

**ALAT PENDETEKSI JATUH PADA LANSIA DALAM
KEADAAN RAWAT JALAN BERBASIS
*INTERNET OF THINGS (IOT)***

PROYEK AKHIR

laporan akhir ini dibuat dan diajukan untuk memenuhi salah satu syarat kelulusan Sarjana Terapan/Diploma VI Politeknik Manufaktur Negeri Bangka Belitung



Disusun oleh :

Muhammad Rezi Madiza Arradea NIM : 1052120

Septia Eriska NIM : 1052125

**POLITEKNIK MANUFaktur NEGERI
BANGKA BELITUNG
TAHUN 2024**

LEMBAR PENGESAHAN

ALAT PENDETEKSI JATUH PADA LANSIA DALAM KEADAAN RAWAT JALAN BERBASIS *INTERNET OF THINGS (IOT)*

Oleh :

Muhammad Rezi Madiza Arradea / 1052120

Septia Eriska / 1052125

Laporan akhir ini disetujui dan disahkan sebagai salah satu syarat kelulusan
program Sarjana Terapan/Diploma III Politeknik Manufaktur Negeri
Bangka Belitung

Menyetujui

Pembimbing 1



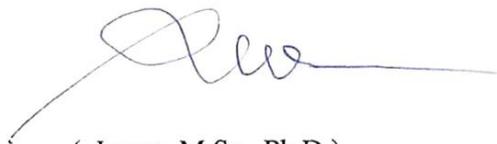
(Zanu Saputra, M.Tr.T.)

Pembimbing 2



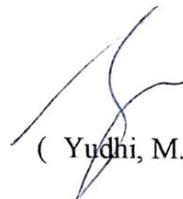
(Nur Khasanah, M.Si)

Penguji 1



(Irwan, M.Sc., Ph.D)

Penguji 2



(Yudhi, M.T)

PERNYATAAN BUKAN PLAGIAT

Yang bertanda tangan di bawah ini:

Nama Mahasiswa : Muhammad Rezi Madiza Arradea NIM : 1052120

Nama Mahasiswa : Septia Eriska NIM : 1052125

Dengan Judul : Alat Pendeteksi Jatuh Pada Lansia Dalam Keadaan Rawat
Jalan Berbasis *Internet of Things (IoT)*

Menyatakan bahwa laporan akhir ini adalah hasil kerja kami sendiri dan bukan merupakan plagiat. Pernyataan kami buat dengan sebenarnya dan bila ternyata dikemudian hari ternyata melanggar pernyataan ini, kami bersedia menerima sanksi yang berlaku.

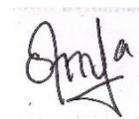
Sungailiat, 30 Juli 2024

Nama Mahasiswa
1. Muhammad Rezi Madiza Arradea

Tanda Tangan



2. Septia Eriska



ABSTRAK

Tingginya risiko jatuh pada lansia dapat menyebabkan cedera serius hingga kematian. Untuk mengatasi masalah ini, dikembangkan alat yang mampu mendeteksi jatuh secara real-time dan mengirimkan notifikasi kepada keluarga atau pengasuh. Alat ini menggunakan sensor MPU6050 untuk mendeteksi gerakan dan orientasi tubuh serta sensor MAX30100 untuk memantau kondisi detak jantung dan saturasi oksigen dalam darah. Data yang dikumpulkan oleh sensor diproses oleh mikrokontroler ESP32 yang kemudian mengirimkan informasi tersebut ke aplikasi seluler yang dibuat menggunakan Mitt App Inventor. Hasil uji coba menunjukkan bahwa alat pendeteksi jatuh memiliki akurasi yang baik dalam mendeteksi kejadian jatuh dan perubahan detak jantung. Alat ini berhasil memberikan notifikasi real-time yang efektif kepada pengguna melalui aplikasi seluler. Pengembangan alat ini diharapkan dapat meningkatkan kualitas hidup lansia dengan memberikan respons cepat terhadap kejadian jatuh dan memonitor kondisi kesehatan mereka secara berkelanjutan.

Kata Kunci: Pendeteksi jatuh, lansia, MPU6050, MAX30100, ESP32, Mitt App Inventor.

ABSTRACT

The high risk of falls in the elderly can cause serious injuries or even death. To address this issue, we developed a device that can detect falls in real-time and send notifications to family members or caregivers. The device uses an MPU6050 sensor to detect body movements and orientation and a MAX30100 sensor to monitor heart rate and blood oxygen saturation. Data collected by the sensors are processed by an ESP32 microcontroller, which then sends the information to a mobile application created using MIT App Inventor. Test results show that the fall detection device has good accuracy in detecting falls and changes in heart rate. The device successfully provides effective real-time notifications to users through the mobile application. The development of this device is expected to improve the quality of life for the elderly by providing a quick response to fall incidents and continuously monitoring their health conditions.

Keywords: Fall detection, elderly, MPU6050, MAX30100, ESP32, MIT App Inventor

KATA PENGANTAR

Assalamu'alaikum Warahmatullahi Wabarakatuh

Puji dan syukur penulis panjatkan ke hadirat Tuhan Yang Maha Esa atas segala rahmat dan karunianya sehingga penulis dapat menyelesaikan laporan proyek akhir yang berjudul “ Alat Pendeteksi Jatuh Pada Lansia Dalam Keadaan Rawat Jalan Berbasis *Internet Of Things(IoT)*”. Sholawat serta salam semoga senantiasa tercurahkan kepada junjungan kita Nabi Muhammad SAW yang telah membawa kita dari zaman kegelapan menuju zaman yang penuh dengan ilmu pengetahuan.

Adapun Tujuan penulis membuat Laporan proyek akhir ini adalah untuk memenuhi syarat salah satu kelulusan Pendidikan Sarjana Terapan di Politeknik Manufaktur Negeri Bangka Belitung. Proyek akhir ini bertujuan untuk merancang dan mengembangkan sebuah perangkat yang mampu mendeteksi jatuh pada lansia dalam keadaan rawat jalan, dengan menggunakan teknologi *internet of things(IoT)*. Perangkat ini diharapkan dapat memberikan kontribusi dalam meningkatkan kualitas hidup lansia dengan memberikan peringatan dini kepada keluarga atau pengasuh jika terjadi kejadian jatuh.

Penulis menyadari bahwa dalam penyusunan laporan ini tidak lepas dari bantuan dan dukungan berbagai pihak. Oleh karena itu, penulis ingin menyampaikan terima kasih yang sebesar-besarnya kepada pihak-pihak yang telah membantu baik secara langsung maupun tidak langsung, diantaranya:

1. Orang Tua tercinta penulis, Bapak Ahmad Jumadi dan Ibu Erliza selaku orang tua dari Muhammad Rezi Madiza Arradea serta Bapak Sobirin dan ibu Yulita selaku orang tua dari Septia Eriska yang selalu memberikan doa, dukungan, dan kasih sayang yang tiada henti. Terima kasih atas segala pengorbanan dan semangat yang diberikan selama ini.
2. Keluarga penulis, yang selalu memberikan doa, dukungan, dan kasih sayang yang tiada henti.

3. Bapak Zanu Saputra, M.Tr.T. Selaku Kepala Jurusan Teknik Elektro dan Informatika sekaligus sebagai dosen pembimbing 1 yang telah memberikan bimbingan, arahan, dan dukungan dalam penyelesaian proyek ini.
4. Ibu Nur Khasanah, M.Si. Selaku dosen pembimbing 2 yang telah memberikan bimbingan, arahan, dan dukungan dalam penyelesaian proyek ini.
5. Bapak I Made Andik Setiawan, M.Eng., Ph.D. Selaku Direktur di Politeknik Manufaktur Negeri Bangka Belitung.
6. Bapak Indra Dwisaputra, S.ST., M.T. Selaku Kaprodi D-IV Teknik Elektro dan Informatika di Politeknik Manufaktur Negeri Bangka Belitung.
7. Teman-teman kelas 3 TE A, yang selalu memberikan dukungan serta kebesamaan yang tak ternilai harganya sela proses penyusunan laporan ini.
8. Kepada teman kami yang bernama Febrian Athala yang sudah mau membantu, mengajarkan dan direpotkan selama pembuatan proyek akhir ini.
9. Seluruh responden yang telah dapat memberikan waktu dan informasi untuk membantu penyelesaian proyek akhir ini.
10. Semua pihak yang tidak dapat disebutkan satu per satu, yang telah dengan tulus ikhlas memberikan doa dan motivasi sehingga dapat terselesaikannya proyek akhir ini.

