**APLIKASI MONITORING KESEHATAN DENGAN MEMANFAATKAN SMARTWATCH BERBASIS ANDROID**

**SKRIPSI**

Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana (S1)

**ZULFIAN FACHRU REZA**

**10118170**



**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS TEKNIK DAN ILMU KOMPUTER**

**UNIVERSITAS KOMPUTER INDONESIA**

**2022**

# DAFTAR ISI

[DAFTAR ISI ii](#_Toc107043759)

[DAFTAR TABEL iv](#_Toc107043760)

[DAFTAR GAMBAR v](#_Toc107043761)

[BAB 1 PENDAHULUAN 6](#_Toc107043762)

[1.1 Latar Belakang Masalah 6](#_Toc107043763)

[1.2 Identifikasi Masalah 7](#_Toc107043764)

[1.3 Batasan Masalah 8](#_Toc107043765)

[1.4 Maksud dan Tujuan 8](#_Toc107043766)

[1.4.1 Maksud 8](#_Toc107043767)

[1.4.2 Tujuan 8](#_Toc107043768)

[1.5 Sistematika Penulisan 9](#_Toc107043769)

[BAB 2 TINJAUAN PUSTAKA DAN LANDASAN TEORI 11](#_Toc107043770)

[2.1 Tinjauan Pustaka 11](#_Toc107043771)

[2.2 Landasan Teori 12](#_Toc107043772)

[2.2.1 Node.js 12](#_Toc107043773)

[2.2.2 Express.js 12](#_Toc107043774)

[2.2.3 JSON Web Token (JWT) 13](#_Toc107043775)

[2.2.4 PostgreSQL 14](#_Toc107043776)

[2.2.5 Flutter 15](#_Toc107043777)

[2.2.5.1. Kenapa Harus Flutter? 16](#_Toc107043778)

[2.2.6 Heroku 17](#_Toc107043779)

[2.2.7 Firebase 18](#_Toc107043780)

[2.2.8 Denyut Jantung 18](#_Toc107043781)

[2.2.8.1 Takikardia 19](#_Toc107043782)

[2.2.8.2 Bradikardia 20](#_Toc107043783)

[2.2.9 Saturasi Oksigen 20](#_Toc107043784)

[2.2.9.1 Hipoksemia 22](#_Toc107043785)

[2.2.9.2 Happy Hypoxia 22](#_Toc107043786)

[BAB 3 ANALISIS DAN PERANCANGAN SISTEM 23](#_Toc107043787)

[3.1 Analisis Masalah 23](#_Toc107043788)

[3.1.1. Analisis Sistem Sedang Berjalan 23](#_Toc107043789)

[3.1.2. Deskripsi Kebutuhan Non Fungsional 23](#_Toc107043790)

[3.1.2.1. Deskripsi Kebutuhan Perangkat Keras 24](#_Toc107043791)

[3.1.2.2. Analisis Kebutuhan Perangkat Lunak 24](#_Toc107043792)

[3.1.3. Analisis Kebutuhan Fungsional 24](#_Toc107043793)

[3.1.3.1. Use Case Diagram 26](#_Toc107043794)

[3.1.3.2. Use Case Description 27](#_Toc107043795)

[3.1.3.3. Activity Diagram 33](#_Toc107043796)

[3.1.3.4. Class Diagram 39](#_Toc107043797)

[3.1.3.5. Sequence Diagram 40](#_Toc107043798)

[3.2 Perancangan 44](#_Toc107043799)

[3.2.1 Perancangan Struktur Tabel 44](#_Toc107043800)

[3.2.2 Perancangan Struktur Menu 45](#_Toc107043801)

[3.2.3 Perancangan Antar Muka 45](#_Toc107043802)

[DAFTAR PUSTAKA 57](#_Toc107043803)

# DAFTAR TABEL

[Tabel 2.1 Denyut jantung Berdasarkan Usia 18](#_Toc107006706)

[Tabel 2.2 Nilai Saturasi Oksigen 20](#_Toc107006707)

[Tabel 3. 1 Kebutuhan Fungsional 24](#_Toc107006697)

[Tabel 3. 2 Use Case Description Register 26](#_Toc107006698)

[Tabel 3. 3 Use Case Description Login 27](#_Toc107006699)

[Tabel 3. 4 User Case Description Deteksi Denyut Jantung 28](#_Toc107006700)

[Tabel 3. 5 User Case Description Deteksi Saturasi Oksigen 29](#_Toc107006701)

[Tabel 3. 6 User Case Description User Setting 31](#_Toc107006702)

[Tabel 3. 7 User Case Description Logout 31](#_Toc107006703)

[Tabel 3. 8 Collection User 42](#_Toc107006704)

[Tabel 3. 9 Tabel Data Kesehatan 43](#_Toc107006705)

# DAFTAR GAMBAR

[Gambar 3.1 Use Case Diagram 26](#_Toc107006724)

[Gambar 3. 2 Activity Diagram Register 33](#_Toc107006725)

[Gambar 3. 3 Activity Diagram Login 34](#_Toc107006726)

[Gambar 3. 4 Activity Diagram Deteksi Denyut jantung 35](#_Toc107006727)

[Gambar 3. 5 Activity Diagram Deteksi Saturasi Oksigen 36](#_Toc107006728)

[Gambar 3. 6 Activity Diagram User Setting 37](#_Toc107006729)

[Gambar 3. 7 Activity Diagram Logout 38](#_Toc107006730)

[Gambar 3.8 Class Diagram 38](#_Toc107006731)

[Gambar 3. 9 Sequence Diagram Register 39](#_Toc107006732)

[Gambar 3. 10 Sequence Diagram Login 39](#_Toc107006733)

[Gambar 3. 11 Sequence Diagram Deteksi Denyut Jantung 40](#_Toc107006734)

[Gambar 3. 12 Sequence Diagram Deteksi Saturasi Oksigen 40](#_Toc107006735)

[Gambar 3. 13 Sequence Diagram User Setting 41](#_Toc107006736)

[Gambar 3. 14 Sequence Diagram Logout 42](#_Toc107006737)

[Gambar 3.15 Struktur Menu 43](#_Toc107006738)

[Gambar 3.16 Rancangan Antarmuka Halaman Daftar 44](#_Toc107006739)

[Gambar 3.17 Rancangan Antarmuka Halaman Home 45](#_Toc107006740)

[Gambar 3.18 Rancangan Antarmuka Halaman Tanda Vital 46](#_Toc107006741)

[Gambar 3.19 Rancangan Antarmuka Halaman Denyut Jantung 47](#_Toc107006742)

[Gambar 3.20 Rancangan Antarmuka Halaman Data Denyut Jantung 48](#_Toc107006743)

[Gambar 3.21 Rancangan Antarmuka Halaman Tentang Denyut Jantung 49](#_Toc107006744)

[Gambar 3.22 Rancangan Antarmuka Halaman Saturasi Oksigen 50](#_Toc107006745)

[Gambar 3.23 Rancangan Antarmuka Halaman Data Saturasi Oksigen 51](#_Toc107006746)

[Gambar 3.24 Rancangan Antarmuka Halaman Tentang Saturasi Oksigen 52](#_Toc107006747)

[Gambar 3.25 Rancangan Antarmuka Halaman Tips dan Penanganan Pertama 53](#_Toc107006748)

[Gambar 3.26 Rancangan Antarmuka Halaman User Profile 54](#_Toc107006749)

# BAB 1 PENDAHULUAN

## Latar Belakang Masalah

Kesehatan merupakan salah satu hal yang sangat berharga di dalam hidup kita. Dengan tubuh yang sehat, kita sanggup melakukan bermacam-macam pekerjaan dan kesibukan tanpa adanya kendala dan dengan mempunyai raga yang bugar, tentu akan menghasilkan jiwa yang damai dan perasaan yang seimbang. Karena itulah menjaga kesehatan tubuh adalah salah satu hal penting yang harus selalu kita upayakan. Apalagi disaat tingginya kasus harian Covid-19 belakangan ini, alangkah baiknya kita harus meningkatkan kewaspadaan kita dengan menerapkan protokol kesehatan yang telah dianjurkan pemerintah. Salah satu upaya dalam menjaga kesehatan adalah dengan memantau tanda-tanda kesehatan vital, seperti detak jantung dan tingkat saturasi oksigen dalam darah [1].

Jantung merupakan organ yang vital, penting dan esensial. Fungsinya memengaruhi hampir seluruh organ tubuh yang lain. Jika kita mempunyai permasalahan pada jantung maka kelangsungan hidup kita juga akan terancam. Oleh karena itu, penting untuk menjaga kesehatan jantung untuk hidup yang lebih berkualitas [2]. [Detak](https://www.halodoc.com/artikel/detak-jantung-tidak-teratur-apa-penyebabnya#:~:text=Normalnya%20detak%20jantung%20sekitar%2050,artinya%20ia%20sedang%20mengalami%20aritmia.) jantung atau denyut nadi adalah berapa kali jantung kamu berdetak dalam 1 menit. Aktivitas yang kamu lakukan bisa mengubah seberapa cepat atau lambat detak jantung berfluktuasi, mulai dari detak lambat dan stabil saat beristirahat atau tidur, hingga detak jantung yang cepat selama kamu berolahraga. Detak jantung normal orang dewasa dengan anak-anak berbeda. Jika detak jantung normal orang dewasa berkisar antara 60-100 detak per menit, anak-anak biasanya memiliki detak jantung yang lebih tinggi. Dengan mengetahui detak jantung normal pada anak-anak dan orang dewasa, kamu bisa mengetahui kondisi kesehatan jantung kamu dan anak lebih baik lagi, sehingga bisa segera mencari penanganan bila menemukan adanya kejanggalan [3].

Selain detak jantung, ada saturasi oksigen yang tak kalah penting yang harus tetap harus kita jaga. Saturasi oksigen merupakan nilai yang menunjukkan [kadar oksigen di dalam darah](https://www.alodokter.com/penting-diketahui-ini-kadar-oksigen-normal-dalam-darah" \t "_blank). Nilai ini sangat berpengaruh terhadap berbagai fungsi organ dan jaringan tubuh. Pengukuran nilai saturasi oksigen dapat dilakukan dengan beberapa cara, yakni dengan [analisis gas darah](https://www.alodokter.com/analisa-gas-darah-dan-hal-hal-penting-yang-ada-di-dalamnya" \t "_blank) (AGD), menggunakan [alat oximeter](https://www.alodokter.com/pentingnya-oximeter-bagi-pasien-isolasi-mandiri-covid-19" \t "_blank) atau yang sekarang lagi hype yaitu menggunakan *smartwatch*. Meskipun *smartwatch* tak bisa dianggap sebagai alat medis, namun hasil yang didapatkan dinilai cukup akurat mendekati standar alat medis. Nilai saturasi oksigen normal pada orang dengan kondisi paru-paru yang sehat atau tidak memiliki kondisi medis tertentu memiliki nilai SpO2 95-100%. Sementara itu, pada orang yang memang memiliki penyakit paru-paru seperti Penyakit Paru Obstruktif Kronik (PPOK), memiliki nilai saturasi oksigen yang berbeda, berada pada 88-92% [4].

