

**IMPLEMENTASI MODEL *GATED RECURRENT UNIT* (GRU) UNTUK
PEMANTAUAN KESEHATAN BERKELANJUTAN PADA
SMARTWATCH ANDROID BERBASIS IOT**

SKRIPSI

Oleh:
KHOLILATUL IZZAH
NIM. 220605110125



**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2026**

**IMPLEMENTASI MODEL *GATED RECURRENT UNIT* (GRU) UNTUK
PEMANTAUAN KESEHATAN BERKELANJUTAN PADA
SMARTWATCH ANDROID BERBASIS IOT**

SKRIPSI

Diajukan kepada:
Universitas Islam Negeri Maulana Malik Ibrahim Malang
Untuk memenuhi Salah Satu Persyaratan dalam
Memperoleh Gelar Sarjana Komputer (S.Kom)

Oleh:
KHOLILATUL IZZAH
NIM. 220605110125

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2026**

HALAMAN PERSETUJUAN

**IMPLEMENTASI MODEL *GATED RECURRENT UNIT* (GRU) UNTUK
PEMANTAUAN KESEHATAN BERKELANJUTAN PADA
SMARTWATCH ANDROID BERBASIS IOT**

SKRIPSI

Oleh :
KHOLILATUL IZZAH
NIM. 220605110125

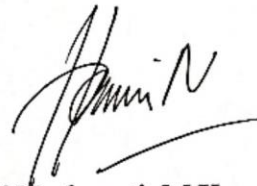
Telah Diperiksa dan Disetujui untuk Diuji:
Tanggal: 23 Juni 2026

Pembimbing I,



Prof. Dr. Muhammad Faisal, M.T
NIP. 19740510 200501 1 007

Pembimbing II,



Hani Nurhayati, M.Kom
NIP. 19780625 200801 2 006

Mengetahui,
Ketua Program Studi Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang



Supriyanto, M.Kom
NIP. 19841010 201903 1 012

HALAMAN PENGESAHAN

IMPLEMENTASI *GATED RECURRENT UNIT* (GRU) UNTUK PEMANTAUAN KESEHATAN BERKELANJUTAN PADA *SMARTWATCH* ANDROID BERBASIS IOT

SKRIPSI

Oleh :
KHOLILATUL IZZAH
NIM. 220605110125

Telah Dipertahankan di Depan Dewan Penguji Skripsi
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
Untuk Memperoleh Gelar Sarjana Komputer (S.Kom)
Tanggal: 23 Juni 2026

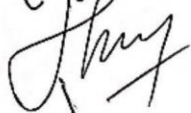
Susunan Dewan Penguji

Ketua Penguji : Ahmad Fabmi Karami, M.Kom
NIP. 19870909 202012 1 001

Anggota Penguji I : Shoffin Nahwa Utama, M.T
NIP. 19860703 202012 1 003

Anggota Penguji II : Prof. Dr. Muhammad Faisal, M.T
NIP. 19740510 200501 1 007

Anggota Penguji III : Hani Nurhayati, M.T
NIP. 19780625 200801 2 006

()
()
()
()

Mengetahui dan Mengesahkan,
Ketua Program Studi Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang



Supriyono, M.Kom
NIP. 19801010 201903 1 012

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama : Kholilatul Izzah
NIM : 220605110125
Fakultas / Program Studi : Sains dan Teknologi / Teknik Informatika
Judul Skripsi : IMPLEMENTASI *MODEL GATED RECURRENT UNIT (GRU)* UNTUK PEMANTAUAN KESEHATAN BERKELANJUTAN PADA *SMARTWATCH* ANDROID BERBASIS IOT

Menyatakan dengan sebenarnya bahwa Skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambil alihan data, tulisan, atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka.

Apabila dikemudian hari terbukti atau dapat dibuktikan skripsi ini merupakan hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 25 Juni 2026

Yang membuat pernyataan,



Kholilatul Izzah
NIM.220605110125

MOTTO

"Fall seven times, stand up eight."

— *Japanese Proverb*

"It always seems impossible until it's done."

— *Nelson Mandela*

HALAMAN PERSEMBAHAN

Karya ini saya persembahkan dengan penuh rasa hormat, cinta, dan kasih sayang kepada kedua orang tua tercinta, Bapak Suliyanto dan Ibu Siti Khofsah beserta kedua adik saya Nia dan Yaqin. Terima kasih atas doa yang selalu mengiringi setiap langkah saya, rida yang tulus, serta segala pengorbanan waktu, tenaga, dan kasih sayang yang telah diberikan tanpa henti. Meskipun perjalanan ini membutuhkan waktu yang lebih panjang dari yang saya bayangkan, berkat dukungan, kesabaran, serta restu Bapak dan Ibu, dan atas izin Allah Subhanahu wa Ta'ala, saya akhirnya mampu menyelesaikan skripsi ini hingga meraih gelar sarjana. Semoga Bapak dan Ibu senantiasa dianugerahi kesehatan, umur yang penuh keberkahan, serta kebahagiaan yang tiada henti.

Karya ini juga saya persembahkan kepada sahabat-sahabat terdekat saya di kampus, yaitu Lailatul, Humayroh, Intan, dan Deshinta, serta sahabat-sahabat lainnya yang tidak dapat saya sebutkan satu per satu. Terima kasih atas kebersamaan, dukungan, dan kepedulian yang selalu kalian berikan selama ini. Kalian selalu hadir dalam setiap fase perjalanan, baik dalam suka maupun duka, dengan sabar mendengarkan segala keluh kesah dan memberikan semangat di saat saya merasa lelah atau kehilangan arah. Dari kalian, saya banyak belajar tentang arti ketulusan, keikhlasan, serta pentingnya rasa syukur dalam menjalani kehidupan. Kehadiran kalian menjadi salah satu anugerah yang sangat berarti dalam perjalanan ini.

Karya ini turut saya persembahkan kepada teman-teman ABA 46, rekan-rekan PKL dan KKN, serta seluruh mahasiswa Teknik Informatika angkatan 2022. Perjalanan selama menempuh pendidikan ini terasa lebih berwarna karena kebersamaan, kerja sama, dan pengalaman yang kita lalui bersama. Setiap tawa, perjuangan, kerja keras, serta kenangan yang telah tercipta akan selalu menjadi bagian berharga yang tidak akan terlupakan. Semoga kita semua senantiasa diberikan kemudahan dan kelancaran dalam setiap langkah ke depan, serta mampu meraih cita-cita dan kesuksesan masing-masing dengan penuh kebanggaan.

Terakhir, karya ini saya persembahkan untuk diri saya sendiri. Untuk setiap malam panjang yang telah dilalui, untuk setiap keraguan yang pernah hadir dan sempat melemahkan langkah, serta untuk setiap air mata yang jatuh dalam keheningan. Perjalanan ini bukanlah sesuatu yang mudah, dan ada begitu banyak fase di mana saya hampir memilih untuk menyerah. Namun, pada akhirnya saya memutuskan untuk terus bertahan dan melangkah maju.

Rasa syukur yang mendalam saya panjatkan kepada Allah Subhanahu wa Ta'ala atas segala kekuatan, kesabaran, dan keteguhan hati yang telah dianugerahkan, sehingga saya mampu sampai pada titik ini. Karya ini menjadi bukti nyata bahwa saya mampu melewati setiap proses, bertahan dalam setiap ujian, dan akhirnya sampai pada tujuan yang telah diperjuangkan.

KATA PENGANTAR

Segala puji bagi Allah Subhānahu wa Ta‘ālā atas segala nikmat dan kasih sayang-Nya yang telah memudahkan penulis untuk menyelesaikan skripsi yang berjudul “IMPLEMENTASI MODEL *GATED RECURRENT UNIT* (GRU) UNTUK PEMANTAU KESEHATAN BERKELANJUTAN PADA *SMARTWATCH* ANDROID BERBASIS IOT”. Shalawat serta salam senantiasa tercurah kepada junjungan kita Nabi Muhammad ﷺ ‘Alaihi Wasallam, keluarga, sahabat, serta para pengikutnya hingga akhir zaman. Semoga kita semua mendapatkan syafaat beliau di hari kiamat kelak, āmīn yā Rabbal ‘ālamīn.

Dengan rasa syukur yang mendalam, penulis menyadari bahwa penyelesaian skripsi ini tidak lepas dari bantuan, bimbingan, dan doa dari berbagai pihak. Oleh karena itu, penulis ingin menyampaikan ucapan terima kasih yang sebesar-besarnya kepada:

1. Prof. Dr. Hj. Ilfi Nurdiana, M.Si., CAHRM., CRMP, selaku Rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang.
2. Dr. Agus Mulyono, M.Kes, selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
3. Supriyono, M.Kom., selaku Ketua Program Studi Teknik Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang.
4. Prof. Dr. Muhammad Faisal, M.T., selaku Dosen Pembimbing I, yang dengan sabar, telaten, dan penuh dedikasi telah memberikan bimbingan, arahan, serta masukan yang sangat berharga sehingga penulis dapat menyelesaikan skripsi ini.

5. Hani Nurhayati, M.T, selaku Dosen Pembimbing II, yang telah meluangkan waktu dan pikiran untuk membimbing, mengarahkan, serta memberikan kritik dan saran yang konstruktif selama proses pengerjaan skripsi.
6. Ahmad Fahmi Karami, M.Kom dan Shoffin Nahwa Utama, M.T, selaku penguji yang telah memberikan masukan dan saran sehingga skripsi ini dapat disempurnakan.
7. Segenap dosen, staf administrasi, laboran, dan teman-teman mahasiswa Program Studi Teknik Informatika Fakultas Sains dan Teknologi yang telah memberikan dukungan moril maupun materiil selama penulis menempuh pendidikan dan mengerjakan skripsi ini.

Penulis menyadari bahwa skripsi ini masih jauh dari sempurna. Oleh karena itu, segala kritik, saran, dan masukan yang membangun sangat penulis harapkan demi kesempurnaan di masa mendatang. Semoga skripsi ini bermanfaat bagi penulis khususnya, dan bagi pembaca pada umumnya.

Malang, 25 Juni 2026

Penulis

DAFTAR ISI

HALAMAN PENGAJUAN	ii
HALAMAN PERSETUJUAN	iii
HALAMAN PENGESAHAN	iv
PERNYATAAN KEASLIAN TULISAN	v
MOTTO	vi
HALAMAN PERSEMBAHAN	vii
KATA PENGANTAR.....	ix
DAFTAR ISI.....	xi
DAFTAR GAMBAR.....	xiii
DAFTAR TABEL	xiv
ABSTRAK	xv
ABSTRACT	xvi
مستخلص البحث.....	xvii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	4
1.3 Batasan Masalah	4
1.4 Tujuan Penelitian.....	5
1.5 Manfaat Penelitian.....	6
BAB II STUDI PUSTAKA	7
2.1 Penelitian Terdahulu.....	7
2.2 Teknologi <i>Wearable</i>	10
2.2.1 Teknologi <i>Wearable</i> dalam Pemantauan Kesehatan.....	10
2.2.2 Integrasi IoT dalam Pemantauan Kesehatan	11
2.2.3 Tantangan dalam Pemrosesan dan Analisis Data Kesehatan.....	11
2.3 <i>Gated Recurrent Unit</i> (GRU)	12
BAB III METODOLOGI PENELITIAN	14
3.1 Analisis dan Perancangan	14
3.2 Sumber Data	17
3.3 Rancangan Perhitungan	22
3.3.1 Penggabungan dan Penyelarasan Format Data	23
3.3.2 Pembersihan Data (<i>Data Cleaning</i>)	23
3.3.3 Standardisasi Data	24
3.3.4 Labeling Data	25
3.3.5 Pembagian Data (<i>Data Splitting</i>)	26
3.4 Perancangan Sistem	27
3.4.1 Arsitektur Sistem.....	27
3.4.2 Diagram Arsitektur Sistem.....	29
3.4.3 Komponen Sistem	31
3.4.4 Alur Proses Sistem	32
3.4.5 Target Performa Sistem dan Model	33
3.5 Implementasi Model GRU.....	34
3.5.1 Arsitektur Model	35
3.5.2 Parameter Pelatihan.....	36

3.5.3	Proses Pelatihan Model	37
3.5.4	Integrasi Model ke Sistem.....	37
3.5.5	Diagram Alur Implementasi GRU	38
3.6	Integrasi Sistem IoT dan Aplikasi Android	39
3.6.1	Struktur Umum Sistem IoT	39
3.6.2	Alur Kerja Sistem.....	41
3.6.3	Komunikasi Antar Komponen	43
3.6.4	Antarmuka Pengguna (<i>User Interface</i>)	44
3.7	Rencana Pengujian Sistem	45
3.7.1	Tujuan Pengujian	45
3.7.2	Jenis Pengujian.....	46
3.7.3	Evaluasi dan Analisis Hasil.....	49
3.7.4	Pengujian Usability Menggunakan SUS (<i>System Usability Scale</i>).....	50
BAB IV HASIL DAN PEMBAHASAN		52
4.1	Implementasi Sistem.....	52
4.1.1	Perangkat Keras yang Digunakan	52
4.1.2	Perangkat Lunak yang Digunakan	53
4.1.3	Implementasi Database	54
4.1.4	Implementasi Data	55
4.1.5	Implementasi Integrasi IoT dan Protokol Komunikasi Data	57
4.1.6	Implementasi Model GRU	62
4.1.7	Implementasi Antarmuka	77
4.1.8	Hasil Uji Coba.....	81
4.1.9	Kendala Implementasi Sistem.....	88
4.2	Pembahasan	92
4.3	Integrasi Penelitian dengan Islam.....	97
BAB V KESIMPULAN DAN SARAN		102
5.1	Kesimpulan.....	102
5.2	Saran	103
DAFTAR PUSTAKA		

DAFTAR GAMBAR

Gambar 3.1 Diagram Arsitektur Sistem.....	17
Gambar 3.2 Diagram Arsitektur Sistem Pemantauan Kesehatan Berkelanjutan ..	29
Gambar 3.3 Diagram Alur Implementasi GRU	38
Gambar 3.4 Diagram Alur Sistem IoT	42
Gambar 3.5 Skema Pengujian Pengguna terhadap Aplikasi Android.....	48
Gambar 4.1 Struktur <i>Cloud Firestore</i>	54
Gambar 4.2 Grafik <i>Accuracy</i> dan <i>Loss</i>	76
Gambar 4.3 <i>Confusion Matrix</i>	77
Gambar 4.4 Tampilan <i>Smartwatch</i>	78
Gambar 4.5 Halaman <i>Dashboard</i>	79
Gambar 4.6 Halaman <i>Activity</i>	80
Gambar 4.7 Halaman Notifikasi	81

DAFTAR TABEL

Tabel 2.1 Penelitian terkait	7
Tabel 3.1 Label kategori kondisi.....	36
Tabel 3.2 Parameter Pelatihan.....	36
Tabel 3.3 Pengujian Fungsional.....	46
Tabel 3.4 Kategori skor SUS	51
Tabel 4. 1 Susunan field data raw pada firebase.....	59
Tabel 4. 2 Data yang dikirim dan tidak dikirim ke cloud	59
Tabel 4. 3 Fitur Input Model.....	63
Tabel 4.4 Parameter penelitian.....	64
Tabel 4. 5 Contoh data awal sebelum ekstraksi fitur	66
Tabel 4. 6 Konfigurasi Parameter Pelatihan	71
Tabel 4. 7 Hasil Evaluasi Model GRU.....	75
Tabel 4.8 Hasil Pengujian Fungsional	82
Tabel 4. 9 Informasi responden.....	84
Tabel 4.10 Hasil Perhitungan Skor SUS	85
Tabel 4.11 Analisis jawaban pertanyaan SUS	86
Tabel 4.12 Perbandingan penelitian dengan penelitian sebelumnya	94

ABSTRAK

Izzah, Kholilatul. 2026. **Implementasi Model Gated Recurrent Unit (GRU) untuk pemantauan kesehatan berkelanjutan** . Skripsi. Jurusan Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Prof. Dr. Muhammad Faisal, M.T (II) Hani Nurhayati, M.T.

Kata kunci: GRU, *smartwatch* Android, IoT, pemantauan kesehatan, WESAD, klasifikasi kondisi fisiologis.

Sebagian besar *smartwatch* yang tersedia saat ini hanya menyajikan data fisiologis dalam bentuk visualisasi sederhana tanpa analisis mendalam, sehingga belum mampu memberikan pemahaman komprehensif terhadap kondisi kesehatan pengguna. Penelitian ini mengembangkan sistem pemantauan kesehatan berkelanjutan berbasis *Internet of Things* (IoT) dengan mengimplementasikan model *Gated Recurrent Unit* (GRU) pada *smartwatch* Android menggunakan dataset WESAD (*Wearable Stress and Affect Detection*). Sistem mengadopsi arsitektur tiga lapis dengan pendekatan *on-device inference* berbasis TensorFlow Lite, memanfaatkan dua sensor bawaan *smartwatch* yaitu *heart rate* (HR) dan *accelerometer* (ACC). Sebanyak 13 fitur statistik diekstraksi menggunakan metode *sliding window* 60 detik dengan *overlap* ~92%, menghasilkan 879 *window* untuk tiga kelas kondisi: *Baseline*, *Stress*, dan *Amusement*. Hasil evaluasi menunjukkan model GRU mencapai akurasi 94%, *weighted average F1-score* 0,93, serta *F1-score* per kelas sebesar 0,96 (*Baseline*), 0,93 (*Stress*), dan 0,87 (*Amusement*). Pengujian *usability* menggunakan *System Usability Scale* (SUS) menghasilkan rata-rata skor 77,5 yang termasuk kategori *Good*. Penelitian ini menunjukkan bahwa kombinasi teknologi *wearable*, IoT, dan *deep learning* berbasis GRU berpotensi diterapkan sebagai solusi pemantauan kesehatan yang praktis pada perangkat *wearable* di kehidupan nyata.

ABSTRACT

Izzah, Kholilatul. 2026. **Implementation of the Gated Recurrent Unit (GRU) Model for Continuous Health Monitoring.** Thesis. Department of Informatics Engineering, Faculty of Science and Technology, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Supervisors: (I) Prof. Dr. Muhammad Faisal, M.T (II) Hani Nurhayati, M.T.

Most commercially available smartwatches only present physiological data through simple visualizations without in-depth analysis, limiting users' ability to comprehensively understand their health status. This research develops a continuous health monitoring system based on the Internet of Things (IoT) by implementing a Gated Recurrent Unit (GRU) model on an Android smartwatch using the WESAD (Wearable Stress and Affect Detection) dataset. The system adopts a three-layer IoT architecture with a TensorFlow Lite-based on-device inference approach, utilizing two built-in smartwatch sensors: heart rate (HR) and accelerometer (ACC). A total of 13 statistical features were extracted using a 60-second sliding window method with ~92% overlap, producing 879 windows across three condition classes: Baseline, Stress, and Amusement. Evaluation results show that the GRU model achieved an accuracy of 94%, a weighted average F1-score of 0.93, and per-class F1-scores of 0.96 (Baseline), 0.93 (Stress), and 0.87 (Amusement). Usability testing using the System Usability Scale (SUS) yielded an average score of 77.5, classified as Good. This research demonstrates that the combination of wearable technology, IoT, and GRU-based deep learning holds strong potential as a practical health monitoring solution on real-world wearable devices

Key words: GRU, Android smartwatch, IoT, health monitoring, WESAD, physiological condition classification.

مستخلص البحث

عزة، خليل. 2026. *تطبيق نموذج الوحدة المتكررة البوابية (GRU) لمراقبة الصحة المستمرة*. رسالة جامعية. قسم هندسة المعلوماتية، كلية العلوم والتكنولوجيا، جامعة الدولة الإسلامية مولانا مالك إبراهيم ماليايغ. المشرفون (I): الأستاذ الدكتور محمد فيصل، مهندس (II) هاني نورهايايغ، مهندس.

الكلمات المفتاحية: GRU ، ساعة ذكية أندرويد، إنترنت الأشياء، مراقبة الصحة، WESAD ، تصنيف الحالات الفسيولوجية.

معظم الساعات الذكية المتوفرة حاليًا تعرض البيانات الفسيولوجية بشكل بسيط دون تحليل معمق، لذلك لا تستطيع تقديم فهم شامل لحالة صحة المستخدم. تهدف هذه الدراسة إلى تطوير نظام مراقبة صحة مستمر قائم على الإنترنت للأشياء (IoT) من خلال تطبيق نموذج وحدة الغيتد المتكررة (GRU) على ساعات أندرويد الذكية باستخدام مجموعة بيانات WESAD (كشف التوتر والعاطفة للأجهزة القابلة للارتداء). يعتمد النظام على بنية ثلاثية الطبقات مع نهج الاستنتاج على الجهاز باستخدام TensorFlow Lite، ويستفيد من حساسين مدمجين في الساعة الذكية وهما معدل ضربات القلب (HR) ومقياس التسارع (ACC). تم استخراج 13 ميزة إحصائية باستخدام طريقة النافذة الانزلاقية لمدة 60 ثانية مع تداخل حوالي 92٪، مما نتج عنه 879 نافذة لثلاث فئات من الحالات: الحالة الأساسية، والتوتر، والمتعة أظهرت نتائج التقييم أن نموذج GRU حقق دقة 94٪، ومتوسط F1-الموزون 0.93، و F1 لكل فئة على النحو التالي 0.96 (الخط الأساسي)، 0.93 (الضغط النفسي)، و 0.87 (المرح). وأظهر اختبار قابلية الاستخدام باستخدام مقياس قابلية الاستخدام للنظام (SUS) متوسط درجة 77.5 والذي يندرج ضمن فئة جيد. تُظهر هذه الدراسة أن الجمع بين تقنيات الأجهزة القابلة للارتداء، وإنترنت الأشياء، والتعلم العميق القائم على GRU لديه القدرة على أن يُطبق كحل لمراقبة الصحة بشكل عملي على الأجهزة القابلة للارتداء في الحياة اليومية.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Kemampuan manusia dalam menjalani aktivitas sehari-hari dan berkontribusi secara produktif sangat bergantung pada kondisi kesehatannya. Di tengah pesatnya transformasi digital, sektor kesehatan turut mengalami perubahan signifikan dengan hadirnya perangkat wearable, khususnya smartwatch, sebagai solusi pemantauan kesehatan personal (Żarnowski et al., 2022). Melalui perangkat ini, berbagai parameter fisiologis — mulai dari detak jantung, jumlah langkah, kualitas tidur, aktivitas fisik, hingga saturasi oksigen — dapat dipantau secara real-time tanpa gangguan aktivitas pengguna.

Popularitas *smartwatch* juga terus meningkat seiring kesadaran masyarakat terhadap pemantauan kesehatan mandiri. Berdasarkan data Statista (2024), jumlah pengguna *smartwatch* di Indonesia mencapai 16,9 juta pada tahun 2024 dan diproyeksikan bertambah menjadi 26,76 juta pada tahun 2029. Peningkatan tersebut menunjukkan peluang besar pengembangan sistem pemantauan kesehatan berbasis perangkat *wearable*. Meskipun demikian, sebagian besar informasi dari *smartwatch* masih ditampilkan dalam bentuk grafik sederhana tanpa analisis mendalam sehingga belum membantu pengguna memahami pola kesehatan secara komprehensif (Smuck et al., 2021).

Permasalahan utama dalam pemanfaatan data kesehatan dari smartwatch terletak pada karakteristik data yang bersifat temporal dan berubah dari waktu ke waktu (Hughes et al., 2023). Data seperti ini memerlukan model analisis yang

mampu menangkap pola berurutan dan dinamika fisiologis pengguna. Salah satu pendekatan yang sesuai adalah *Gated Recurrent Unit* (GRU), varian dari *Recurrent Neural Network* (RNN) yang dirancang untuk menangani permasalahan *vanishing gradient*, dengan struktur yang lebih kompak dan komputasi yang lebih ringan dibandingkan *Long Short-Term Memory* (LSTM) (Dewi et al., 2023; Sun et al., 2024; Li et al., 2022).

Berbagai penelitian menunjukkan efektivitas GRU dalam pemrosesan data kesehatan. Tan et al., (2022; 2023) memanfaatkannya untuk pengenalan aktivitas, Liyakathunisa et al., (2022) mengimplementasikannya pada sistem perawatan lansia berbasis IoMT, sementara El Hajj & Kyriacou, (2021) serta Rizal & Yuniarti, (2023) menggunakannya untuk estimasi tekanan darah berbasis sinyal PPG. Selain itu, beberapa penelitian juga telah menggunakan dataset WESAD (*Wearable Stress and Affect Detection*) sebagai basis pengembangan model deteksi stres. Bajpai et al. (2023) menggunakan LSTM pada dataset WESAD dan memperoleh akurasi sebesar 98%, namun menggunakan seluruh modalitas sensor yang tidak tersedia pada *smartwatch* umum. Tiwari et al. (2024) menggunakan *Dilated CNN* berbasis sinyal PPG khusus, sementara Melchiades et al. (2022) dan Sah et al. (2021) menggunakan RNN dan CNN dengan pendekatan multi-sensor. Meskipun hasil yang diperoleh cukup baik, penelitian-penelitian tersebut belum mengintegrasikan sistem secara langsung pada *smartwatch* Android berbasis IoT.

Sebagian besar penelitian sebelumnya masih terbatas pada satu parameter fisiologis, menggunakan sensor khusus, atau belum menerapkan pemantauan secara langsung berbasis IoT secara *end-to-end*. Kesenjangan ini membuka ruang bagi

penelitian dalam pengembangan sistem yang mampu mengintegrasikan data dual-sensor *smartwatch* Android dan memprosesnya secara langsung menggunakan GRU. Penelitian ini hadir untuk mengisi celah tersebut melalui perancangan sistem pemantauan kesehatan yang adaptif dan berkelanjutan bagi pengguna umum.

Dari perspektif Islam, menjaga kesehatan merupakan bentuk ikhtiar yang dianjurkan. Allah SWT berfirman dalam Surah Al-Baqarah ayat 195:

وَأَنْفِقُوا فِي سَبِيلِ اللَّهِ وَلَا تُلْقُوا بِأَيْدِيكُمْ إِلَى التَّهْلُكَةِ وَأَحْسِنُوا إِنَّ اللَّهَ يُحِبُّ الْمُحْسِنِينَ ﴿١٩٥﴾

“Berinfaklah di jalan Allah, janganlah jerumuskan dirimu ke dalam kebinasaan, dan berbuatbaiklah. Sesungguhnya Allah menyukai orang-orang yang berbuat baik.” (QS. Al-Baqarah: 195).