Penulis menyadari bahwa laporan ini masih jauh dari sempurna, oleh karena itu kritik dan saran yang membangun sangat diharapkan untuk perbaikan di masa mendatang.

Wassalamu'alaikum Warahmatullahi Wabarakatuh

DAFTAR ISI

	Halaman
HALAMAN JUDUL	
LEMBAR PENGESAHAN	i
PERNYATAAN BUKAN PLAGIAT	ii
ABSTRAK	iii
ABSTRACT	iv
KATA PENGANTAR.....	v
DAFTAR ISI	vii
DAFTAR TABEL	viii
DAFTAR TABEL	ix
DAFTAR GAMBAR	x
DAFTAR LAMPIRAN	xi
BAB I PENDAHULUAN	1
1.1 Latar Belakang Masalah	1
1.2 Perumusan Masalah	2
1.3 Batasan Masalah	3
1.4 Tujuan.....	3
BAB II DASAR TEORI.....	4
2.1 Definisi Lansia.....	4
2.2 Sistem Deteksi Jatuh.....	5
2.2.1 Internet Of Things (IoT).....	8
2.2.2 Mikrokontroler ESP32	11
2.2.3 Sensor MPU6050	15
2.2.4 Sensor MAX30100.....	19
2.3 Software Arduino IDE (Integrated Development Environment).....	24
2.4 Mitt App Inventor.....	25
BAB III METODE PENELITIAN	27
3.1 Studi Literatur.....	28

3.2 Perancangan Hardware dan Software.....	28
3.2.1 Sistem Kerja Alat	28
3.2.2 Perancangan Hardware.....	29
3.2.3 Perancangan Software	34
3.3 Pembuatan Alat	36
3.4 Pengujian Alat	38
3.5 Perbaikan	39
3.6 Pengambilan Data.....	39
3.7 Analisis Hasil.....	39
3.8 Pembuatan Laporan	40
BAB IV HASIL DAN PEMBAHASAN.....	41
4.1 Bentuk Alat.....	41
4.2 Pengujian Sensor	42
4.2.1 Pengujian Sensor MPU6050	43
4.2.2 Pengujian Sensor MAX30100.....	52
4.3 Pengujian aktivasi Sistem Melalui Smartphone	58
BAB V KESIMPULAN DAN SARAN	61
5.1 Kesimpulan.....	61
5.2 Saran.....	61
DAFTAR PUSTAKA.....	63
LAMPIRAN 1 Daftar Riwayat Hidup.....	68
LAMPIRAN 2 Program	69

DAFTAR TABEL

Tabel 2 1 PinOut ESP32 [21]	11
Tabel 2 2 Perbedaan ESP32 dengan Mikrokontroler lainnya.....	12
Tabel 2 3 Spesifikasi Sensor Gyroscope MPU6050.....	16
Tabel 2 4 Spesifikasi Pin Konfigurasi MAX30100[30]	20
Tabel 2 5 Perbedaan Sensor MAX30100 dengan sensor lainnya.....	21
Tabel 3 1 Penelitian Sebelumnya	28
Tabel 3 2 Komponen yang Digunakan	31
Tabel 4 1 Pengujian Deteksi Jatuh Pertama	48
Tabel 4 2 Pengujian Deteksi Jatuh Ke Dua.....	50
Tabel 4 3 Perbandingan Pengujian Pertama	56

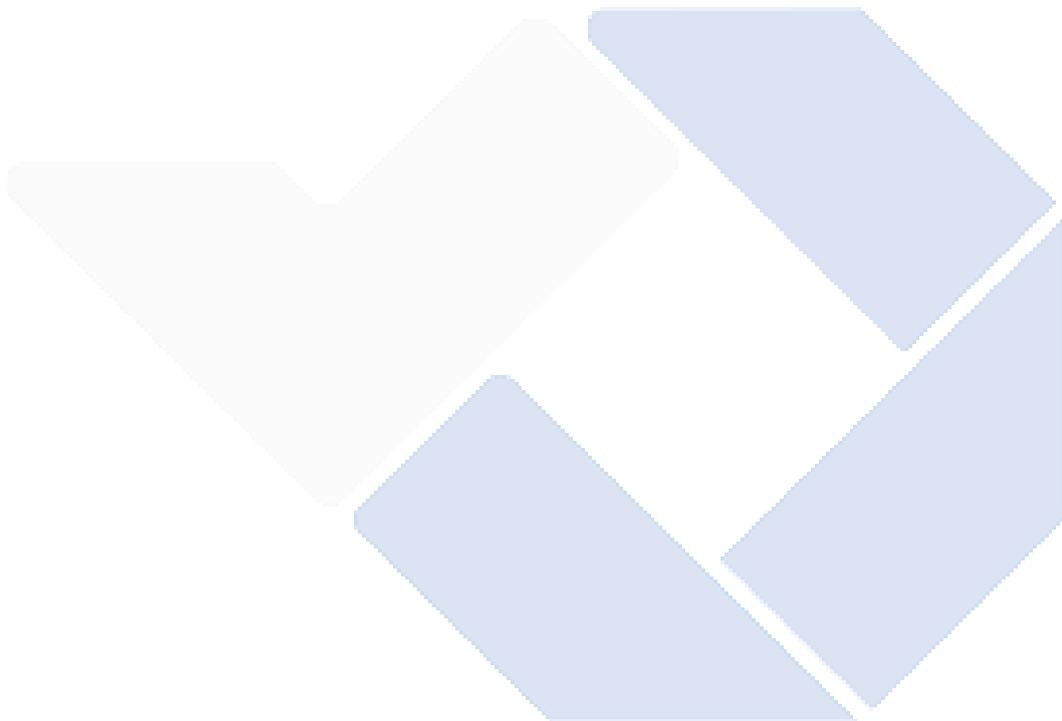
DAFTAR GAMBAR

<i>Gambar 2 1 Diagram Rancangan Sistem Deteksi Jatuh.....</i>	<i>6</i>
<i>Gambar 2 2 Prototype Alat</i>	<i>6</i>
<i>Gambar 2 3 Prototype Alat</i>	<i>10</i>
<i>Gambar 2 4 Bagian-bagian Pin ESP32</i>	<i>11</i>
<i>Gambar 2 5 Spesifikasi Sensor Gyroscope MPU6050.....</i>	<i>16</i>
<i>Gambar 2 6 Titik Kemiringan atau Putaran Pada MPU6050</i>	<i>17</i>
<i>Gambar 2 7 (a) Sensor MAX30100 (b) Tampilan Pin Konfigurasi MAX30100</i>	<i>20</i>
<i>Gambar 2 8 Software Arduino IDE.....</i>	<i>24</i>
<i>Gambar 3 1 FlowChart Proses Pelaksanaan Proyek Akhir</i>	<i>27</i>
<i>Gambar 3 2 Blok Diagram Sistem Kerja Alat.....</i>	<i>29</i>
<i>Gambar 3 3 Desain Alat.....</i>	<i>31</i>
<i>Gambar 3 4 Wiring Diagram</i>	<i>31</i>
<i>Gambar 3 5 Block Diagram Pada Aplikasi Mitt App Inventor</i>	<i>34</i>
<i>Gambar 3 6 Firebase</i>	<i>35</i>
<i>Gambar 3 7 Pembuatan Box Menggunakan 3D Printing</i>	<i>36</i>
<i>Gambar 3 8 Penyusunan Komponen-komponen dalam Box.....</i>	<i>38</i>
<i>Gambar 4 1 Bentuk Fisik Alat.....</i>	<i>41</i>
<i>Gambar 4 2 Penggunaan Alat Pada Pengguna</i>	<i>42</i>
<i>Gambar 4 3 Wiring Diagram Sensor MPU6050 dengan Arduino Nano.....</i>	<i>43</i>
<i>Gambar 4 4 Program MPU6050 dengan Arduino Nano.....</i>	<i>45</i>
<i>Gambar 4 5 Posisi Pengujian Alat Pada Pengguna.....</i>	<i>47</i>
<i>Gambar 4 6 Serial Monitor Pada Pengujian Pertama.....</i>	<i>50</i>
<i>Gambar 4 7 Tampilan Serial Monitor Pengujian Ke Dua</i>	<i>51</i>
<i>Gambar 4 8 Wiring Diagram Sensor MAX30100 dengan Arduino Nano</i>	<i>52</i>
<i>Gambar 4 9 Sensor MAX30100 dengan Arduino Nano</i>	<i>53</i>
<i>Gambar 4 10 Pengujian Menggunakan Tensimeter</i>	<i>56</i>
<i>Gambar 4 11 Tampilan Mitt App Inventor.....</i>	<i>59</i>