Berdasarkan uraian permasalah di atas, dibutuhkan solusi yang dapat membantu orang-orang dapat memantau tanta-tanda kesehatan vital, seperti detak jantung, dan saturasi oksigen. Maka dari itu akan dibangun sebuah aplikasi yang dapat memonitoring kesehatan berbasis android dengan memanfaatkan *smartwatch*. Pada aplikasi tersebut, sistem akan mencatat data-data seperti kondisi detak jatung, nilai saturasi oksigen, dari pengguna setiap kali pengguna tersebut mengeceknya menggunakan *smartwatch*. Dari data-data yang sudah didapatkan, nantinya pengguna akan mendapatkan informasi terkait kondisi kesehatannya. Seperti adanya kelainan pada detak jantung atau tidak, dan menampilkan nilai saturasi oksigen apakah berada pada kondisi normal atau tidak. Nantinya, aplikasi akan memberikan informasi terkait penanganan pertama sesuai dengan kondisi kesehatan yang didapatkan.

## Identifikasi Masalah

Berdasarkan latar belakang di atas, maka didapatkan masalah yang dapat diidentifikasi, yaitu adalah sebagai berikut..

1. Pentingnya memantau tanda-tanda vital kesehatan seperti detak jantung dan saturasi oksigen dalam darah terlebih di saat pandemi seperti ini sebagai upaya untuk meningkatkan kewaspadaan terhadap Covid-19.
2. Masih minimnya aplikasi yang dapat memantau kesehatan dengan menggunakan *smartwatch* yang langsung memberikan informasi terkait penanganan pertama ataupun saran atas masalah dari kondisi kesehatan tersebut.

## Batasan Masalah

Adapun batasan masalah dalam penelitian ini, yaitu sebagai berikut:

1. Sistem yang dibangun hanya akan mendukung sistem operasi Android dengan versi *sdk* minimal 5.0 *lollipop* atau API Level 21
2. Sistem yang dibangun saat ini hanya dapat berjalan pada *smartwatch* Samsung saja, yaitu Samsung Galaxy Watch 4.
3. Sistem yang dibangun hanya akan berjalan jika terdapat aplikasi Google Fit pada *smartphone* yang digunakan.
4. Database yang digunakan yaitu Firebase Cloud Firestore.
5. Untuk autentikasinya yaitu menggunakan Firebase Authentication.
6. Untuk layanan push notification-nya menggunakan Firebase Cloud Messaging (FCM)
7. Aplikasi harus terkoneksi dengan internet.

## Maksud dan Tujuan

Adapun maksud dan tujuan dari penelitian ini adalah sebagai berikut :

### Maksud

Maksud dari penelitian ini adalah untuk membangun aplikasi berbasis android yang dapat memonitoring kesehatan dengan menggunakan *smartwatch,* dimana nantinya aplikasi akan memberikan informasi terkait kondisi kesehatan pengguna dan juga memberikan informasi bagaimana penangan pertama jika terdeteksi kondisi kesehatan berada pada kondisi yang tidak normal.

### Tujuan

Adapun tujuan yang akan dicapai dalam penelitian ini diantaranya sebagai berikut :

1. Membuat aplikasi yang dapat memonitoring kesehatan dengan menggunakan smartwatch
2. Membantu orang-orang dalam upaya menjaga kesehatan dengan memantau tanda-tanda kesehatan vital seperti detak jantung dan saturasi oksigen.
3. Memberikan informasi terkait penanganan pertama jika terdapat kondisi kesehatan diluar normal.

## Sistematika Penulisan

Pada bab ini berisi uraian latar belakang masalah, identifikasi masalah, maksud dan tujuan, batasan masalah, metodologi penelitian, tahap pengumpulan data, model pengembangan perangkat lunak dan sistematika penulisan.

**BAB I PENDAHULUAN**

Bab ini menerangkan tentang latar belakang, maksud dan tujuan, manfaat, batasan masalah dan metode penelitian.

**BAB II TINJAUAN PUSTAKA DAN LANDASAN TEORI**

Pada bab ini membahas mengenai tinjauan umum mengenai Yayasan Al Muttaqin di Medono Pekalongan dan pembahasan berbagai konsep dasar mengenai Sistem Informasi Manajemen dan teori-teori pendukung lainnya yang berkaitan dengan topik pembangunan perangkat lunak.

**BAB III ANALISIS DAN PERANCANGAN SISTEM**

Bab ini berisi analisis kebutuhan dalam membangun aplikasi ini, analisis sistem yang sedang berjalan pada aplikasi ini sesuai dengan metode pembangunan perangkat lunak yang digunakan, selain itu juga terdapat perancangan antarmuka untuk aplikasi yang dibangun sesuai dengan hasil analisis yang telah dibuat.

**BAB IV IMPLEMENTASI DAN PENGUJIAN SISTEM**

Bab ini membahas implementasi dalam bahasa pemrograman yaitu implementasi kebutuhan perangkat keras dan perangkat lunak, implementasi basis data, implementasi antarmuka dan tahap - tahap dalam melakukan pengujian perangkat lunak.

**BAB V KESIMPULAN DAN SARAN**

Bab ini membahas tentang kesimpulan yang sudah diperoleh dari hasil penulisan tugas akhir dan saran mengenai pengembangan aplikasi untuk masa yang akan datang.

# BAB 2 TINJAUAN PUSTAKA DAN LANDASAN TEORI

## Tinjauan Pustaka

Penelitian yang berhubungan dengan topik yang penulis bahas adalah “Pengembangan Aplikasi Sistem Pendeteksi Denyut Jantung Menggunakan Mikrokontroler Arduino Berbasis Android” yang dilakukan oleh Yose (2019). Pada penelitian ini dikembangkan suatu sistem yang berfungsi untuk mendeteksi dan memberikan laporan terkait kondisi kesehatan jantung seorang pasien atau user untuk membantu mengambil keputusan dalam menjalankan aktivitasnya sehari-hari. Aplikasi ini dibuat menggunakan Arduino UNO dan Pulse sensor sebagai alat yang digunakan untuk mendeteksi jantung dan juga device android sebagai hasil output dari sistem. Pada penelitian ini memiliki perbedaan dengan sistem yang penulis buat, yaitu alat yang digunakan untuk mendeteksi jantung menggunakan Arduino UNO dengan pulse sensor. Sedangkan dalam sistem yang penulis buat yaitu menggunakan sensor yang ada pada smartwatch.

Selanjutnya penelitian yang berhubungan dengan topik yang penulis bawa selanjutnya adalah “Alat Pengukur Saturasi Oksigen Dalam Darah Menggunakan Metode PPG Reflectance Pada Sensor MX30100” yang dilakukan oleh Chandra (2019). Penelitian ini bertujuan untuk merancang dan membangun sebuah alat yang dapat mengukur saturasi oksigen dalam darah digital menggunakan metode PPG reflectance yang terdapat pada sensor MX30100 dan menggunakan arduino Nano sebagai mikroprosesor. Alat ini juga menggunakan device android sebagai tampilan keluarannya.

Meninjau dari penelitian yang sudah dilakukan, penulis akan membuat sebuah “Aplikasi Monitoring Kesehatan dengan Memanfaatkan Smartwatch Berbasis Android” yang bertujuan untuk memonitoring atau memantau tanda-tanda kesehatan vital seperti detak jantung, dan juga tingkat saturasi oksigen dalam darah dengan memanfaatkan sensor-sensor yang ada pada smartwatch.

## Landasan Teori

### Node.js

Node.js adalah runtime environment untuk JavaScript yang bersifat open-source dan cross-platform. Dengan Node.js kita dapat menjalankan kode JavaScript di mana pun, tidak hanya terbatas pada lingkungan browser.

Node.js menjalankan V8 JavaScript engine (yang juga merupakan inti dari Google Chrome) di luar browser. Ini memungkinkan Node.js memiliki performa yang tinggi.

Node.js juga menyediakan banyak library/module JavaScript yang membantu menyederhanakan pengembangan aplikasi web. Berikut ini adalah beberapa fitur penting dari Node.js yang menjadikannya pilihan utama dalam pengembangan aplikasi:

* Asynchronous & Event-driven

Semua API dari Node.js bersifat asynchronous, artinya tidak memblokir proses lain sembari menunggu satu proses selesai. Server Node.js akan melanjutkan ke ke pemanggilan API berikutnya lalu memanfaatkan mekanisme event notification untuk mendapatkan respon dari panggilan API sebelumnya.

* Very Fast

Eksekusi kode dengan Node.js sangat cepat karena berjalan pada V8 JavaScript Engine dari Google Chrome.

* Single Thread but Highly Scalable.

Node.js menggunakan model single thread dengan event looping. Mekanisme ini membantu server untuk merespon secara asynchronous dan menjadikan server lebih scalable dibandingkan server tradisional yang menggunakan banyak thread untuk menangani permintaan.

### Express.js

ExpressJS adalah salah satu framework javascript untuk pengembangan backend terbaik yang di desain secara minimalis dan fleksibel yang menyediakan serangkaian fitur menarik untuk pengembangan aplikasi web maupun mobile (react). Anda dapat melihat website resminya di [expressjs.com](https://expressjs.com/) yang didalamnya terdapat beberapa penjelasan singkat tentang expressjs.

ExpressJS memudahkan anda dalam pembuatan aplikasi web dimana dengan expressjs, anda tidak perlu menggunakan module http bawaan dari NodeJS. Framework ini menawarkan beberapa fitur seperti routing, rendering view, dan mendukung middleware dengan kata lain anda akan banyak menghemat waktu dalam pengembangan aplikasi Node.js.

Pada dasarnya Express JS berupa library javascript dimana kita dapat membangun sistem backend dengan mudah mulai dari pengaturan NPM, instal express package, membuat model, routes, dan lain-lain. Banyak perusahaan besar yang menggunakan framework expressjs seperti IBM, [Paypal](https://www.paypal.com/), [GoDaddy](https://www.godaddy.com/), Walmart, Storylens, Flickr dan masih banyak lagi.

Selain itu, anda juga bisa membuat web server HTML, server file statik, aplikasi chat, search engine, social media, layanan web dengan akses melalui REST API atau aplikasi hybrid yaitu selain pengguna mempunyai akses melalui REST API juga mempunyai akses ke HTML page. Hal ini sesuai dengan penggunaan utamanya yaitu membuat Restful API yang meneripa permintaan dari frontend dan mengirim respons yang sesuai.