Ayat di atas memperkuat pentingnya menjaga diri dari hal yang membahayakan (Departemen Agama Republik Indonesia, 2019). Di samping itu, pada Surah Asy-Syu'ara ayat 80, Allah SWT berfirman:

وَإِذَا مَرِضْتُ فَهُوَ يَشْفِينِ ﴿٨٠﴾

“Dan apabila aku sakit, Dialah (Allah) yang menyembuhkanku” (QS. Asy-Syu'ara: 80)

Tafsir menjelaskan bahwa manusia wajib berikhtiar menjaga kesehatan, sedangkan kesembuhan tetap berada dalam kekuasaan Allah {Formatting Citation} .

Dua ayat tersebut menegaskan bahwa menjaga kesehatan adalah tanggung jawab manusia, sedangkan kesembuhan adalah kehendak Allah. Pemanfaatan teknologi seperti *smartwatch* dan IoT dapat dipandang sebagai salah satu bentuk ikhtiar manusia dalam menjaga kesehatan dan mencegah penyakit.

Penelitian Nurhayati & Fitriyana, (2020) juga mengungkapkan bahwa menjaga kesehatan merupakan wujud syukur atas nikmat kehidupan sekaligus bentuk ibadah, sebab dengan kondisi tubuh yang prima memungkinkan seseorang untuk menjalankan berbagai aktivitas kebaikan secara maksimal.

Bertolak dari uraian tersebut, penelitian ini berupaya mengembangkan pada perancangan dan pengembangan sistem kesehatan berbasis IoT dan model GRU yang mampu menganalisis data dual-sensor secara berkelanjutan guna memberikan rekomendasi kesehatan adaptif bagi pengguna.

1.2 Rumusan Masalah

Mengacu pada latar belakang yang telah dijabarkan di atas, penelitian ini merumuskan pokok permasalahan sebagai berikut: “Bagaimana mengimplementasikan model GRU serta menilai performanya untuk memprediksi dan menganalisis parameter kesehatan berbasis data dual-sensor *smartwatch* Android dalam sistem pemantauan berbasis IoT?”

1.3 Batasan Masalah

Guna menjaga fokus dan ketercapaian penelitian, perlu ditetapkan beberapa batasan masalah sebagai berikut:

1. Perangkat yang digunakan terbatas pada *smartwatch* berbasis Android dengan sensor bawaan seperti detak jantung dan akselerometer.
2. Jenis data yang digunakan adalah data kesehatan *time-series*, yaitu data yang memiliki keterurutan waktu. Data mencakup *heart rate* dan *accelerometer* yang tersedia pada *Smartwatch* Android.

3. Model pembelajaran yang digunakan terbatas pada GRU tanpa membandingkan metode lain.
4. Sistem yang dikembangkan bukan alat diagnosis medis, sehingga hasil prediksi hanya berupa informasi pendukung kesehatan dan tidak dapat menggantikan pemeriksaan tenaga medis profesional.
5. Integrasi sistem terbatas pada penggunaan arsitektur IoT sederhana, yaitu koneksi antara *smartwatch* dengan aplikasi Android dan server. Tidak mencakup arsitektur IoT skala industri.
6. Evaluasi model difokuskan pada akurasi prediksi dan efisiensi pemrosesan.

1.4 Tujuan Penelitian

Berdasar pada rumusan masalah yang telah ditetapkan di atas, sehingga arah penelitian menjadi jelas dan terukur. Di bawah ini merupakan tujuan penelitian adalah sebagai berikut:

1. Mengembangkan arsitektur sistem pemantauan kesehatan berbasis IoT yang memanfaatkan *smartwatch* Android sebagai sumber data kesehatan *time-series*.
2. Menerapkan model *Gated Recurrent Unit* (GRU) untuk menganalisis pola data dual-sensor dari *smartwatch*, sehingga mampu mengenali perubahan kondisi fisiologis pengguna.
3. Mengintegrasikan model GRU ke dalam aplikasi Android, sehingga hasil analisis dan prediksi dapat ditampilkan secara langsung kepada pengguna dalam bentuk informasi kesehatan yang mudah dipahami.
4. Melakukan evaluasi terhadap performa model GRU dan kinerja sistem, meliputi akurasi prediksi serta stabilitas proses pengiriman data.

1.5 Manfaat Penelitian

Dari sisi teoretis, penelitian ini memberikan kontribusi dalam memperluas khasanah literatur ilmiah mengenai pemrosesan data kesehatan temporal menggunakan model *Gated Recurrent Unit* (GRU). Selain itu, penelitian ini memperkuat landasan kajian terkait pengembangan sistem untuk pemantauan kesehatan berbasis IoT menggunakan data dual-sensor perangkat *wearable* seperti *smartwatch* Android, pengembangan sistem pemantauan kesehatan berbasis IoT dengan arsitektur tiga lapis pada ekosistem Android/Wear OS; dan penerapan *on-device inference* menggunakan TFLite sebagai pendekatan efisien untuk klasifikasi kondisi fisiologis secara *real-time* pada perangkat berkapasitas terbatas.

Secara praktis, penelitian ini menghadirkan sistem yang dapat dimanfaatkan masyarakat luas untuk memantau kondisi tubuh secara mandiri dan berkesinambungan. Sistem ini dirancang untuk menghasilkan rekomendasi kesehatan yang adaptif berdasarkan pola data fisiologis pengguna, sehingga diharapkan mampu meningkatkan kesadaran kesehatan, mendukung pemantauan rutin sehari-hari, serta berperan dalam deteksi dini potensi gangguan kesehatan. Di samping itu, temuan penelitian ini dapat menjadi rujukan bagi para pengembang aplikasi kesehatan, tenaga medis, maupun institusi pendidikan dalam merancang solusi kesehatan digital berbasis *smartwatch* dan IoT.

BAB II

STUDI PUSTAKA

2.1 Penelitian Terdahulu

Pada bagian ini membahas penelitian terkait pemanfaatan *Gated Recurrent Unit* (GRU) dalam analisis data kesehatan berbasis sensor dan IoT. Secara umum, penelitian terdahulu memperlihatkan keunggulan GRU dalam mengolah data sekuensial, termasuk sinyal fisiologis dan rekaman aktivitas pengguna. Namun, sebagian besar penelitian masih memiliki keterbatasan seperti hanya mengolah satu parameter kesehatan, menggunakan perangkat atau sensor khusus, atau belum menerapkan arsitektur IoT secara penuh.

Untuk memberikan gambaran posisi penelitian ini, beberapa penelitian relevan dirangkum pada tabel berikut.

Tabel 2.1 Penelitian terkait

No.	Peneliti	Judul	Persamaan	Perbedaan/ Kelemahan	Solusi/ Kebaruan
1.	Liyakathunisa et al. (2022)	<i>Ambient Assisted Living Framework for Elderly Care Using Internet of Medical Things, Smart Sensors, and GRU Deep Learning Techniques</i>	Menggunakan GRU untuk pemantauan tanda vital berbasis IoMT.	Fokus pada lansia; sensor terbatas; tidak menggunakan <i>smartwatch</i> Android.	Sistem untuk pengguna umum dengan dual-sensor <i>smartwatch</i> Android dan integrasi IoT.

2.	Rizal & Yuniarti (2023)	<i>GRU-Based Fusion Models for Enhanced Non-Invasive Blood Pressure Estimation from PPG Signals</i>	Menggunakan GRU untuk estimasi parameter fisiologis.	Parameter tunggal; tidak <i>real-time</i> ; tidak dual-sensor; tidak memanfaatkan <i>smartwatch</i> .	Prediksi dual-parameter, dengan data dual-sensor <i>smartwatch</i> Android.
3.	El Hajj & Kyriacou (2021)	<i>RNN Models for Blood Pressure Monitoring Using PPG Morphological Features</i>	Memanfaatkan GRU untuk estimasi tekanan darah berbasis PPG.	Membutuhkan sensor PPG khusus; tidak dual-sensor; tidak memakai perangkat umum.	Memanfaatkan sensor bawaan <i>smartwatch</i> tanpa alat tambahan.
4.	Tan et al. (2023)	<i>Using a Hybrid Neural Network and a Regularized Extreme Learning Machine for Human Activity Recognition with Smartphone and Smartwatch</i>	Menggunakan CNN-GRU pada data <i>smartphone</i> dan <i>smartwatch</i> .	Fokus pada pengenalan aktivitas (HAR); bukan analisis kesehatan; tidak berbasis IoT.	GRU digunakan untuk analisis kesehatan dual-parameter berbasis IoT.
5.	Tan et al. (2022)	<i>Human Activity Recognition Using an Ensemble Learning Algorithm with Smartphone Sensor Data</i>	Memakai GRU untuk klasifikasi aktivitas manusia.	Tidak memantau kesehatan; tidak IoT; hanya HAR.	Sistem pemantauan kesehatan berkelanjutan berbasis dual-sensor <i>smartwatch</i> Android.
6.	Bajpai et al. (2023)	<i>Stress Detection Using LSTM on WESAD Dataset</i>	Menggunakan dataset WESAD untuk deteksi stres berbasis data fisiologis.	Menggunakan seluruh modalitas sensor (EDA, ECG, EMG, ACC, Resp); tidak diimplementasikan pada <i>smartwatch</i> Android; tidak berbasis IoT.	Penelitian ini hanya menggunakan HR dan ACC yang tersedia langsung pada <i>smartwatch</i> Android tanpa sensor tambahan.

7.	Tiwari et al. (2024)	<i>Stress Detection Using Dilated CNN on PPG Signal from WESAD</i>	Menggunakan dataset WESAD untuk klasifikasi kondisi fisiologis.	Bergantung pada sensor PPG khusus; tidak menggunakan <i>smartwatch</i> Android; tidak berbasis IoT.	Memanfaatkan sensor bawaan <i>smartwatch</i> tanpa memerlukan sensor khusus tambahan
8.	Brihadiswaran et al. (2023)	<i>Wrist-Based Stress Detection Using Federated Learning on WESAD</i>	Menggunakan dataset WESAD dengan data pergelangan tangan (<i>wrist-based</i>).	Fokus pada privasi data menggunakan <i>Federated Learning</i> ; tidak diintegrasikan dengan aplikasi Android; tidak berbasis IoT.	Sistem terintegrasi penuh dengan aplikasi Android dan <i>cloud</i> berbasis IoT untuk pemantauan berkelanjutan.
9.	Melchiades et al. (2022)	<i>Stress Classification Using RNN on WESAD Dataset</i>	Menggunakan dataset WESAD dan model sekuensial (RNN) untuk klasifikasi kondisi fisiologis.	Menggunakan RNN standar yang lebih rentan terhadap <i>vanishing gradient</i> ; tidak diimplementasikan pada perangkat <i>wearable</i> nyata.	Menggunakan GRU yang lebih efisien dan stabil dibandingkan RNN, diintegrasikan langsung pada <i>smartwatch</i> Android.
10.	Sah et al. (2021)	<i>Classification of Mental Stress from Wearable Physiological Sensors Using Image-Encoding Based Deep Neural Network</i>	Menggunakan dataset WESAD untuk klasifikasi kondisi mental berbasis sensor <i>wearable</i> .	Menggunakan pendekatan <i>image-encoding</i> yang memerlukan komputasi lebih tinggi; tidak diimplementasikan secara <i>real-time</i> pada <i>smartwatch</i> .	Model GRU lebih ringan dan cocok untuk <i>on-device inference</i> pada <i>smartwatch</i> Android secara <i>real-time</i> .

Berdasarkan tabel 2.1 tersebut, bisa disimpulkan bahwa meskipun GRU sudah banyak digunakan dalam pengolahan sinyal fisiologis dan aktivitas, sebagian besar penelitian masih memiliki keterbatasan ruang lingkup. Banyak studi berfokus pada parameter tunggal, menggunakan sensor khusus yang tidak praktis, atau tidak menerapkan sistem pemantauan secara langsung berbasis IoT. Selain itu,

pemanfaatan *smartwatch* Android sebagai perangkat multi-sensor yang mudah diakses masyarakat masih jarang dilakukan.

Pada penelitian ini datang untuk mengisi *gap* tersebut dengan mengembangkan sistem pemantauan kesehatan berbasis IoT yang mengintegrasikan berbagai sensor *smartwatch* Android dan memprosesnya menggunakan GRU untuk menghasilkan analisis dan rekomendasi kesehatan adaptif.

2.2 Teknologi *Wearable*

2.2.1 Teknologi *Wearable* dalam Pemantauan Kesehatan

Teknologi *wearable*, khususnya *smartwatch*, telah berkembang menjadi perangkat yang dapat merekam berbagai indikator fisiologis secara *real-time* seperti detak jantung, aktivitas fisik, kualitas tidur, dan suhu tubuh. Kemampuan tersebut menjadikan *smartwatch* sebagai alat pemantauan kesehatan yang praktis dan mudah digunakan dalam kehidupan sehari-hari.

Perkembangan *Internet of Things* (IoT) semakin memperkuat fungsi *smartwatch* melalui kemampuannya mengirim data kesehatan ke server atau *cloud* untuk dianalisis lebih lanjut (Ahmadzadeh et al., 2022). Data yang dikumpulkan secara terus-menerus memberikan gambaran yang lebih akurat mengenai kondisi tubuh pengguna, namun juga memerlukan teknik analisis yang mampu mengolah data time series secara efektif.

Karakteristik data dari perangkat *wearable* yang tidak terstruktur serta dipengaruhi aktivitas pengguna menjadikan model GRU relevan untuk digunakan, karena model ini dirancang untuk memproses data berurutan secara efisien.

2.2.2 Integrasi IoT dalam Pemantauan Kesehatan

Integrasi IoT dalam *smartwatch* memungkinkan proses pengumpulan, pengiriman, dan sinkronisasi data kesehatan secara *real-time*. Dengan arsitektur IoT, data yang diperoleh dapat dianalisis secara berkelanjutan sehingga memberikan informasi kesehatan yang lebih komprehensif, termasuk pemantauan jarak jauh dan penyimpanan data jangka panjang (Liyakathunisa et al., 2022).

Meskipun menawarkan banyak manfaat, integrasi IoT juga memunculkan tantangan seperti kebutuhan manajemen *big data* dan analisis cepat. GRU menjadi salah satu model yang sesuai karena mampu mengolah data temporal dan memberikan prediksi berdasarkan pola historis pengguna. Dalam konteks penelitian ini, IoT berfungsi sebagai penghubung antara *smartwatch* Android dan sistem analisis pada server sehingga seluruh proses berlangsung otomatis dan berkelanjutan.

2.2.3 Tantangan dalam Pemrosesan dan Analisis Data Kesehatan

Pemrosesan data kesehatan dari perangkat *wearable* menghadapi berbagai tantangan, seperti:

- a. adanya *noise* pada sensor,
- b. variabilitas aktivitas harian pengguna,
- c. perubahan kondisi fisik yang dinamis,
- d. ketergantungan waktu yang kuat.

Data yang kompleks ini tidak dapat dianalisis secara manual. Oleh karena itu, dibutuhkan model yang mampu menangkap ketergantungan temporal secara

efektif. GRU terbukti mampu memproses data berurutan dengan efisien dan menghasilkan prediksi yang akurat (Zhang, 2022). Dengan memanfaatkan pola historis, model ini dapat memberikan rekomendasi kesehatan personal dan membantu pengguna memahami dinamika kondisi tubuhnya.

2.3 Gated Recurrent Unit (GRU)

Gated Recurrent Unit (GRU) merupakan salah satu pengembangan dari arsitektur *Recurrent Neural Network* (RNN) yang dikembangkan oleh Cho et al. (2014) untuk mengatasi masalah *vanishing gradient* dalam RNN tradisional (Lu et al., 2020). GRU dikendalikan oleh dua gerbang utama, yaitu *update gate* dan *reset gate*, yang berfungsi mengatur aliran informasi antara data lama dan data baru.

Dibandingkan LSTM, GRU memiliki jumlah parameter lebih sedikit sehingga lebih efisien dan cepat dilatih. Hal ini menjadikan GRU sangat cocok untuk aplikasi seperti pemantauan kesehatan berbasis IoT.

Rumus dasar GRU adalah sebagai berikut:

$$z_t = \sigma(W_z \cdot [h_{t-1}, x_t] + b_z) \quad (2.1)$$

$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t] + b_r) \quad (2.2)$$

$$\tilde{h}_t = \tanh(W_h \cdot [r_t * h_{t-1}, x_t] + b_h) \quad (2.3)$$

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t \quad (2.4)$$

Keterangan:

- a. h_{t-1} : output pada waktu sebelumnya,
- b. x_t : input saat ini
- c. W_z, W_r , dan W_h : matriks bobot yang dipelajari selama pelatihan
- d. b_z, b_r, b_h : vektor bias
- e. σ : fungsi aktivasi sigmoid

Keunggulan GRU meliputi efisiensi komputasi, stabilitas selama proses pelatihan, serta kemampuan mengatasi *vanishing gradient*. Dengan demikian, GRU

merupakan salah satu metode paling efektif untuk analisis deret waktu dalam sistem pemantauan kesehatan yang membutuhkan pemrosesan cepat dan akurat.

BAB III

METODOLOGI PENELITIAN

Bab ini menjelaskan metode dan tahapan yang digunakan dalam mengembangkan sistem pemantauan kesehatan berkelanjutan berbasis *Internet of Things* (IoT) dengan model *Gated Recurrent Unit* (GRU). Metodologi penelitian disusun secara sistematis, mencakup perancangan arsitektur sistem, pengumpulan data kesehatan dari *smartwatch* Android, pengolahan data, pelatihan model GRU, serta integrasi model ke dalam aplikasi. Pada bagian akhir juga dijelaskan rencana pengujian untuk mengevaluasi kinerja sistem.

3.1 Analisis dan Perancangan

Desain sistem dalam penelitian ini bertujuan untuk menggambarkan alur kerja dan interaksi antara komponen-komponen utama pada sistem pemantauan kesehatan berkelanjutan berbasis *Internet of Things* (IoT) dengan penerapan model *Gated Recurrent Unit* (GRU). Sistem ini dikembangkan agar dapat melakukan pengambilan data fisiologis pengguna secara otomatis melalui *smartwatch* Android, kemudian mengirimkan data tersebut untuk dianalisis oleh model GRU yang telah dilatih sebelumnya.

Sistem yang dikembangkan terdiri atas beberapa lapisan utama, yaitu lapisan perangkat keras (*hardware layer*), lapisan perangkat lunak (*software layer*), dan lapisan pemrosesan data (*data processing layer*).

1. Lapisan Perangkat Keras (*Hardware Layer*)

Lapisan ini terdiri dari *smartwatch* Android yang telah terintegrasi dengan sensor-sensor kesehatan seperti:

- a. *Heart rate sensor*
- b. *Accelerometer*

Sensor ini digunakan untuk memperoleh data detak jantung dan percepatan gerak pada tiga sumbu (X, Y dan Z). Pengambilan data dilakukan menggunakan Wear OS Data Layer API dengan metode *MessageClient*, yang menyediakan akses *real-time* terhadap sensor dengan interval tertentu, pada penelitian ini menggunakan 5Hz. Semua proses mengikuti persetujuan dari pengguna.

2. Lapisan Perangkat Lunak (*Software Layer*)

Lapisan ini mencakup aplikasi *smartwatch* Android yang berfungsi sebagai:

- a. antarmuka pengguna untuk menampilkan informasi kesehatan secara visual,
- b. pengambilan dan pengiriman data sensor dari *smartwatch* ke aplikasi Android melalui Wear OS Data Layer API dengan metode *MessageClient*,
- c. pemrosesan awal data fisiologis yang diterima dari *smartwatch*,
- d. analisis data menggunakan model *machine learning Gated Recurrent Unit* (GRU) yang telah diintegrasikan pada aplikasi Android,
- e. penampil hasil analisis GRU dalam bentuk status kesehatan, dan rekomendasi kesehatan bagi pengguna.

Dengan arsitektur ini, proses analisis dapat dilakukan langsung pada perangkat Android (*on-device processing*), sehingga sistem mampu

memberikan respons secara lebih cepat tanpa bergantung sepenuhnya pada koneksi server.

3. Lapisan Pemrosesan Data (*Data Processing Layer*)

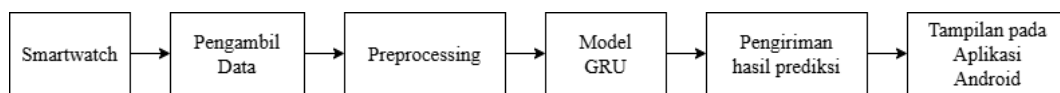
Lapisan pemrosesan data bertanggung jawab terhadap proses analisis data fisiologis yang diperoleh dari *smartwatch*. Proses ini dilakukan langsung pada aplikasi Android sebelum data disimpan ke *cloud*. Tahapan pemrosesan data meliputi:

- a. *preprocessing* data untuk membersihkan dan menyiapkan data sensor,
- b. standarisasi data agar berada pada skala yang seragam,
- c. ekstraksi fitur dari sinyal fisiologis, seperti fitur statistik detak jantung dan aktivitas gerak,
- d. pemrosesan data menggunakan model *Gated Recurrent Unit* (GRU) untuk mempelajari pola temporal dari data fisiologis.

Model GRU kemudian menghasilkan prediksi kondisi kesehatan pengguna, seperti relaksasi, stres ringan, atau aktivitas tinggi. Hasil prediksi tersebut selanjutnya ditampilkan pada aplikasi Android dalam bentuk visualisasi, seperti grafik tren kesehatan, status kondisi tubuh, serta notifikasi atau rekomendasi kesehatan yang adaptif bagi pengguna. Selain itu, data fisiologis dan hasil analisis juga disimpan pada layanan *Firebase Cloud Firestore* sebagai riwayat pemantauan kesehatan.

Alur kerja keseluruhan sistem ditunjukkan pada Gambar 3.1, yang menunjukkan interaksi antara komponen utama, yaitu perangkat *smartwatch* Android sebagai sumber data, *Wear OS Data Layer API* sebagai penyedia data

sensor, aplikasi pengguna sebagai pusat pemrosesan data menggunakan model GRU serta sebagai media visualisasi hasil analisis, dan *server cloud* sebagai tempat menyimpan data hasil analisis.



Gambar 3.1 Diagram Arsitektur Sistem

Dengan rancangan sistem yang terdiri dari tiga lapisan utama tersebut, diharapkan proses akuisisi, pengiriman, dan analisis data kesehatan dapat dilakukan secara otomatis dan berkelanjutan. Model GRU akan berperan penting dalam mengolah data berurutan dari sensor untuk menghasilkan analisis dan prediksi kondisi kesehatan pengguna. Tahapan selanjutnya dalam penelitian ini mencakup implementasi sistem, pelatihan model, serta evaluasi kinerja sistem berdasarkan akurasi dan efisiensi.

3.2 Sumber Data

Dua jenis sumber data digunakan pada penelitian ini, yaitu data primer dan data sekunder, yang keduanya berperan penting pada proses pengembangan dan evaluasi model *Gated Recurrent Unit (GRU)* untuk pemantauan kesehatan berkelanjutan.

1. Data Sekunder

Penggunaan data sekunder pada penelitian ini bersumber dari dataset publik atau dataset penelitian terdahulu terkait pemantauan kesehatan berbasis sensor. Dataset ini digunakan untuk pelatihan awal (*pretraining*) model GRU sebelum diadaptasikan dengan data dari pengguna *smartwatch*.

Penggunaan data sekunder dalam penelitian ini berasal dari WESAD Dataset (*Wearable Stress and Affect Detection Dataset*) yang dikembangkan oleh *Universität Siegen, Germany*, dan dapat diakses melalui tautan <https://uni-siegen.sciebo.de/s/HGdUkoNIW1Ub0Gx>. Dataset ini memuat data fisiologis dari 15 partisipan yang dikumpulkan dengan menggunakan dua perangkat *wearable*, yaitu Empatica E4 Wristband dan RespiBAN.

Pemilihan WESAD didasarkan pada kelengkapan parameter fisiologis serta kesesuaian karakteristiknya dengan data kesehatan dari perangkat *wearable* modern. Dataset ini digunakan untuk memperkuat proses pelatihan awal model GRU, sekaligus menjadi acuan pembandingan terhadap performa model yang diterapkan pada data nyata pengguna *smartwatch*.

2. Data Primer

Di samping data sekunder, pada penelitian ini juga memanfaatkan data primer yang didapat langsung dari pengguna *smartwatch* Android menggunakan aplikasi yang dikembangkan peneliti. Data primer ini tidak dipergunakan pada proses pelatihan model, melainkan digunakan sebagai data input untuk pengujian dan implementasi model GRU yang telah dilatih menggunakan dataset WESAD. Data yang dikumpulkan merupakan data sensor *real-time* yang secara umum tersedia pada perangkat *wearable*, diantaranya:

- a. Detak jantung (*Heart Rate*): merepresentasikan kondisi kardiovaskular pengguna.
- b. Sensor percepatan (*Accelerometer*): mengukur percepatan atau perubahan gerakan suatu objek pada tiga sumbu, yaitu sumbu x,y dan z.

Setiap parameter direkam dengan interval tertentu yaitu setiap 60 detik untuk detak jantung dan percepatan agar menghasilkan data berurutan yang sesuai untuk pelatihan model GRU. Data tersebut dikumpulkan secara kontinu dengan kecepatan 5Hz, disimpan dalam penyimpanan lokal perangkat dan dilakukan tahapan *preprocessing*. Seluruh proses pengambilan data diikuti oleh persetujuan pengguna sebelum data digunakan untuk penelitian.

3.2.1 Rencana Pengambilan Data Primer

Pengambilan data primer dilakukan dengan memanfaatkan sensor *real-time* pada perangkat *smartwatch* Android. Data fisiologis yang direkam berupa detak jantung (*heart rate*) dan akselerometer (*accelerometer*) dalam konteks penggunaan sehari-hari (*free-living*). Pendekatan ini dipilih agar data yang terkumpul merepresentasikan kondisi nyata pengguna, bukan sekadar hasil eksperimen terstruktur.