DAFTAR LAMPIRAN

Lampiran 1 : Daftar Riwayat Hidup

Lampiran 2 : Program



BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Jatuh merupakan kejadian yang berbahaya bagi manusia karena dapat berpengaruh terhadap kerusakan serta kesehatan tubuh. Terutama bagi orang yang sudah lanjut usia (lansia), jatuh dapat menimbulkan dampak yang sangat fatal. Resiko jatuh pada lansia terbilang lebih tinggi dibandingkan dengan usia muda. Hal ini dikarenakan fungsi otak pada lansia sudah mulai menurun dan terjadi keterbatasan gerak [1]. Jatuh menjadi ancaman yang paling diantisipasi pada lansia karena berpotensi menyebabkan kematian.

Jatuh pada lansia dapat menyebabkan cedera serius seperti patah tulang, cedera , trauma internal dan psikis. Data dari WHO menunjukkan bahwa satu dari setiap lansia berusia 65 tahun ke atas mengalami jatuh setiap tahunnya, dan angka tersebut meningkat seiring bertambahnya usia. Di Indonesia, laporan dari Riskesdas 2018 menunjukkan prevalensi jatuh pada lansia mencapai 28,8% [2].

Dengan berkembangnya teknologi internet menawarkan peluang untuk menciptakan solusi inovatif untuk membantu mengatasi permasalahan tersebut. Oleh karena itu, penelitian ini dilakukan dengan judul “Alat pendeteksi jatuh pada lansia dalam keadaan rawat jalan berbasis *internet of things*” menggunakan sensor MPU6050 dan ESP32 serta tampilan pada LCD dan MIT App Inventor, judul ini dipilih karena urgensinya untuk memberikan solusi praktis yang efektif terhadap masalah jatuh dikalangan masyarakat.

Untuk mengatasi permasalahan ini, diperlukan sebuah alat yang dapat mendeteksi jatuh secara *real-time* dan memberikan notifikasi kepada keluarga atau pengasuh. Alat yang diusulkan terdiri dari beberapa komponen utama berupa sensor MPU6050, sensor ini digunakan untuk mendeteksi gerakan dan orientasi. MPU6050 dilengkapi akselerometer dan *gyroscope* yang mampu mengukur akselerasi dan rotasi pada tiga sumbu, memungkinkan deteksi jatuh secara akurat. Sensor detak jantung, sensor ini digunakan untuk status kesehatan jantung orang

lanjut usia. ESP32 modul mikrokontroler ini berfungsi sebagai pusat pemrosesan data. ESP32 dilengkapi dengan konektivitas *Wi-Fi* dan *Bluetooth* memungkinkan transmisi data *real-time* ke aplikasi seluler. Layar *LCD* digunakan untuk menampilkan status peralatan dan notifikasi langsung ke layar. MIT App Inventor, Platform ini digunakan untuk membuat aplikasi seluler yang dapat menerima notifikasi dari perangkat pendeteksi jatuh. Aplikasi ini juga memungkinkan pemantauan kondisi orang lanjut usia secara *real-time* oleh keluarga.

Alat ini bekerja dengan cara terus memantau pergerakan dan kesehatan jantung para lansia. MPU6050 akan mengirimkan data akselerasi dan rotasi ke ESP32, sedangkan sensor detak jantung akan mengirimkan data detak jantung. Jika pola gerakan yang terdeteksi sesuai dengankriteria jatuh atau terdapat terjadi perubahan detak jantung yang signifikan, ESP32 akan mengirimkan sinyal pada *LCD* ke aplikasi seluler. Pengembangan alat pendeteksi jatuh *Internet of things* ini merupakan solusi yang efektif untuk meningkatkan taraf hidup dan kualitas hidup para lansia. Dengan menggabungkan sensor MPU6050, sensor detak jantung, mikrokontroler ESP32 dan aplikasi seluler yang mudah digunakan, hal ini dapat memberikan respons cepat terhadap jatuh dan kondisi kesehatan, dan memungkinkan pemantauan berkelanjutan terhadap kondisi lansia.

1.2 Perumusan Masalah

Adapun rumusan masalah yang diangkat berdasarkan latar belakang proyek akhir ini antara lain :

1. Bagaimana membuat alat yang dapat mendeteksi jatuh pada lansia yang sedang rawat jalan?
2. Bagaimana cara Aplikasi MIT App Inventor bisa *auto running* ketika lansia terdeteksi jatuh?
3. Bagaimana merancang sistem yang dapat memonitoring kondisi detak jantung dan saturasi oksigen dalam darah pada lansia?

1.3 Batasan Masalah

1. Alat ini dirancang khusus untuk memantau kejadian jatuh dan tidak memantau aktivitas lainnya.
2. Penempatan posisi alat di pinggang pengguna
3. Alat ini bergantung pada koneksi internet untuk mengirim notifikasi
4. Daya tahan baterai alat ini terbatas dan perlu diisi ulang secara berkala.

1.4 Tujuan

1. Membuat alat yang mampu mendeteksi berupa jatuh ke depan dan belakang dengan kemiringan 45° , jatuh kedepan dan belakang dengan kemiringan 90° , jatuh ke kiri dan jatuh ke kanan.
2. Membantu keluarga lansia dengan mengirimkan peringatan berupa suara pada buzzer dan tampilan di aplikasi Mitt App Inventor.
3. Merancang sistem yang dapat memonitoring kondisi detak jantung dan saturasi oksigen dalam darah pada lansia.

BAB II

DASAR TEORI

1.1 Definisi Lansia

Salah satu indikator keberhasilan pembangunan nasional dari sisi kesehatan adalah semakin meningkatnya usia harapan hidup penduduk. Meningkatnya usia harapan hidup penduduk, menyebabkan jumlah penduduk lanjut usia (lansia) terus meningkat dari tahun ke tahun. Definisi lansia menurut UU Nomor 13 Tahun 1998 tentang kesejahteraan, lansia adalah penduduk yang telah mencapai usia 60 tahun ke atas [3].

Jumlah penduduk lansia di Indonesia pada tahun 2019 sebanyak 25 juta jiwa. Jumlah lansia diprediksi setiap tahunnya akan terus meningkat. Nilai tersebut diperoleh berdasarkan perhitungan jumlah penduduk yang berusia lebih dari 60 tahun sesuai dengan Peraturan Pemerintah Republik Indonesia Nomor 43 tahun 2004. Di Indonesia angka kesakitan Lansia pada tahun 2019 sebesar 25%, artinya setiap 4 orang lansia 1 orang diantaranya mengalami sakit. Lansia memiliki keterbatasan fisik antara lain seperti keterbatasan penglihatan , kemampuan berjalan, dan pendengaran. Berdasarkan kondisi tersebut potensi terjadinya jatuh pada lansia cukup tinggi [4-7].