### JSON Web Token (JWT)

JSON Web Token, yang berarti token ini menggunakan JSON (Javascript Object Notation) berbentuk string panjang yang sangat random, lalu token ini memungkinkan kita untuk mengirimkan data yang dapat diverifikasi oleh dua pihak atau lebih. Sama halnya seperti password jadi ketika users berhasil melakukan Login maka server akan memberikan sebuah Token. Nanti Token tersebut akan disimpan oleh users pada Local Storage atau Cookies Browser dan bila users ingin mengakses halaman halaman tertentu maka harus menyertakan token tersebut. Untuk itu users akan mengirim balik token yang dikasih diawal tadi sebagai bukti bila user ini, sudah melakukan login. Sekarang kita akan lihat struktur dasarnya Tokennya dimana terdiri dari tiga bagian yaitu yang pertama header lalu kedua bagian payloadnya atau datanya dan yang ketiga adalah bagian verify signature.

1. Header

Header biasanya terdiri dari dua bagian: jenis token, yaitu JWT, dan algoritma penandatanganan yang digunakan, seperti HMAC SHA256 atau RSA.

1. Payload

sebagai infomasi atau data yang ingin kita kirimkan. Dalam penerapannya di otentikasi atau pun otorisasi, biasanya data ini berupa data yang sifatnya unik bagi user, seperti: email, id/uuid, dan juga data yang berkaitan dengan otorisasi seperti role, karena data tersebut akan digunakan sebagai tanda pengenal si pengirim token.

1. Verify Signature

Verify Signature adalah hasil dari Hash atau gabungan dari isi encode Header dan Payloadnya lalu ditambahkan kode secretnya. Signature ini berguna untuk memverifikasi bahwa header maupun payload yang ada dalam token tidak berubah dari nilai aslinya (karena untuk membuat payload dan header palsu itu cukup mudah).

Signature-nya sendiri tidak mungkin dapat diakali, karena sudah dalam berbentuk hash; yang mana adalah fungsi satu arah (tidak dapat dikembalikan ke nilai semula), dan meski kita tahu algoritma hashing-nya, kita juga memerlukan secret key yang mana hanya si pembuat aplikasi yang tahu.

### PostgreSQL

PostgreSQL (atau dikenal juga sebagai Postgres) adalah sebuah RDBMS *open-source* (didistribusikan secara *free*) yang menekankan pada pemenuhan standar teknis dan fleksibilitas (keluwesan) data. PostgreSQL didistribusikan dengan lisensi bebas/gratis, sehingga dapat digunakan, dimodifikasi, dan didistribusikan kembali kepada publik secara bebas/gratis untuk tujuan pribadi, komersial, ataupun akademik.

PostgreSQL dirancang untuk menangani beban kerja terhadap data dari sebuah mesin menuju layanan web yang diakses banyak orang secara bersamaan. PostgreSQL merupakan sebuah sistem manajemen database yang didesain agar bekerja dengan baik pada sistem MacOS Server, namun *development*-nya juga diberlakukan untuk [platform Linux](https://www.nesabamedia.com/pengertian-linux-beserta-sejarah-dan-perkembangan-linux/), FreeBSD, OpenBSD, dan Windows 32-bit dan 64 bit sampai pada versi PostgreSQL 10. Selanjutnya, mulai dari versi PostgreSQL 11 dikembangkan untuk platform MacOS 64-bit dan Windows 64-bit saja.

Potsgres pada awal pengembangannya dilakukan oleh seorang mahasiswa dan staff programmer dari University of California, Berkeley, di bawah bimbingan Professor Michael Stonebraker pada tahun 1982.

RDBMS ini tadinya diberi nama Postgres, namun seiring perkembangannya dengan fungsi-fungsi query SQL yang semakin hari terus ditambahkan, maka pada tahun 1995 namanya dirombak menjadi Postgres95. Terakhir pada tahun 1996 dirombak lagi menjadi PostgreSQL, yang terus digunakan sampai saat ini.

### Flutter

Flutter adalah sebuah bahasa pemrograman buatan Google pada tahun 2011 yang populer pada akhir-akhir ini. Flutter adalah sebuah platform yang banyak dipakai oleh mobile developer untuk pembuatan aplikasi multiplatform menggunakan satu basis coding. Yang berarti flutter adalah sebuah teknologi open source berasal dari Google yang digunakan untuk pembuatan aplikasi pada sistem operasi Android dan iOS.

Flutter adalah sebuah Software Development Kit (SDK) yang membantu para mobile developer, sehingga sudah pasti dilengkapi dengan berbagai macam fitur berguna buat mengembangkan aplikasi pada lintas platform.

Contoh pada rendering engine, widget flutter adalah yang siap pakai, pengujian dan integrasi pada API, serta command-line tools. Meskipun ada desain serupa berupa teknologi seperti react native tetapi para mobile developer tetap memakai flutter karena kemudahan yang ditawarkan. Inilah salah satu faktor kenapa mobile developer suka flutter, yaitu kemudahan membuat aplikasi dengan menggunakan satu basis kode saja.

Hal tersebut sesuai dengan klaim dari pihak Google sebagai pembuat aplikasi. Jika kamu sebagai mobile developer dan ingin mengembangkan aplikasi berbasis flutter, maka kamu harus menguasai bahasa pemrograman dart terlebih dahulu.

Hubungan dart dan flutter adalah dart merupakan bahasa pemrograman pada flutter yang berfokus untuk mengembangkan front end sehingga dapat dimanfaatkan dalam pembuatan aplikasi mobile dan website.

#### Kenapa Harus Flutter?

1. Cross Platform

Flutter mendukung cross platform alias dapat dijalankan di beberapa platform yang berbeda. Dengan menggunakan Flutter, kita dapat membuat aplikasi Android dan iOS sekaligus. Selain mobile, kita juga dapat membuat aplikasi [web](https://www.google.com/url?q=https://flutter.dev/web&sa=D&ust=1587033113225000)dan [desktop](https://www.google.com/url?q=https://flutter.dev/desktop&sa=D&ust=1587033113225000). Tentunya hal ini akan menghemat waktu. Kita tidak perlu mempelajari bahasa native yang digunakan di masing-masing platform.

1. Fast Development (Hot Reload)

Flutter memiliki sebuah fitur bernama [hot reload](https://www.google.com/url?q=https://flutter.dev/docs/development/tools/hot-reload&sa=D&ust=1587033113226000). Dengan fitur ini, proses pengembangan aplikasi dapat berjalan lebih cepat dan mudah.. Setelah melakukan perubahan pada kode program, cukup tekan hot reload. Aplikasi akan diperbarui dalam kurun waktu kurang dari 1 detik. Sangat cepat bukan?

Hot reload bekerja dengan cara menginjeksi kode program yang mengalami perubahan ke dalam Dart Virtual Machine. Setelah virtual machine memperbarui tiap kelas dengan kode program versi terbaru, maka framework Flutter secara otomatis membangun kembali susunan komponen widget sehingga kita dapat dengan cepat melihat perubahan yang terjadi.

1. Beautiful UI

Flutter dirancang untuk mempermudah developer dalam membangun tampilan user interface. Keseluruhan UI pada Flutter dibangun menggunakan [widget](https://www.google.com/url?q=https://flutter.dev/docs/development/ui/widgets-intro&sa=D&ust=1587033113228000). Sebagai contoh, jika kita menambahkan sebuah text field, text field tersebut adalah widget. Button dan Image juga merupakan widget. Bahkan untuk mengatur posisi komponen menjadi rata tengah, kita menggunakan center widget. Kita dapat melakukan kustomisasi pada tiap widget. Widget akan menggambarkan seperti apa tampilan yang akan dibuat berdasarkan konfigurasi dan state yang ada. Terdapat 2 set widget, [Material Design](https://www.google.com/url?q=https://flutter.dev/docs/development/ui/widgets/material&sa=D&ust=1587033113229000) (Android) dan [Cupertino](https://www.google.com/url?q=https://flutter.dev/docs/development/ui/widgets/cupertino&sa=D&ust=1587033113229000)(iOS).Teman-teman dapat melihat daftar lengkap widget yang tersedia di tautan [ini](https://www.google.com/url?q=https://flutter.dev/docs/development/ui/widgets&sa=D&ust=1587033113229000). Selain itu, di tiap pekan, tim Flutter juga membahas Widget of the Week yang dapat teman-teman saksikan lewat channel [Youtube Flutter](https://www.google.com/url?q=https://www.youtube.com/channel/UCwXdFgeE9KYzlDdR7TG9cMw&sa=D&ust=1587033113230000).

### Heroku

Heroku adalah sebuah cloud platform yang menjalankan bahasa pemrograman tertentu, Heroku mendukung bahasa pemrograman seperti Ruby, Node.js, Python, Java, PHP, dan lain-lain.

Heroku termasuk ke dalam kriteria Platform As A Service (PaaS), sehingga bagi anda yang ingin melakukan deploy aplikasi ke heroku cukup hanya dengan melakukan konfigurasi aplikasi yang ingin di deploy dan menyediakan platform yang memungkinkan pelanggan untuk mengembangkan, menjalankan, dan mengelola aplikasi tanpa kompleksitas membangun dan memelihara infrastruktur yang biasanya terkait dengan pengembangan dan peluncuran aplikasi.

Heroku pada awalnya dikembangkan oleh James Lindenbaum, Adam Wiggins, dan Orion Henry untuk proyek-proyek pendukung yang kompatibel dengan platform pemrograman Ruby yang dikenal sebagai Rack. Pengembangan prototipe memakan waktu sekitar enam bulan. Kemudian, Heroku menghadapi kelemahan karena kurangnya pelanggan pasar yang tepat karena banyak pengembang aplikasi menggunakan alat dan lingkungan mereka sendiri. Pada Januari 2009 platform baru diluncurkan yang dibangun hampir dari awal setelah upaya tiga bulan. Pada Oktober 2009, Byron Sebastian bergabung dengan Heroku sebagai CEO. Pada 8 Desember 2010, Salesforce.com mengakuisisi Heroku sebagai anak perusahaan yang sepenuhnya dimiliki oleh Salesforce.com. Pada 12 Juli 2011, Yukihiro "Matz" Matsumoto, kepala desainer bahasa pemrograman Ruby, bergabung dengan perusahaan sebagai Chief Architect, Ruby. Pada bulan yang sama, Heroku menambahkan dukungan untuk Node.js dan Clojure. Pada 15 September 2011, Heroku dan Facebook memperkenalkan Heroku untuk Facebook. Saat ini Heroku mendukung database Redis sebagai tambahan untuk PostgreSQL standarnya.

### Firebase

Firebase adalah suatu layanan dari Google untuk memberikan kemudahan bahkan mempermudah para developer aplikasi dalam mengembangkan aplikasinya. Firebase alias BaaS (*Backend as a Service*) merupakan solusi yang ditawarkan oleh Google untuk mempercepat pekerjaan developer.