1. Partisipan Penelitian

Untuk memperoleh keragaman pola fisiologis dan meningkatkan generalisasi model, partisipan dibagi menjadi dua kelompok usia:

- a. Kelompok A (15–25 tahun)
- b. Kelompok B (26–50 tahun)

Jumlah partisipan berjumlah 6 orang, dengan 3 orang per kelompok. Seluruh partisipan memenuhi kriteria berikut:

- a) tidak memiliki riwayat penyakit jantung serius,
- b) tidak sedang mengonsumsi obat yang memengaruhi detak jantung,
- c) bersedia memakai *smartwatch* selama periode pengambilan data.

2. Metode Penggunaan *Smartwatch* (*Free-Living Monitoring*)

Pengumpulan data dilakukan secara alami tanpa skenario laboratorium. Setiap partisipan diminta memakai *smartwatch* selama 3–5 hari, dengan durasi minimal 4–8 jam per hari pada aktivitas harian normal, seperti:

- a. duduk, belajar, bekerja, atau istirahat,
- b. berjalan santai, berjalan cepat, atau naik tangga,
- c. aktivitas sosial seperti berbicara, tertawa, atau berkumpul,
- d. aktivitas ringan lainnya.

Penggunaan metode *free-living* memungkinkan data yang diperoleh mencerminkan variabilitas fisiologis yang terjadi secara natural pada berbagai kondisi fisik dan emosional.

3. Kategori Aktivitas Pengguna

Karena pengumpulan data dilakukan menggunakan pendekatan *free-living monitoring*, aktivitas pengguna tidak ditentukan melalui skenario eksperimen tertentu. Namun demikian, data fisiologis yang terekam selama penggunaan harian dapat diklasifikasikan secara otomatis ke dalam tiga kategori utama, yaitu:

- a) *Relax* (kondisi rileks / *baseline*)
 - a. partisipan duduk atau istirahat,
 - b. pergerakan sangat rendah,
 - c. detak jantung mendekati *baseline* harian.
- b) *Stress*
 - a. kondisi mental tertekan ringan, fokus tinggi, atau beban tugas,

- b. atau kondisi fisik yang memicu peningkatan detak jantung tanpa pergerakan signifikan,
- c. misalnya berjalan cepat singkat, tegang karena tugas, terburu-buru, dan sejenisnya.

c) *Amusement*

- a. kondisi emosional positif, seperti tertawa atau antusias,
- b. peningkatan fisiologis bersamaan dengan aktivitas tubuh,
- c. disertai pola akselerometer yang lebih aktif.

Klasifikasi ini tidak ditetapkan oleh partisipan, melainkan ditentukan oleh *pipeline* pemrosesan data.

4. Pelabelan Otomatis (*Automatic Annotation*)

Untuk menghasilkan label *baseline*, *stress* ringan, dan *amusement* tanpa intervensi partisipan, digunakan *pipeline* otomatis berikut:

a) Akuisisi Data Mentah (*Raw Data Acquisition*)

- a. *heart rate*
- b. *accelerometer*

b) *Resampling* dan standardisasi

Semua data diseragamkan ke frekuensi 5 Hz dan dihaluskan.

c) Pendeteksian *baseline* fisiologis

Baseline HR dihitung dari 10% nilai HR terendah per individu.

d) Segmentasi berbasis window

Data dibagi menjadi jendela 60 detik dengan overlap ~92% 5 detik.

- e) Ekstraksi fitur per window, seperti:
 - a. mean HR, std HR, *slope* HR, RMSSD (HRV)
 - b. mean ACC *magnitude*, *acc energy*, *activity fraction*
- f) *Clustering* otomatis (KMeans, k=3) pada fitur window untuk menentukan tiga kelompok fisiologis alami.
- g) Mapping klaster → label, berdasarkan karakteristik centroid:
 - a. HR rendah + ACC rendah → *Baseline*
 - b. HR tinggi + ACC rendah → *Stress* ringan
 - c. HR tinggi + ACC tinggi → *Amusement*
- h) *Smoothing* temporal

Menggunakan median filter/HMM agar label tidak berubah terlalu cepat. Metode ini memungkinkan proses pelabelan data dilakukan secara otomatis tanpa intervensi langsung dari pengguna, dengan tetap mempertimbangkan karakteristik fisiologis yang umum pada data *wearable*.

5. Keamanan, Etika, dan Privasi Data

- a. Data hanya berisi sinyal sensor (HR & ACC), tanpa identitas pribadi.
- b. Setiap peserta diberi ID anonim.
- c. Data digunakan secara terbatas untuk penelitian dan pengembangan model.

3.3 Rancangan Perhitungan

Tahapan *preprocessing* data dilakukan untuk memastikan bahwa data yang digunakan dalam pelatihan dan pengujian model GRU berada dalam kondisi bersih, konsisten, dan siap diolah. Tahapan ini penting karena data sensor mentah sering mengandung *noise*, ketidakteraturan, dan perbedaan format antarperangkat.

3.3.1 Penggabungan dan Penyelarasan Format Data

Langkah pertama adalah menyesuaikan struktur dan format data dari kedua sumber agar dapat diproses secara seragam. Data mentah yang diperoleh dari *smartwatch* Android biasanya berupa rekaman *time series* dengan interval tertentu, yaitu lima data per detik. Sebaliknya, data WESAD memiliki *sampling rate* yang jauh lebih tinggi, yaitu hingga 700 Hz, yang berarti sensornya merekam sekitar 700 data dalam satu detik. Tingginya frekuensi pengambilan sampel ini membuat data WESAD jauh lebih detail dan halus seperti BVP yang dibutuhkan untuk data HR, sehingga ukuran datanya lebih besar dan memerlukan proses pengolahan yang lebih kompleks.

Oleh karena itu, dilakukan proses *resampling* agar seluruh data berada dalam satuan waktu yang sama. Setiap kolom data diberi label waktu (*timestamp*) yang seragam untuk memudahkan sinkronisasi antarparameter. Kemudian penyelarasan nama kolom dan struktur tabel sangat penting agar dapat diproses dalam *pipeline* yang sama.

3.3.2 Pembersihan Data (*Data Cleaning*)

Data dari perangkat *wearable* sering kali mengandung nilai yang hilang (*missing values*), *outlier*, atau sinyal yang tidak stabil akibat gangguan sensor. Oleh karena itu, dilakukan langkah-langkah berikut:

- a. Menghapus atau mengganti nilai kosong menggunakan metode interpolasi linier dan *forward fill*.
- b. Menghilangkan *outlier* berdasarkan ambang batas fisiologis (misalnya detak jantung di bawah 40 bpm atau di atas 200 bpm dianggap tidak valid).

- c. Menyaring sinyal bising dengan menggunakan *Butterworth low-pass filter* untuk data seperti EDA dari WESAD.

3.3.3 Standardisasi Data

Agar model GRU dapat belajar secara optimal, seluruh fitur numerik diproses menggunakan metode standardisasi *Z-Score* (atau *Standard Serialization*). Berbeda dengan *Min-Max Normalization* yang membatasi data pada rentang tertentu, *Z-Score* mengubah distribusi data sehingga memiliki rata-rata (*mean*) bernilai 0 dan standar deviasi bernilai 1 (Han et al., 2012).

Metode ini dipilih karena sangat efektif dalam menangani data sensor biologis (*Heart Rate*) dan pergerakan (*Accelerometer*) yang seringkali memiliki pencilan (*outlier*) atau lonjakan nilai yang tiba-tiba saat kondisi stres terjadi. Dengan memusatkan data pada nilai rata-rata, model GRU dapat lebih stabil dalam mempelajari variasi temporal tanpa terganggu oleh perbedaan satuan antara BPM (*Beats Per Minute*) dan m/s^2 . Proses ini membantu mencegah terjadinya *gradient vanishing* atau *exploding* pada jaringan saraf rekuren.

Persamaan *Z-Score Standardization* yang digunakan dalam penelitian ini adalah:

$$z = \frac{x - \mu}{\sigma} \quad (3.1)$$

Keterangan:

- z = nilai fitur hasil standardisasi (skor-z)
- x = nilai asli (*raw data*) fitur sebelum diproses
- μ = nilai rata-rata (*mean*) dari fitur tersebut pada dataset training
- σ = Nilai simpangan baku (*standard deviation*) dari fitur tersebut pada dataset training.

3.3.4 Labeling Data

Pada data WESAD, label kelas sudah tersedia berdasarkan kondisi emosional peserta, yaitu:

- a. 0 = *Baseline* (keadaan normal)
- b. 1 = *Stress*
- c. 2 = *Amusement*

Masing-masing divalidasi melalui kombinasi sensor fisiologis dan prosedur induksi emosi. Label tersebut telah digunakan secara luas dalam penelitian mengenai deteksi stres dan emosi dengan berbasis sinyal fisiologis, sehingga dapat menjadi rujukan yang kuat untuk pengembangan model berbasis *deep learning*.

Untuk data primer yang dikumpulkan dari *smartwatch*, proses pelabelan dilakukan dengan mengacu pada parameter fisiologis yang tersedia pada perangkat, seperti detak jantung (*Heart Rate*/HR) dan tingkat percepatan aktivitas fisik (*accelerometer*). Penentuan label ini tidak dilakukan secara subjektif, melainkan berdasarkan acuan yang konsisten dengan literatur sebelumnya yang membahas hubungan antara perubahan fisiologis dan kondisi emosional.

Beberapa penelitian mendukung hubungan tersebut, Kim & André (2006) menunjukkan bahwa kondisi emosi dapat direpresentasikan melalui kombinasi HR dan parameter fisiologis lain seperti ACC. Shaffer & Ginsberg (2017) menjelaskan rentang HR normal dalam kondisi istirahat, yang umum digunakan sebagai indikator *baseline* fisiologis. Healey & Picard (2005) menunjukkan bahwa peningkatan HR pada aktivitas fisik yang rendah sering berhubungan dengan stres fisiologis atau beban emosional. Pada konteks *wearable* modern, Gjoreski et al.

(2017) juga mengonfirmasi bahwa HR, HRV, dan aktivitas gerak dapat membedakan kondisi stres dalam *setting* kehidupan sehari-hari. Selain itu, Schmidt et al. (2018) dalam studi WESAD menekankan bahwa kondisi *amusement* ditandai oleh peningkatan moderat pada HR serta perubahan pola gerakan ringan sebagai respons arousal positif.

Dengan merujuk pada dasar-dasar tersebut, kategori label pada data smartwatch diselaraskan dengan tiga label yang ada pada WESAD, yaitu 0 (*Baseline*), 1 (*Stress*), dan 2 (*Amusement*). Penyelarasan ini penting agar model GRU yang dilatih menggunakan data WESAD dapat digunakan kembali (transfer) untuk mengevaluasi sinyal fisiologis pengguna *smartwatch* tanpa melakukan pelatihan ulang berbasis emosi manusia.

Penting untuk dicatat bahwa proses pelabelan ini tidak dimaksudkan untuk memberikan diagnosis medis atau memprediksi kondisi klinis tertentu. Pelabelan ini hanya bertujuan menyediakan representasi relatif dari kondisi fisiologis pengguna, sehingga model pembelajaran terawasi dapat mempelajari pola dinamika sinyal yang berkaitan dengan tiga kategori yang relevan. Dengan demikian, kesetaraan label antara WESAD dan data *smartwatch* memungkinkan model GRU untuk melakukan prediksi kondisi fisiologis secara konsisten, baik pada data laboratorium maupun data penggunaan sehari-hari.

3.3.5 Pembagian Data (*Data Splitting*)

Setelah data bersih dan terlabel, dataset dibagi menjadi dua bagian:

- a. *Training set* (80%) untuk melatih model GRU.
- b. *Testing set* (20%) untuk mengatur parameter dan mencegah *overfitting*.

Pembagian dilakukan secara acak menggunakan *stratified sampling* agar proporsi kelas tetap seimbang pada setiap subset data. Proses ini memastikan evaluasi model dilakukan secara objektif dan tidak bergantung pada data pelatihan.

Melalui proses *preprocessing* yang sistematis, baik data primer maupun sekunder dapat digunakan secara efisien dalam pelatihan model GRU. Dengan data yang bersih, seragam, dan terlabel dengan baik, sistem mampu melakukan prediksi kondisi kesehatan pengguna secara lebih akurat.

3.4 Perancangan Sistem

Perancangan sistem bertujuan untuk menggambarkan alur kerja dan interaksi antar komponen pada sistem pemantauan kesehatan berkelanjutan berbasis *smartwatch* Android dengan penerapan model *Gated Recurrent Unit* (GRU). Sistem ini dirancang agar mampu memantau, mengolah, dan menganalisis data fisiologis pengguna melalui integrasi antara perangkat keras, perangkat lunak, dan layanan berbasis IoT.

3.4.1 Arsitektur Sistem

Arsitektur sistem yang dibangun pada penelitian ini terdiri atas tiga lapisan utama, yaitu Lapisan Akuisisi Data, Lapisan Pengolahan dan Analisis, dan Lapisan Presentasi. Setiap lapisan memiliki peran berbeda mulai dari pengambilan data sensor, pemrosesan oleh model GRU, hingga penyajian hasil kepada pengguna. Secara garis besar, proses *training* dilakukan menggunakan dataset WESAD, sedangkan data dari *smartwatch* hanya digunakan untuk *inference* atau prediksi secara langsung.

1. Lapisan Akuisisi Data (*Data Acquisition Layer*)

Lapisan ini berfungsi mengumpulkan data fisiologis pengguna secara langsung melalui sensor pada *smartwatch*. Melalui Wear OS Data Layer API dengan metode `MessageClient` digunakan untuk mengirim data dari sensor bawaan, yaitu detak jantung (*Heart Rate*) dan tingkat aktivitas (*accelerometer*). Data dikumpulkan secara periodik, disimpan sementara di aplikasi, kemudian dikirim ke server IoT untuk diproses lebih lanjut melalui koneksi internet yang aman.

2. Lapisan Pengolahan dan Analisis (*Processing & Analysis Layer*)

Lapisan ini merupakan inti dari sistem. Data mentah yang diterima dari *smartwatch* akan melalui beberapa tahapan, yaitu:

- a. *Preprocessing*, meliputi pembersihan data (*data cleaning*), penyamaan format waktu (*resampling*), dan standardisasi dengan metode *z-score*. Tahapan ini memastikan data dari *smartwatch* memiliki format yang konsisten dengan data pelatihan (WESAD).
- b. *Feature extraction* tanpa labeling, pada tahap penggunaan user, sistem hanya mengekstrak fitur yang diperlukan (misalnya HR rata-rata *window*, variabilitas HR, *magnitude accelerometer*). Tidak dilakukan *labeling*, karena label hanya ada pada dataset WESAD saat training.
- c. Analisis dengan Model GRU (*Inference*), Model GRU yang sebelumnya dilatih menggunakan dataset WESAD dengan label *baseline*, *stress*, *amusement* akan digunakan untuk mempelajari pola temporal HR dan ACC serta memprediksi kondisi fisiologis pengguna berdasarkan pola tersebut. Model tersebut akan menghasilkan keluaran kategori kondisi, yaitu *Baseline/ Relaksasi*, *Stress* dan

Amusement / Positif. Hasil prediksi disimpan sementara di server dan dikirim kembali ke aplikasi pengguna.

3. Lapisan Presentasi (*Presentation Layer*)

Lapisan ini akan menghasilkan analisis dikirim kembali ke aplikasi Android dan ditampilkan dalam bentuk visual seperti tingkat kondisi, serta notifikasi atau rekomendasi kesehatan. Pengguna dapat melihat:

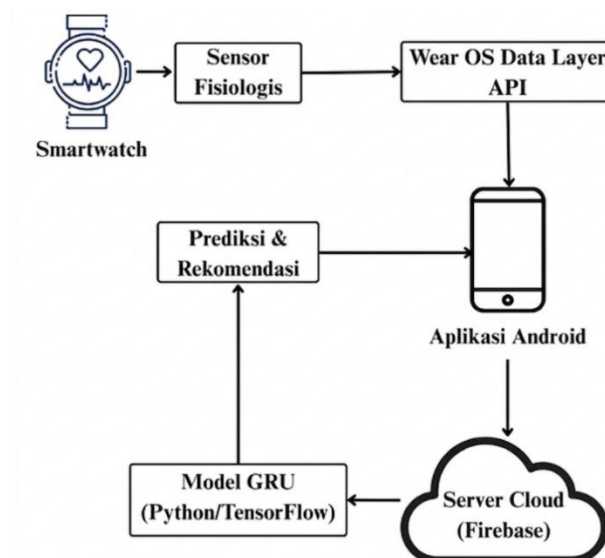
- a. Kategori kondisi tubuh terkini (misalkan: normal, stress ringan, senang).
- b. Rekomendasi adaptif berbasis hasil prediksi model GRU, seperti:

“Kondisi tubuh stabil, lanjutkan aktivitas.”

“Tanda stres terdeteksi, lakukan pernapasan dalam.”

Notifikasi cerdas diberikan untuk mendukung pemantauan kesehatan secara mandiri dan berkelanjutan.

3.4.2 Diagram Arsitektur Sistem



Gambar 3.2 Diagram Arsitektur Sistem Pemantauan Kesehatan Berkelanjutan

Diagram yang ditunjukkan pada Gambar 3.2 yaitu hubungan antar komponen utama dalam sistem pemantauan kesehatan berbasis *smartwatch* Android dengan penerapan model *Gated Recurrent Unit* (GRU). Sistem ini terdiri dari empat komponen utama, yaitu *smartwatch*, Wear OS Data Layer API, *server cloud* (Firebase), dan modul *machine learning* (GRU) yang terintegrasi dengan aplikasi Android sebagai antarmuka pengguna.

Alur data dimulai dari sensor fisiologis pada *smartwatch* yang mengukur parameter seperti detak jantung dan aktivitas gerak. Data tersebut diambil secara *real-time* melalui *Wear OS Data Layer API*, yang memiliki fungsi untuk menghubungkan antara perangkat *wearable* dan aplikasi Android yang resmi.

Selanjutnya, pada aplikasi Android akan diproses lebih lanjut oleh modul *machine learning* berbasis GRU (*Python/TensorFlow*) untuk melakukan analisis sekuensial terhadap data fisiologis pengguna. Model GRU mengenali pola temporal untuk mendeteksi kondisi kesehatan pengguna berdasarkan tiga kategori kondisi fisiologis, yaitu relaksasi, stres ringan, dan senang. Data fisiologis dan hasil analisis selanjutnya akan dikirim dan disimpan di *server cloud* yang menggunakan *Firebase Firestore SDK*.

Hasil analisis berupa prediksi dan rekomendasi kesehatan dikirim kembali dari modul GRU ke aplikasi Android melalui server IoT. Aplikasi kemudian menampilkan hasil tersebut dalam bentuk visualisasi data (probabilitas, status tubuh) serta notifikasi adaptif seperti “Kondisi tubuh stabil”, “Tubuh menunjukkan gejala stress ringan” atau “Tanda vital menunjukkan kondisi positif.”

Dengan desain ini, sistem mampu menjalankan proses pemantauan kesehatan secara langsung dan berkelanjutan, serta memberikan informasi yang lebih adaptif, personal, dan berbasis data untuk pengguna.

3.4.3 Komponen Sistem

a) *Smartwatch* Android

Smartwatch berfungsi sebagai perangkat utama pengumpul data fisiologis pengguna melalui sensor bawaan seperti *heart rate sensor* dan *accelerometer*. Data ini mencakup detak jantung dan akselerometer.

b) *Wear OS Data Layer API*

API ini berperan sebagai jembatan antara *smartwatch* dan aplikasi Android, memungkinkan pengambilan data kesehatan dari sensor secara *real-time* dengan izin pengguna.

c) Aplikasi Android

Aplikasi Android berfungsi sebagai pusat pengelolaan sistem sekaligus antarmuka pengguna (*user interface*). Aplikasi ini menerima data fisiologis dari *smartwatch*, melakukan pemrosesan awal data, serta menjalankan model *machine learning Gated Recurrent Unit* (GRU) untuk menganalisis kondisi kesehatan pengguna. Selain itu, aplikasi juga menampilkan hasil analisis dalam bentuk status kondisi tubuh, dan rekomendasi kesehatan.

d) *Firebase Cloud Firestore*

Firebase Cloud Firestore digunakan sebagai layanan basis data cloud untuk menyimpan data fisiologis pengguna serta hasil analisis model GRU. Penyimpanan

ini memungkinkan pencatatan riwayat kesehatan pengguna secara terpusat serta mendukung sinkronisasi data antar perangkat apabila diperlukan.

e) Model GRU (*Python / Tensorflow*)

Model *Gated Recurrent Unit* (GRU) merupakan komponen pembelajaran mesin yang digunakan untuk menganalisis data fisiologis secara berurutan (*time series*). Model ini ditanamkan pada aplikasi Android dengan menggunakan *TensorFlow Lite* sehingga proses inferensi dapat dijalankan langsung pada perangkat pengguna (*on-device inference*). Model GRU mempelajari pola temporal dari data fisiologis untuk memprediksi kondisi kesehatan pengguna, seperti relaksasi, stres ringan, atau aktivitas tinggi. Hasil prediksi kemudian ditampilkan pada aplikasi Android sebagai riwayat pemantauan kesehatan.

3.4.4 Alur Proses Sistem

Secara garis besar, alur kerja sistem adalah sebagai berikut:

a. Pengumpulan Data dari *Smartwatch*

Smartwatch Android mengumpulkan data fisiologis pengguna yaitu detak jantung dan akselerometer melalui sensor bawaan perangkat. Data ini diakses menggunakan *Wear OS Data Layer API*, yang menjembatani komunikasi antara sensor dan aplikasi Android dengan tetap menjaga privasi pengguna.

b. Penerimaan dan *Preprocessing Data*

Data fisiologis yang diterima oleh aplikasi Android kemudian melalui tahap *preprocessing* agar dapat memastikan kualitas data sebelum dianalisis. Proses ini meliputi standarisasi data, penyelarasan format waktu (*timestamp*), serta pembersihan data dari nilai yang tidak valid atau anomali.

c. Analisis Menggunakan Model GRU

Data yang telah diproses selanjutnya dianalisis menggunakan model *Gated Recurrent Unit* (GRU) yang telah ditanamkan ke dalam aplikasi Android menggunakan *TensorFlow Lite*. Model ini mempelajari pola temporal dari data fisiologis untuk mengenali kondisi pengguna, seperti relaksasi, stres ringan, atau senang, berdasarkan hasil pelabelan tiga kategori kondisi tubuh.

d. Penyimpanan Data ke Cloud

Data fisiologis yang dihasilkan kemudian disimpan pada layanan *Firebase Cloud Firestore*. Penyimpanan ini bertujuan untuk mencatat riwayat kondisi kesehatan pengguna serta memungkinkan sinkronisasi data pada aplikasi.

e. Penyajian Hasil Analisis pada Aplikasi

Aplikasi Android menampilkan hasil analisis dalam bentuk visualisasi yang informatif, seperti status kondisi tubuh, serta notifikasi atau rekomendasi kesehatan adaptif. Dengan demikian, pengguna dapat memantau kondisi kesehatannya secara langsung melalui aplikasi.

3.4.5 Target Performa Sistem dan Model

Sebelum dilakukan evaluasi, penelitian ini menetapkan target performa sistem dan model GRU sebagai acuan keberhasilan. Target ini merujuk pada standar umum penelitian *time-series* berbasis *deep learning* serta kemampuan pemrosesan perangkat *wearable*.

Target performa yang ditetapkan:

1. Target performa model GRU:

- a. Akurasi minimal 85% pada data *testing*,

- b. *F1-score* minimal 0.80 untuk setiap kelas (*baseline, stress, amusement*),
 - c. *Loss* model stabil pada *epoch* akhir tanpa tanda *overfitting* signifikan.
- 2. Target performa sistem IoT:
 - a. Keandalan pengiriman data *real-time* minimal 95% berhasil terkirim.
 - b. Konsumsi daya aplikasi tidak mengganggu penggunaan normal *smartwatch*.
- 3. Target fungsi aplikasi Android:
 - a. Data fisiologis tampil tanpa delay berarti (<1 detik setelah diterima).
 - b. Notifikasi rekomendasi muncul maksimal 3 detik setelah model mengeluarkan prediksi.

Target ini digunakan sebagai tolok ukur untuk menilai apakah integrasi model GRU, aplikasi Android, dan server IoT telah memenuhi standar kelayakan.

3.5 Implementasi Model GRU

Implementasi model *Gated Recurrent Unit* (GRU) bertujuan untuk mempelajari pola sekuensial dari data fisiologis pengguna seperti detak jantung dan percepatan aktivitas fisik, sehingga sistem mampu memberikan analisis dan rekomendasi kesehatan adaptif. Model GRU dipilih karena memiliki kemampuan untuk menangani data *time series* dengan efisien, menggunakan lebih sedikit parameter disbanding dengan *Long Short-Term Memory* (LSTM), tetapi tetap mempertahankan performa tinggi dalam mengenali ketergantungan jangka panjang antar data fisiologis.

3.5.1 Arsitektur Model

Model GRU yang digunakan terdiri dari beberapa lapisan utama sebagai berikut:

1. Input Layer

Lapisan ini menerima data hasil *preprocessing* dari Wear OS Data Layer API dan dataset WESAD yang sudah disesuaikan formatnya. Fitur utama yang digunakan meliputi:

- a) *Heart rate* (detak jantung)
- b) *Accelerometer* (percepatan aktivitas fisik)

Data diatur dalam bentuk *sequence* dengan panjang jendela waktu tertentu (60 detik terakhir) agar GRU dapat mengenali dinamika temporal dari perubahan data kesehatan pengguna.

2. GRU Layers

Model menggunakan satu lapisan GRU yang berfokus mengenali pola jangka pendek, seperti fluktuasi detak jantung atau perubahan aktivitas harian. Setiap unit GRU menggunakan fungsi aktivasi tanh dan mekanisme update gate serta reset gate untuk mengontrol aliran informasi.