Menua adalah proses yang ilmiah, hilangnya perlahan-lahan kemampuan jaringan organ tubuh, memperbaiki atau mengganti untuk mempertahankan struktur dan fungsi normalnya, bersifat irreversibel dan dialami oleh semua makhluk. Bertambahnya usia, timbul perubahan-perubahan sebagai akibat proses menua (*aging process*), meliputi perubahan fisik, mental, spiritual dan psikososial. Lansia merupakan proses tumbuh kembang manusia yang tidak secara tiba-tiba menjadi tua, tetapi berkembang dari bayi dan akhirnya menjadi tua. Hal ini normal, dengan perubahan mencapai usia tahap perkembangan kronologis tertentu [8].

Kecelakaan jatuh dan akibatnya memerlukan perhatian medis dan seringkali merupakan penyebab patah tulang, cedera otak traumatik dan lesi pada anggota

tubuh bagian atas yang menginduksi, dalam banyak kasus, dengan hilangnya kemandirian atau kematian. Seringkali orang tua tidak dapat kembali ke posisi berdiri setelah jatuh dan ada hubungan erat antara penundaan dalam membantu cedera dan tingkat kematian [9].

Akibat dari jatuh pada lansia adalah rusaknya jaringan lunak yang terasa sakit berupa robek atau tertariknya jaringan otot, robeknya arteri/vena, patah tulang, hematoma, kecacatan dan meninggal. Untuk mencegah agar tidak jatuh pada lansia dengan cara mengidentifikasi faktor resiko, menilai dan mengawasi. Pada prinsipnya mencegah terjadinya jatuh sangat penting pada usia lanjut [10].

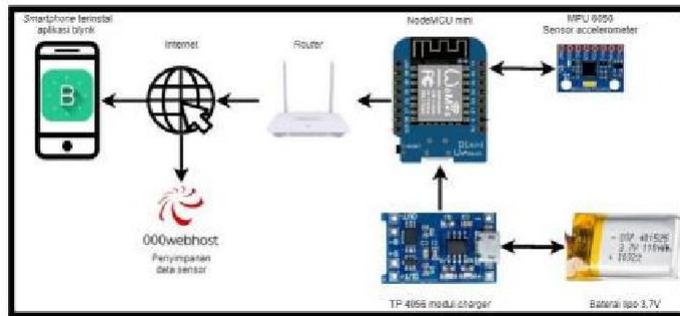
Pengawasan sangat bagi lanjut usia yang potensi jatunya tinggi. Pengawasan dapat dilakukan oleh anggota keluarga atau orang terdekat. Orang yang bertugas mengawasi tentunya dituntut untuk selalu berada di dekat lanjut usia agar lebih mudah mengetahui dan memberikan pertolongan jika lanjut usia mengalami jatuh. Akan tetapi, sebagai anggota keluarga tidak mungkin selamanya berada di dekat lansia untuk melakukan pengawasan secara langsung. Oleh karena itu, pengawasan dapat dilakukan secara tidak langsung dengan memanfaatkan teknologi yang ada saat ini. Salah satunya adalah dengan menggunakan perangkat yang dapat mendeteksi jatuh [11].

1.2 Sistem Deteksi Jatuh

Kemajuan teknologi telah membuka jalan bagi berbagai inovasi yang memiliki tujuan jelas dalam meningkatkan kualitas hidup dan kesehatan lansia. Salah satu teknologi yang mendapat perhatian besar adalah alat pendeteksi jatuh. Sistem ini dirancang untuk mendeteksi lansia ketika terjatuh, mengingat bahaya serius yang diimbulkan jika tidak dilakukan tindakan pertolongan dengan cepat.

Penelitian tentang alat pendeteksi jatuh berbasis IoT telah berkembang pesat dalam beberapa tahun terakhir. Salah satu studi yang dilakukan oleh Ryan Ady Putera Effendy yang berjudul “Perancangan Prototipe Pendeteksi Gerakan Jatuh Pada Lansia Menggunakan Sensor *Accelerometer* Berbasis IoT”, Metode penelitian ini menggunakan perangkat dengan sensor MPU6050 yang dikalibrasi dalam beberapa posisi berbeda untuk mengukur akurasi sensor. Pengujian

dilakukan untuk mengukur ketahanan baterai serta menentukan tingkat akurasi alat menggunakan rumus tertentu. Diagram rancangan sistem deteksi jatuh pada Gambar 2.1 di bawah ini.



Gambar 2 1 Diagram Rancangan Sistem Deteksi Jatuh

Mengenai diagram perancangan sistem deteksi jatuh pada gambar 2.1 Sensor MPU6050 mendeteksi gerakan kemudian nodeMCU mengolah data dan mendeteksi jatuh, setelah itu data akan dikirim melalui router ke aplikasi Blynk, kemudian aplikasi blynk menampilkan notifikasi jatuh dan akan disimpan di *000WebHost* untuk akses lebih lanjut. TP4556 memastikan baterai LiPo 3.7V tetap terisi untuk mendukung daya sistem. Kajian penelitian ini menunjukkan bahwa penggunaan sensor akselerometer pada perangkat pendeteksi jatuh cukup akurat dalam berbagai kondisi postur tubuh. Berikut dibawah ini Gambar Prototipe Alat dan Pemasangan Alat



Gambar 2 2 Prototipe Alat[12]

Pada Gambar 2.2 (a) *Prototype* Alat deteksi jatuh dirancang agar portabel dan mudah digunakan oleh lansia. Dengan desain yang ringan dan komponen yang ringkas, alat ini dapat dengan mudah dibawa kemana saja, memastikan keamanan lansia dalam berbagai situasi.

Sementara itu penelitian yang dilakukan oleh Syifa Dzikri Tsani dan Indra Herdian Mulyadi yang berjudul “ Sistem Pendeteksi Jatuh *Wearable* untuk Lanjut Usia Menggunakan *Accelerometer* Dan *Gyroscope*”. Penelitian ini mengembangkan sistem pendeteksi jatuh menggunakan sensor akselerometer dan giroskop berbasis mikrokontroler dengan metode *threshold*. Sistem ini mengumpulkan data percepatan dan kecepatan sudut dari sensor, yang diproses untuk mendeteksi kejadian jatuh. Alat ini bersifat *wearable*, dipasang di pinggang pengguna, dan terhubung ke *smartphone* melalui *Bluetooth* untuk memberikan notifikasi jatuh. Hasil penelitian menunjukkan bahwa sistem ini memiliki sensitivitas sebesar 82,50%, spesifisitas sebesar 91,67%, dan akurasi sebesar 88%. Selain itu, tingkat keberhasilan *smartphone* dalam menampilkan notifikasi jatuh mencapai 96,97%. Kelebihan dari sistem ini adalah kemampuannya untuk memantau kondisi pengguna secara *real-time* dan mengirim notifikasi secara cepat jika terjadi jatuh. Namun, kekurangannya termasuk ketergantungan pada koneksi *Bluetooth* dan daya tahan baterai yang terbatas. Secara keseluruhan, IoT pada penelitian ini menunjukkan potensi besar dalam membantu pengawasan dan penanganan insiden jatuh pada lanjut usia, meskipun masih terdapat beberapa tantangan teknis yang perlu diatasi untuk peningkatan lebih lanjut[13].