Dengan menggunakan Firebase, apps developer bisa fokus dalam mengembangkan aplikasi tanpa memberikan *effort*yang besar untuk urusan *backend*.

Singkat cerita mengenai sejarah dari Firebase didirikan pertama kali pada tahun 2011 oleh Andrew Lee dan James Tamplin. Produk Firebase yang pertama kali adalah Realtime Database. Realtime Database digunakan developer untuk menyimpan data dan *synchronize* ke banyak *user*. Kemudian ia berkembang sebagai layanan pengembang aplikasi. Pada bulan Oktober 2014, perusahaan tersebut diakuisisi oleh Google.

Mengenai segi layanan, dulu Firebase memberikan *service trial*(percobaan), namun saat ini kamu bisa memanfaatkan dan menggunakan layanan Firebase secara *free* (gratis). Tentu saja dengan adanya batasan-batasan tertentu.

### Denyut Jantung

Jantung adalah organ vital yang bertugas untuk memompa darah ke seluruh tubuh. Hal ini membuat [peredaran darah](https://hellosehat.com/kelainan-darah/sistem-peredaran-darah-manusia/) yang kaya oksigen dapat mencapai seluruh sel-sel dalam tubuh Anda. Untuk mengecek kesehatan jantung, dokter biasanya akan mengamati [seberapa normal tekanan darah](https://hellosehat.com/jantung/tekanan-darah-normal/) dan detak jantung.

Detak jantung atau yang sering disebut juga sebagai denyut nadi adalah berapa kali jantung Anda berdetak per menit. Detak jantung atau denyut nadi yang normal berkisar antara 60 hingga 100 detak per menit (BPM). Berikut merupakan tabel denyut jantung normal yang dikategorikan berdasarkan usia:

|  |  |
| --- | --- |
| **Usia** | **Denyut Jantung (bpm)** |
| < 1 bulan | 70 – 190 |
| 1 – 11 bulan | 80 – 150 |
| 1 – 2 tahun | 80 – 130 |
| 3 – 4 tahun | 80 – 120 |
| 5 – 6 tahun | 75 – 115 |
| 7 – 9 tahun | 70 – 110 |
| > 10 tahun | 60 - 100 |

Tabel 2.1 Denyut jantung Berdasarkan Usia

Secara medis, gangguan irama jantung disebut dengan istilah [aritmia](https://www.alodokter.com/aritmia" \t "_blank). Kondisi ini ditandai dengan denyut jantung yang terlalu cepat, lambat, tidak teratur, atau bahkan terhenti sama sekali.

Aritmia dapat disebabkan oleh berbagai hal, seperti riwayat penyakit jantung, tekanan darah tinggi, penyakit katup jantung, gangguan tiroid, [gangguan elektrolit](https://www.alodokter.com/gangguan-elektrolit" \t "_blank), atau sedang dalam masa pemulihan setelah menjalani [operasi jantung](https://www.alodokter.com/memahami-operasi-jantung" \t "_blank).

Gaya hidup tidak sehat, seperti konsumsi minuman beralkohol secara berlebihan dan kebiasaan merokok, serta efek samping obat-obatan, juga dapat menyebabkan aritmia.

Penyakit aritmia secara garis besar dibagi menjadi dua, yaitu takikardia dan bradikardia.

#### Takikardia

[Takikardia](https://www.alodokter.com/takikardia) adalah kondisi ketika detak jantung berdetak lebih cepat saat istirahat. Belum diketahui secara pasti penyebab kondisi ini, tetapi ada beberapa faktor yang diduga dapat memicu takikardia.

Faktor tersebut meliputi faktor keturunan, riwayat penyakit tertentu seperti penyakit jantung dan anemia, efek samping obat-obatan, atau kebiasaan seperti merokok dan mengonsumsi minuman beralkohol.

Takikardia dapat menimbulkan keluhan berupa nyeri dada, pusing, kelelahan, dan sesak napas. Namun, ada kalanya takikardia tidak menimbulkan gejala atau keluhan apa pun.

#### Bradikardia

Detak jantung yang terlalu lambat disebut [bradikardia](https://www.alodokter.com/bradikardia" \t "_blank). Normalnya, jantung berdetak 60–100 kali per menit ketika istirahat. Namun, pada kondisi bradikardia, detak jantung kurang dari 60 kali dalam satu menit.

Kondisi ini dapat dipengaruhi oleh faktor pertambahan usia, kebiasaan merokok, efek samping obat-obatan, atau riwayat penyakit seperti [tekanan darah tinggi](https://www.alodokter.com/hipertensi" \t "_blank) atau kelainan tiroid.

Pada sebagian orang, mungkin detak jantung yang terlalu lambat tidak menimbulkan masalah. Namun, kondisi ini bisa menjadi tanda adanya masalah pada sistem listrik jantung.

Bradikardia dapat menimbulkan keluhan berupa sesak napas, sulit konsentrasi, [pingsan](https://www.alodokter.com/pingsan" \t "_blank), pusing, dan [mudah lelah](https://www.alodokter.com/agar-tubuh-tidak-mudah-lelah" \t "_blank) walau hanya melakukan sedikit aktivitas.

### Saturasi Oksigen

Saturasi oksigen merupakan nilai yang menunjukkan [kadar oksigen di dalam darah](https://www.alodokter.com/penting-diketahui-ini-kadar-oksigen-normal-dalam-darah" \t "_blank). Nilai ini sangat berpengaruh terhadap berbagai fungsi organ dan jaringan tubuh. Pengukuran nilai saturasi oksigen dapat dilakukan dengan 2 cara, yakni dengan [analisis gas darah](https://www.alodokter.com/analisa-gas-darah-dan-hal-hal-penting-yang-ada-di-dalamnya" \t "_blank) (AGD) atau menggunakan [alat oximeter](https://www.alodokter.com/pentingnya-oximeter-bagi-pasien-isolasi-mandiri-covid-19" \t "_blank).

Analisis gas darah adalah metode pengukuran saturasi oksigen yang dilakukan dengan cara mengambil sampel darah dari pembuluh darah arteri. Hasil analisis gas darah sangat akurat, karena pengukurannya dilakukan di rumah sakit dan dikerjakan oleh tenaga medis profesional.

Sementara itu, oximeter adalah alat pengukur saturasi oksigen yang berbentuk klip. Pengukurannya dilakukan dengan cara menjepitkan oximeter pada jari tangan. Saturasi oksigen kemudian akan diukur berdasarkan jumlah cahaya yang dipantulkan oleh sinar inframerah, yang dikirim ke pembuluh darah kapiler.

Berbeda dengan analisis gas darah, pengukuran saturasi oksigen dengan oximeter bisa dilakukan sendiri dengan mudah di rumah. Oximeter bahkan kini direkomendasikan oleh Badan Kesehatan Dunia (WHO) untuk dimiliki di setiap rumah guna mengukur nilai saturasi oksigen secara berkala.

Hasil pengukuran saturasi oksigen yang dilakukan dengan analisis gas darah ditunjukkan dengan istilah PaO2 (tekanan parsial oksigen). Sementara itu, hasil pengukuran saturasi oksigen dengan menggunakan oximeter ditunjukkan dengan istilah SpO2.

Berikut merupakan tabel untuk nilai saturasi oksigen.

|  |  |
| --- | --- |
| Nilai SpO2 | Kategori |
| 95 – 100% | Normal |
| < 95% | Rendah |

Tabel 2.2 Nilai Saturasi Oksigen

Orang yang memiliki saturasi oksigen rendah atau [hipoksemia](https://www.alodokter.com/penyebab-hipoksemia-dan-cara-mengukurnya" \t "_blank) bisa merasakan berbagai gejala, seperti nyeri dada, sesak napas, batuk, sakit kepala, detak jantung cepat, kebingungan, dan kulit membiru.

Kendati demikian, orang yang mengalami hipoksemia juga bisa tidak merasakan gejala apa pun. Kondisi ini yang disebut dengan [happy hypoxia](https://www.alodokter.com/mengenal-happy-hypoxia-pada-penyakit-covid-19) ini bisa terjadi pasien COVID-19.

Hipoksemia, baik yang menimbulkan gejala maupun tidak, bisa menganggu kerja organ dan jaringan tubuh. Bila dibiarkan, hal ini dapat menyebabkan kerusakan pada organ vital, seperti jantung, otak, dan ginjal, dan berisiko menyebabkan komplikasi yang berbahaya.

#### Hipoksemia

Hipoksemia adalah kondisi di mana kadar oksigen di dalam darah di bawah batas normal. Padahal, oksigen sangat diperlukan untuk menjaga organ dan jaringan tubuh, termasuk jantung, otak, ginjal, dan lainnya, agar tetap berfungsi dengan baik. Hipoksemia bisa terdeteksi melalui pemeriksaan fisik serta [tes darah](https://www.alodokter.com/kenali-jenis-dan-fungsi-tes-darah" \t "_blank).

#### Happy Hypoxia

**Istilah *happy hypoxia* digunakan untuk menunjukkan kondisi berkurangnya kadar oksigen di dalam tubuh tanpa menimbulkan gejala. Meski sulit dikenali, kondisi ini perlu diwaspadai karena dapat berakibat fatal, terutama bagi penderita COVID-19.**

Hingga saat ini, penyebab [h*appy hypoxia*](https://www.alodokter.com/mengenal-happy-hypoxia-pada-penyakit-covid-19) belum diketahui secara pasti. Namun, ada teori yang menyebutkan bahwa *happy hypoxia* terjadi akibat peradangan pada jaringan paru-paru yang disebabkan oleh infeksi [virus Corona](https://www.alodokter.com/virus-corona).

# BAB 3 ANALISIS DAN PERANCANGAN SISTEM

## Analisis Masalah

Pada tahap ini akan dijelaskan masalah yang ditemui dan akan menjadi sebuah dasar dalam pembuatan aplikasi. Seperti yang kita tahu, kesehatan merupakan salah satu hal yang sangat berharga di dalam hidup kita. Dengan tubuh yang sehat, kita sanggup melakukan pekerjaan dan kesibukan tanpa adanya kendala. Salah satu upaya dalam menjaga kesehatan adalah dengan memantau tanda-tanda kesehatan vital seperti detak jantung dan saturasi oksigen. Akan tetapi, saat ini belum terdapat aplikasi yang dapat memberikan informasi terkait kondisi kesehatannya dan juga memberikan informasi tips dan juga penanganan pertama jika teridentifikasi kondisi kesehatannya dibawah rata-rata.

### Analisis Sistem Sedang Berjalan

Sebelum membangun suatu aplikasi, tahap awal yang wajib dilakukan adalah mempelajari dan menganalisis aplikasi yang ada atau sedang berjalan. Saat ini kebanyakan aplikasi hanya menambahkan data kesehatan lalu melihat riwayat penambahan data tanpa adanya informasi apakah data tersebut termasuk normal atau diluar normal dan juga tidak diberikannya informasi terkait tips dan penanganan terhadap kondisi kesehatannya jika teridentifikasi berada diluar normal.