3. Dropout Layer

Untuk mengurangi risiko *overfitting*, diterapkan *dropout* sebesar 0.3, di mana sebagian neuron akan dinonaktifkan secara acak selama proses pelatihan. Langkah ini membantu model agar lebih general dan tidak hanya menghafal pola dari data pelatihan.

4. Dense Layer (Output Layer)

Lapisan akhir digunakan fungsi aktivasi *softmax* untuk menghasilkan probabilitas dari setiap kategori kondisi tubuh. Berdasarkan rancangan pelabelan yang telah disusun sebelumnya, kategori keluaran model terdiri dari tiga kelas pada tabel 3.1.

Tabel 3.1 Label kategori kondisi

Label	Kategori Kondisi	Deskripsi Singkat
0	Relaksasi / Istirahat	Aktivitas rendah, detak jantung normal
1	Stres Fisiologi Ringan	HR tinggi, aktivitas rendah
2	<i>Amusement</i> / Senang	Aktivitas rendah-sedang, HR naik ringan

3.5.2 Parameter Pelatihan

Pelatihan model dilakukan menggunakan dataset gabungan dari data primer (*Wear OS Data Layer API*) dan data sekunder (WESAD) yang telah melalui tahap sinkronisasi dan *preprocessing*. Tujuan dari pelatihan ini adalah agar model GRU mampu mengenali pola temporal dalam data fisiologis dan mengklasifikasikan kondisi kesehatan pengguna ke dalam tiga kategori utama. Parameter pelatihan yang digunakan tertera pada tabel 3.2.

Tabel 3. 2 Parameter Pelatihan

No	Parameter	Nilai
1	<i>Epoch</i>	200 (<i>Early Stopping</i>)
2	<i>Batch Size</i>	64
3	<i>Optimizer</i>	Adam
4	<i>Learning rate</i>	0.001
5	<i>Loss function</i>	<i>Sparse Categorical Cross-Entropy</i>
6	<i>Validation Split</i>	0.2
7	<i>Dropout</i>	0.3
8	GRU <i>units</i>	128

Proses pelatihan dilakukan menggunakan lingkungan *Python* dengan *TensorFlow/Keras framework*. Model yang telah dilatih kemudian disimpan dalam format *.tflite* untuk diintegrasikan ke dalam aplikasi Android.

3.5.3 Proses Pelatihan Model

Tahapan pelatihan dilakukan sebagai berikut:

- a) Data hasil *preprocessing* dibagi menjadi *training set* dan *testing set*.
- b) Model GRU dilatih menggunakan *training set* dengan *backpropagation through time (BPTT)*.
- c) Model diuji dengan *testing set* untuk menilai akurasi dan kemampuan generalisasi.

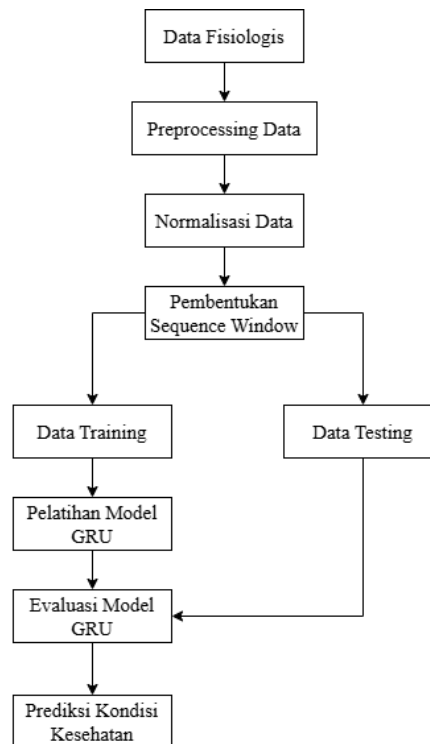
Proses pelatihan ini dilakukan secara iteratif hingga nilai loss stabil dan akurasi pada *validation set* tidak meningkat signifikan, menandakan model telah mencapai konvergensi.

3.5.4 Integrasi Model ke Sistem

Model GRU yang telah dilatih ditanamkan dengan sistem pemantauan berbasis IoT melalui pendekatan *On-device inference*. Model dikonversi ke format *TensorFlow Lite* (.tflite) untuk memungkinkan eksekusi lokal pada perangkat Android (*smartphone* pengguna) guna menjaga fungsi dasar sistem saat koneksi internet tidak stabil.

Pendekatan *on-device inference* dipilih sebagai metode utama dalam sistem ini karena memungkinkan proses analisis data fisiologis dilakukan langsung pada perangkat pengguna. Dengan pendekatan ini, model *machine learning* dapat memproses data tanpa bergantung pada koneksi server. Selain meningkatkan kecepatan respons sistem, metode ini juga membantu menjaga privasi data pengguna karena sebagian besar proses analisis dilakukan secara lokal pada perangkat Android.

3.5.5 Diagram Alur Implementasi GRU



Gambar 3.3 Diagram Alur Implementasi GRU

Diagram pada Gambar 3.3 menggambarkan tahapan mulai dari input data fisiologis, proses standardisasi, pembentukan *sequence window*, pelatihan model GRU, hingga hasil prediksi kondisi tubuh pengguna.

Melalui penerapan model GRU, sistem pemantauan kesehatan dapat mengenali pola fisiologis pengguna secara dinamis dan memberikan rekomendasi berbasis data yang lebih akurat. Implementasi ini menjadikan *smartwatch* tidak hanya sebagai alat pengumpul data, tetapi juga sebagai perangkat cerdas yang mampu menganalisis dan merespons kondisi tubuh pengguna secara berkelanjutan.

3.6 Integrasi Sistem IoT dan Aplikasi Android

Integrasi sistem IoT dan aplikasi Android dilakukan untuk memastikan seluruh komponen dalam sistem pemantauan kesehatan berkelanjutan dapat berkomunikasi secara sinkron, mulai dari proses pengambilan data sensor, pengiriman ke server *cloud*, hingga penampilan hasil analisis kepada pengguna. Sistem ini menggabungkan konsep *Internet of Things (IoT)*, *machine learning*, dan komputasi untuk memberikan pemantauan kesehatan yang akurat dan adaptif.

3.6.1 Struktur Umum Sistem IoT

Sistem IoT dalam penelitian ini dibangun dengan empat komponen utama yang saling terhubung untuk memungkinkan pemantauan kesehatan secara langsung dan berkelanjutan, yaitu:

1) *Smartwatch* Android

Smartwatch berfungsi sebagai perangkat utama pengumpul data fisiologis pengguna melalui berbagai sensor internal seperti *heart rate sensor* dan *accelerometer*. Akses data dilakukan melalui sensor bawaan dari *smartwatch* kemudian dikirim menggunakan *Wear OS Data Layer API*, yang memungkinkan aplikasi memperoleh data seperti:

- a. Detak jantung (*Heart Rate*)
- b. Gerakan fisik (*Accelerometer*)

Data ini secara periodik dan dikirim ke aplikasi Android untuk diteruskan ke server.

2) *Smartphone* Android

Smartphone Android berfungsi sebagai perangkat penghubung antara *smartwatch* dan sistem pemrosesan data pada perangkat. Aplikasi Android

menerima data fisiologis dari *smartwatch* melalui *Wear OS Data Layer API*, kemudian menampilkan informasi kesehatan pengguna dalam bentuk antarmuka yang interaktif. Selain itu, aplikasi juga menjalankan proses analisis menggunakan model GRU berbasis *on-device inference* sehingga seluruh proses pemrosesan data dilakukan secara lokal pada perangkat tanpa memerlukan koneksi ke server eksternal. Pendekatan ini memungkinkan sistem tetap berfungsi secara langsung, offline, serta meningkatkan privasi dan keamanan data pengguna.

3) Server IoT dan Model GRU

Server cloud berperan sebagai pusat pemrosesan data dan analisis *machine learning*. Data fisiologis yang dikirim dari aplikasi Android disimpan pada layanan *Firebase Cloud Firestore*. Selanjutnya, data tersebut diakses oleh modul analisis yang menjalankan model *Gated Recurrent Unit* (GRU) yang diimplementasikan menggunakan *Python* dengan *framework TensorFlow* untuk melakukan proses inferensi atau prediksi kondisi kesehatan pengguna. Hasil prediksi yang dihasilkan oleh model GRU kemudian disimpan kembali ke *Firebase Cloud Firestore* dan dikirimkan ke aplikasi Android untuk ditampilkan kepada pengguna dalam bentuk informasi kondisi tubuh dan rekomendasi kesehatan.

4) Database Cloud

Sistem menggunakan layanan *Firebase Cloud Firestore* sebagai basis data cloud untuk menyimpan dan mengelola data pengguna secara terpusat. *Firestore* merupakan database NoSQL berbasis dokumen yang mampu menangani penyimpanan data secara *real-time*, skalabel, dan mudah diintegrasikan dengan

aplikasi Android. Database ini digunakan untuk menyimpan berbagai informasi yang diperlukan oleh sistem, antara lain:

- a. Riwayat data fisiologis pengguna
- b. Hasil prediksi dan rekomendasi model GRU
- c. Parameter sistem untuk analisis lebih lanjut

3.6.2 Alur Kerja Sistem

Secara garis besar, alur kerja sistem pemantauan kesehatan berkelanjutan berbasis IoT ini dapat dijelaskan sebagai berikut:

1) *Smartwatch* Android

Smartwatch mengumpulkan data fisiologis pengguna secara *real time* dengan sensor internal, yaitu detak jantung dan akselerometer. Akses data dilakukan menggunakan aplikasi yang sudah peneliti buat pada *smartwatch* yang berfungsi untuk membaca sensor guna menjembatani komunikasi antara sensor dan aplikasi Android.

2) Pengiriman Data ke aplikasi Android

Data hasil pengukuran dikumpulkan melalui *Wear OS Data Layer API* pada *smartwatch* Android dan diteruskan ke aplikasi Android. Aplikasi berfungsi sebagai *gateway* yang menampung sementara data sebelum dikirimkan ke server *cloud* melalui koneksi Bluetooth.

3) Pemrosesan dan Analisis Data pada Aplikasi Android

Data fisiologis yang diterima kemudian diproses langsung pada aplikasi Android. Tahapan ini meliputi preprocessing awal seperti standarisasi data dan

penyusunan data dalam bentuk *time-series* sebelum dianalisis menggunakan model machine learning.

4) Analisis Data dengan Model GRU

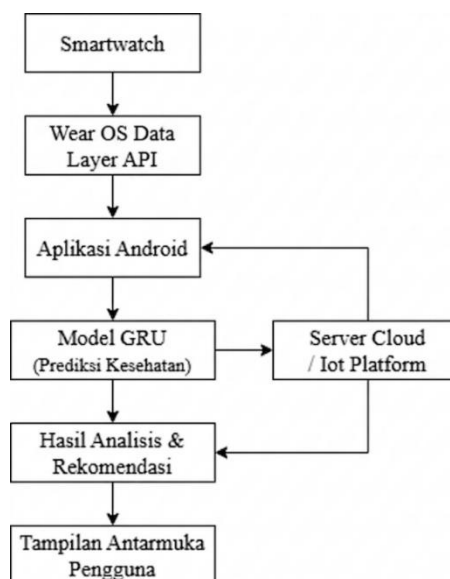
Model *Gated Recurrent Unit* (GRU) yang telah diintegrasikan ke dalam aplikasi Android digunakan untuk menganalisis pola temporal dari data fisiologis pengguna. Model ini menghasilkan prediksi kondisi kesehatan pengguna.

5) Penyimpanan Data Ke *Cloud*

Data fisiologis serta hasil prediksi yang dihasilkan oleh model GRU kemudian disimpan ke *Firebase Cloud Firestore* untuk keperluan pencatatan riwayat kesehatan pengguna dan sinkronisasi data antar perangkat.

6) Tampilan pada Antarmuka Pengguna

Aplikasi Android menampilkan analisis dalam bentuk visual yang informatif, seperti grafik tren detak jantung, tingkat stres, aktivitas harian, serta notifikasi adaptif (misalnya: “*Perlu istirahat,*” atau “*Kondisi tubuh stabil*”).



Gambar 3.4 Diagram Alur Sistem IoT

Diagram pada Gambar 3.4 menunjukkan hubungan antar komponen mulai dari *smartwatch*, aplikasi Android, *server cloud (Firebase)*, Model GRU (*Python*), hingga hasil akhir yang ditampilkan dalam antarmuka pengguna. Panah data menunjukkan aliran dua arah antara aplikasi dan server untuk memastikan sinkronisasi.

3.6.3 Komunikasi Antar Komponen

- a. Komunikasi *Smartwatch* – Aplikasi Android: *Smartwatch* mengirimkan data sensor kesehatan kepada aplikasi Android menggunakan *Wear OS Data Layer API* dengan metode *MessageClient*. Melalui mekanisme ini, data seperti detak jantung dan nilai percepatan dari sensor akselerometer dapat dikirim secara langsung dari perangkat *smartwatch* ke aplikasi Android yang terhubung.
- b. Komunikasi Aplikasi Android – Firebase Cloud Firestore (*IoT Platform*): Aplikasi Android mengirimkan data sensor yang diterima dari *smartwatch* ke layanan *Firebase Cloud Firestore* menggunakan *Firebase Firestore SDK*. Data tersebut disimpan dalam bentuk dokumen pada koleksi *Firestore* dengan struktur yang menyerupai format JSON. Penyimpanan ini memungkinkan data kesehatan pengguna tersimpan secara terpusat di *cloud* serta dapat diakses kembali untuk keperluan pemantauan dan analisis kesehatan secara berkelanjutan.
- c. Komunikasi *Firebase Cloud Firestore* – Sistem Analisis Data: Data yang telah tersimpan pada *Firebase Cloud Firestore* selanjutnya digunakan oleh sistem analisis untuk memproses dan melakukan prediksi menggunakan model *Gated Recurrent Unit (GRU)*. Hasil analisis berupa prediksi kondisi kesehatan

kemudian dapat disimpan kembali ke *Firestore* sebagai bagian dari riwayat kesehatan pengguna sehingga dapat ditampilkan pada aplikasi untuk keperluan monitoring.

3.6.4 Antarmuka Pengguna (*User Interface*)

Aplikasi Android berada di *smartwatch* dan *smartphone* dirancang dengan antarmuka yang sederhana, interaktif, dan informatif. Beberapa tampilan utama antara lain:

- a. Halaman ambil data (*smartwatch*): Menampilkan halaman pengambilan data pada perangkat *smartwatch* dirancang untuk menampilkan data sensor (HR dan ACC) secara *real-time* langsung dari pergelangan tangan pengguna.
- b. Halaman Utama (*Home*): Menampilkan data fisiologis seperti detak jantung, kalori dan langkah. Halaman ini juga memuat grafik aktivitas harian, notifikasi rekomendasi otomatis dari model GRU, serta pengingat kesehatan agar pengguna dapat segera merespons kondisi tubuhnya.
- c. Halaman *Activity*: Menampilkan berbagai fitur aktivitas pendukung kesehatan mental dan fisik, seperti *breathing exercise*, meditasi, afirmasi harian, dan *focus mode* yang dapat digunakan pengguna untuk membantu mengurangi stres dan menjaga kestabilan kondisi tubuh.
- d. Halaman Profil dan Pengaturan: Halaman ini berfungsi sebagai pusat informasi dan pengaturan akun pengguna.

Integrasi antara komponen IoT, model GRU, dan aplikasi Android memungkinkan sistem untuk melakukan pemantauan kesehatan secara terus-menerus dan adaptif. Sistem ini tidak hanya mencatat data kesehatan pengguna,

tetapi juga mampu memberikan rekomendasi berbasis analisis prediktif yang dapat membantu menjaga keseimbangan kondisi tubuh sehari-hari.

3.7 Rencana Pengujian Sistem

Rencana pengujian sistem bertujuan untuk mengevaluasi performa keseluruhan sistem pemantauan kesehatan berkelanjutan berbasis *IoT smartwatch* Android yang menggunakan model *Gated Recurrent Unit* (GRU). Pengujian dilakukan untuk memastikan bahwa sistem berjalan stabil, data terbaca dengan benar, hasil analisis akurat, serta rekomendasi kesehatan yang diberikan sesuai dengan kondisi fisiologis pengguna.

3.7.1 Tujuan Pengujian

Tujuan utama pengujian sistem ini adalah:

- a. Memastikan integrasi antara *smartwatch*, aplikasi Android, dan server *cloud* berjalan tanpa gangguan.
- b. Mengukur akurasi model GRU dalam mengenali pola fisiologis pengguna berdasarkan data secara langsung.
- c. Menilai performa sistem dari segi kecepatan pemrosesan, keandalan komunikasi data, serta responsivitas aplikasi.
- d. Mengevaluasi tampilan antarmuka dan kemudahan navigasi, kejelasan informasi, serta kenyamanan penggunaan (*usability*) aplikasi bagi pengguna umum.

3.7.2 Jenis Pengujian

Pengujian sistem dibagi menjadi beberapa kategori berikut:

a. Pengujian Fungsional (*Functional Testing*)

Dilakukan untuk memastikan setiap komponen sistem berjalan sesuai rancangan.

Beberapa skenario yang diuji antara lain:

Tabel 3.3 Pengujian Fungsional

No	Komponen	Aspek yang Diuji	Kriteria Keberhasilan
1	<i>Smartwatch</i>	Sensor detak jantung dan aktivitas tubuh	Data terbaca dan terkirim ke aplikasi
2	Aplikasi Android	Tampilan data, koneksi ke server, notifikasi rekomendasi	Semua fitur dapat diakses tanpa error
3	Server <i>Cloud</i>	Penerimaan dan pemrosesan data	Data tersimpan dan terproses oleh model GRU
4	Model GRU	Hasil prediksi kondisi tubuh	Output kategori sesuai label referensi

Tabel 3.3 menampilkan hasil rancangan skenario pengujian fungsional untuk setiap komponen utama sistem. Pengujian dilakukan dengan tujuan memastikan bahwa alur integrasi antar komponen berjalan stabil, mulai dari pengambilan data melalui di *smartwatch*, pengiriman data ke server *cloud*, hingga pemrosesan oleh model GRU yang menghasilkan prediksi kondisi kesehatan pengguna. Setiap komponen dianggap berhasil apabila memenuhi kriteria keberhasilan yang telah ditetapkan tanpa terjadi error atau kehilangan data selama proses pengujian.

b. Pengujian Akurasi Model (*Model Evaluation*)

Pengujian ini dilakukan terhadap model GRU menggunakan dataset *testing* hasil *preprocessing*. Evaluasi dilakukan menggunakan beberapa metrik utama, yaitu:

- Accuracy* (Akurasi): Persentase prediksi benar terhadap total data uji.
- Precision*: Proporsi prediksi benar dari seluruh prediksi positif.

c) *Recall*: Kemampuan model mengenali kondisi sebenarnya.

d) F1-Score: Rata-rata harmonik dari *precision* dan *recall*.

Rumus-rumus perhitungan:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

Keterangan:

- 1) TP (*True Positive*): Jumlah sampel berlabel positif yang berhasil diklasifikasikan secara tepat oleh model.
- 2) TN (*True Negative*): Jumlah sampel berlabel negatif yang berhasil diklasifikasikan secara tepat oleh model.
- 3) FP (*False Positive*): Jumlah sampel berlabel negatif yang keliru diklasifikasikan sebagai positif oleh model.
- 4) FN (*False Negative*): Jumlah sampel berlabel positif yang keliru diklasifikasikan sebagai negatif oleh model.
- 5) *Accuracy* mengukur proporsi keseluruhan prediksi yang tepat dibandingkan total seluruh data pengujian.
- 6) *Precision* mengukur sejauh mana model mampu mendeteksi seluruh sampel yang sesungguhnya termasuk kelas positif.
- 7) *Recall* menilai kemampuan model dalam mengenali seluruh data positif yang sebenarnya.
- 8) F1-Score dihitung sebagai rata-rata harmonik dari Precision dan Recall, menjadikannya indikator yang merepresentasikan keseimbangan antara ketepatan dan kelengkapan prediksi model.

c. Pengujian Kinerja Sistem (*Performance Testing*)

Fokus pada kecepatan dan stabilitas sistem dalam kondisi operasional nyata.

Parameter yang diuji:

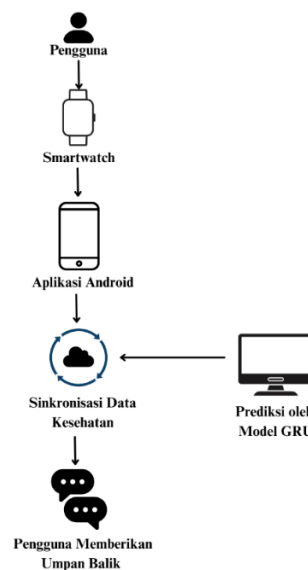
- 1) *Response time*: waktu dari pengiriman data sampai hasil rekomendasi muncul di aplikasi.
- 2) *Data throughput*: jumlah data yang dapat diproses per satuan waktu.
- 3) Stabilitas koneksi: kemampuan sistem menjaga sinkronisasi data saat jaringan tidak stabil.

Hasil pengujian dicatat menggunakan *logging system* untuk menilai kehandalan sistem secara keseluruhan.

d. Pengujian Pengguna (*User Testing*)

Pengujian ini melibatkan sejumlah pengguna untuk menilai aspek kegunaan sistem secara langsung. Kriteria yang dinilai meliputi:

- 1) Kemudahan navigasi aplikasi
- 2) Kejelasan tampilan data kesehatan
- 3) Kecepatan respon terhadap perubahan kondisi tubuh
- 4) Tingkat kepuasan pengguna terhadap rekomendasi yang diberikan



Gambar 3.5 Skema Pengujian Pengguna terhadap Aplikasi Android

Skema pada Gambar 3.5 ini menggambarkan proses pengujian aplikasi Android yang terintegrasi dengan sistem pemantauan kesehatan berbasis GRU. Pengujian dimulai dari pengguna yang memakai *smartwatch* untuk merekam data fisiologis seperti detak jantung dan aktivitas fisik.

Smartwatch terhubung ke aplikasi Android melalui koneksi *Wear OS Data Layer API*.

Aplikasi kemudian memproses data fisiologis yang diterima dari smartwatch menggunakan model GRU yang dijalankan secara *on-device inference* pada perangkat smartphone. Model tersebut menganalisis data secara berurutan untuk menghasilkan prediksi kondisi kesehatan pengguna. Hasil analisis dan rekomendasi kemudian ditampilkan secara pada antarmuka aplikasi Android. Setelah itu, pengguna dapat memberikan umpan balik terkait keakuratan prediksi dan kenyamanan penggunaan aplikasi. Umpan balik tersebut digunakan sebagai dasar untuk evaluasi performa sistem serta penyempurnaan fitur pada tahap pengembangan berikutnya.

3.7.3 Evaluasi dan Analisis Hasil

Hasil pengujian dari seluruh tahap dikumpulkan dan dianalisis untuk menilai apakah sistem memenuhi kriteria berikut:

1. Akurasi prediksi model GRU minimal 85%.
2. Waktu respon sistem maksimal 3 detik untuk setiap permintaan analisis.
3. Data sensor terbaca minimal 95% dari total waktu pemantauan.

Jika hasil pengujian belum memenuhi standar, dilakukan proses *fine-tuning* pada model GRU, optimasi algoritma komunikasi, atau penyempurnaan desain antarmuka aplikasi.

3.7.4 Pengujian Usability Menggunakan SUS (*System Usability Scale*)

Untuk menilai aspek kegunaan (*usability*) aplikasi Android, penelitian ini menggunakan metode *System Usability Scale* (SUS). SUS merupakan standar internasional yang dikembangkan oleh Brooke (1996) untuk mengukur persepsi pengguna terhadap kegunaan sistem melalui 10 item pertanyaan dengan skala Likert 1–5.

Langkah-langkah pengujian SUS:

1. Pemilihan Responden

Pengguna mencoba aplikasi selama 1-3 hari penggunaan normal.

2. Prosedur Uji

- a. Pengguna memakai *smartwatch* untuk merekam data fisiologis.
- b. Aplikasi menampilkan status kondisi, dan rekomendasi GRU.
- c. Setelah mencoba fitur utama, pengguna mengisi kuesioner SUS.

3. Instrumen SUS

Kuesioner berisi 10 pertanyaan:

- a. “Saya merasa akan sering menggunakan aplikasi pemantauan kesehatan ini.”
- b. “Saya merasa aplikasi ini terlalu rumit untuk digunakan.”
- c. “Saya merasa aplikasi ini mudah digunakan.”
- d. “Saya membutuhkan bantuan teknisi untuk dapat menggunakan aplikasi ini.”
- e. “Saya merasa fitur-fitur dalam aplikasi pemantauan kesehatan ini terintegrasi dengan baik.”

- f. “Saya merasa terdapat terlalu banyak ketidakkonsistensinan pada aplikasi ini.”
- g. “Saya merasa kebanyakan orang akan dengan mudah mempelajari aplikasi pemantauan kesehatan ini.”
- h. “Saya merasa aplikasi ini sangat rumit dan membingungkan untuk digunakan.”
- i. “Saya merasa sangat percaya diri dan nyaman menggunakan aplikasi pemantauan kesehatan ini.”
- j. “Saya perlu mempelajari banyak hal terlebih dahulu sebelum dapat menggunakan aplikasi ini.”

4. Perhitungan Skor SUS

- a. Skor dihitung berdasarkan rumus:

$$\text{SUS Score} = \sum(\text{Adjusted Score}) \times 2.5$$

- b. Skor akhir berada pada rentang 0–100.
- c. Interpretasi pada gambar 3.4:

Tabel 3.4 Kategori skor SUS

No	Skor	Kategori
1	> 80	<i>Excellent</i>
2	70 – 80	<i>Good</i>
3	50 – 70	OK
4	< 50	<i>Poor</i>

5. Output Pengujian SUS

Hasil SUS digunakan untuk menilai tingkat kenyamanan, kemudahan, serta kepuasan pengguna terhadap sistem, serta menjadi dasar perbaikan antarmuka dan fitur aplikasi.