Kemudian Penelitian yang dilakukan oleh Fajar Akbar Maulana yang berjudul “ Sistem Pendeteksi Jatuh Untuk Manusia Lanjut Usia Berbasis Arduino” sistem penelitian ini menggunakan metode prototipe, yang meliputi tahapan pengumpulan kebutuhan perangkat keras, pembangunan prototipe perangkat keras, evaluasi prototipe perangkat keras, pengkodean sistem, pengujian sistem, evaluasi sistem, dan penggunaan sistem. Hasil Pengujian menunjukkan bahwa sistem deteksi jatuh yang dikembangkan memiliki sensitivitas sebesar 91% dan spesifisitas sebesar 96%, yang berarti sistem ini cukup baik dalam mendeteksi

aktivitas jatuh dan aktivitas biasa. Sistem ini membutuhkan rata-rata waktu 6 detik setelah aktivitas jatuh untuk mengirimkan notifikasi kepada anggota keluarga pengguna. Sistem deteksi jatuh berbasis arduino dengan sensor MPU6050, modul GSM, sensor mic, modul micro SD, dan sumber daya dari baterai 9V, serta menggunakan perangkat lunak Arduino IDE 1.8.5, berhasil mendeteksi jatuh dengan tingkat akurasi yang tinggi dan memberikan notifikasi dalam waktu yang relatif cepat [14].

1.2.1 *Internet of Things(IoT)*

Internet of Things adalah suatu konsep yang bertujuan untuk memanfaatkan teknologi internet yang terus berkembang agar dapat diimplementasikan ke dalam benda fisik sehingga manusia dapat berinteraksi langsung dengan benda tersebut seperti mengirim data dan melakukan kendali jarak jauh secara *real-time*. Makna lain serupa, *Internet of Things (IoT)* adalah sebuah konsep di mana suatu objek yang memiliki kemampuan untuk mentransfer data melalui jaringan internet tanpa melakukan interaksi manusia atau manusia ke komputer [15].

Alat pendeteksi jatuh berbasis *Internet of Things* merupakan aplikasi penting dari teknologi ini untuk meningkatkan kehidupan individu, khususnya mereka yang rentan terhadap risiko jatuh, seperti pasien lanjut usia yang menderita kondisi medis tertentu. Pada dasarnya, teknologi ini menggunakan sensor yang tertanam pada perangkat atau lingkungan untuk mendeteksi perubahan pada posisi tubuh yang dapat mengindikasikan terjatuh. Penelitian mengenai IoT pada alat pendeteksi jatuh sudah banyak dilakukan oleh para peneliti dari berbagai perguruan tinggi di Indonesia, terutama untuk keperluan tugas akhir. Pada penelitian sebelumnya, alat pendeteksi jatuh berbasis IoT ini sudah dibahas dalam berbagai pokok permasalahan, mikrokontroler yang digunakan, hingga komponen penyusun lainnya.

Dalam penelitian yang dilakukan oleh Shandy Dwijoseputra yang berjudul “Sistem Deteksi Jatuh Berbasis *Internet of Things*”. Penelitian ini mengembangkan sistem deteksi jatuh menggunakan sensor akselerometer yang

terintegrasi dengan teknologi *Internet of Things (IoT)*. IoT memungkinkan berbagai perangkat berkomunikasi dan bertukar data melalui internet. Dalam penelitian ini, sensor akselerometer dipasang pada pengguna dan terhubung ke internet menggunakan modul *NodeMCU* dan SIM808. Sistem ini mengirimkan data lokasi dan kondisi jatuh secara *real-time* ke server web, yang kemudian dapat diakses melalui aplikasi Android. Hasil pengujian menunjukkan bahwa sistem memiliki akurasi tinggi dalam mendeteksi jatuh dan pengiriman data rata-rata memerlukan waktu 5,5 detik. Kelebihan dari penggunaan IoT dalam sistem ini termasuk kemampuan untuk memantau kondisi pengguna secara terus-menerus dan memberikan respons cepat saat terjadi insiden jatuh. Namun, tantangannya adalah ketergantungan pada koneksi internet yang stabil untuk memastikan data dikirim dan diterima dengan tepat waktu [16].

Penelitian yang dilakukan oleh Gigih Gumilar dan Hendi Handian Rachmat dengan judul "*Sistem Pendeteksi Jatuh Wireless Berbasis Sensor Accelerometer*". Metode Penelitian Penelitian oleh Gigih Gumilar dan Hendi Handian Rachmat menggunakan sensor *accelerometer* untuk mendeteksi jatuh secara *real-time* pada lansia. Sistem ini terdiri dari dua bagian: pengirim dan penerima, yang berkomunikasi secara *wireless* menggunakan frekuensi radio 2,4 GHz. Sensor *accelerometer* pada bagian pengirim mendeteksi tiga sumbu percepatan gravitasi dan nilai magnitudo saat terjadi jatuh, yang kemudian diolah oleh Arduino Nano. Pengujian dilakukan dengan menempatkan sistem pengirim pada pinggang naracoba untuk mendeteksi dua tipe jatuh: jatuh ke depan dan jatuh ke belakang. Data yang diolah kemudian dikirimkan ke bagian penerima untuk dianalisis lebih lanjut. Bagian penerima akan ditampilkan hasil deteksi dan memberikan notifikasi jika jatuh terdeteksi, memastikan respons cepat terhadap insiden jatuh.

Berikut dibawah ini pada Gambar 2.3 *Prototype Alat*



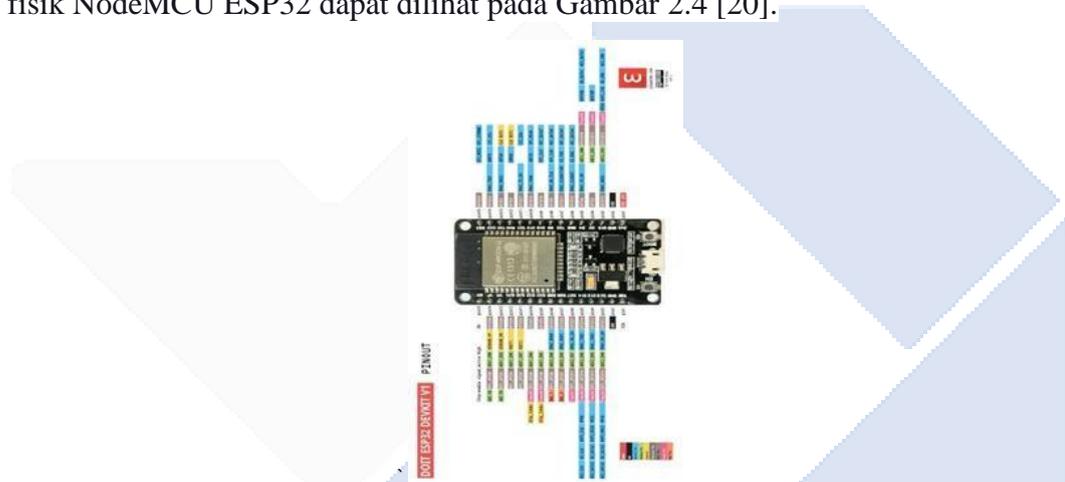
Gambar 2 3 Prototype Alat[17]

Pada Gambar 2.3 *prototype* alat sistem ini juga membuktikan bahwa pengawasan lansia menggunakan perangkat *wearable* dan *wireless* dapat memberikan respons cepat dan efektif saat terjadi jatuh. Hasil Keseluruhan Penelitian ini berhasil mengembangkan sistem pendeteksi jatuh yang *real-time*, *wearable*, dan *ambulatory* dengan menggunakan komunikasi *wireless*.

Dalam hal ini, IoT berperan sebagai penghubung utama yang memungkinkan sensor dan modul komunikasi berinteraksi secara efisien untuk mendeteksi kejadian jatuh dan mengirimkan data tersebut secara *real-time*. IoT memungkinkan perangkat keras yang terdiri dari sensor MPU6050, NodeMCU ESP32 untuk bekerja sama dan memberikan informasi penting mengenai kondisi pengguna ke aplikasi Mit App Inventor. Data ini kemudian dapat diakses melalui aplikasi Android, yang menyediakan alat pemantau yang efektif bagi keluarga atau pengasuh untuk memberikan respons cepat saat terjadi insiden jatuh. Melihat demikian besarnya potensi dari pemanfaatannya, IoT sangat dimungkinkan untuk diimplementasikan untuk alat pendeteksi jatuh pada lansia, karena kemampuannya mengintegrasikan sensor yang mampu memantau aktivitas dan pergerakan lansia tepat waktu.