### Deskripsi Kebutuhan Non Fungsional

Deskripsi kebutuhan non-fungsional merupakan analisis yang dibutuhkan untuk menentukan spesifikasi kebutuhan sistem. Spesifikasi ini juga meliputi elemen atau komponen-komponen apa saja yang dibutuhkan untuk sistem yang akan dibangun sampai dengan sistem tersebut diimplementasikan. Analisis kebutuhan ini juga menentukan spesifikasi masukan yang diperlukan sistem, keluaran yang akan dihasilkan sistem dan proses yang dibutuhkan untuk mengolah masukan sehingga menghasilkan suatu keluaran yang diinginkan.

#### Deskripsi Kebutuhan Perangkat Keras

Spesifikasi perangkat keras yang digunakan untuk membangun aplikasi adalah sebagai berikut :

1. Procesor : Amd A8
2. RAM : 8 GB RAM
3. Harddisk : 500GB
4. Keyboard
5. Mouse

#### Analisis Kebutuhan Perangkat Lunak

Spesifikasi perangkat lunak pendukung yang dibutuhkan dalam pembangunan aplikasi yaitu :

1. Sistem Operasi : Windows 11
2. IDE : Android Studio
3. Code Editor : Visual Studio Code
4. Google Chrome
5. PgAdmin 4
6. Genymotion
7. Postman

### Analisis Kebutuhan Fungsional

Analisis kebutuhan fungsional dapat mendefinisikan sebagai gambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen untuk dijadikan satu kesatuan yang utuh dan berfungsi. Tahap ini menyangkut konfigurasi dari komponen perangkat lunak dan perangkat keras dari suatu sistem sehingga instalasi dari sistem bisa berfungsi.

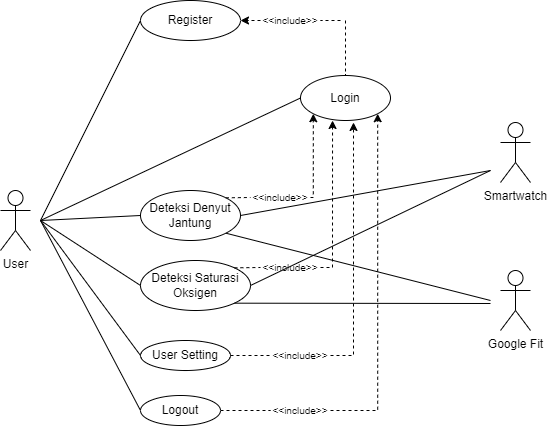
Alat bantu yang digunakan untuk menggambarkan sistem secara umum menggunakan diagram konteks. Untuk menjelaskan bagaimana suatu masukan diproses pada sistem, maka dibutuhkan spesifikasi proses untuk mengetahui aliran data pada sistem.

Tabel 3. 1 Kebutuhan Fungsional

|  |  |  |  |
| --- | --- | --- | --- |
| No | Kode Fungsi | Nama Fungsi | Deskripsi |
| 1 | SKPL-F-01 | Halaman Login | Pada halaman ini aplikasi akan menampilkan halaman login untuk otentikasi agar bisa masuk ke halaman utama aplikasi. |
| 2 | SKPL-F-02 | Halaman Register | Pada halaman ini aplikasi akan menampilkan halaman register untuk pembuatan akun agar user bisa masuk ke aplikasi. |
| 3 | SKPL-F-03 | Halaman Home | Pada halaman ini akan informasi seperti nama user, widget yang akan menampilkan data terbaru dari denyut jantung dan juga saturasi oksigen, widget yang akan menampilkan informasi terkait kondisi denyut jantung apakah termasuk rendah, normal, atau tinggi, dan juga widget yang menampilkan informasi kondisi saturasi oksigen. |
| 4 | SKPL-F-04 | Halaman Denyut Jantung | Pada halaman ini akan ditampilkan data-data terkait denyut jantung seperti data terbaru, data maksimum, data minimum dan juga data riwayat pengecekan denyut jantung. |
| 5 | SKPL-F-05 | Halaman Saturasi Oksigen | Pada halaman ini akan ditampilkan data-data terkait saturasi oksigen seperti data terbaru, data maksimum, data minimum dan juga data riwayat pengecekan saturasi oksigen. |
| 6 | SKPL-F-06 | Halaman User | Pada halaman ini akan ditampilkan informasi dari user yang login seperti nama, jenis kelamin, umur, tinggi badan, berat badan dan juga ada tombol yang digunakan untuk logout dari aplikasi. |
| 7 | SKPL-F-07 | Logout | Aplikasi akan mengakhiri sesi login user. |

#### Use Case Diagram

Use Case Diagram adalah diagram yang menggambarkan hubungan interaksi antara sistem dan actor. Use case dapat mendeskripsikan tipe interaksi antara si pengguna sistem dengan sistemnya.



Gambar 3.1 Use Case Diagram

#### Use Case Description

Use case description berisi penjelasan mengenai usecase diagram. Penjelasan terdiri dari deskripsi mengenai use case, aktor, pre-kondisi, alur use case tersebut, post-condition, dan alternatif yang terjadi dari use case.

*Use case description register*

Tabel 3. 2 Use Case Description Register

|  |  |
| --- | --- |
| Nama Use Case | Register |
| Deskripsi | Aktor melakukan registrasi akun terlebih dahulu untuk bisa menggunakan aplikasi. Data-data yang digunakan untuk registrasi diantaranya nama, email, password, jenis kelamin, umur, tinggi badan dan juga berat badan. |
| Aktor dan Interface | Aktor: User  Interface: Halaman Register |
| Pre-Condition | User harus registrasi akun terlebih dahulu untuk dapat menggunakan aplikasi. |
| Basic Flow | * User memilih menu daftar * Sistem menampilkan halaman register. * User mengisi data-data yang dibutuhkan. * User mengklik tombol daftar * Sistem akan mengecek apakah email sudah digunakan sebelumnnya atau belum, dan akan memvalidasi data lainnya. * Jika gagal, akan pesan error. * Jika berhasil, user akan diarahkan ke halaman login. |
| Post-Condition | User berhasil registrasi akun. |
| Alternative Flow | Authentikasi gagal dan akan menampilkan pesan error. |

*Use case description login*

Tabel 3. 3 Use Case Description Login

|  |  |
| --- | --- |
| Nama Use Case | Login |
| Deskripsi | User melakukan login untuk dapat memasuki aplikasi. Login menggunakan akun yang sebelumnya sudah di registrasi. |
| Aktor dan Interface | Aktor: User  Interface: Halaman Login |
| Pre-Condition | User harus sudah memiliki akun yang sebelumnya sudah dibuat pada saat proses registrasi akun. |
| Basic Flow | * User membuka aplikasi * Sistem akan menampilkan halaman login. * User mengisi form login. * User mengklik tombol login. * User akan diarahkan ke halaman utama aplikasi. |
| Post-Condition | User berhasil login dan data user login akan disimpan agar ketika user membuka aplikasi tidak perlu untuk melakukan lagi. |
| Alternative Flow | Authentikasi gagal dan akan memunculkan pesan error. |

*Use case description deteksi denyut jantung*

Tabel 3. 4 User Case Description Deteksi Denyut Jantung

|  |  |
| --- | --- |
| Nama Use Case | Deteksi Denyut Jantung |
| Deskripsi | Proses ini dilakukan untuk melakukan pengecekan apakah kondisi denyut jantung dari user termasuk ke dalam kategori normal, dibawah nilai normal atau termasuk tinggi. Dan jika denyut jantung terdeteksi rendah atau tinggi, aplikasi akan mengirimkan notifikasi ke smartwatch. |
| Aktor dan Interface | Aktor: User, Smartwatch  Interace: Halaman Denyut Jantung |
| Pre-Condition | User sudah login ke dalam aplikasi menggunakan Firebase Authentication |
| Basic Flow | * User melakukan pengecekan denyut jantung. * User melakukan sinkronisasi data menggunakan aplikasi Health Sync. * User membuka halaman denyut jantung dan melakukan refresh halaman. * Sistem akan menampilkan data denyut jantung dari smartwatch. * Sistem akan mengirimkan notifikasi ke smartwatch jika denyut jantung berada dibawah normal ataupun diatas normal. |
| Post-Condition | Data denyut jantung berhasil ditampilkan. |
| Alternative Flow | Denyut jantung gagal ditampilkan, karena aplikasi tidak terhubung dengan smartwatch. |

*Use case description deteksi saturasi oksigen*

Tabel 3. 5 User Case Description Deteksi Saturasi Oksigen

|  |  |
| --- | --- |
| Nama Use Case | Deteksi Saturasi Oksigen |
| Deskripsi | Proses ini dilakukan untuk melakukan pengecekan apakah nilai saturasi oksigen dari user termasuk ke dalam kategori normal, atau rendah. Dan jika nilai saturasi oksigen terdeteksi rendah, aplikasi akan mengirimkan notifikasi ke smartwatch. |
| Aktor dan Interface | Aktor: User, Smartwatch  Interace: Halaman Denyut Jantung |
| Pre-Condition | User sudah login ke dalam aplikasi menggunakan Firebase Authentication |
| Basic Flow | * User melakukan pengecekan saturasi oksigen. * User melakukan sinkronisasi data menggunakan aplikasi Health Sync. * User membuka halaman saturasi oksigen dan melakukan refresh halaman. * Sistem akan menampilkan data saturasi oksigen dari smartwatch. * Sistem akan mengirimkan notifikasi ke smartwatch jika nilai saturasi oksigen rendah. |
| Post-Condition | Data nilai saturasi oksigen berhasil ditampilkan. |
| Alternative Flow | Saturasi oksigen gagal ditampilkan, karena aplikasi tidak terhubung dengan smartwatch. |

*Use case description user setting*

Tabel 3. 6 User Case Description User Setting

|  |  |
| --- | --- |
| Nama Use Case | User Setting |
| Deskripsi | Proses ini digunakan untuk mengubah data-data profile dari user seperti nama, jenis kelamin, berat badan, tinggi badan. |
| Aktor dan Interface | Aktor: User  Interface: Halaman User |
| Pre-Condition | User sudah login ke dalam aplikasi menggunakan Firebase Authentication |
| Basic Flow | * User memilih menu halaman user. * User memilih data yang akan diubah. * User mengubah data yang akan diubah. * Aplikasi akan memperbarui data profile dari user dan akan menyimpannya di database. |
| Post-Condition | Data user berhasil diperbarui dan berhasil disimpan ke dalam database. |
| Alternative Flow | Gagal memperbarui data user. |