BAB IV

HASIL DAN PEMBAHASAN

4.1 Implementasi Sistem

Implementasi sistem pada penelitian ini bertujuan untuk merealisasikan perancangan sistem yang telah dijelaskan pada bab sebelumnya. Sistem dirancang untuk mengidentifikasi kondisi yang dilakukan oleh subjek berdasarkan data sensor yang diperoleh dari perangkat *smartwatch*. Label yang dikenali dalam penelitian ini meliputi *Baseline*, *Stress* dan *Amusement*. Metode klasifikasi yang digunakan adalah *Gated Recurrent Unit* (GRU) yang mampu mengolah data deret waktu secara efektif.

Implementasi sistem mencakup beberapa komponen utama, yaitu perangkat keras, perangkat lunak, basis data, pengelolaan data, analisis data menggunakan model GRU, antarmuka pengguna, serta pengujian sistem. Penjelasan detail dari masing-masing komponen disajikan pada subbab berikut.

4.1.1 Perangkat Keras yang Digunakan

Perangkat keras yang digunakan dalam implementasi sistem ini terdiri dari beberapa komponen utama, yaitu:

a. *Smartwatch*

Smartwatch digunakan sebagai perangkat utama untuk mengumpulkan data sensor dari subjek. Perangkat ini dilengkapi dengan sensor *accelerometer* untuk merekam percepatan gerak pada tiga sumbu (X, Y, dan Z), serta sensor *Heart Rate* (HR) yang berfungsi untuk mengukur denyut jantung subjek selama melakukan

aktivitas. *Smartwatch* dikenakan pada pergelangan tangan subjek untuk memastikan akuisisi data sensor yang stabil dan konsisten.

b. *Smartphone* Android

Smartphone berfungsi sebagai perangkat perantara untuk menerima data dari *smartwatch* serta mengirimkan data tersebut ke sistem penyimpanan dan pemrosesan. *Smartphone* juga digunakan untuk menampilkan hasil klasifikasi kondisi kepada pengguna.

c. Komputer/Laptop

Komputer digunakan untuk proses pengembangan aplikasi, pelatihan model GRU, serta analisis data hasil pengujian sistem.

4.1.2 Perangkat Lunak yang Digunakan

Perangkat lunak yang digunakan dalam implementasi sistem ini adalah sebagai berikut:

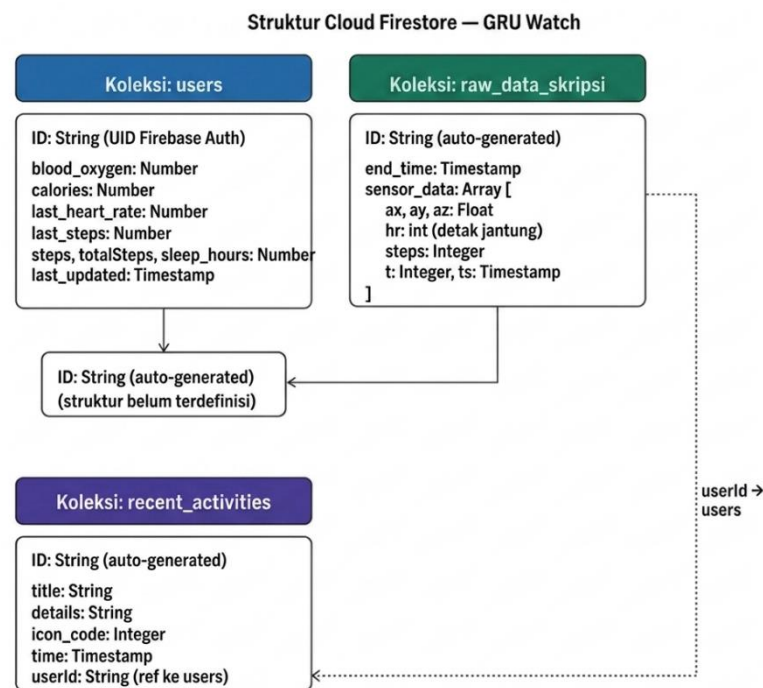
- a. *Wear OS* 16
- b. *WINDOWS* 11
- c. Android 16
- d. *Visual Studio Code* 1.113.0
- e. *Android Studio* 2024.1
- f. *Firebase*

Pada penelitian ini menggunakan beberapa Bahasa pemrograman, yaitu *Python* yang digunakan untuk proses pengolahan data dan pelatihan model GRU. Kemudian bahasa pemrograman *Dart* 3.10.7 dengan menggunakan *framework Flutter*. *Framework* ini memungkinkan pembuatan antarmuka pengguna yang

responsif serta mendukung komunikasi data antara *smartwatch* dan sistem *backend*. Pada *smartwatch* menggunakan bahasa pemrograman Kotlin untuk membaca dan mengirim data sensor ke perangkat *smartphone* atau sistem penyimpanan. Kotlin dipilih karena merupakan bahasa pemrograman resmi Android yang memiliki performa baik dan kompatibilitas tinggi dengan *Wear OS*.

4.1.3 Implementasi Database

Dalam proses implementasi database, sistem menggunakan layanan *Cloud Firestore* dari *Firebase* sebagai penyimpanan data berbasis cloud. *Firestore* menggunakan model dokumen-koleksi (*document-collection*) yang bersifat NoSQL, sehingga struktur data dapat didefinisikan secara fleksibel sesuai kebutuhan sistem. Gambaran mengenai struktur dan relasi basis data disajikan pada Gambar 4.1.



Gambar 4. 1 Struktur Cloud Firestore

Sistem menggunakan empat koleksi utama. Pertama, koleksi `users` yang menyimpan data profil pengguna sekaligus data kesehatan terkini, mencakup field seperti `last_heart_rate`, `last_steps`, `blood_oxygen`, `calories`, `sleep_hours`, dan `last_updated`. Koleksi ini juga memiliki sub-koleksi `health_logs` untuk pencatatan riwayat data kesehatan pengguna. Kedua, koleksi `raw_data_skripsi` yang menyimpan data sensor mentah yang dikirim dari smartwatch, berupa array objek dengan field `ax`, `ay`, `az` (akselerometer), `hr` (detak jantung), `steps`, serta `ts` sebagai timestamp setiap sampel dan `end_time` sebagai waktu akhir sesi pengambilan data. Ketiga, koleksi `recent_activities` yang mencatat log aktivitas pengguna di dalam aplikasi, dengan field `title`, `details`, `icon_code`, `time`, dan `userId` sebagai referensi ke koleksi `users`. Keempat, koleksi `users_skripsi` yang digunakan sebagai koleksi pengembangan dan pengujian selama proses implementasi berlangsung.

Adapun hasil prediksi model GRU tidak disimpan ke *Cloud Firestore*, melainkan diproses dan disimpan secara lokal di perangkat Android menggunakan pendekatan *on-device inference*. Hal ini memungkinkan sistem tetap memberikan hasil analisis kondisi pengguna tanpa ketergantungan penuh terhadap koneksi internet.

4.1.4 Implementasi Data

Implementasi data pada penelitian ini mencakup seluruh proses pengelolaan data, mulai dari pengambilan data sensor, pengiriman data, penyimpanan data, hingga data siap digunakan dalam tahap analisis. Data yang digunakan merupakan data deret waktu (*time-series*) yang diperoleh secara langsung dari sensor pada *smartwatch* saat subjek melakukan aktivitas tertentu.

Data sensor diambil dari sensor *accelerometer* dan sensor *Heart Rate* (HR) yang terdapat pada *smartwatch*. Sensor *accelerometer* menghasilkan data percepatan gerak pada tiga sumbu, yaitu sumbu X, Y, dan Z, sedangkan sensor HR menghasilkan data denyut jantung subjek dalam satuan denyut per menit (*beats per minute*). Pengambilan data dilakukan secara kontinu dengan interval waktu tertentu selama subjek melakukan aktivitas sehari-hari (*free-living*).

Data sensor yang diperoleh dari *smartwatch* kemudian dikirimkan ke perangkat *smartphone* melalui aplikasi yang dikembangkan menggunakan bahasa pemrograman Flutter. Setiap data yang dikirimkan dilengkapi dengan informasi waktu pengambilan data (*timestamp*) untuk menjaga keterurutan data deret waktu. Selanjutnya, data tersebut disimpan ke dalam basis data Firebase.

Pada tahap pengelolaan data, data sensor yang telah tersimpan dalam basis data diberi label sesuai dengan jenis kondisi subjek. Proses pelabelan ini bertujuan untuk membedakan masing-masing kondisi sehingga data dapat digunakan sebagai data pelatihan dan data pengujian pada model GRU. Label kondisi ditentukan berdasarkan kondisi aktual pada saat pengambilan data dilakukan.

Sebelum digunakan dalam proses analisis, data sensor terlebih dahulu melalui tahap *preprocessing*. Tahap ini meliputi pembersihan data dari nilai yang tidak valid, standardisasi data untuk menyamakan skala antar fitur, serta segmentasi data deret waktu menggunakan metode *sliding window*. Segmentasi data dilakukan untuk membentuk potongan data dengan panjang waktu tertentu agar sesuai dengan kebutuhan input model GRU.

Data yang telah diproses selanjutnya dibagi menjadi dua bagian, yaitu data pelatihan (*training data*) dan data pengujian (*testing data*). Data pelatihan digunakan untuk melatih model GRU dalam mengenali pola kondisi, sedangkan data pengujian digunakan untuk mengevaluasi kinerja model dalam mengklasifikasikan kondisi yang belum pernah dilihat sebelumnya.

Dengan implementasi data yang terstruktur dan sistematis ini, diharapkan data yang digunakan dalam penelitian memiliki kualitas yang baik serta mampu merepresentasikan pola kondisi subjek secara akurat, sehingga mendukung proses analisis dan klasifikasi kondisi menggunakan metode GRU.

4.1.5 Implementasi Integrasi IoT dan Protokol Komunikasi Data

Implementasi integrasi IoT pada penelitian ini bertujuan untuk menghubungkan seluruh komponen sistem secara fungsional, mulai dari perangkat *smartwatch* sebagai sumber data, aplikasi Android sebagai pusat pemrosesan, hingga *Firebase Cloud Firestore* sebagai penyimpanan data berbasis *cloud*. Integrasi ini memungkinkan data fisiologis pengguna dikirim, diproses, dan disimpan secara otomatis dan berkelanjutan dalam satu alur yang terintegrasi.

Sistem IoT yang dikembangkan mengadopsi arsitektur tiga lapis, yaitu lapisan persepsi (*perception layer*), lapisan jaringan (*network layer*), dan lapisan aplikasi (*application layer*). Lapisan persepsi terdiri dari *smartwatch* yang mengumpulkan data sensor HR dan ACC. Lapisan jaringan mencakup Wear OS Data Layer API dan Firebase SDK yang mengelola komunikasi antar komponen. Lapisan aplikasi mencakup aplikasi Android yang memproses data, menjalankan inferensi model GRU, dan menampilkan hasil kepada pengguna.

A. Protokol Komunikasi Smartwatch – Aplikasi Android

Komunikasi antara *smartwatch* dan aplikasi Android dilakukan menggunakan Wear OS Data Layer API dengan metode `MessageClient`. Mekanisme ini merupakan antarmuka resmi yang disediakan oleh Google untuk memfasilitasi pertukaran data antara perangkat *wearable* berbasis Wear OS dan aplikasi Android yang terpasang pada *smartphone* yang terhubung.

`MessageClient` bekerja melalui koneksi Bluetooth yang secara otomatis dikelola oleh sistem Wear OS. Data sensor yang telah dikumpulkan oleh aplikasi pada *smartwatch* dikemas dalam format *byte array* sebelum dikirimkan. Di sisi penerima, aplikasi Android melakukan *parsing* terhadap *byte array* tersebut untuk memperoleh kembali nilai HR, ACC X, ACC Y, ACC Z, dan *timestamp*.

Mekanisme ini dipilih karena bersifat *fire-and-forget* dengan latensi rendah, sehingga sesuai untuk kebutuhan pengiriman data sensor secara periodik. Pengiriman data dilakukan setiap satu sesi pengukuran selesai, yaitu setelah aplikasi *smartwatch* mengumpulkan data selama interval yang ditentukan. Seluruh proses pengiriman ini berlangsung di dalam ekosistem perangkat pengguna tanpa melibatkan koneksi internet, sehingga tidak bergantung pada ketersediaan jaringan eksternal.

B. Protokol Komunikasi Aplikasi Android – Firebase Cloud Firestore

Setelah data diterima oleh aplikasi Android, data fisiologis mentah dikirimkan ke *Firebase Cloud Firestore* menggunakan *Firebase Firestore SDK* untuk Android. Komunikasi berlangsung melalui protokol HTTPS yang dikelola secara otomatis oleh *Firebase SDK*, sehingga seluruh data yang dikirimkan terenkripsi selama proses transmisi.

Data dikirim secara asinkron menggunakan pendekatan *non-blocking*, artinya proses pengiriman ke *cloud* tidak menghambat jalannya proses lain pada aplikasi Android, termasuk proses inferensi model GRU dan pembaruan tampilan antarmuka pengguna. Setiap pengiriman data menghasilkan satu dokumen baru pada koleksi *raw_data_skripsi* di Firestore dengan struktur sebagai berikut:

Tabel 4. 1 Susunan *field* data raw pada firebase

No	<i>Field</i>	Tipe Data	Keterangan
1	hr	<i>Integer</i>	Nilai detak jantung dalam BPM
2	ax	<i>Float</i>	Percepatan sumbu X (m/s ²)
3	ay	<i>Float</i>	Percepatan sumbu Y (m/s ²)
4	az	<i>Float</i>	Percepatan sumbu Z (m/s ²)
5	steps	<i>Integer</i>	Jumlah langkah kaki
6	ts	<i>Timestamp</i>	Waktu pengambilan data per sampel
7	end time	<i>Timestamp</i>	Waktu akhir sesi pengambilan data

Struktur pada tabel 4.1 ini konsisten dengan perancangan basis data yang telah dijelaskan pada subbab 4.1.3. Selain koleksi *raw_data_skripsi*, aplikasi Android juga memperbarui dokumen pada koleksi *users* setiap sesi selesai, mencakup field *last_heart_rate*, *last_steps*, *calories*, dan *last_updated* sebagai ringkasan data terkini pengguna.

C. Data yang Dikirim dan Tidak Dikirim ke Cloud

Tidak semua data yang diproses oleh sistem dikirimkan ke *Firestore*. Tabel 4.2 berikut merangkum data yang dikirim dan tidak dikirim ke *cloud* beserta alasannya:

Tabel 4. 2 Data yang dikirim dan tidak dikirim ke cloud

No	Data	Dikirim ke Cloud	Keterangan
1	HR mentah (BPM)	✓ Ya	Disimpan pada koleksi <i>raw_data_skripsi</i>
2	ACC X, Y, Z	✓ Ya	Disimpan pada koleksi <i>raw_data_skripsi</i>
3	Jumlah langkah	✓ Ya	Disimpan pada koleksi <i>raw_data_skripsi</i>

4	Timestamp sesi	✓ Ya	Disimpan pada koleksi raw data skripsi
5	Ringkasan data pengguna	✓ Ya	Diperbarui pada koleksi users
6	Hasil prediksi GRU (label & probabilitas)	✗ Tidak	Diproses dan disimpan secara lokal (<i>on-device</i>)
7	Fitur hasil ekstraksi (13 fitur statistik)	✗ Tidak	Digunakan hanya untuk inferensi lokal

Hasil prediksi model GRU tidak dikirimkan ke *cloud* karena seluruh proses inferensi dilakukan secara lokal pada perangkat Android menggunakan pendekatan *on-device inference*. Pendekatan ini memberikan dua keuntungan utama: pertama, sistem tetap dapat memberikan analisis kondisi pengguna meskipun koneksi internet tidak tersedia; kedua, privasi pengguna lebih terjaga karena data yang paling sensitif, yaitu hasil analisis kondisi fisiologis, tidak meninggalkan perangkat pengguna.

D. Mekanisme *On-Device Inference* sebagai Bagian dari Arsitektur IoT

Proses inferensi model GRU dilakukan sepenuhnya di perangkat Android menggunakan TensorFlow Lite (TFLite). Mekanisme ini berperan sebagai bagian dari lapisan aplikasi (*application layer*) dalam arsitektur IoT sistem ini. Model .tflite dimuat ke dalam memori aplikasi pada saat aplikasi diinisialisasi, sehingga tersedia untuk digunakan kapan pun dibutuhkan tanpa memerlukan pemuatan ulang dari server.

Alur proses inferensi pada aplikasi Android adalah sebagai berikut:

1. Data sensor yang diterima dari *smartwatch* dikumpulkan selama satu *window* (60 detik).
2. Data distandarisasi menggunakan parameter Z-Score (*mean* dan *std*) yang disimpan sebagai file pendamping model.

3. Sebanyak 13 fitur statistik diekstraksi dari setiap *window* dan diatur dalam format tensor $[1 \times \text{timesteps} \times 13]$.
4. Tensor dimasukkan ke dalam TFLite Interpreter yang menjalankan model GRU.
5. Model menghasilkan vektor probabilitas tiga kelas: *Baseline*, *Stress*, dan *Amusement*.
6. Kelas dengan probabilitas tertinggi dipilih sebagai hasil prediksi (*argmax*) dan ditampilkan pada antarmuka pengguna.

Dengan mekanisme ini, model GRU berfungsi secara mandiri di sisi perangkat (*edge computing*), sementara Firebase Firestore berperan sebagai penyimpanan riwayat data yang diakses secara asinkron. Kombinasi keduanya membentuk arsitektur IoT yang seimbang antara kecepatan respons lokal dan persistensi data jangka panjang di *cloud*.

E. Alur Kerja Integrasi IoT Secara Keseluruhan

Berikut adalah alur kerja integrasi IoT sistem secara lengkap dari awal hingga akhir:

1. Akuisisi data: *Smartwatch* membaca nilai HR dan ACC secara kontinu melalui sensor bawaan perangkat.
2. Pengiriman ke Android: Aplikasi Wear OS pada *smartwatch* mengirim data ke aplikasi Android melalui MessageClient (Wear OS Data Layer API) menggunakan koneksi Bluetooth.
3. Penerimaan dan penyimpanan sementara: Aplikasi Android menerima data dan menyimpannya sementara di memori lokal selama satu sesi pengukuran.

4. *Preprocessing* dan ekstraksi fitur: Setelah data terkumpul selama 60 detik, aplikasi menjalankan standardisasi Z-Score dan mengekstraksi 13 fitur statistik dari *window* tersebut.
5. Inferensi model GRU: Fitur hasil ekstraksi dimasukkan ke model GRU dalam format TFLite yang berjalan secara *on-device*. Model menghasilkan vektor probabilitas dan label kelas dipilih berdasarkan nilai tertinggi.
6. Tampilan hasil: Hasil prediksi ditampilkan pada antarmuka aplikasi Android dalam bentuk label kondisi (*Baseline / Stress / Amusement*), nilai probabilitas masing-masing kelas, serta rekomendasi kesehatan adaptif.
7. Pengiriman ke cloud: Secara asinkron dan terpisah dari proses inferensi, data sensor mentah dikirim ke *Firebase Cloud Firestore* melalui HTTPS menggunakan *Firebase SDK* untuk disimpan sebagai riwayat kesehatan pengguna.

Alur kerja ini memastikan bahwa sistem dapat memberikan hasil analisis secara cepat kepada pengguna melalui *on-device inference*, sekaligus menyimpan data secara terpusat di *cloud* untuk keperluan pemantauan riwayat kesehatan jangka panjang.

4.1.6 Implementasi Model GRU

A. Penentuan Fitur Input Model

Fitur input yang digunakan dalam model GRU berasal dari data sensor yang telah melalui tahap *preprocessing*. Fitur input yang digunakan pada model GRU terdiri dari data *accelerometer* pada sumbu X, Y, dan Z, serta data *Heart Rate* (HR).

Kombinasi fitur ini digunakan untuk membedakan kondisi yang memiliki pola gerakan serupa namun kondisi fisiologis yang berbeda.

Data sensor disusun dalam bentuk deret waktu menggunakan metode *sliding window*, di mana setiap segmen data merepresentasikan interval waktu tertentu. Setiap segmen dilengkapi dengan label kondisi, yaitu 0 (*baseline*), 1 (*stress*) dan 2 (*amusement*). Penyusunan fitur input ini bertujuan agar model GRU dapat mempelajari pola temporal dari masing-masing kondisi secara lebih efektif.

Pada Tabel 4.3 menunjukkan semua sensor input yang digunakan untuk melatih model GRU. Sensor *Accelerometer* merepresentasikan percepatan gerak pergelangan tangan dalam tiga sumbu yaitu X, Y dan Z, sementara HR digunakan untuk membedakan kondisi dengan pola gerakan serupa tetapi kondisi fisiologis berbeda. *Timestamp* hanya digunakan untuk menjaga urutan data deret waktu.

Tabel 4. 3 Fitur Input Model

No	Nama Fitur	Satuan	Sumber Sensor	Keterangan
1	Mean HR	BPM	<i>Heart Rate</i> (HR)	Rata-rata detak jantung dalam 60 detik
2	Std HR	BPM	<i>Heart Rate</i> (HR)	Variasi/fluktuasi detak jantung
3	Min HR	BPM	<i>Heart Rate</i> (HR)	Titik detak jantung terendah
4	Max HR	BPM	<i>Heart Rate</i> (HR)	Titik detak jantung tertinggi
5	<i>Range</i> HR	BPM	<i>Heart Rate</i> (HR)	Selisih antara Max dan Min HR
6	Median HR	BPM	<i>Heart Rate</i> (HR)	Nilai tengah dari data detak jantung
7	RMSSD	ms	<i>Blood Volume Pulse</i> (BVP)	Indikator variabilitas detak jantung jangka pendek (HRV)
8	Mean ACC	m/s ²	<i>Accelerometer</i> (Mag)	Rata-rata kekuatan gerak (<i>magnitude</i>)
9	Std ACC	m/s ²	<i>Accelerometer</i> (Mag)	Intensitas getaran/tremor pada tangan
10	Min ACC	m/s ²	<i>Accelerometer</i> (Mag)	Gerakan minimal yang terdeteksi
11	Max ACC	m/s ²	<i>Accelerometer</i> (Mag)	Gerakan maksimal (lonjakan gerak)

12	Range ACC	m/s ²	Accelerometer (Mag)	Rentang intensitas pergerakan
13	Median ACC	m/s ²	Accelerometer (Mag)	Nilai tengah dari intensitas gerak

Pada tabel 4.4 menjelaskan parameter segmentasi data menggunakan metode *sliding window*. *Window size* 60 detik dipilih untuk menangkap pola kondisi secara utuh, baik pada kondisi dengan gerakan minimal maupun kondisi yang memiliki fluktuasi HR tinggi seperti stres/tegang. *Overlap* ~92% memastikan pola transisi antar segmen tidak hilang, sedangkan jumlah sampel per window bersifat dinamis mengikuti sistem Wear OS.

Tabel 4.4 Parameter penelitian

No	Parameter	Nilai	Keterangan
1	<i>Window Size</i>	60 detik	Panjang satu segmen data
2	<i>Sampling Rate</i>	Dinamis	Menyesuaikan sistem <i>Wear OS</i>
3	Jumlah Sampel per <i>Window</i>	300 sampel	Total data dalam satu kali perhitungan fitur
4	<i>Overlap</i>	~92%	Tumpang tindih antar window
5	Jumlah Fitur	13	Mean HR, Std HR, Min HR, Max HR, Range HR, Median HR, RMSSD, Mean ACC, Std ACC, Min ACC, Max ACC, Range ACC, Median ACC

Dengan pengaturan fitur dan parameter *window* pada tabel 4.2 di atas, setiap segmen data siap digunakan sebagai input ke model GRU untuk melakukan klasifikasi tiga kondisi subjek secara akurat dan konsisten.

B. Penyusunan Data Deret Waktu

Setelah fitur input dan parameter *sliding window* ditentukan, langkah berikutnya adalah menyusun data sensor menjadi bentuk deret waktu (*time-series*)

yang sesuai dengan kebutuhan model GRU. Penyusunan ini bertujuan agar model dapat mempelajari pola temporal dari setiap aktivitas secara efektif.

1. Segmentasi Data Menggunakan *Sliding Window*

Data sensor dipecah menjadi beberapa segmen berdasarkan *window size* yang telah ditentukan, yaitu 60 detik, dengan *overlap* ~92% antar segmen. Setiap segmen mewakili satu unit input untuk model GRU dan dilengkapi dengan label kondisi yang sesuai (*baseline*, *stress* dan *amusement*).

- 1) *Window Size*: 60 detik
- 2) *Overlap*: ~92%
- 3) Jumlah Fitur: 13 (Mean HR, Std HR, Min HR, Max HR, Range HR, Median HR, RMSSD, Mean ACC, Std ACC, Min ACC, Max ACC, Range ACC, Median ACC)
- 4) Jumlah Sampel per *Window*: 300 sampel

Dengan metode ini, setiap segmen data berbentuk matriks dengan dimensi:

$$[\text{jumlah sampel per window}] \times [\text{jumlah fitur}]$$

Setiap baris mewakili satu *timestamp* data, sedangkan setiap kolom mewakili fitur sensor.

2. Pembagian Data Pelatihan dan Pengujian

Untuk melatih dan mengevaluasi model GRU secara objektif, data yang telah disegmentasi dibagi menjadi:

- a) Data pelatihan (*training set*): 80% dari seluruh segmen
- b) Data pengujian (*testing set*): 20% dari seluruh segmen

Pembagian ini dilakukan secara acak namun tetap menjaga distribusi tiap aktivitas agar masing-masing kelas terwakili dengan proporsional.

3. Standardisasi Data

Sebelum dimasukkan ke dalam model, data sensor distandardisasi untuk memastikan bahwa skala fitur yang berbeda tidak mempengaruhi pelatihan model, yaitu sensor detak jantung dan akselerometer. Standardisasi ini membantu GRU mempelajari pola temporal lebih cepat dan stabil.

4. Contoh Data Awal sebelum Ekstraksi Fitur

Data yang diperoleh dari sensor *smartwatch* masih berupa data mentah (*raw data*) hasil akuisisi secara *real-time* yang mencakup data *accelerometer* (Acc X, Acc Y, dan Acc Z) serta *heart rate* (HR). Data tersebut belum dapat langsung digunakan sebagai masukan model GRU karena masih memerlukan tahapan *preprocessing* dan ekstraksi fitur. Contoh bentuk data awal sebelum proses ekstraksi fitur ditunjukkan pada Tabel 4.5.