1.2.2 Mikrokontroler ESP32

Mikrokontroler ESP32 dibuat oleh perusahaan bernama *Espressif Systems*, perusahaan berbasis di Shanghai, Tiongkok [18]. Salah satu kelebihan yang dimiliki oleh ESP32 yaitu sudah terdapat *Wi-Fi* dan *Bluetooth* di dalamnya, sehingga akan sangat memudahkan ketika belajar membuat sistem IoT yang memerlukan koneksi *wireless* [19]. Modul ini dapat digunakan untuk aplikasi lain seperti kontrol sistem, monitoring, dan lainnya. ESP32 memiliki fitur *deep sleep* untuk menghemat daya dengan mematikan modul saat tidak digunakan. Tampilan fisik NodeMCU ESP32 dapat dilihat pada Gambar 2.4 [20].



Gambar 2.4 Bagian-bagian Pin ESP32

Tabel 2.1 PinOut ESP32 [21]

15 ADC Channels	15 channels dengan 12 bit SAR ADC rentang 0-1v, 0-1, 4V, 0-2V atau 0-4V
2 UART Interface	2 interface UART dengan kontrol aliran dan suport IrDA
25 PWM Outputs	25 pin PWM untuk mengontrol hal-hal seperti kecepatan motor atau kecerahan LED
2 DAC Channels	Dua DAC 8 bit untuk menghasilkan <i>voltage</i> analog
SPL, I2C, dan I2S Interface	Tiga interface SPI dan satu I2C untuk

	menguhungkan berbagai sensor dan periferel, sera dua antarmuka 12S untuk menambahkan suara ke projek
9 Touch Pads	9 GPIO

Pada board ESP32, terdapat beberapa bagian penting yang harus diketahui. Pin VIN digunakan untuk menyambungkan tegangan 5V yang memberikan daya kepada ESP32. Pin 3V3 menyediakan tegangan 3.3V untuk komponen lain yang membutuhkan tegangan ini. Pin GND adalah pin ground yang digunakan sebagai referensi tegangan nol. Selain itu, ada pin ADC yang berfungsi membaca sinyal analog dari sensor dan mengubahnya menjadi data digital. ESP32 juga memiliki banyak pin GPIO yang dapat diatur sebagai input atau output, sehingga bisa mengendalikan atau membaca status perangkat seperti LED atau tombol.

Ada juga pin khusus untuk mendeteksi sentuhan, yang disebut *touch pins*, mirip seperti layar sentuh pada *smartphone*. Untuk komunikasi, ESP32 dilengkapi dengan pin UART yang memungkinkan komunikasi serial, pin SPI untuk komunikasi cepat dengan perangkat lain seperti sensor atau layar, dan pin I2C yang memungkinkan komunikasi dengan beberapa perangkat menggunakan hanya dua kabel. Pin EN digunakan untuk mengaktifkan atau menonaktifkan ESP32, sementara pin BOOT digunakan untuk memasukkan ESP32 ke mode khusus saat memulai. Terakhir, ada pin CH_PD yang digunakan untuk menonaktifkan *chip* ESP32 saat tidak digunakan. Bagian-bagian ini membuat ESP32 sangat fleksibel untuk berbagai aplikasi IoT dan proyek elektronik lainnya. Perbedaan ESP32 dengan mikrokontroler lain dipaparkan pada tabel 2.2.

Tabel 2 2 Perbedaan ESP32 dengan Mikrokontroler lainnya

	Arduino Uno	Node MCU (ESP8266)	ESP32
Tegangan	5 Volt	3.3 Volt	3.3 Volt
CPU	Atmega328- 16MHz	<i>Xtensa single core</i> L106-60 MHz	<i>Xtens a dual core</i> LX6-160MHz

Arsitektur	8 bit	32 bit	32 bit
<i>Flash Memory</i>	32KB	16MB	16MB
SRAM	2KB	160KB	512KB
GPIO Pin (ADC/DAC)	14(6/-)	17(1/-)	36(18/2)
<i>Bluetooth</i>	Tidak ada	Tidak ada	Ada
WiFi	Tidak ada	Ada	Ada
SPI/I2C/UART	1/1/1	2/1/2	4/2/2

Terlihat perbedaan pada tabel 2.2 yang menjadi keunggulan mikrokontroler ESP32 dibanding dengan mikrokontroler yang lain, mulai dari pin out nya yang lebih banyak, pin analog lebih banyak, memori yang lebih besar, terdapat *bluetooth 4.0 low energy* serta tersedia *WiFi* yang memungkinkan untuk mengaplikasikan *Internet of Things* dengan mikrokontroler ESP32.

Penelitian yang dilakukan oleh Leonardo Hermaditya Chesa dan Agustinus Bayu Primawan yang berjudul “ Monitoring Kemiringan Benda Berbasis *SMS Gateway*”. Dalam penelitian ini ESP32 digunakan sebagai pusat kontrol utama untuk sistem pemantauan kemiringan. ESP32 adalah mikrokontroler dengan integrasi tinggi yang menyediakan konektivitas WiFi dan *Bluetooth*, namun dalam penelitian ini lebih menonjolkan kemampuannya dalam mengolah data sensor dan berkomunikasi modul GSM. Konektivitas dengan sensor ESP32 memiliki banyak *General Purpose Input/Output* (GPIO) yang memungkinkan koneksi langsung dengan sensor *gyroscope* MPU6050. Data dari sensor ini terus-menerus dibaca dan dianalisis oleh ESP32 untuk memantau perubahan sudut kemiringan objek. Meskipun ESP32 memiliki kemampuan konektivitas WiFi dan *Bluetooth*, penelitian ini menggunakan modul SIM900A GSM untuk mengirimkan notifikasi melalui SMS. ESP32 mengirimkan perintah ke modul SIM900A menggunakan *AT Command* untuk mengirim, memeriksa, dan menghapus pesan SMS. Ini menunjukkan fleksibilitas ESP32 dalam berintegrasi dengan berbagai modul komunikasi. Dengan semua fitur dan

kemampuan ini, ESP32 memainkan peran penting dalam memungkinkan sistem monitoring kemiringan yang efisien dan dapat diandalkan, serta menawarkan potensi untuk pengembangan lebih lanjut dengan memanfaatkan fitur konektivitas nirkabelnya yang lain [22].

Penelitian yang dilakukan oleh Dadan Darmawan, Arnisa Stefanie, dan Dian Budhi Santoso yang berjudul “Perancangan Sistem Instrumentasi Berbasis *Internet Of Things* Pada Alat Pendeteksi Bangkitan Kejang Pengidap Epilipsis” berfokus pada pengembangan alat pendeteksi kejang yang dapat dipantau dari jarak jauh oleh keluarga pasien. Alat ini menggunakan beberapa komponen utama, yaitu sistem sensor EEG, sensor LDR, sensor SW-520D (pendeteksi kemiringan), GPS Ublok Neo-6M, dan mikrokontroler ESP32. Penelitian ini menggunakan metode penelitian dan pengembangan yang melibatkan perancangan perangkat keras dan perangkat lunak. Perangkat keras mencakup sensor EEG yang dipasang di kepala pasien, sensor SW-520D untuk mendeteksi jatuh, GPS untuk mendeteksi lokasi, dan sensor LDR untuk mendeteksi intensitas cahaya. Data dari sensor-sensor ini diproses oleh mikrokontroler ESP32 dan dikirimkan ke aplikasi Android melalui database yang dirancang menggunakan Mitt App Inventor. Pengujian dilakukan untuk mengukur akurasi setiap sensor dan komponen.