*Use case description logout*

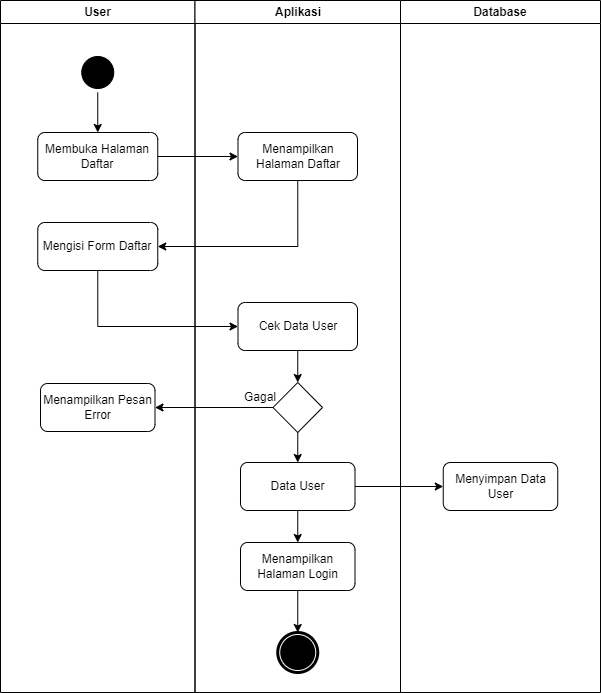
Tabel 3. 7 User Case Description Logout

|  |  |
| --- | --- |
| Nama Use Case | Logout |
| Deskripsi | Proses ini dilakukan untuk melakukan logout dan mengakhiri sesi login yang aktif. |
| Aktor dan Interface | Aktor: User  Interface: Halaman User |
| Pre-Condition | User sudah login ke dalam aplikasi menggunakan Firebase Authentication |
| Basic Flow | * User masuk ke halaman user * User mengklik tombol logout * Aplikasi akan melakukan logout dan mengakhri sesi login yang aktif. |
| Post-Condition | Berhasil logout dan akan diarahkan ke halaman login. |
| Alternative Flow | - |

#### Activity Diagram

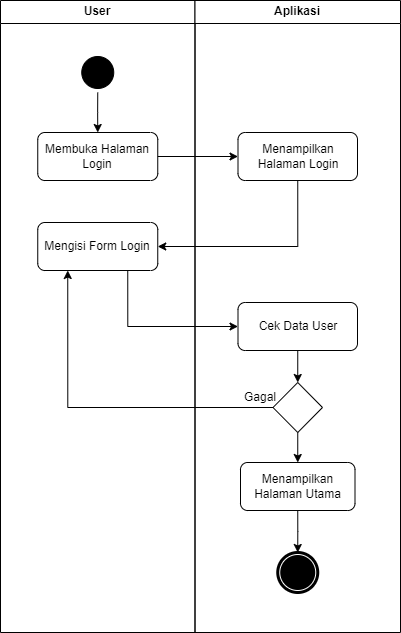
Activity diagram merupakan penggambaran proses dari aktivitas yang dilakukan. Activity diagram pada aplikasi ini yaitu sebagai berikut:

Activity diagram register dapat dilihat pada gambar 3.2



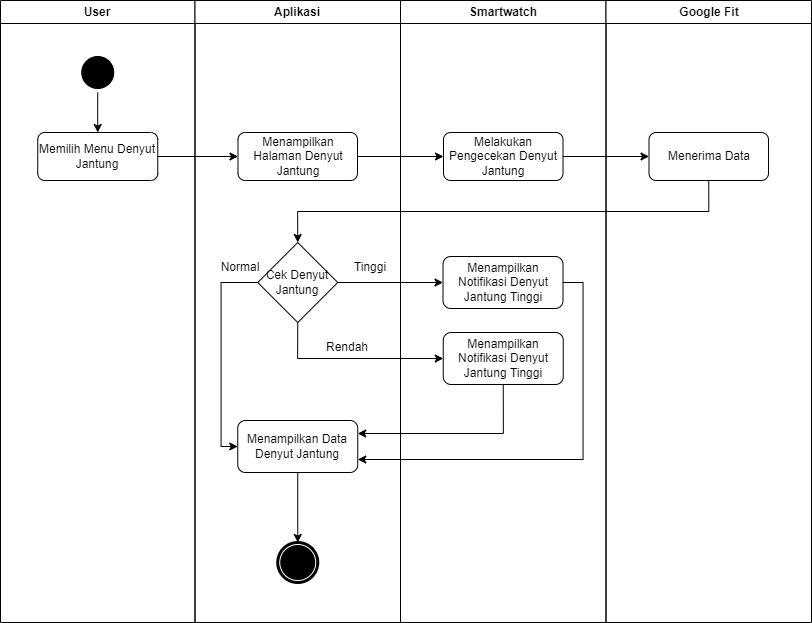
Gambar 3. 2 Activity Diagram Register

Activity diagram login dapat dilihat pada gambar 3.3



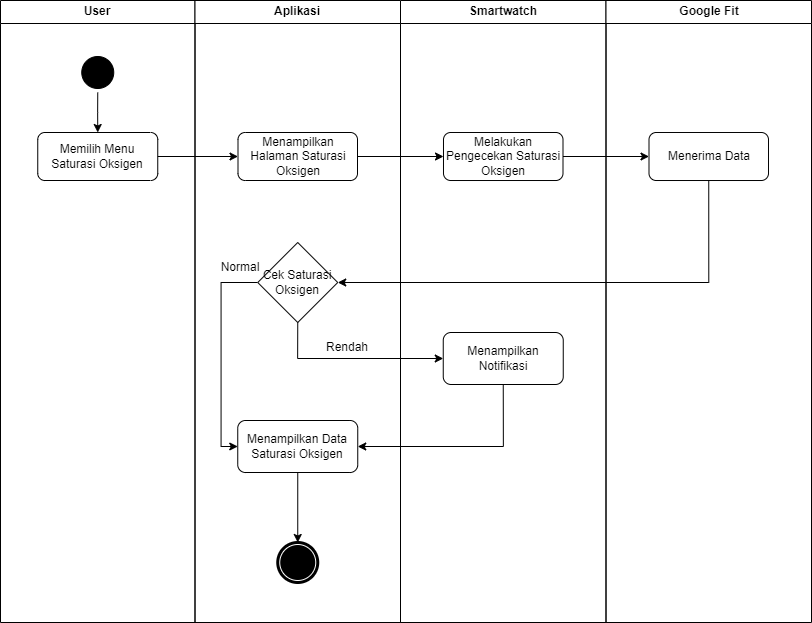
Gambar 3. 3 Activity Diagram Login

Activity diagram deteksi denyut jantung dapat dilihat pada gambar 3.4



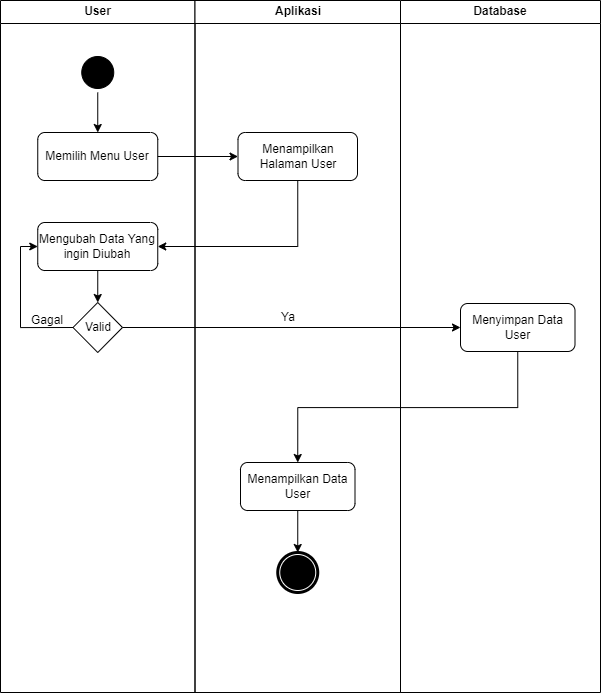
Gambar 3. 4 Activity Diagram Deteksi Denyut jantung

Activity diagram deteksi saturasi oksigen dapat dilihat pada gambar 3.5



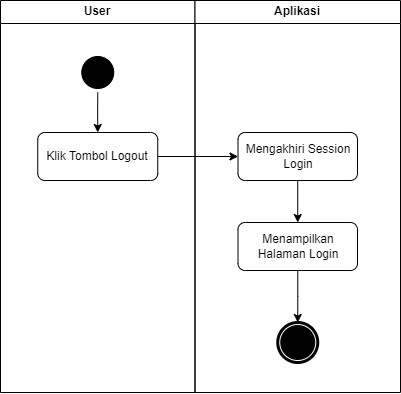
Gambar 3. 5 Activity Diagram Deteksi Saturasi Oksigen

Activity diagram user setting dapat dilihat pada gambar 3.6



Gambar 3. 6 Activity Diagram User Setting

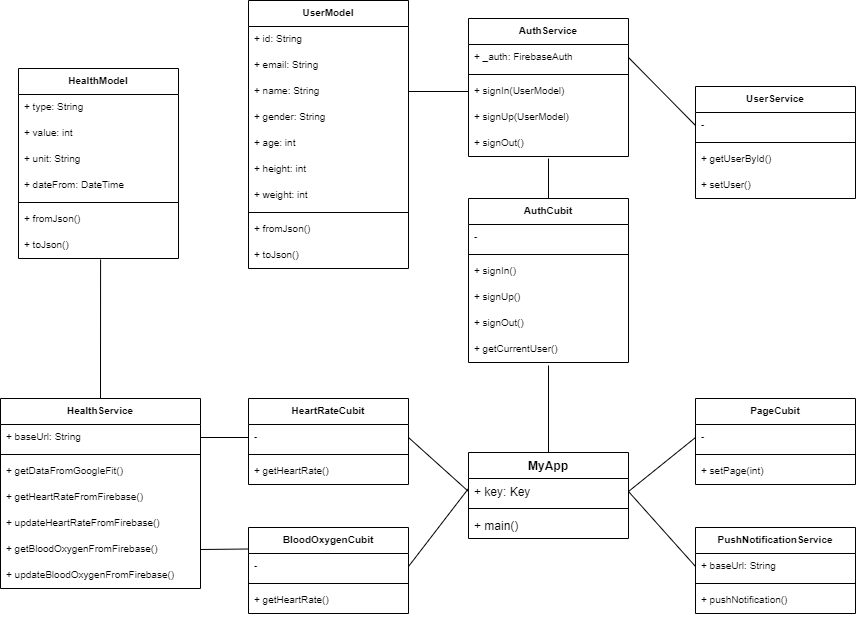
Activity diagram logout dapat dilihat pada gambar 3.7