Tabel 4. 5 Contoh data awal sebelum ekstraksi fitur

Time Step	Acc X	Acc Y	Acc Z	HR
2026-02-09 01:49:44.304579+00:00	6.708553	-6.67503	-2.45645	84
2026-02-09 01:49:44.306247+00:00	6.708553	-6.67503	-2.45645	84
2026-02-09 01:49:44.307585+00:00	6.708553	-6.67503	-2.45645	84
2026-02-09 01:49:44.381301+00:00	6.761226	-6.64391	-2.35829	77
2026-02-09 01:49:44.382437+00:00	6.761226	-6.64391	-2.35829	77

C. Perancangan Arsitektur Model GRU

Setelah data disusun dalam bentuk deret waktu menggunakan metode *sliding window*, tahap selanjutnya adalah merancang arsitektur model GRU (*Gated Recurrent Unit*) yang digunakan untuk melakukan klasifikasi kondisi. Model GRU dipilih karena mampu mempelajari ketergantungan temporal pada data deret waktu dengan kompleksitas yang lebih rendah dibandingkan LSTM, sehingga sesuai untuk data sensor *wearable*.

1. Struktur Input Model

Input model GRU berupa data deret waktu hasil segmentasi *sliding window* dengan karakteristik sebagai berikut:

- a) Panjang window: 60 detik
- b) Jumlah fitur: 13 (Mean HR, Std HR, Min HR, Max HR, Range HR, Median HR, RMSSD, Mean ACC, Std ACC, Min ACC, Max ACC, Range ACC, Median ACC)
- c) Bentuk input:
(time steps, jumlah fitur)

Setiap window merepresentasikan 13 fitur dan digunakan sebagai input ke dalam model GRU.

2. Lapisan GRU

Lapisan utama pada model adalah GRU layer yang berfungsi untuk mengekstraksi pola temporal dari data sensor. GRU menggunakan mekanisme update gate dan reset gate untuk menentukan informasi mana yang dipertahankan atau dilupakan pada setiap langkah waktu.

Secara umum, proses ini memungkinkan model:

- a) Mengingat pola gerakan berulang,
- b) Menangkap perubahan kondisi fisiologis melebihi nilai *heart rate* (HR) pada kondisi stres/tegang,
- c) Mengabaikan *noise* yang tidak relevan pada data sensor.

3. Lapisan *Fully Connected (Dense Layer)*

Output dari lapisan GRU diteruskan ke lapisan *fully connected (Dense)* yang berfungsi sebagai pengklasifikasi. Lapisan ini mengubah representasi temporal menjadi probabilitas masing-masing kelas kondisi.

- a) Jumlah neuron output: 3
- b) Fungsi aktivasi: *Softmax*

Ketiga neuron output merepresentasikan kelas kondisi:

- 1. *Baseline*
- 2. *Stress*
- 3. *Amusement*

Nilai output berupa probabilitas, dan kelas dengan probabilitas tertinggi dipilih sebagai hasil prediksi aktivitas.

4. Alur Kerja Model GRU

Secara umum, alur kerja model dapat dijelaskan sebagai berikut:

- a) Data sensor hasil preprocessing disegmentasi menjadi window 60 detik
- b) Setiap window dimasukkan ke lapisan GRU sebagai data deret waktu
- c) GRU mengekstraksi pola temporal dari data sensor
- d) Output GRU diteruskan ke *Dense layer*

e) Model menghasilkan prediksi kelas kondisi

Model GRU tidak secara langsung menghasilkan label kelas sebagai keluaran akhir, melainkan menghasilkan vektor probabilitas yang merepresentasikan tingkat keyakinan model terhadap masing-masing kelas kondisi. Vektor probabilitas ini dihasilkan oleh fungsi aktivasi softmax pada lapisan Dense di akhir arsitektur model, dengan jumlah nilai sebanyak tiga buah yang masing-masing merepresentasikan peluang input termasuk dalam kelas Baseline (indeks 0), Stress (indeks 1), dan Amusement (indeks 2). Ketiga nilai probabilitas tersebut selalu berjumlah 1,0 karena merupakan distribusi probabilitas yang dinormalisasi oleh fungsi softmax.

Sebagai contoh, apabila model menghasilkan output [0.72, 0.21, 0.07], maka dapat diinterpretasikan bahwa model memperkirakan probabilitas kondisi Baseline sebesar 72%, Stress sebesar 21%, dan Amusement sebesar 7%. Nilai-nilai inilah yang ditampilkan pada antarmuka aplikasi Android sebagai informasi probabilitas kondisi kepada pengguna.

Kelas yang dipilih sebagai hasil prediksi akhir adalah kelas dengan nilai probabilitas tertinggi, menggunakan operasi argmax:

$$\text{predicted_class} = \text{argmax}([0.72, 0.21, 0.07]) = 0 \text{ (Baseline)}$$

Pendekatan ini merupakan standar pada sistem klasifikasi multi-kelas berbasis deep learning, di mana lapisan softmax menghasilkan distribusi probabilitas dan kelas dengan probabilitas tertinggi dipilih sebagai prediksi akhir (Goodfellow et al., 2016). Pendekatan serupa juga digunakan pada penelitian deteksi kondisi fisiologis

berbasis sensor yang menggunakan dataset WESAD, termasuk Bajpai et al. (2023) dan Melchiades et al. (2022).

Fungsi model GRU berakhir pada tahap menghasilkan vektor probabilitas ini. Seluruh proses selanjutnya, yaitu penentuan label kondisi, pemberian rekomendasi kesehatan adaptif, penampilan visualisasi, dan pengiriman notifikasi kepada pengguna, dilakukan oleh lapisan aplikasi Android di luar model GRU itu sendiri.

5. Alasan Pemilihan Arsitektur GRU

Arsitektur GRU dengan satu lapisan GRU dan satu lapisan *Dense* dipilih karena:

- a) Cukup untuk menangkap pola aktivitas berbasis *wearable*,
- b) mengurangi risiko *overfitting* pada jumlah data terbatas,
- c) lebih efisien secara komputasi dibandingkan model yang lebih kompleks.

Dengan arsitektur tersebut, model GRU diharapkan mampu mempelajari pola temporal dari kombinasi data *accelerometer* dan *heart rate* untuk mengklasifikasikan tiga jenis kondisi subjek secara akurat.

D. Proses Pelatihan Model GRU

Setelah arsitektur model GRU dirancang, tahap selanjutnya adalah melakukan proses pelatihan (*training*) untuk memperoleh model yang mampu mengklasifikasikan kondisi secara akurat. Proses pelatihan dilakukan menggunakan data hasil segmentasi *sliding window* yang telah dibagi menjadi data pelatihan dan data pengujian.

1. Pembagian Data Pelatihan dan Pengujian

Data yang telah disusun dalam bentuk *window* dibagi menjadi dua bagian, yaitu:

- a) Data pelatihan (*training set*): digunakan untuk melatih model GRU
- b) Data pengujian (*testing set*): digunakan untuk mengevaluasi kinerja model

Proporsi pembagian data yang digunakan adalah:

- a) 80% data pelatihan
- b) 20% data pengujian

Pembagian data dilakukan secara acak dengan tetap menjaga distribusi masing-masing kelas kondisi agar tidak terjadi ketimpangan kelas (*class imbalance*).

2. Konfigurasi Parameter Pelatihan

Tabel 4. 6 Konfigurasi Parameter Pelatihan

No	Parameter	Nilai	Keterangan
1	<i>Optimizer</i>	Adam	<i>Optimizer</i> adaptif untuk pembaruan bobot
2	<i>Learning Rate</i>	0.001	Nilai standar untuk stabilitas pelatihan
3	<i>Loss Function</i>	<i>Sparse Categorical Cross-Entropy</i>	Digunakan untuk klasifikasi multi-kelas
4	<i>Batch Size</i>	64	Jumlah sampel per iterasi pelatihan
5	<i>Epoch</i>	Maksimum 200 <i>Epoch</i> dengan <i>Early Stopping</i>	Jumlah pengulangan pelatihan
6	<i>Validation Split</i>	20%	Data validasi dari data training
7	<i>Dropout</i>	0.3	Regularisasi untuk mencegah <i>overfitting</i> dengan menonaktifkan 30% neuron secara acak saat pelatihan
8	GRU <i>units</i>	128	Jumlah unit/sel pada lapisan GRU untuk mengekstraksi pola temporal

3. Proses Pelatihan Model

Proses pelatihan dilakukan secara iteratif selama sejumlah *epoch* yang telah ditentukan. Pada setiap *epoch*, model GRU melakukan:

- a) *Forward propagation* untuk menghasilkan prediksi
- b) Perhitungan *loss* berdasarkan perbedaan antara prediksi dan label actual
- c) *Backward propagation* untuk memperbarui bobot model menggunakan optimizer Adam

Proses ini berlangsung hingga model mencapai konvergensi atau jumlah *epoch* maksimum tercapai.

4. Pencegahan *Overfitting*

Untuk mencegah terjadinya *overfitting*, beberapa strategi diterapkan, antara lain:

- a. Penggunaan *validation set*
- b. Pemilihan arsitektur GRU yang sederhana bertujuan untuk membatasi kapasitas model, sehingga model tidak mudah menghafal data training (*overfitting*) dan lebih mampu menangkap pola umum yang dapat digeneralisasi ke data baru
- c. Penghentian pelatihan saat performa validasi tidak meningkat secara signifikan

Langkah-langkah ini bertujuan agar model memiliki kemampuan generalisasi yang baik terhadap data baru.

5. Output Model

Model GRU hasil pelatihan disimpan dan digunakan pada tahap pengujian untuk mengklasifikasikan kondisi subjek berdasarkan data sensor yang diterima. Melalui proses pelatihan ini, model GRU diharapkan mampu mempelajari hubungan temporal antara data akselerometer dan detak jantung sehingga dapat menghasilkan prediksi kondisi yang akurat dan konsisten.

E. Evaluasi dan Hasil Kinerja Model GRU

Setelah proses pelatihan selesai, model GRU dievaluasi untuk mengetahui kemampuan klasifikasi kondisi subjek berdasarkan data sensor *smartwatch*. Evaluasi dilakukan menggunakan data pengujian yang sebelumnya telah dipisahkan dari data pelatihan dengan proporsi 80% untuk training dan 20% untuk testing. Beberapa metrik digunakan untuk menilai kinerja model, yaitu akurasi, *precision*, *recall*, dan *F1-score*. Akurasi menunjukkan persentase prediksi yang benar dari seluruh sampel, sedangkan *precision* dan *recall* mengukur kemampuan model dalam memprediksi setiap kelas kondisi secara spesifik. *F1-score* digunakan sebagai kombinasi dari *precision* dan *recall* untuk memberikan gambaran keseluruhan mengenai performa model per kelas kondisi.

Nilai metrik evaluasi diperoleh dari proses pengujian model menggunakan testing set sebanyak 879 sampel (20% dari total data). Berikut ilustrasi perhitungan untuk kelas Stress sebagai contoh berdasarkan confusion matrix yang diperoleh:

- a. TP (*True Positive*, diprediksi Stress dan benar) = 241
- b. FP (*False Positive*, diprediksi Stress tetapi bukan Stress) = 13
- c. FN (*False Negative*, sebenarnya Stress tetapi diprediksi kelas lain) = 21

d. $TN \text{ (True Negative)} = 604$

Berdasarkan nilai tersebut, metrik per kelas dihitung sebagai berikut:

a. $Precision = TP / (TP + FP) = 241 / (241 + 13) = 0,95$

b. $Recall = TP / (TP + FN) = 241 / (241 + 21) = 0,92$

c. $F1\text{-Score} = 2 \times (0,95 \times 0,92) / (0,95 + 0,92) = 0,93$

Akurasi keseluruhan dihitung berdasarkan jumlah seluruh prediksi yang benar dari semua kelas dibagi total sampel, yaitu $826 / 879 \approx 0,94$ (94%). Nilai *macro average* merupakan rata-rata tidak tertimbang dari nilai per kelas, sedangkan *weighted average* mempertimbangkan jumlah sampel tiap kelas (*support*), sehingga kelas *baseline* dengan 469 sampel memiliki bobot lebih besar dalam perhitungan *weighted average*. Seluruh perhitungan dilakukan secara otomatis menggunakan fungsi `classification_report` dari pustaka `scikit-learn` pada Python berdasarkan matriks konfusi hasil prediksi model pada testing set.

Hasil evaluasi menunjukkan bahwa model GRU mampu membedakan tiga kondisi yang ditentukan, yaitu *baseline*, *stress* dan *amusement*, dengan akurasi tinggi. Kondisi yang pola gerakannya mirip tetapi memiliki perbedaan kondisi fisiologis, seperti *baseline* dan *stress*, juga dapat dibedakan meskipun akurasi sedikit lebih rendah. Hal ini menunjukkan bahwa kombinasi fitur detak jantung dan akselerometer memungkinkan model GRU menangkap perbedaan pola temporal dan fisiologis antar kondisi.

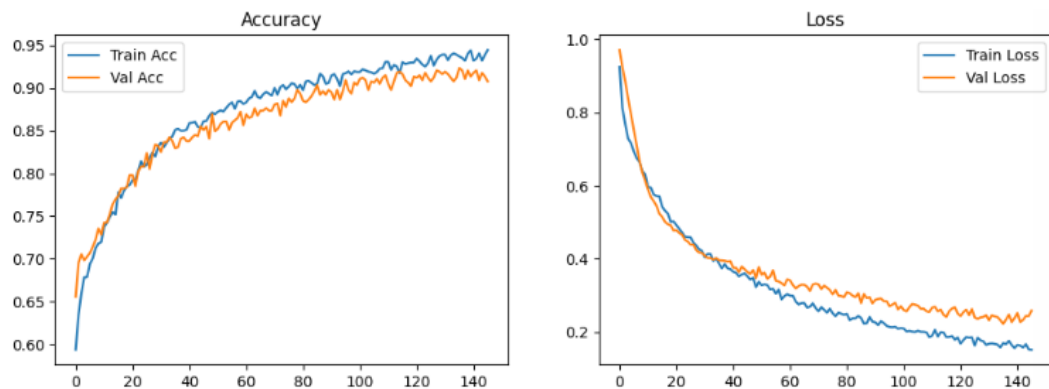
Tabel 4. 7 Hasil Evaluasi Model GRU

	<i>Precision</i>	<i>Recall</i>	<i>F1-Score</i>	<i>Support</i>
<i>Baseline</i>	0.95	0.97	0.96	469
<i>Stress</i>	0.95	0.92	0.93	262
<i>Amusement</i>	0.88	0.86	0.87	148
<i>Accuracy</i>			0.94	879
<i>Macro avg</i>	0.92	0.92	0.92	879
<i>Weighted avg</i>	0.93	0.94	0.93	879

Berdasarkan hasil evaluasi pada Tabel 4.5, model *Gated Recurrent Unit* (GRU) memperoleh nilai *accuracy* sebesar 94%, yang menunjukkan bahwa model memiliki kemampuan klasifikasi yang baik dalam mengenali kondisi fisiologis pengguna. Nilai *precision*, *recall*, dan *F1-score* pada kelas *baseline* masing-masing sebesar 0.95, 0.97, dan 0.96, menunjukkan bahwa model mampu mengenali kondisi normal dengan tingkat konsistensi yang tinggi. Pada kelas *stress*, model memperoleh *precision* sebesar 0.95 dan *recall* sebesar 0.92, yang mengindikasikan bahwa pola fisiologis stres dapat dikenali secara cukup efektif oleh model.

Sementara itu, kelas *amusement* memiliki nilai *recall* yang lebih rendah yaitu sebesar 0.86 dibandingkan kelas lainnya. Kondisi ini menunjukkan bahwa sebagian data *amusement* masih mengalami salah klasifikasi ke kelas lain. Hal tersebut kemungkinan disebabkan oleh adanya kemiripan pola fisiologis antara kondisi *amusement* dan *baseline* maupun *stress*, terutama ketika perubahan detak jantung dan aktivitas tubuh tidak menunjukkan perbedaan yang signifikan. Selain itu, jumlah data pada kelas *amusement* yang lebih sedikit dibandingkan kelas lainnya juga dapat memengaruhi kemampuan model dalam mempelajari pola secara optimal.

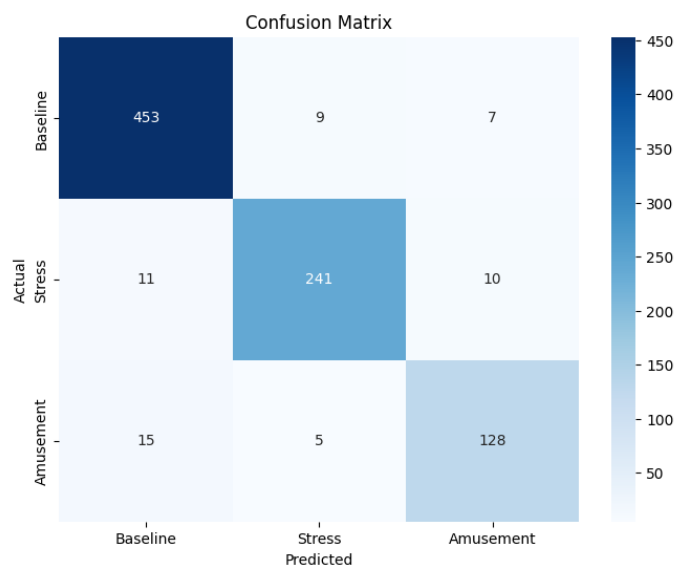
Secara keseluruhan, model GRU memperoleh nilai *weighted average precision* sebesar 0.93, *weighted average recall* sebesar 0.94, dan *weighted average F1-score* sebesar 0.93. Pada tingkat per kelas, model mencapai *F1-score* tertinggi pada kelas *baseline* sebesar 0.96, diikuti kelas *stress* sebesar 0.93, dan kelas *amusement* sebesar 0.87. Perbedaan performa antar kelas ini sejalan dengan distribusi data yang tidak seimbang, di mana kelas *amusement* memiliki jumlah sampel paling sedikit yakni 148 sampel dibandingkan *baseline* dengan 469 sampel dan *stress* dengan 262 sampel.



Gambar 4. 2 Grafik *Accuracy* dan *Loss*

Berdasarkan hasil pelatihan tersebut, model GRU dapat dikatakan memiliki performa yang baik dalam mengenali pola temporal dari data *time-series*. Hal ini dibuktikan dengan nilai training *accuracy* yang mencapai sekitar 94% dan *validation accuracy* sekitar 91% pada akhir pelatihan, yang telah memenuhi target performa penelitian yang ditetapkan sebelumnya (minimal 85%). Pada *epoch-epoch* awal, *validation accuracy* sempat lebih tinggi dibandingkan training *accuracy*, yang merupakan perilaku umum akibat penerapan *Dropout* selama proses pelatihan — lapisan *Dropout* aktif menekan performa saat training namun

dinonaktifkan saat validasi. Kurva *accuracy* yang terus meningkat dan kurva *loss* yang menurun secara stabil hingga *training loss* ~ 0.17 dan *validation loss* ~ 0.25 menunjukkan bahwa model berhasil mempelajari karakteristik data tanpa mengalami masalah *underfitting* maupun *overfitting* yang signifikan.



Gambar 4. 3 *Confusion Matrix*

Berdasar pada *confusion matrix* di atas, sebagian besar data berhasil diklasifikasikan dengan benar pada masing-masing kelas kondisi. Kesalahan klasifikasi masih ditemukan pada beberapa data dengan pola fisiologis yang mirip. Hasil evaluasi tersebut menunjukkan bahwa model GRU efektif digunakan untuk sistem monitoring kondisi berbasis *smartwatch* dan mampu melakukan klasifikasi kondisi pengguna dengan performa yang baik.

4.1.7 Implementasi Antarmuka

Tahapan ini adalah tahapan implementasi desain antarmuka yang sudah dirancang. Berikut merupakan implementasi *interface* pada penelitian ini:

A. Implementasi tampilan *smartwatch*

Halaman pengambilan data pada perangkat *smartwatch* dirancang untuk menampilkan metrik kesehatan dan data sensor secara *real-time* langsung dari pergelangan tangan pengguna. Visualisasi dari implementasi antarmuka halaman pengambilan data ini dapat dilihat pada Gambar 4.4 di bawah ini.



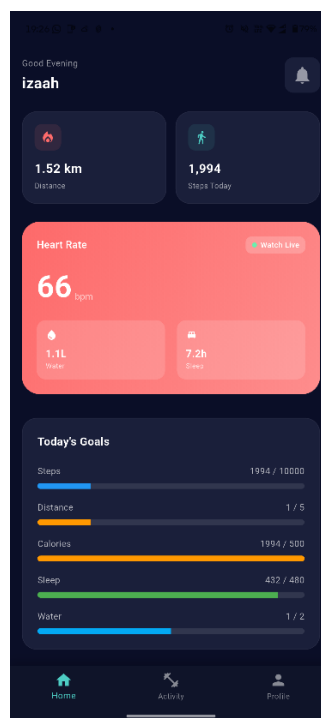
Gambar 4. 4 Tampilan *Smartwatch*

Panel pertama (sebelah kiri) merupakan halaman pemantauan detak jantung (*Heart Rate*) yang menampilkan angka *Beats Per Minute* (BPM) secara aktual dilengkapi dengan indikator status "LIVE". Sementara itu, panel kedua (sebelah kanan) merupakan halaman *Accelerometer* yang memvisualisasikan grafik pergerakan sensor spasial pada sumbu X, Y, dan Z dalam bentuk *progress bar* berwarna serta nilai numeriknya.

B. Implementasi halaman Dashboard

Halaman utama (*home*) berfungsi sebagai pusat monitoring pada sistem pemantauan kesehatan berkelanjutan berbasis *smartwatch* Android. Halaman ini menampilkan parameter fisiologis pengguna secara terintegrasi, seperti detak

jantung, jumlah langkah, dan kalori yang terbakar secara langsung dari data sensor *smartwatch*. Selain itu, halaman utama juga menampilkan grafik aktivitas harian, hasil prediksi kondisi fisiologis dari model GRU, serta notifikasi rekomendasi kesehatan adaptif. Tampilan pada gambar 4.5 dirancang untuk membantu pengguna memantau kondisi tubuh secara lebih mudah, cepat, dan informatif sehingga pengguna dapat merespons perubahan kondisi fisiologis secara mandiri.

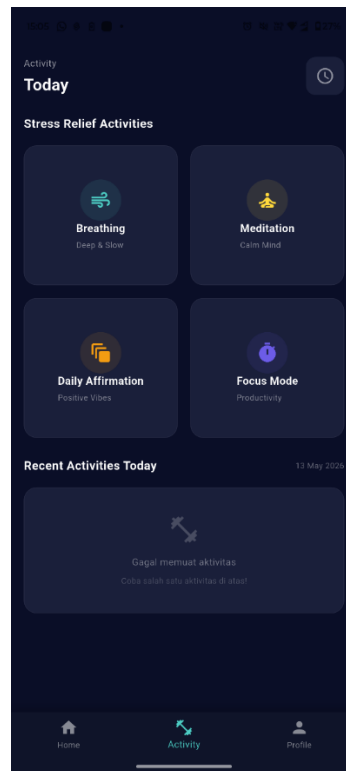


Gambar 4.5 Halaman Dashboard

C. Implementasi halaman *Activity*

Halaman *Activity* berfungsi sebagai media pendukung manajemen aktivitas dan relaksasi pengguna dalam sistem pemantauan kesehatan berbasis *smartwatch* Android. Pada gambar 4.6 halaman ini menampilkan berbagai fitur aktivitas pendukung kesehatan mental dan fisik, seperti *breathing exercise*, meditasi,

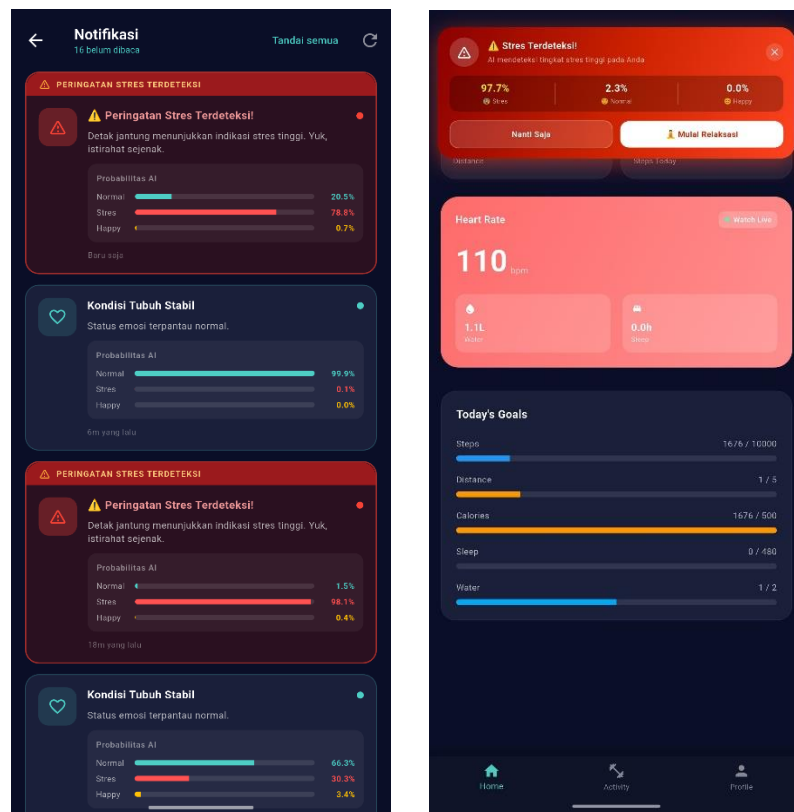
afirmasi harian, dan *focus mode* yang dapat digunakan pengguna untuk membantu mengurangi stres dan menjaga kestabilan kondisi tubuh.