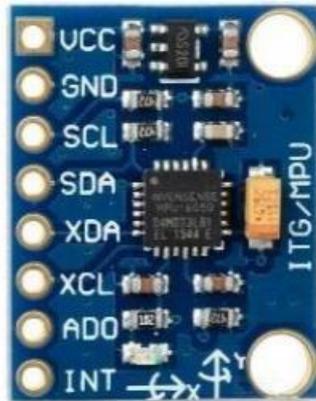
Penelitian ini meninjau beberapa studi terkait sebelumnya, termasuk penelitian tentang alat pendeteksi kejang berbasis IoT dan aplikasi monitoring penderita epilepsi menggunakan sensor akselerometer pada *smartphone* Android. Kajian literatur ini membantu dalam merumuskan metode penelitian dan perancangan alat yang lebih efektif. Hasil pengujian menunjukkan bahwa sensor EEG mampu mendeteksi perubahan amplitudo sinyal otak dengan penguatan sebesar 4213 kali. Sensor LDR memiliki tingkat akurasi 71.12% dalam mendeteksi intensitas cahaya, sementara sensor SW-520D memiliki akurasi 96.7% dalam mendeteksi jatuh. GPS Ublox Neo-6M menunjukkan perbedaan rata-rata jarak sebesar 2.90 meter dibandingkan dengan GPS *smartphone*. Pembahasan dalam penelitian ini menekankan pentingnya akurasi dan respons

cepat dari setiap komponen untuk mendeteksi kejang secara efektif. Mikrokontroler ESP32 dipilih karena kemampuannya untuk mengolah data sensor secara cepat dan mengirimkan informasi *real-time* melalui internet. ESP32 memiliki keunggulan dalam hal konektivitas Wi-Fi, daya rendah, dan kapasitas pemrosesan yang memadai untuk aplikasi IoT [23].

ESP32 dipilih dalam pembuatan alat pendeteksi jatuh pada lansia berbasis IoT karena kemampuannya yang baik dalam mengelola sensor-sensor yang berbeda, fleksibilitasnya dalam komunikasi nirkabel, biaya yang relatif rendah. Platform ini menawarkan solusi yang efektif dan efisien untuk memonitor aktivitas harian lansia secara *real-time* dan memberikan respons cepat terhadap situasi darurat seperti jatuh, membuatnya ideal untuk implementasi dalam konteks perawatan kesehatan rawat jalan.

1.2.3 Sensor MPU6050

MPU6050 adalah *chip IC inverse* yang di dalamnya terdapat sensor *accelerometer* dan *gyroscope* yang sudah terintegrasi. *Accelerometer* digunakan untuk mengukur percepatan, percepatan gerakan dan juga percepatan gravitasi. *Accelerometer* sering digunakan untuk menghitung sudut kemiringan, dan hanya dapat melakukan dengan nyata ketika statis dan tidak bergerak. Untuk mendapatkan sudut akurat kemiringan, sering dikombinasikan dengan satu atau lebih *gyroscope* dan kombinasi data yang digunakan untuk menghitung sudut. *Groscope* adalah perangkat untuk mengukur atau mempertahankan orientasi, yang berlandaskan pada prinsip-prinsip momentum sudut [24]. Tampilan fisik MPU6050 dapat dilihat pada gambar 2.5.



Gambar 2.5 Spesifikasi Sensor Gyroscope MPU6050 [24]

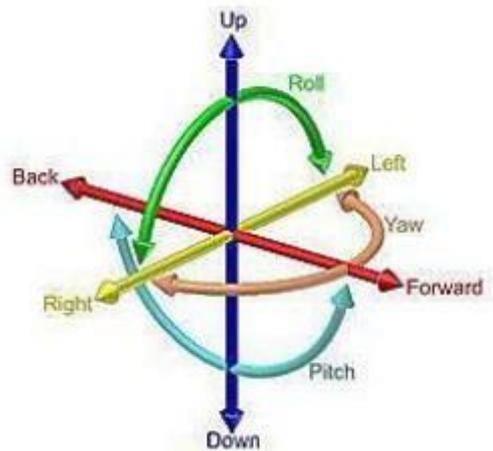
Gambar 2.5 menunjukkan tampilan fisik dan spesifikasi teknis sensor gyroscope MPU6050, yang merupakan komponen utama untuk mendeteksi gerakan dan orientasi dalam berbagai aplikasi mikrokontroler.

Tabel 2.3 Spesifikasi Sensor Gyroscope MPU6050

Tegangan Sensor	5Volt
Dimensi	23mm x 23mm
Output Tegangan	0,25 Volt-3,75 Volt

MPU6050 sendiri sudah memiliki *Digital Motion Processors (DMP)*, yang akan mengolah data mentah dari masing-masing sensor. DMP pada MPU6050 juga berfungsi meminimalisasi eror yang dihasilkan. *Chip IC Inverse* yang didalamnya terdapat sensor *Accelerometer* dan *Gyroscope* yang sudah terintegrasi. *Accelerometer* digunakan untuk mengukur percepatan, *Accelerometer* sering digunakan untuk menghitung sudut kemiringan, dan hanya dapat melakukan dengan nyata ketika tidak bergerak. Untuk mendapatkan sudut kemiringan, sering dikombinasikan data yang digunakan untuk menghitung sudut [25].

Pada Gambar 2.6 merupakan titik kemiringan atau putaran pada sensor MPU650.



Gambar 2.6 Titik Kemiringan atau Putaran Pada MPU6050

Pada Gambar 2.8 di atas merupakan letak putaran atau kemiringan dari sensor MPU6050. *Roll*, *pitch*, dan *yaw* merupakan titik acuan dari kemiringan pada sistem. Gerakan memutar ke samping merupakan istilah dari *Roll*. Kemudian gerakan memutar ke bawah dan atas merupakan *pitch*. Serta gerakan memutar ke samping adalah *yaw*[26].

Penelitian yang dilakukan oleh Steven Pandelaki, Lanny Sitanayah, dan Micael Liem yang berjudul “Sistem Pendeteksi Jatuh Berbasis *Internet Of Things*”, metode yang digunakan melibatkan penggunaan sensor MPU6050 yang terintegrasi dengan mikrokontroler NodeMCU ESP8266. Sensor MPU6050 ini terdiri dari *accelerometer* dan *gyroscope* yang mampu mengukur percepatan dan rotasi pada tiga sumbu. Sensor dan mikrokontroler ditempatkan pada pinggang pengguna untuk mendeteksi kejadian jatuh. Penelitian ini dimulai dengan identifikasi masalah, yaitu kebutuhan akan sistem yang dapat mendeteksi jatuh pada orang lanjut usia atau orang dengan penyakit tertentu. Setelah itu, dilakukan studi literatur untuk memahami teknologi yang ada dan memilih alat yang tepat. Tahapan berikutnya meliputi perancangan sistem, implementasi, dan pengujian. Sistem yang dirancang memanfaatkan keluaran data dari sensor MPU6050 untuk mendeteksi kejadian jatuh. Data yang dikumpulkan oleh sensor kemudian diproses oleh mikrokontroler NodeMCU ESP8266 yang mengirimkan ke database

untuk dianalisis lebih lanjut menggunakan algoritma C4.5. Algoritma ini digunakan untuk mengklasifikasikan data apakah termasuk kejadian jatuh atau tidak. Studi literatur yang dilakukan mengulas berbagai sistem pendeteksi jatuh yang telah ada, termasuk sistem yang menggunakan sensor *accelerometer* dan *gyroscope* serta berbagai metode klasifikasi data. Kajian ini menunjukkan bahwa kombinasi penggunaan sensor MPU6050 dan algoritma C4.5 mampu memberikan hasil yang akurat dalam mendeteksi kejadian jatuh.

Berdasarkan pengujian yang telah dilakukan, sistem yang dibangun memiliki tingkat akurasi sebesar 95% dalam mendeteksi dan memberikan peringatan kejadian jatuh. Sensor MPU6050 yang digunakan dapat mendeteksi perubahan percepatan dan rotasi tubuh dengan baik, memungkinkan sistem untuk mengidentifikasi kejadian jatuh secara efektif. Hasil penelitian menunjukkan bahwa sistem pendeteksi jatuh berbasis IoT yang menggunakan sensor ini dapat berfungsi dengan sangat baik dan memiliki potensi untuk digunakan dalam aplikasi nyata [27].