Gambar 3. 7 Activity Diagram Logout

#### Class Diagram

Class diagram adalah penggambaran kelas dari setiap objek yang dibuat dalam aplikasi dan menunjukkan hubungan tiap kelas. Class diagram pada aplikasi ini adalah sebagai berikut:

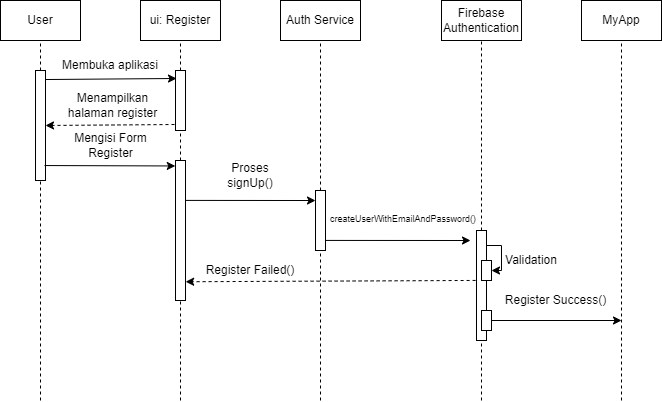


Gambar 3.8 Class Diagram

#### Sequence Diagram

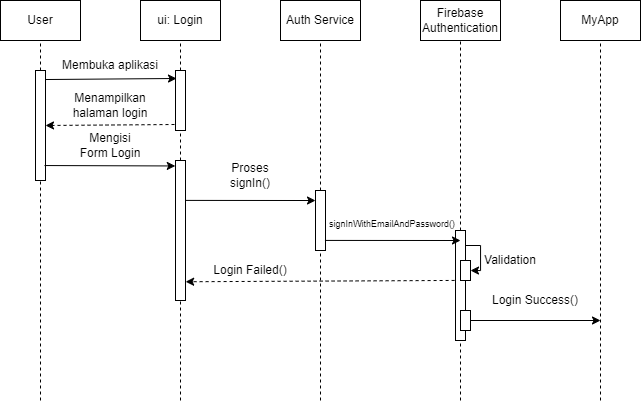
Sequence Diagram adalah sebuah diagram yang digunakan untuk menjelaskan dan menampilkan interaksi antar objek-objek dalam sebuah aplikasi secara terperinci. Selain itu sequence diagram juga akan menampilkan pesan atau perintah yang dikirim, beserta waktu pelaksanaanya.

Sequence diagram register dapat dilihat pada gambar 3.9



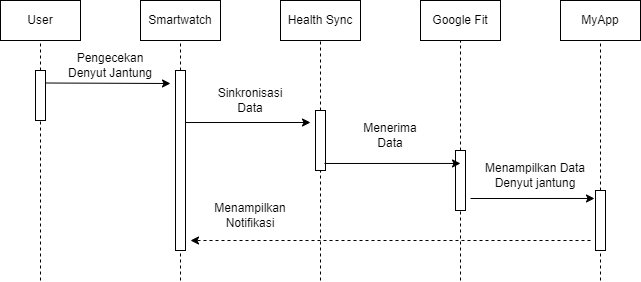
Gambar 3. 9 Sequence Diagram Register

Sequence diagram login dapat dilihat pada gambar 3.10



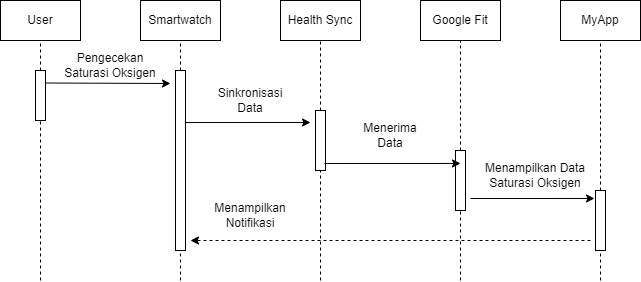
Gambar 3. 10 Sequence Diagram Login

Sequence diagram deteksi denyut jantung dapat dilihat pada gambar 3.11



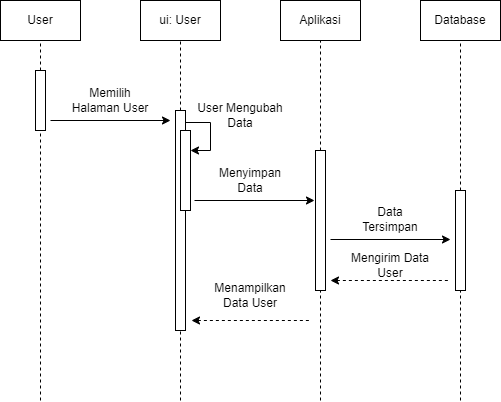
Gambar 3. 11 Sequence Diagram Deteksi Denyut Jantung

Sequence diagram deteksi saturasi oksigen dapat dilihat pada gambar 3.12



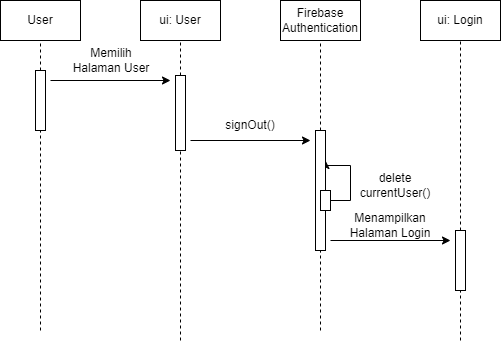
Gambar 3. 12 Sequence Diagram Deteksi Saturasi Oksigen

Sequence diagram user setting dapat dilihat pada gambar 3.13



Gambar 3. 13 Sequence Diagram User Setting

Sequence diagram logout dapat dilihat pada gambar 3.14



Gambar 3. 14 Sequence Diagram Logout

## Perancangan

### Perancangan Struktur Tabel

Dikarenakan basis data akan menggunakan Firebase Cloud Firestore. Maka struktur basis data pada server berupa json, yang mana per node yang bisa dibilang fieldnya dinamakan child. Selain itu tidak ada primary key ataupun foreign key pada struktur database ini. Berikut merupakan perancangan basis data yang akan dibuat ;

1. Collection User

Tabel 3. 8 Collection User

|  |  |
| --- | --- |
| Nama Child | Tipe Data |
| id | String |
| name | String |
| email | String |
| password | String |
| gender | String |
| age | Number |
| height | Number |
| weight | Number |

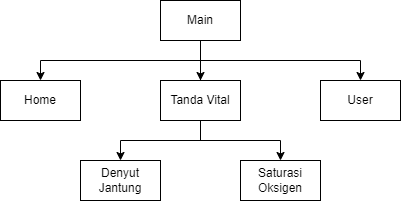
1. Collection Health

Tabel 3. 9 Tabel Data Kesehatan

|  |  |
| --- | --- |
| Nama Child | Tipe Data |
| tipe | String |
| value | Number |
| unit | String |
| dateFrom | String |

### Perancangan Struktur Menu

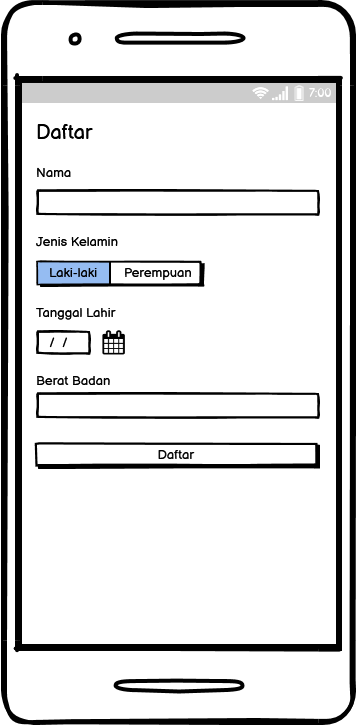
Pembuatan aplikasi monitoring kesehatan memiliki struktur menu sebagai berikut:



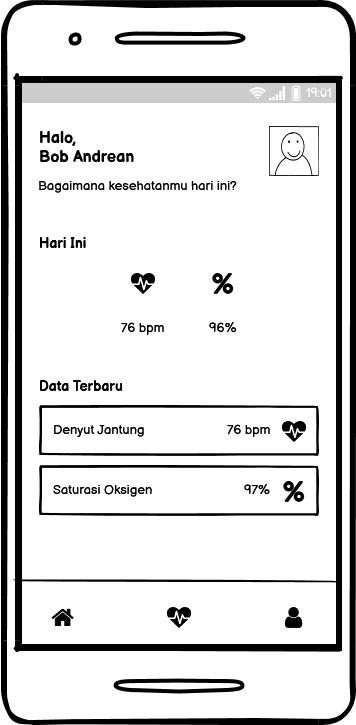
Gambar 3.15 Struktur Menu

### Perancangan Antar Muka

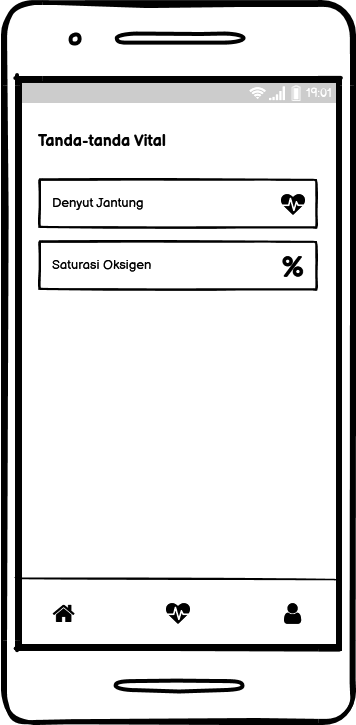
Perancangan antarmuka ini untuk membuat gambar sistem yang akan dibuat nantinya.