Gambar 4.6 Halaman *Activity*

D. Implementasi halaman notifikasi

Halaman notifikasi digunakan untuk memberikan informasi terkait kondisi kesehatan pengguna berdasarkan hasil pemantauan sistem. Sistem menampilkan notifikasi berupa peringatan ketika terdeteksi indikasi stres tinggi serta informasi kondisi tubuh stabil apabila parameter kesehatan berada dalam kondisi normal. Informasi yang ditampilkan pada gambar 4.7 mencakup probabilitas hasil analisis kecerdasan buatan terhadap kondisi emosional pengguna, sehingga dapat membantu pengguna dalam memantau kondisi kesehatan secara berkala.



Gambar 4.7 Halaman Notifikasi

4.1.8 Hasil Uji Coba

Tahap hasil uji coba dilakukan untuk memastikan bahwa sistem monitoring aktivitas berbasis *smartwatch* yang dikembangkan dapat berjalan dengan baik, baik dari sisi fungsionalitas sistem, performa model klasifikasi GRU, maupun tingkat kenyamanan penggunaan aplikasi oleh pengguna. Pengujian dilakukan secara menyeluruh mulai dari proses pengambilan data sensor, pengiriman data ke perangkat *smartphone*, proses klasifikasi kondisi menggunakan model GRU, hingga tampilan hasil prediksi pada antarmuka aplikasi.

A. Pengujian Fungsional Sistem

Pengujian fungsional dilakukan untuk memastikan setiap komponen sistem berjalan sesuai dengan perancangan. Pengujian mencakup proses pengambilan data akselerometer dan detak jantung pada *smartwatch*, pengiriman data ke *smartphone*, penyimpanan data pada database, serta proses klasifikasi kondisi menggunakan model GRU.

Pengujian dilakukan dengan metode *black-box testing*, yaitu menguji fungsi sistem tanpa melihat kode program secara langsung. Hasil pengujian menunjukkan bahwa:

1. *Smartwatch* berhasil merekam data *accelerometer* (sumbu X, Y, Z) dan *heart rate* secara kontinu.
2. Data sensor berhasil dikirim dari *smartwatch* ke *smartphone* melalui koneksi *Wear OS* tanpa kehilangan data signifikan.
3. Data berhasil diproses menggunakan metode *sliding window*.
4. Model GRU dapat menghasilkan prediksi kondisi.
5. Hasil prediksi ditampilkan dengan benar pada antarmuka aplikasi.

Berikut ringkasan hasil pengujian fungsional:

Tabel 4.8 Hasil Pengujian Fungsional

No	Fitur yang Diuji	Hasil	Keterangan
1	Pengambilan data <i>accelerometer</i>	Berhasil	Data terbaca normal
2	Pengambilan data <i>heart rate</i>	Berhasil	Data BPM sesuai kondisi
3	Pengiriman data ke <i>smartphone</i>	Berhasil	Tidak ada error koneksi
4	Proses klasifikasi GRU	Berhasil	Prediksi muncul
5	Tampilan hasil prediksi	Berhasil	Sesuai kondisi

Berdasarkan pengujian pada tabel 4.8 tersebut, sistem dinyatakan berjalan sesuai dengan spesifikasi yang dirancang.

B. Pengujian Performa Model pada Kondisi Nyata

Pengujian performa dilakukan dengan meminta subjek melakukan tiga kondisi aktivitas yang telah ditentukan, yaitu kondisi netral/santai, kondisi stres/tegang, dan aktivitas fisik ringan. Model kemudian memprediksi kondisi aktivitas berdasarkan data sensor yang diterima secara *real-time* dari *smartwatch*.

Hasil pengujian menunjukkan bahwa model *Gated Recurrent Unit* (GRU) mampu mengenali ketiga kondisi tersebut dengan tingkat akurasi yang baik. Aktivitas fisik ringan memiliki pola gerakan yang lebih jelas sehingga dapat dikenali dengan baik berdasarkan data *Accelerometer*. Sementara itu, kondisi netral dan stres memiliki pola gerakan yang relatif mirip, sehingga perbedaan keduanya lebih banyak dibantu oleh informasi fisiologis dari *Heart Rate* (HR).

Pada beberapa kondisi pengujian, notifikasi rekomendasi muncul melebihi 3 detik setelah model mengeluarkan prediksi. Hal ini dipengaruhi oleh beberapa faktor, di antaranya latensi koneksi internet saat proses sinkronisasi data ke *Firebase*, beban komputasi perangkat Android saat menjalankan inferensi model *TFLite* secara bersamaan dengan proses pengambilan data sensor, serta variasi performa perangkat yang digunakan selama pengujian. Keterlambatan ini tidak bersifat konsisten dan umumnya terjadi pada kondisi jaringan yang tidak stabil.

C. Pengujian *Usability* Menggunakan *System Usability Scale* (SUS)

Selain melakukan pengujian teknis terhadap performa model GRU, penelitian ini juga melakukan evaluasi dari sisi pengalaman dan kemudahan

penggunaan sistem melalui metode *System Usability Scale* (SUS). Pengujian *usability* penting dilakukan untuk memastikan bahwa sistem tidak hanya akurat secara teknis, tetapi juga mudah digunakan dan dapat diterima oleh pengguna dalam kondisi nyata.

System Usability Scale (SUS) adalah instrumen evaluasi *usability* yang diperkenalkan oleh John Brooke pada tahun 1996 dan hingga kini telah diadopsi secara luas dalam berbagai studi untuk mengukur persepsi pengguna terhadap kemudahan penggunaan sebuah sistem. Instrumen ini terdiri dari 10 butir pernyataan yang dinilai menggunakan skala Likert 1–5, dengan rentang jawaban dari sangat tidak setuju hingga sangat setuju. SUS dipilih dalam penelitian ini karena penerapannya yang praktis, waktu pengisian yang singkat, serta kemampuannya menghasilkan penilaian *usability* yang valid dan dapat diandalkan.

Pada penelitian ini, pengujian SUS dilakukan dengan melibatkan sejumlah responden yang terbagi menjadi 2 kelompok umur dan berbagai latar belakang profesi yang telah mencoba menggunakan aplikasi monitoring aktivitas berbasis *smartwatch*. Responden pada tabel 4.9 diminta untuk menggunakan sistem mulai dari proses memulai perekaman data, melakukan aktivitas, melihat hasil prediksi aktivitas, hingga menghentikan sistem. Setelah itu, responden diminta mengisi kuesioner SUS berdasarkan pengalaman mereka selama menggunakan aplikasi.

Tabel 4. 9 Informasi responden

No	Inisial Responden	Umur	Kelompok Umur	Latar Belakang Profesi
1	R1	22	A	Mahasiswa
2	R2	23	A	Mahasiswa
3	R3	24	A	<i>Web Developer</i>
4	R4	26	B	<i>Mobile Developer</i>
5	R5	46	B	Guru
6	R6	50	B	Guru

Penghitungan skor SUS mengikuti prosedur berikut:

1. Pernyataan bernomor ganjil (1, 3, 5, 7, 9): nilai skor jawaban dikurangi dengan 1.
2. Pernyataan bernomor genap (2, 4, 6, 8, 10): nilai skor diperoleh dari 5 dikurangi skor jawaban responden.
3. Seluruh nilai hasil penyesuaian dari kesepuluh pernyataan kemudian diakumulasikan.
4. Total akumulasi tersebut dikalikan dengan 2,5 guna mengkonversi skor ke dalam rentang 0–100.

Tabel 4.10 berikut menyajikan rincian skor SUS yang diperoleh masing-masing responden yang berpartisipasi dalam sesi pengujian:

Tabel 4.10 Hasil Perhitungan Skor SUS

No	Responden	1	2	3	4	5	6	7	8	9	10	Skor SUS
1	R1	3	3	4	2	3	2	4	4	3	2	75
2	R2	2	3	2	2	3	2	3	3	1	1	55
3	R3	3	4	3	4	2	3	3	3	3	3	77.5
4	R4	3	4	2	3	3	2	3	3	3	3	72.5
5	R5	3	3	3	3	4	3	2	3	3	3	77.5
6	R6	4	4	3	3	3	3	4	3	4	3	87.5

Berdasarkan tabel tersebut, diperoleh rata-rata skor SUS sebesar:

$$\text{Rata-rata Skor SUS} = 77,5$$

Menurut interpretasi standar SUS, skor di atas 70 termasuk dalam kategori *Good* (Baik) dan berada pada tingkat *Acceptable*. Dengan demikian, skor rata-rata sebesar 77,5 menunjukkan bahwa sistem memiliki tingkat *usability* yang baik dan dapat diterima oleh pengguna.

Tabel 4.11 Analisis jawaban pertanyaan SUS

No	Pertanyaan	Tipe	Rata-rata Jawaban	Keterangan
1	Akan sering digunakan	Positif	3,0	Pengguna cukup berniat menggunakan secara rutin
2	Terlalu kompleks	Negatif	1,5	Mayoritas tidak merasa aplikasi rumit
3	Mudah digunakan	Positif	3,5	Aplikasi dinilai cukup mudah digunakan
4	Butuh bantuan orang lain	Negatif	2,0	Mayoritas tidak butuh bantuan khusus
5	Tampilan mudah dipahami	Positif	3,8	Tampilan dinilai cukup intuitif
6	Terlalu banyak inkonsistensi	Negatif	2,5	Ada sedikit inkonsistensi yang dirasakan
7	Mudah dipelajari pengguna baru	Positif	3,8	Aplikasi dinilai mudah dipelajari
8	Membingungkan	Negatif	2,0	Mayoritas tidak merasa bingung
9	Nyaman dan percaya diri	Positif	3,2	Cukup nyaman digunakan
10	Perlu pembiasaan	Negatif	2,5	Sebagian merasa perlu waktu untuk terbiasa

Dari hasil analisis per pertanyaan pada tabel 4.11, secara keseluruhan pengguna memberikan penilaian yang positif terhadap sistem. Pertanyaan nomor 1 mengenai intensitas penggunaan memperoleh rata-rata jawaban tertinggi yaitu 3,0, yang menunjukkan bahwa pengguna berniat menggunakan aplikasi secara rutin dalam kehidupan sehari-hari. Pertanyaan nomor 5 dan 7 masing-masing memperoleh rata-rata 3,8, yang mengindikasikan bahwa tampilan dan navigasi aplikasi dinilai cukup intuitif serta mudah dipelajari oleh pengguna baru.

Sementara itu, pertanyaan nomor 2 mengenai kerumitan aplikasi memperoleh rata-rata jawaban terendah yaitu 1,5. Karena pertanyaan ini bersifat negatif, nilai rendah tersebut justru menunjukkan hasil yang positif, yaitu mayoritas pengguna tidak merasa aplikasi terlalu kompleks untuk digunakan. Hal serupa juga terlihat pada pertanyaan nomor 4 dan 8 yang masing-masing memperoleh rata-rata

2,0, yang berarti mayoritas pengguna tidak membutuhkan bantuan orang lain dan tidak merasa bingung saat menggunakan aplikasi.

Namun demikian, terdapat beberapa aspek yang masih perlu ditingkatkan. Pertanyaan nomor 6 mengenai inkonsistensi tampilan dan pertanyaan nomor 10 mengenai kebutuhan pembiasaan masing-masing memperoleh rata-rata 2,5. Meskipun nilai tersebut masih tergolong rendah untuk pertanyaan negatif, angka ini mengindikasikan bahwa sebagian responden masih merasakan adanya ketidakkonsistenan pada antarmuka aplikasi dan memerlukan waktu adaptasi sebelum dapat menggunakan aplikasi secara optimal. Selain itu, pertanyaan nomor 9 mengenai kenyamanan dan kepercayaan diri pengguna memperoleh rata-rata 3,2, yang merupakan nilai terendah di antara pertanyaan positif, sehingga aspek kenyamanan penggunaan masih perlu menjadi perhatian dalam pengembangan selanjutnya.

Skor terendah diperoleh dari responden dengan latar belakang mahasiswa sebesar 55, sementara skor tertinggi diperoleh dari responden berlatar belakang guru sebesar 87,5. Hal ini mengindikasikan bahwa pengguna non-teknis justru memberikan penilaian yang lebih positif terhadap kemudahan penggunaan aplikasi. Secara keseluruhan, hasil pengujian *usability* menggunakan metode SUS menunjukkan bahwa sistem monitoring aktivitas berbasis *smartwatch* yang dikembangkan tidak hanya memiliki performa klasifikasi yang baik, tetapi juga memiliki tingkat kenyamanan penggunaan yang tinggi. Kombinasi antara akurasi model GRU dan *usability* sistem yang baik menjadikan aplikasi ini layak untuk diterapkan sebagai solusi monitoring aktivitas berbasis *wearable*.

4.1.9 Kendala Implementasi Sistem

Selama proses implementasi dan pengujian sistem pemantauan kesehatan berkelanjutan berbasis IoT dengan model GRU, ditemukan beberapa kendala teknis yang memengaruhi proses pengembangan. Kendala-kendala tersebut diidentifikasi dan ditangani melalui berbagai pendekatan teknis agar sistem dapat berjalan sesuai dengan tujuan penelitian.

A. Inkonsistensi *Sampling Rate* Sensor Wear OS

Sensor *heart rate* pada Wear OS tidak memberikan data dengan frekuensi tetap. Nilai HR hanya diperbarui ketika terjadi perubahan yang terdeteksi oleh sensor, bukan pada setiap interval waktu yang seragam. Kondisi ini menyebabkan jumlah sampel yang tersedia dalam satu *window* 60 detik tidak selalu konsisten. Sebagai contoh, pada kondisi pengguna yang relatif diam, pembaruan nilai HR dapat terjadi hanya beberapa kali dalam satu menit, sedangkan pada kondisi aktif pembaruan dapat jauh lebih sering.

Kendala ini ditangani dengan menerapkan mekanisme *forward fill*, yaitu mengisi nilai kosong pada interval waktu yang tidak memiliki pembacaan sensor baru dengan menggunakan nilai terakhir yang tersedia. Pendekatan ini umum digunakan dalam pemrosesan data sensor biologis yang memiliki frekuensi sampling tidak tetap (Tăuțan et al., 2021). Dengan demikian, setiap segmen *window* tetap memiliki jumlah representasi data yang konsisten sebelum tahap ekstraksi fitur dilakukan.

B. Perbedaan Karakteristik Data WESAD dengan Data *Smartwatch* Android

Dataset WESAD dikumpulkan menggunakan perangkat Empatica E4 *Wristband* yang memiliki sensor *Blood Volume Pulse* (BVP) dengan frekuensi sampling hingga 64 Hz serta sensor akselerometer dengan frekuensi 32 Hz. Sementara itu, *smartwatch* Android hanya menyediakan nilai HR dalam satuan BPM yang merupakan hasil estimasi perangkat, bukan sinyal BVP mentah, dengan frekuensi pembaruan yang bersifat dinamis.

Perbedaan ini menimbulkan tantangan dalam menjaga konsistensi antara data pelatihan dan data inferensi. Untuk mengatasi hal tersebut, pipeline *preprocessing* dirancang khusus agar fitur statistik yang diekstraksi dari data *smartwatch* memiliki representasi yang sebanding dengan fitur yang diekstraksi dari data WESAD. Fitur seperti RMSSD yang pada WESAD dihitung dari sinyal BVP mentah, pada *smartwatch* didekati menggunakan variasi nilai HR antarwaktu dalam satu *window*. Penyelarasan ini memungkinkan model GRU yang dilatih pada data WESAD tetap dapat digunakan untuk melakukan inferensi pada data *smartwatch* secara konsisten.

C. Tantangan Konversi Model GRU ke Format TFLite

Model GRU yang dilatih menggunakan TensorFlow/Keras dikonversi ke format .tflite agar dapat dijalankan secara *on-device* pada aplikasi Android. Proses konversi ini menghadapi beberapa tantangan teknis. Tidak semua operasi TensorFlow didukung secara penuh oleh *runtime* TFLite, terutama pada beberapa operasi internal lapisan GRU. Kendala ini diatasi dengan memastikan

kompatibilitas versi antara TensorFlow yang digunakan saat pelatihan dan versi TFLite *runtime* yang diintegrasikan pada aplikasi Android.

Selain itu, parameter standardisasi Z-Score berupa nilai *mean* dan *standard deviation* dari data pelatihan tidak dapat disertakan langsung di dalam file model .tflite. Parameter tersebut disimpan secara terpisah dalam format JSON dan dimuat oleh aplikasi Android pada saat inisialisasi. Ketidakcocokan antara parameter scaler saat pelatihan dan saat inferensi akan menghasilkan input yang tidak terstandardisasi dengan benar, sehingga dapat menurunkan akurasi prediksi secara signifikan. Oleh karena itu, konsistensi parameter scaler antara tahap pelatihan dan tahap inferensi menjadi salah satu aspek kritis yang harus dijaga dalam implementasi sistem ini.

Kendala lain yang ditemukan selama integrasi adalah ketidaksesuaian antara format input yang diharapkan oleh model TFLite dan format data yang disiapkan oleh aplikasi Android. Model GRU yang dilatih menggunakan TensorFlow/Keras mengharapkan input dalam bentuk tensor tiga dimensi dengan susunan [batch_size, timesteps, jumlah_fitur], yaitu [1, 1, 13] pada saat inferensi satu sampel. Namun pada implementasi awal di Android, data fitur disiapkan dalam bentuk array satu dimensi sehingga menyebabkan ketidakcocokan dimensi yang mengakibatkan hasil prediksi tidak keluar atau tidak sesuai ekspektasi.

Selain itu, urutan fitur dalam array input harus benar-benar identik dengan urutan fitur yang digunakan saat pelatihan model. Model GRU mempelajari bahwa posisi ke-0 merepresentasikan *Mean HR*, posisi ke-1 merepresentasikan *Std HR*, dan seterusnya hingga posisi ke-12 sebagai *Median ACC*. Perbedaan urutan fitur

pada saat inferensi, meskipun seluruh nilai fitur sudah benar, akan menyebabkan model menghasilkan prediksi yang tidak akurat tanpa memunculkan pesan *error* apapun karena model tetap menerima input dengan jumlah fitur yang sama.

Kendala ketiga berkaitan dengan format output TFLite Interpreter pada Android. Output model tidak langsung berupa tiga nilai probabilitas yang siap digunakan, melainkan berupa struktur `Array<FloatArray>` yang harus diakses pada indeks yang tepat. Kesalahan dalam mengakses indeks output menyebabkan nilai probabilitas yang ditampilkan tidak sesuai dengan urutan kelas yang sebenarnya. Seluruh kendala ini berhasil diselesaikan melalui penyesuaian bertahap pada kode aplikasi Android, meliputi restrukturisasi dimensi input tensor, penyeragaman urutan fitur antara pipeline Python dan Dart, serta penyesuaian cara membaca output dari TFLite Interpreter.

D. Manajemen *Thread* dan Beban Komputasi pada Perangkat Android

Proses inferensi TFLite dijalankan pada *background thread* agar tidak memblokir *UI thread* aplikasi Android. Namun, eksekusi bersamaan antara beberapa proses, yaitu penerimaan data dari *smartwatch*, *preprocessing*, ekstraksi fitur, dan inferensi model, pada perangkat dengan spesifikasi terbatas dapat menyebabkan peningkatan penggunaan memori dan penurunan responsivitas antarmuka.

Kendala ini ditangani dengan membatasi frekuensi inferensi menjadi satu kali per *window* 60 detik dan mengoptimalkan alokasi memori TFLite Interpreter agar tidak terjadi *memory leak* selama aplikasi berjalan dalam jangka panjang. Pembatasan frekuensi inferensi ini juga sejalan dengan karakteristik sistem

pemantauan kesehatan yang tidak memerlukan prediksi setiap detik, melainkan cukup memberikan analisis kondisi pengguna secara periodik.

E. Latensi Notifikasi Rekomendasi

Target performa sistem yang ditetapkan pada Bab 3.4.5 menyebutkan bahwa notifikasi rekomendasi harus muncul maksimal 3 detik setelah model mengeluarkan prediksi. Pada kondisi normal, target ini dapat dipenuhi. Namun, pada kondisi jaringan yang tidak stabil, proses sinkronisasi data ke Firebase Cloud Firestore yang berjalan bersamaan dengan proses inferensi dapat menambah beban komputasi perangkat sehingga memperlambat kemunculan notifikasi.

Kendala ini bersifat kondisional dan bergantung pada kondisi jaringan serta spesifikasi perangkat yang digunakan. Karena proses inferensi dan proses pengiriman data ke *cloud* dilakukan secara asinkron dan terpisah, keterlambatan sinkronisasi *cloud* tidak memengaruhi hasil prediksi yang ditampilkan kepada pengguna. Notifikasi rekomendasi tetap dihasilkan dari hasil inferensi lokal, sehingga fungsi utama sistem tidak terganggu meskipun koneksi internet mengalami ketidakstabilan.

4.2 Pembahasan

Berdasarkan hasil evaluasi model, metode *Gated Recurrent Unit* (GRU) menunjukkan performa yang baik dalam mengklasifikasikan kondisi pengguna berdasarkan data sensor *smartwatch*. Model dilatih menggunakan kombinasi data *accelerometer* dan *heart rate*, kemudian dievaluasi menggunakan metrik *accuracy*, *precision*, *recall*, dan *F1-score* untuk menilai kemampuan klasifikasi pada setiap kelas kondisi.

Hasil evaluasi menunjukkan bahwa model mampu mencapai akurasi keseluruhan sebesar 94%, yang menunjukkan bahwa sebagian besar data pengujian dapat diklasifikasikan dengan benar oleh model. Nilai *precision*, *recall*, dan *F1-score* pada masing-masing kelas juga menunjukkan performa yang seimbang, sehingga dapat disimpulkan bahwa model tidak hanya baik dalam memprediksi satu kelas tertentu, tetapi juga mampu mengenali seluruh kelas kondisi secara konsisten.

Model GRU dipilih karena memiliki arsitektur yang lebih sederhana dibandingkan *Long Short-Term Memory* (LSTM), namun tetap mampu menangkap dependensi temporal pada data deret waktu (*time-series*). Dengan jumlah parameter yang lebih sedikit, model GRU lebih efisien dalam proses pelatihan dan inferensi, sehingga cocok untuk diimplementasikan pada perangkat *mobile* seperti *smartwatch* Android dengan keterbatasan sumber daya komputasi.

Jika dibandingkan dengan penelitian lain yang menggunakan dataset WESAD, model GRU pada penelitian ini memperoleh akurasi sebesar 94% dengan hanya menggunakan dua fitur yaitu *Heart Rate* (HR) dan *Accelerometer* (ACC). Bajpai et al. (2023) yang menggunakan LSTM pada dataset WESAD berhasil memperoleh akurasi sebesar 98%, *precision* 97,01%, *recall* 97,00%, dan *F1-score* 97,03%. Namun demikian, penelitian tersebut menggunakan seluruh modalitas sensor yang tersedia pada WESAD seperti EDA, ECG, EMG, dan respirasi, bukan hanya HR dan ACC. Penelitian lain oleh Tiwari et al. (2024) menggunakan *Dilated CNN* pada sinyal PPG dari WESAD dan memperoleh akurasi sebesar 93,56% serta AUC sebesar 96,52%. Meskipun akurasinya relative sepadan, penelitian tersebut bergantung pada sensor PPG khusus yang tidak selalu tersedia pada *smartwatch*

Android secara umum. Sementara itu, Brihadiswaran et al. (2023) menggunakan *Federated Learning* berbasis DNN pada dataset WESAD dan memperoleh akurasi sebesar 86,82%. Meskipun akurasinya lebih rendah dari penelitian ini, pendekatan tersebut memiliki keunggulan dalam aspek privasi data pengguna. Melchiades et al. (2022) yang menggunakan RNN pada dataset WESAD memperoleh akurasi sebesar 86%, sedangkan Sah et al. (2021) menggunakan CNN dan memperoleh akurasi sebesar 92,85%. Perbandingan hasil tersebut dapat dilihat pada Tabel 4.10 berikut.

Tabel 4.12 Perbandingan penelitian dengan penelitian sebelumnya

No	Penelitian	Model	Fitur yang Digunakan	Akurasi
1	Bajpai et al. (2023)	LSTM	Multi-sensor (EDA, ECG, EMG, ACC, Resp)	98,00%
2	Sah et al. (2021)	CNN	Multi-sensor	92,85%
3	Tiwari et al. (2024)	<i>Dilated CNN</i>	PPG	93,56%
4	Melchiades et al. (2022)	RNN	Multi-sensor	86,00%
5	Brihadiswaran et al. (2023)	<i>Federated DNN</i>	EDA	86,82%
6	Penelitian ini	GRU	HR + ACC	94,00%

Perbandingan pada tabel 4.12 tersebut menunjukkan bahwa akurasi 94% yang diperoleh pada penelitian ini berada pada rentang yang kompetitif dibandingkan penelitian lain yang menggunakan dataset WESAD. Yang membedakan penelitian ini adalah penggunaan hanya dua fitur (HR dan ACC) yang tersedia secara langsung pada *smartwatch* Android tanpa memerlukan sensor tambahan. Hal ini menjadikan pendekatan yang dikembangkan lebih praktis dan aplikatif untuk diterapkan pada perangkat *wearable* di kehidupan nyata, di mana keterbatasan sensor merupakan tantangan utama yang sering dihadapi.

Dari sisi implementasi sistem IoT, data sensor berhasil dikirimkan dari *smartwatch* ke *smartphone* melalui koneksi *Wear OS* dan selanjutnya disimpan ke *Firestore* sebagai basis data cloud. Proses pengiriman data berjalan

dengan baik meskipun pada beberapa kondisi pengujian terdapat keterlambatan notifikasi rekomendasi yang melebihi 3 detik setelah model mengeluarkan prediksi. Keterlambatan ini dipengaruhi oleh latensi koneksi internet saat proses sinkronisasi data ke *Firebase*, beban komputasi perangkat Android saat menjalankan inferensi model *TFLite* secara bersamaan dengan proses pengambilan data sensor, serta variasi performa perangkat yang digunakan selama pengujian. Keterlambatan ini tidak bersifat konsisten dan umumnya terjadi pada kondisi jaringan yang tidak stabil.

