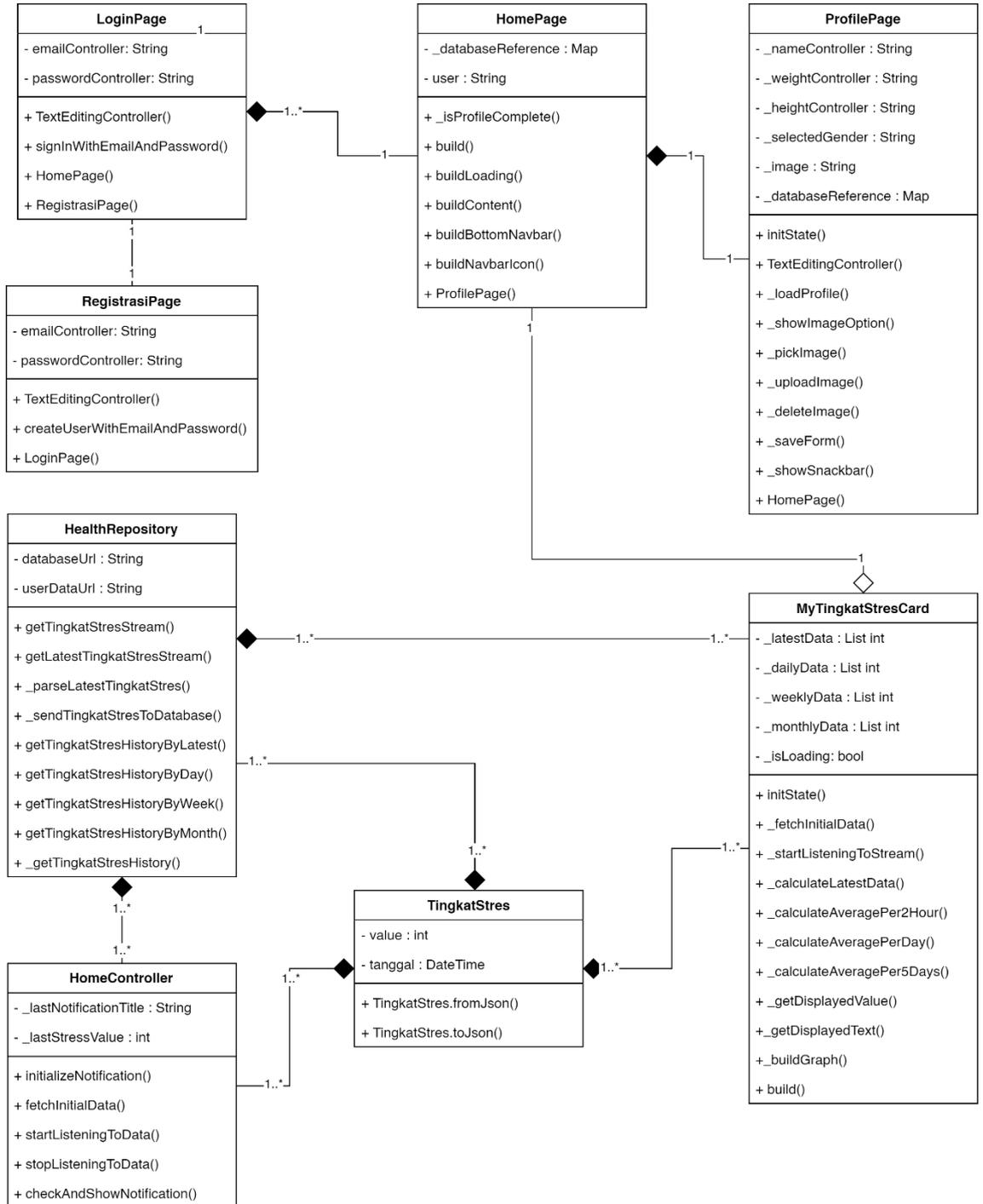
Gambar 2. 17 Sequence Diagram

2.14.4 Class Diagram

Diagram Class (Diagram Kelas) adalah jenis diagram dalam *Unified Modeling Language* (UML) yang digunakan untuk menggambarkan struktur kelas dari sistem perangkat lunak. Diagram ini menggambarkan kelas-kelas yang ada dalam sistem, serta atribut-atribut dan metode-metode yang dimiliki oleh setiap kelas [30]. Kelas direpresentasikan oleh kotak dengan tiga bagian, yaitu bagian atas berisi nama kelas, bagian tengah berisi atribut-atribut kelas, dan bagian bawah berisi metode-metode kelas. Hubungan antara kelas-kelas, seperti pewarisan (*inheritance*), asosiasi, dan agregasi, juga dapat ditunjukkan dalam diagram kelas menggunakan garis-garis yang

menghubungkan kelas-kelas. Contoh dari *Class Diagram* dapat dilihat pada Gambar

2. 18 berikut.



Gambar 2. 18 Diagram Class [30]