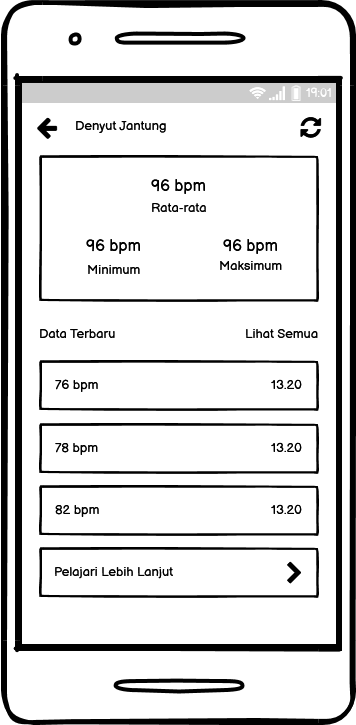
Gambar 3.16 Rancangan Antarmuka Halaman Daftar



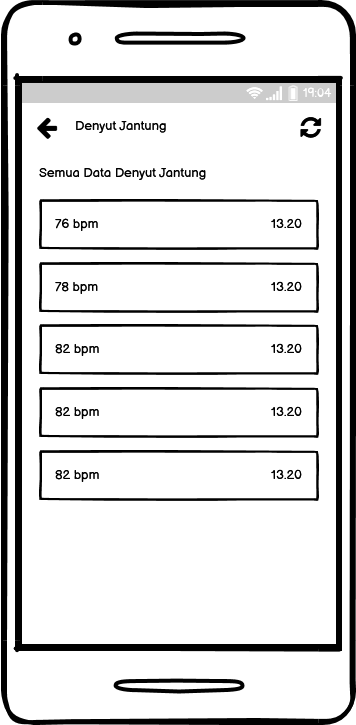
Gambar 3.17 Rancangan Antarmuka Halaman Home



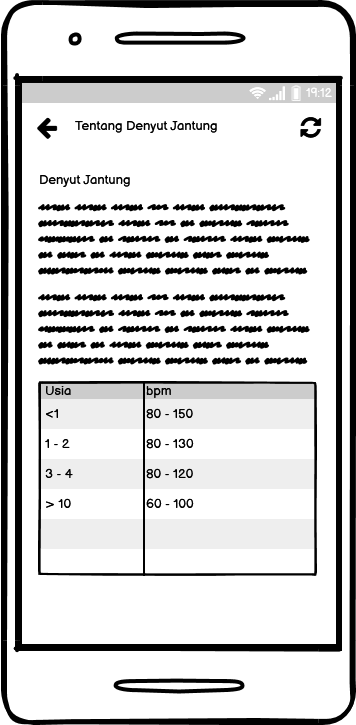
Gambar 3.18 Rancangan Antarmuka Halaman Tanda Vital



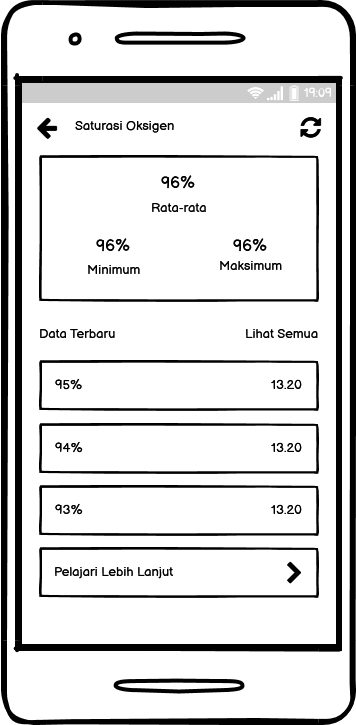
Gambar 3.19 Rancangan Antarmuka Halaman Denyut Jantung



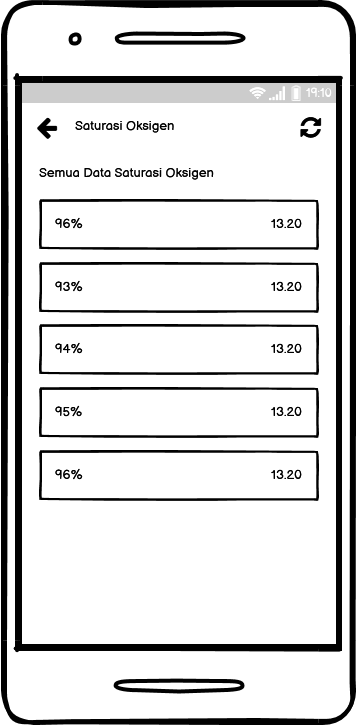
Gambar 3.20 Rancangan Antarmuka Halaman Data Denyut Jantung



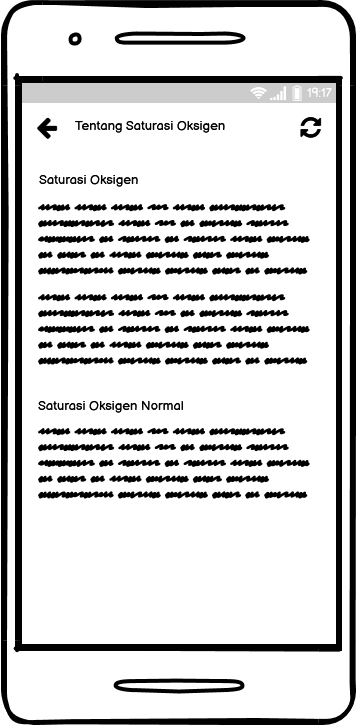
Gambar 3.21 Rancangan Antarmuka Halaman Tentang Denyut Jantung



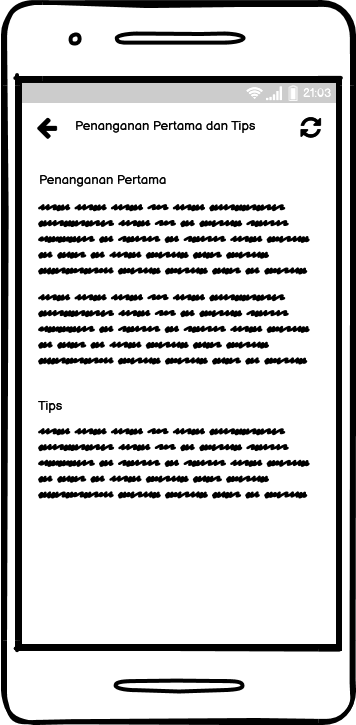
Gambar 3.22 Rancangan Antarmuka Halaman Saturasi Oksigen



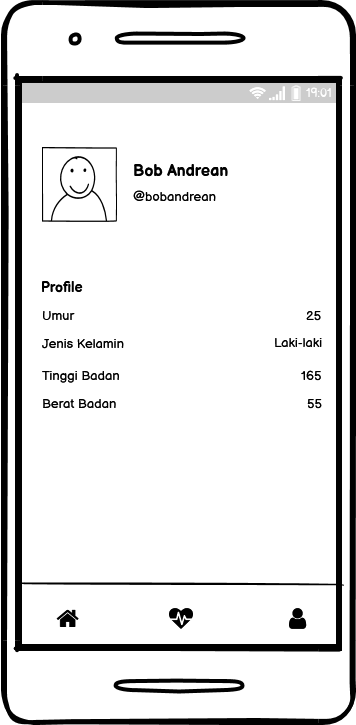
Gambar 3.23 Rancangan Antarmuka Halaman Data Saturasi Oksigen



Gambar 3.24 Rancangan Antarmuka Halaman Tentang Saturasi Oksigen



Gambar 3.25 Rancangan Antarmuka Halaman Tips dan Penanganan Pertama



Gambar 3.26 Rancangan Antarmuka Halaman User Profile

# DAFTAR PUSTAKA

[1] M. Rifqi, “Pentingnya Menjaga Kesehatan”, 16 April 2018, [Online], Available:<https://kumparan.com/muhamad-rifki-a/pentingnya-menjaga-kesehatan/1>, [Accessed 30 Januari 2022]

[2] Satria, “Menjaga Kesehatan Jantung untuk Hidup Lebih Berkualitas”, 7 Oktober 2021, [Online], Available:<https://ugm.ac.id/id/berita/21772-menjaga-kesehatan-jantung-untuk-hidup-lebih-berkualitas>, [Accessed 30 Januari 2022]

[3] F. R. Makarim, “Beda atau Sama Detak Jantung Normal pada Anak dan Dewasa?”, 3 September 2021, [Online], Available:<https://www.halodoc.com/artikel/beda-atau-sama-detak-jantung-normal-pada-anak-dan-dewasa>, [Accessed 31 Januari 2022 ]

[4] K. Adrian, “Mengetahui Nilai Saturasi Oksigen dan Cara Meningkatkannya”, 9 Juli 2021, [Online], Available:<https://www.alodokter.com/mengetahui-nilai-saturasi-oksigen-dan-cara-meningkatkannya>, [Accessed 31 Januari 2022]

[5] D. Archiando, “4 Alasan Pentingnya Menjaga Tekanan Darah Normal”, 9 September 2020, [Online], *Available:https://lifepack.id/4-alasan-pentingnya-menjaga-tekanan-darah-*normal/#:~:text=Tensi%20darah%20yang%20normal%20dapat,yang%20dapat%20berakibat%20serangan%20jantung., [Accessed 6 Februari 2022]

[6] A. S. Utomo, E. H. P. Negoro, M. Sofie, “Monitoring Heart Rate dan Saturasi Oksigen Melalui Smartphone”, Jurnal SIMETRIS, Vol. 10, No. 1, 2019.

[7] M. Nareza, “Penting Diketahui, Ini Kadar Oksigen Normal dalam Darah”, 12 Februari 2021, [Online], Available: <https://www.alodokter.com/penting-diketahui-ini-kadar-oksigen-normal-dalam-darah>, [Accessed 7 Februari 2022]

[8] S. You, “Tergantung Usia, Berapa Tekanan Darah Normal Kamu?”, 26 Juni 2020, [Online], Available: <https://superyou.co.id/blog/kesehatan/tekanan-darah-normal-berdasarkan-usia/>, [Accessed 11 Februari 2022].

[9] S. Pers, “Ini Pentingnya Memantau Kesehatan saat Pandemi”, 28 Juni 2021, [Online], Available: <https://koranbernas.id/ini-pentingnya-memantau-kesehatan-saat-pandemi>, [Accessed 30 Januari 2022].

[10] I. F. Faisal, A. P. Kharisma, Sutrisno, “Pengembangan Aplikasi Pendeteksi Kantuk Pada Pengendara Kendaraan Bermotor Dengan Menggunakan Sensor Detak Jantung Pada Smartwatch”, Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer, Vol. 3, No.10, pp. 9568-9578, 2019.

[11] V. Yonanto, D. W. G. Wisana, T. Rahmawati, “Pemantauan SpO2 Melalui Aplikasi Android di Mobile Phone”, TEKNOKES, Vol. 12, No. 2, pp. 21-27, 2019.

[12] D. P. Nugroho, R. Munadi, I. H. Santoso, “Sistem Pemantauan Kondisi Detak Jantung Berbasis Internet of Things Menggunakan Sensor EKG Dengan Media Aplikasi Android”, e-Proceeding of Engineering, Vol. 8, No. 5, pp. 5530-5536, 2021.

[13] A. Andriani, R. Hartono, “Saturasi Oksigen Dengan Pulse Oximetry Dalam 24 Jam Pada Pasien Dewasa Terpasang Ventilator Di Ruang ICU Rumah Sakit Panti Wilasa Citarum Semarang”, Jendela Nursing Journal, Vol. 2, No. 1, 2013.

[14] K. Adrian, “Pentingnya Oximeter bagi Pasien Isolasi Mandiri COVID-19”, 1 Februari 2021, [Online], Available: <https://www.alodokter.com/pentingnya-oximeter-bagi-pasien-isolasi-mandiri-covid-19>, [Accessed 28 Januari 2022].