Hasil pengujian fungsional menunjukkan bahwa seluruh fitur utama sistem berjalan sesuai dengan yang direncanakan, meliputi pembacaan sensor, pengiriman data, klasifikasi kondisi, hingga pemberian rekomendasi kepada pengguna. Adapun hasil pengujian *usability* menggunakan metode *System Usability Scale* (SUS) memperoleh skor rata-rata sebesar 77,5. Menurut interpretasi standar SUS (Bangor et al., 2009), skor tersebut termasuk dalam kategori *Good* (Baik) dan berada pada tingkat *Acceptable*, yang menunjukkan bahwa sistem memiliki tingkat *usability* yang baik dan dapat diterima oleh pengguna.

Dari hasil analisis per pertanyaan SUS, diketahui bahwa pengguna menilai aplikasi mudah digunakan dan tidak memerlukan bantuan orang lain dalam pengoperasiannya. Tampilan dan navigasi aplikasi juga dinilai cukup intuitif dan mudah dipahami. Namun demikian, terdapat beberapa aspek yang masih perlu ditingkatkan, yaitu konsistensi tampilan antarmuka serta kemudahan adaptasi bagi pengguna baru, mengingat sebagian responden merasa masih memerlukan waktu pembiasaan sebelum dapat menggunakan aplikasi secara optimal. Skor terendah

diperoleh dari responden dengan latar belakang mahasiswa sebesar 55, sementara skor tertinggi diperoleh dari responden berlatar belakang guru sebesar 87,5. Hal ini mengindikasikan bahwa pengguna non-teknis justru memberikan penilaian yang lebih positif terhadap kemudahan penggunaan aplikasi.

Dalam kondisi tertentu, model GRU dapat menghasilkan distribusi probabilitas yang relatif seimbang di antara ketiga kelas, misalnya mendekati $[0.333, 0.333, 0.334]$. Kondisi ini dapat terjadi ketika pola data sensor pada *window* yang diproses tidak memiliki karakteristik yang cukup khas, misalnya saat pengguna sedang dalam transisi antara kondisi fisiologis atau data sensor memiliki kualitas rendah akibat *motion artifact*. Pada implementasi sistem ini, kondisi tersebut tetap ditangani menggunakan operasi *argmax* yang memilih kelas dengan probabilitas tertinggi meskipun selisihnya sangat kecil. Sebagai pengembangan ke depan, sistem dapat dilengkapi dengan mekanisme *confidence threshold*: apabila probabilitas tertinggi berada di bawah nilai tertentu, misalnya kurang dari 0,50, sistem dapat menampilkan keterangan bahwa data belum cukup representatif dan menyarankan pengguna untuk melakukan pengukuran ulang (Goodfellow et al., 2016).

Secara keseluruhan, hasil pengujian menunjukkan bahwa sistem pemantauan kesehatan berkelanjutan berbasis smartwatch Android yang dikembangkan tidak hanya memiliki performa klasifikasi yang baik, tetapi juga memiliki tingkat kenyamanan penggunaan yang tinggi. Kombinasi antara akurasi model GRU sebesar 94% dan nilai *usability* sistem yang baik menjadikan aplikasi

ini layak untuk diterapkan sebagai solusi pemantauan kesehatan berbasis *wearable* yang praktis dan dapat digunakan oleh masyarakat umum.

4.3 Integrasi Penelitian dengan Islam

Berdasarkan nilai-nilai ajaran Islam, pemanfaatan teknologi dalam bidang kesehatan tidak hanya dipandang sebagai bentuk perkembangan ilmu pengetahuan, tetapi juga sebagai sarana untuk mewujudkan kemaslahatan hidup manusia. Penelitian mengenai sistem pemantauan kesehatan berbasis *smartwatch* dan teknologi *Internet of Things* (IoT) ini memiliki keterkaitan dengan nilai-nilai keislaman yang mencakup hubungan manusia dengan Allah SWT, hubungan manusia dengan sesama manusia, serta hubungan manusia dengan lingkungan. Dalam perspektif Islam, ketiga hubungan tersebut dikenal dengan konsep *hablum minallah*, *hablum minannas*, dan *hablum minal 'alam*. Oleh karena itu, penelitian ini dapat dipetakan berdasarkan ketiga aspek tersebut sebagai berikut.

A. Hablum Minallah (Hubungan Manusia dengan Allah SWT)

Dalam perspektif Islam, menjaga kesehatan merupakan salah satu bentuk tanggung jawab manusia terhadap amanah yang diberikan oleh Allah SWT. Kesehatan menjadi nikmat yang sangat berharga karena dengan kondisi tubuh yang sehat, manusia dapat menjalankan aktivitas ibadah, pekerjaan, serta kehidupan sosial secara optimal. Islam mengajarkan umatnya untuk senantiasa menjaga diri dari segala hal yang dapat membahayakan kesehatan, baik kesehatan fisik maupun mental. Hal tersebut sebagaimana firman Allah SWT dalam QS. Al-Baqarah ayat 195:

وَأَنْفِقُوا فِي سَبِيلِ اللَّهِ وَلَا تُلْقُوا بِأَيْدِيكُمْ إِلَى التَّهْلُكَةِ، وَأَحْسِنُوا، إِنَّ اللَّهَ يُحِبُّ الْمُحْسِنِينَ

“Dan belanjakanlah (harta bendamu) di jalan Allah, dan janganlah kamu menjatuhkan dirimu sendiri ke dalam kebinasaan, serta berbuat baiklah. Sungguh, Allah menyukai orang-orang yang berbuat baik.” (QS. Al-Baqarah: 195)

Ayat tersebut mengandung makna bahwa manusia diperintahkan untuk menjaga dirinya dari berbagai hal yang dapat menimbulkan kerusakan atau membahayakan kesehatan. Dalam konteks kehidupan modern, menjaga kesehatan dapat dilakukan melalui pola hidup sehat, menjaga keseimbangan aktivitas, serta memanfaatkan perkembangan teknologi sebagai sarana pendukung dalam pemantauan kondisi tubuh secara dini dan berkelanjutan (Shihab, 2002).

Selain itu, Allah SWT juga berfirman dalam QS. Asy-Syu'ara ayat 80:

وَإِذَا مَرِضْتُ فَهُوَ يَشْفِينِ

“Dan apabila aku sakit, Dialah yang menyembuhkan aku.” (QS. Asy-Syu'ara: 80)

Ayat tersebut menjelaskan bahwa hakikat kesembuhan berasal dari Allah SWT. Namun demikian, manusia tetap diwajibkan untuk melakukan ikhtiar sebagai bentuk usaha dalam menjaga dan memelihara kesehatan. Ikhtiar tersebut dapat dilakukan melalui pemanfaatan ilmu pengetahuan dan teknologi yang berkembang pada era modern saat ini (Shihab, 2002).

Hal ini juga diperkuat dengan hadis Rasulullah SAW:

نِعْمَتَانِ مَغْبُورٌ فِيهِمَا كَثِيرٌ مِنَ النَّاسِ: الصِّحَّةُ وَالْفَرَاغُ

“Dua nikmat yang banyak manusia tertipu di dalamnya, yaitu nikmat sehat dan waktu luang.” (HR. Bukhari)

Hadis tersebut menjelaskan bahwa kesehatan merupakan nikmat besar yang sering kali tidak disadari nilainya oleh manusia hingga nikmat tersebut berkurang atau hilang. Oleh karena itu, menjaga kesehatan menjadi salah satu bentuk rasa syukur kepada Allah SWT atas karunia yang telah diberikan.

B. Hablum Minannas (Hubungan Manusia dengan Sesama Manusia)

Selain itu, Rasulullah SAW juga bersabda:

الْمُؤْمِنُ الْقَوِيُّ خَيْرٌ وَأَحَبُّ إِلَى اللَّهِ مِنَ الْمُؤْمِنِ الضَّعِيفِ وَفِي كُلِّ خَيْرٍ

“Mukmin yang kuat lebih baik dan lebih dicintai Allah daripada mukmin yang lemah, dan pada keduanya terdapat kebaikan.” (HR. Muslim)

Hadis ini menunjukkan pentingnya menjaga kekuatan dan kesehatan tubuh agar manusia dapat menjalankan ibadah, aktivitas, serta tanggung jawab sosial dengan baik. Kekuatan yang dimaksud tidak hanya mencakup kekuatan fisik, tetapi juga kesehatan mental dan emosional (An-Nawawi., 2010).

Selain itu, Allah SWT juga berfirman dalam QS. Yunus ayat 57:

يَا أَيُّهَا النَّاسُ قَدْ جَاءَكُمْ مَوْعِظَةٌ مِنْ رَبِّكُمْ وَشِفَاءٌ لِمَا فِي الصُّدُورِ وَهُدًى وَرَحْمَةٌ لِّلْمُؤْمِنِينَ

“Wahai manusia! Sungguh, telah datang kepadamu pelajaran (Al-Qur'an) dari Tuhanmu, penyembuh bagi penyakit yang ada dalam dada, petunjuk serta rahmat bagi orang-orang yang beriman.” (QS. Yunus: 57)

Ayat tersebut menjelaskan bahwa Islam tidak hanya memperhatikan kesehatan fisik, tetapi juga kesehatan hati, mental, dan emosional. Penyakit dalam dada dapat dimaknai sebagai kegelisahan, stres, kecemasan, maupun gangguan emosional lainnya yang dapat memengaruhi kualitas hidup seseorang (Shihab, 2002). Oleh karena itu, menjaga kesehatan mental menjadi bagian penting dalam

kehidupan manusia agar tercipta keseimbangan antara kondisi fisik, psikis, dan spiritual.

Dengan adanya sistem pemantauan kesehatan, pengguna diharapkan dapat meningkatkan kesadaran terhadap kondisi kesehatannya sehingga mampu mengambil langkah pencegahan dan penanganan yang tepat. Teknologi ini juga dapat membantu pengguna dalam menjaga keseimbangan antara aktivitas, kondisi fisik, dan kesehatan mental sehingga kualitas hidup dapat meningkat secara optimal.

C. Hablum Minal ‘Alam (Hubungan Manusia dengan Lingkungan dan Pemanfaatan Teknologi)

Penelitian ini merupakan salah satu bentuk implementasi ikhtiar manusia dalam menjaga kesehatan melalui pengembangan sistem pemantauan kesehatan berbasis smartwatch dan teknologi *Internet of Things* (IoT). Sistem yang dikembangkan mampu memonitor kondisi fisiologis pengguna secara berkelanjutan dengan memanfaatkan sensor seperti sensor akselerometer dan detak jantung. Data yang diperoleh kemudian digunakan untuk mendeteksi perubahan kondisi tubuh, seperti relaksasi, stres, maupun aktivitas emosional lainnya secara lebih dini.

Pemanfaatan teknologi secara bijak dalam bidang kesehatan merupakan bentuk tanggung jawab manusia dalam menggunakan perkembangan ilmu pengetahuan untuk kemaslahatan bersama tanpa memberikan dampak negatif terhadap lingkungan. Teknologi yang digunakan secara tepat dapat membantu menciptakan kehidupan yang lebih sehat, efisien, dan berkelanjutan.

Dengan demikian, penelitian ini tidak hanya memberikan kontribusi dalam bidang teknologi dan kesehatan digital, tetapi juga sejalan dengan nilai-nilai Islam yang mendorong manusia untuk menjaga kesehatan, memanfaatkan ilmu pengetahuan secara bertanggung jawab, serta menjadikan teknologi sebagai sarana dalam meningkatkan kualitas hidup dan kesejahteraan umat manusia.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil implementasi, pengujian, dan pembahasan yang telah dilakukan, penelitian ini berhasil mengembangkan sistem pemantauan kesehatan secara otomatis dan berkelanjutan berbasis *Internet of Things* (IoT) dengan memanfaatkan *smartwatch* Android sebagai perangkat akuisisi data fisiologis pengguna. Sistem mampu mengintegrasikan sensor *heart rate* dan *accelerometer* pada *smartwatch* dengan aplikasi Android melalui *Wear OS Data Layer API* sehingga proses pengambilan, pengiriman, dan pengolahan data dapat dilakukan dan berkelanjutan. Integrasi ini memungkinkan data fisiologis pengguna diperoleh secara otomatis untuk mendukung pemantauan kesehatan sehari-hari.

Penerapan model *Gated Recurrent Unit* (GRU) pada penelitian ini berhasil digunakan untuk menganalisis data deret waktu (*time-series*) berupa detak jantung dan aktivitas gerak pengguna. Berdasarkan hasil evaluasi model, GRU memperoleh nilai *accuracy* sebesar 94%, *precision* rata-rata sebesar 0.92, *recall* rata-rata sebesar 0.92, dan *weighted average F1-score* sebesar 0.93. Pada masing-masing kelas, model memperoleh *F1-score* sebesar 0.96 untuk kelas *baseline*, 0.93 untuk kelas *stress*, dan 0.87 untuk kelas *amusement*. Hasil tersebut menunjukkan bahwa model mampu mempelajari pola temporal data fisiologis dengan cukup baik untuk mengklasifikasikan kondisi pengguna ke dalam kategori *baseline*/relaksasi, *stress* ringan, dan *amusement*/senang.

Implementasi model menggunakan pendekatan *on-device inference* pada aplikasi Android juga memungkinkan proses analisis dilakukan secara langsung pada perangkat pengguna sehingga sistem dapat memberikan hasil prediksi secara cepat dan responsif tanpa ketergantungan penuh terhadap koneksi server. Selain itu, berdasarkan hasil pengujian fungsional, seluruh komponen sistem mulai dari smartwatch, aplikasi Android, integrasi *Firebase Cloud Firestore*, hingga model GRU dapat berjalan sesuai fungsi yang telah dirancang. Sistem mampu melakukan proses akuisisi data, pemrosesan, analisis, serta penyajian hasil prediksi dalam bentuk visualisasi dan rekomendasi kesehatan adaptif kepada pengguna.

Dengan demikian, penelitian ini menunjukkan bahwa kombinasi teknologi *wearable*, *Internet of Things* (IoT), dan metode *deep learning* berbasis GRU memiliki potensi yang baik untuk diterapkan pada sistem pemantauan kesehatan berkelanjutan. Sistem yang dikembangkan mampu membantu pengguna dalam memahami kondisi fisiologis tubuh secara lebih mandiri melalui proses monitoring dan analisis data kesehatan yang berjalan secara otomatis.

5.2 Saran

Berdasarkan hasil penelitian yang telah dilakukan, masih terdapat beberapa keterbatasan yang dapat menjadi perhatian untuk pengembangan penelitian selanjutnya. penelitian ini menggunakan dataset sekunder WESAD (Wearable Stress and Affect Detection) sebagai data pelatihan model GRU. Dataset tersebut dikumpulkan di Jerman oleh Universität Siegen pada tahun 2018 menggunakan perangkat Empatica E4 Wristband dengan karakteristik sensor yang berbeda dari smartwatch Android, seperti perbedaan sampling rate, kualitas sinyal, dan tingkat

noise. Selain itu, karakteristik fisiologis populasi, pola aktivitas harian, serta kondisi lingkungan seperti suhu dan kelembapan di Indonesia dapat berbeda dari kondisi saat dataset tersebut dikumpulkan, sehingga terdapat kemungkinan penurunan akurasi ketika model diterapkan pada pengguna lokal. Oleh karena itu, penelitian selanjutnya disarankan untuk mengumpulkan data primer langsung dari pengguna smartwatch Android di Indonesia dengan jumlah partisipan yang lebih representatif, rentang usia yang lebih beragam, durasi pengambilan data yang lebih panjang, serta kondisi aktivitas yang bervariasi. Dataset lokal yang dihasilkan tidak hanya akan meningkatkan akurasi dan kemampuan generalisasi model, tetapi juga memberikan kontribusi berupa dataset wearable berbasis populasi Indonesia yang dapat dimanfaatkan oleh peneliti selanjutnya.

Selain itu, pada penelitian ini hanya memanfaatkan dua sensor yang tersedia pada smartwatch Android, yaitu *heart rate* (HR) dan *accelerometer* (ACC). Penambahan parameter sensor lain yang sudah mulai tersedia pada smartwatch modern, seperti SpO₂ (saturasi oksigen), suhu kulit (*skin temperature*), kualitas tidur, maupun *Electrodermal Activity* (EDA), dapat dipertimbangkan untuk memperkaya informasi fisiologis yang dianalisis. Penambahan fitur ini berpotensi meningkatkan kemampuan model dalam membedakan kondisi fisiologis yang memiliki pola HR dan ACC serupa, khususnya pada kelas *amusement* yang pada penelitian ini memperoleh performa lebih rendah dibandingkan kelas lainnya.

Selain itu, pengembangan arsitektur model dapat dilakukan dengan membandingkan performa GRU terhadap metode lain seperti LSTM, CNN-GRU *hybrid*, atau *Transformer*-based model untuk memperoleh performa klasifikasi

yang lebih optimal pada data *wearable* dengan karakteristik lokal. Pendekatan *transfer learning* juga dapat dipertimbangkan, yaitu memanfaatkan model yang telah dilatih pada dataset WESAD kemudian melakukan *fine-tuning* menggunakan data primer lokal agar proses adaptasi model dapat dilakukan secara lebih efisien.

Dari sisi sistem dan antarmuka, fitur rekomendasi kesehatan yang lebih personal dan kontekstual dapat dikembangkan dengan mempertimbangkan riwayat kondisi pengguna secara jangka panjang, bukan hanya berdasarkan prediksi satu *window* tunggal. Integrasi sistem dengan layanan kesehatan digital atau platform *telemedicine* juga dapat menjadi arah pengembangan selanjutnya, sehingga sistem tidak hanya berfungsi sebagai alat pemantauan mandiri, tetapi juga sebagai pendukung pengambilan keputusan kesehatan yang dapat dihubungkan dengan tenaga medis profesional.

DAFTAR PUSTAKA

- Ahmadzadeh, E., Kim, H., Jeong, O., Kim, N., & Moon, I. (2022). A Deep Bidirectional LSTM-GRU Network Model for Automated Ciphertext Classification. *IEEE Access*, 10, 3228–3237. <https://doi.org/10.1109/ACCESS.2022.3140342>
- Al-Maraghi, A. M. (1993). *Tafsir Al-Maraghi*. Dar al-Fikr.
- An-Nawawi. (2010). *Syarah Shahih Muslim*. Dar al-Kutub al-‘Ilmiyyah.
- Bajpai, A. et al. (2023). Mental Stress Level Detection Using LSTM for WESAD Dataset. ResearchGate.
- Bangor, A., Kortum, P., & Miller, J. (2009). Determining what individual SUS scores mean: Adding an adjective rating scale. *Journal of Usability Studies*, 4(3), 114–123.
- Brihadiswaran, G., Haputhanthri, D., Gunathilaka, S., Meedeniya, D., & Jayarathna, S. (2023). EEG-based Stress Recognition Using Federated Deep Learning. *International Journal of Advanced Computer Science and Applications*, 14(1). <https://doi.org/10.14569/IJACSA.2023.0140152>
- Brooke, J. (1996). SUS: A "quick and dirty" usability scale. In P. W. Jordan, B. Thomas, B. A. Weerdmeester, & I. L. McClelland (Eds.), *Usability Evaluation in Industry* (pp. 189–194). Taylor & Francis.
- Departemen Agama Republik Indonesia. (2019). *Al-Qur'an dan Terjemahannya*. Lajnah Pentashihan Mushaf Al-Qur'an.
- Dewi, R. K., Tantular, B., Suprijadi, J., & Pravitasari, A. A. (2023). Analisis Sentimen Ulasan Pengguna Aplikasi E-Samsat Provinsi Jawa Barat Menggunakan Metode BiGRU. *Inferensi*, 1(1), 1. <https://doi.org/10.12962/j27213862.v1i1.19113>
- El Hajj, C., & Kyriacou, P. A. (2021). Recurrent Neural Network Models for Blood Pressure Monitoring Using PPG Morphological Features. *2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, 1865–1868. <https://doi.org/10.1109/EMBC46164.2021.9630319>
- Gjoreski, M., Luštrek, M., Gams, M., & Gjoreski, H. (2017). Monitoring stress with a wrist device using context. *Journal of Biomedical Informatics*, 73, 159–170. <https://doi.org/10.1016/j.jbi.2017.08.006>

- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.
<https://www.deeplearningbook.org>
- Han, J., Kamber, M., & Pei, J. (2012). *Data mining: Concepts and techniques* (3rd ed). Elsevier/Morgan Kaufmann.
- Healey, J. A., & Picard, R. W. (2005). Detecting Stress During Real-World Driving Tasks Using Physiological Sensors. *IEEE Transactions on Intelligent Transportation Systems*, 6(2), 156–166.
<https://doi.org/10.1109/TITS.2005.848368>
- Hughes, A., Shandhi, M. M. H., Master, H., Dunn, J., & Brittain, E. (2023). Wearable Devices in Cardiovascular Medicine. *Circulation Research*, 132(5), 652–670. <https://doi.org/10.1161/CIRCRESAHA.122.322389>
- Kim, J., & André, E. (2006). Emotion Recognition Using Physiological and Speech Signal in Short-Term Observation. In E. André, L. Dybkjær, W. Minker, H. Neumann, & M. Weber (Eds.), *Perception and Interactive Technologies* (Vol. 4021, pp. 53–64). Springer Berlin Heidelberg.
https://doi.org/10.1007/11768029_6
- Li, C., Guo, Q., Shao, L., Li, J., & Wu, H. (2022). Research on Short-Term Load Forecasting Based on Optimized GRU Neural Network. *Electronics*, 11(22), 3834. <https://doi.org/10.3390/electronics11223834>
- Liyakathunisa, Alsaedi, A., Jabeen, S., & Kolivand, H. (2022). Ambient assisted living framework for elderly care using Internet of medical things, smart sensors, and GRU deep learning techniques. *Journal of Ambient Intelligence and Smart Environments*, 14(1), 5–23. <https://doi.org/10.3233/AIS-210162>
- Lu, W., Jin, J., Wang, B., Li, K., Liang, C., Dong, J., & Zhao, S. (2020). Intelligence in Tourist Destinations Management: Improved Attention-based Gated Recurrent Unit Model for Accurate Tourist Flow Forecasting. *Sustainability*, 12(4), 1390. <https://doi.org/10.3390/su12041390>
- Melchiades, F. R., Chicrala, G. M., Santos, M. A. C., & Domingues, A. L. O. (2022). Stress Detection Using Wearable Physiological Sensors and Machine Learning. *Journal of Physics: Conference Series*, 2367(1), 012020. <https://doi.org/10.1088/1742-6596/2367/1/012020>
- Nurhayati, E., & Fitriyana, S. (2020). Determinan Kesehatan dalam Perspektif Islam: Studi Pendahuluan. *Jurnal Integrasi Kesehatan & Sains*, 2(1), 52–56. <https://doi.org/10.29313/jiks.v2i1.5865>
- Rizal, S., & Yuniarti, A. R. (2023). *GRU-Based Fusion Models for Enhanced Non-Invasive Blood Pressure Estimation from PPG Signals*.
<https://doi.org/10.36227/techrxiv.24637416.v1>

- Sah, R. K., Bapat, M., & Bhattacharya, A. (2021). Stress Detection from Wearable Physiological Sensors Using Convolutional Neural Networks. *2021 International Conference on Computational Intelligence and Computing Research (ICCIC)*, 1–6. <https://doi.org/10.1109/ICCIC54684.2021.9763210>
- Schmidt, P., Reiss, A., Duerichen, R., Marberger, C., & Van Laerhoven, K. (2018). Introducing WESAD, a Multimodal Dataset for Wearable Stress and Affect Detection. *Proceedings of the 20th ACM International Conference on Multimodal Interaction*, 400–408. <https://doi.org/10.1145/3242969.3242985>
- Shaffer, F., & Ginsberg, J. P. (2017). An Overview of Heart Rate Variability Metrics and Norms. *Frontiers in Public Health*, 5, 258. <https://doi.org/10.3389/fpubh.2017.00258>
- Shihab, M. Q. (2002). *Tafsir Al-Misbah: Pesan, Kesan dan Keserasian Al-Qur'an*. Lentera Hati.
- Smuck, M., Odonkor, C. A., Wilt, J. K., Schmidt, N., & Swiernik, M. A. (2021). The emerging clinical role of wearables: Factors for successful implementation in healthcare. *Npj Digital Medicine*, 4(1), 45. <https://doi.org/10.1038/s41746-021-00418-3>
- Sun, Y., Liu, Y., Wang, J., Xiang, C., Yin, S., & Li, X. (2024). Intelligent monitoring method for kick risk based on CNN-GRU hybrid network model. *Journal of Physics: Conference Series*, 2901(1), 012007. <https://doi.org/10.1088/1742-6596/2901/1/012007>
- Tan, T.-H., Shih, J.-Y., Liu, S.-H., Alkhaleefah, M., Chang, Y.-L., & Gochoo, M. (2023). Using a Hybrid Neural Network and a Regularized Extreme Learning Machine for Human Activity Recognition with Smartphone and Smartwatch. *Sensors*, 23(6), 3354. <https://doi.org/10.3390/s23063354>
- Tan, T.-H., Wu, J.-Y., Liu, S.-H., & Gochoo, M. (2022). Human Activity Recognition Using an Ensemble Learning Algorithm with Smartphone Sensor Data. *Electronics*, 11(3), 322. <https://doi.org/10.3390/electronics11030322>
- Tăuțan, A.-M., Moço, A. V., Simões-Corrêa, J., de Zeeuw, C., Androne-Naurnescu, I., & Bhattacharya, J. (2021). Artificial intelligence in the Internet of Medical Things: A systematic review. *IEEE Access*, 9, 139914–139952. <https://doi.org/10.1109/ACCESS.2021.3118932>
- Tiwari, P., Chandra, S., & Singh, S. K. (2024). Stress Detection Using PPG Signals and Dilated CNN on WESAD Dataset. *Biomedical Signal Processing and Control*, 89, 105756. <https://doi.org/10.1016/j.bspc.2023.105756>

- Żarnowski, A., Jankowski, M., & Gujski, M. (2022). Use of Mobile Apps and Wearables to Monitor Diet, Weight, and Physical Activity: A Cross-Sectional Survey of Adults in Poland. *Medical Science Monitor*, 28. <https://doi.org/10.12659/MSM.937948>
- Zhang, R. (2022). Personalized Course Recommendation Method Based on Learner Interest Mining in Educational Big Data Environment. *Scientific Programming*, 2022, 1–8. <https://doi.org/10.1155/2022/9943965>