Kemudian penelitian yang dilakukan oleh Muhammad Marsa dan Moch Syaryadi yang berjudul “Penerapan *Wearable Device* untuk Mendeteksi Lansia Jatuh Pada Rumah Aceh” Penelitian ini mengembangkan sistem deteksi jatuh untuk lansia yang tinggal di rumah Aceh, menggunakan chip MPU-6050 yang berisi sensor akselerometer dan giroskop. Sistem terdiri dari dua perangkat: satu ditempatkan pada tubuh lansia untuk mendeteksi pergerakan, dan satu lagi di rumah untuk memantau posisi lansia. Data dari sensor diproses oleh mikrokontroler ATmega328, dan informasi dikirim melalui *Bluetooth* HC-05 ke aplikasi di *smartphone* yang dibuat dengan MIT App Inventor. Penelitian ini menyoroti pentingnya deteksi jatuh bagi lansia, terutama mereka yang tinggal sendirian di rumah yang berisiko seperti rumah Aceh yang memiliki banyak tangga. Kajian teori mencakup penjelasan tentang lansia, rumah Aceh, gravitasi, sensor MPU-6050, sensor ultrasonik, dan penggunaan MIT App Inventor dalam pengembangan aplikasi.

Pengujian menunjukkan bahwa sistem deteksi jatuh dapat mendeteksi pergerakan jatuh dengan akurasi menggunakan set point $\leq -1,5$ atau $\geq 1,5g$. Saat percepatan ini terdeteksi, sistem mengirim informasi melalui *Bluetooth* ke *smartphone* dan mengirim pesan ke keluarga lansia. Sensor ultrasonik berhasil mendeteksi perpindahan lansia antar ruangan di rumah Aceh. Penelitian ini menyoroti pentingnya sensor MPU6050 dalam deteksi jatuh karena kemampuannya mengukur percepatan dan kemiringan pada tiga sumbu (x, y, dan z). Sistem ini diharapkan memberikan respon cepat dalam memberikan pertolongan pertama kepada lansia yang jatuh, terutama di rumah berisiko tinggi seperti rumah Aceh. MPU-6050 dipilih karena andal dalam mendeteksi gerakan dan percepatan, serta mampu mengukur percepatan dan rotasi pada tiga sumbu. Sensor ini memberikan data akurat dan konsisten, yang penting dalam aplikasi kesehatan seperti deteksi jatuh pada lansia. Selain itu, MPU-6050 mudah diintegrasikan dengan mikrokontroler seperti ATmega328 dan memiliki konsumsi daya rendah, menjadikannya pilihan efisien dan efektif untuk perangkat wearable [28].

1.2.4 Sensor Max30100

Sensor MAX30100 adalah perangkat yang mengintegrasikan *pulse oximetry*, pemantauan sinyal detak jantung dan kandungan oksigen dalam darah. Perangkat ini memiliki 12 buah LED dan 1 buah *photodiode* dan bekerja dengan catu daya 1.8V dan 3.5V serta dapat dimatikan melalui perangkat lunak dengan arus *standby* yang dapat diabaikan sehingga memungkinkan catu daya untuk tetap terhubung setiap saat. Untuk pengukuran kadar oksigen di dalam darah, Oximeter bekerja dengan memanfaatkan denyut alami aliran darah di dalam arteri dan sifat hemoglobin yang mampu menyerap cahaya. Dimana cahaya infra merah akan lebih banyak diserap oleh hemoglobin yang tidak memiliki oksigen. Nilai yang telah dideteksi selanjutnya digunakan untuk menentukan jumlah oksigen di dalam darah. Beberapa fitur sensor MAX30100 pulse oximeter adalah kemampuannya mengkonsumsi daya yang sangat rendah (beroperasi dari 1.8V dan 3.5V), arus

shutdown ultranya rendah (0. 7 μ A detik), dan kemampuan output data yang cepat. Tampilan fisik Max30100 dapat dilihat pada gambar 2.7 [29].



Gambar 2 7 Sensor MAX30100

Pada Gambar 2.7 merupakan tampilan dari sensor MAX30100. Untuk tampilan spesifikasi pin konfigurasi MAX30100 akan di jelaskan dalam bentuk tabel 2.4 dibawah ini.

Tabel 2 4 Spesifikasi Pin Konfigurasi MAX30100[30]

Pin	Nama	Fungsi
1,7,8,14	N.C	Untuk stabilitas mekanik dengan menghubungkan ke PCB pad
2	SCL	12C <i>Clock Input</i>
3	SDA	12C Clock Data, <i>Biredirectional</i>
4	PGND	<i>Ground</i>
5	IR_DRV	IR LED katoda dan titik koneksi driver LED
6	R_DRV	Katoda LED merah dan titik koneksi Driver LED
9	R_LED+	Catu Daya untuk LED merah
10	IR_LRD+	Catu Daya untuk LED IR
11	V _{DD}	Input Catu Daya Analog
12	GND	Sebagai Analog <i>Ground</i>
13	INT	<i>Active-low Interrupt</i>

Tabel 2 5 Perbedaan Sensor MAX30100 dengan sensor lainnya

Fitur	Max30100	Max30102	Max30105
Jenis Sensor	Sensor detak jantung dan oksigen dalam darah	Sensor detak jantung dan oksigen dalam darah	Sensor detak jantung, oksigen dalam darah, dan partikel dalam udara.
Fungsi Utama	Mengukur detak jantung dan SpO2	Mengukur detak jantung dan SpO2	Mengukur detak jantung, SpO2, dan deteksi partikel(seperti asap)
LED <i>Emitters</i>	2(Red dan <i>Infrared</i>)	2 (<i>Red</i> dan <i>Infrared</i>)	3 (<i>Red Infrared</i> , dan <i>Green</i>)
<i>Photodectors</i>	1	1	1
Komunikasi	12C	12C	12C
Tegangan Operasi	1.8V hingga 3.3V	1.8V hingga 3.3V	1.8V hingg 3.3V
Fitur Tambahan	-	Menggunakan algoritma pengurang noise untuk hasil lebih akurat	Dapat mendeteksi dalam udara sehingga cocok untuk aplikasi seperti deteksi kualitas udara
Penggunaan Daya	Rendah	Rendah	Rendah
Aplikasi Utama	Pemantauan kesehatan,	Pemantauan kesehatan, <i>wearable</i>	Pemantauan kesehatan,

	<i>wearable devices</i>	<i>devices</i>	<i>wearable devices,</i> dan aplikasi pengukuran kualitas udara.
--	-------------------------	----------------	---

Penelitian oleh Sugiarto dan Rangga Heriansyah berjudul “Perancangan Alat Ukur Saturasi Oksigen dan Detak Jantung Berbasis Arduino dengan Menggunakan Sensor MAX30100 dan LCD” mengkaji penggunaan sensor MAX30100 untuk mengukur saturasi oksigen dalam darah (SpO2) dan detak jantung (BPM). Metode yang dilakukan penelitian ini adalah eksperimen (uji coba) dan *prototype*. Penelitian ini mengusulkan perancangan alat ukur saturasi oksigen dan detak jantung berbasis Arduino dengan metode PPG *reflectance*, berbeda dengan metode PPG *transmittance* yang umum digunakan pada alat komersial. Metode *reflectance* memungkinkan pengukuran pada seluruh bagian kulit tubuh yang terdapat pembuluh darah, tidak terbatas hanya pada jari tangan. Hasil pengujian menunjukkan bahwa alat yang dibuat memiliki akurasi tinggi dalam mengukur SpO2 dan BPM. Perbandingan hasil pengukuran dengan alat komersial menunjukkan akurasi sebesar 98,38% untuk SpO2 dan 98,29% untuk BPM. Alat ini berhasil memenuhi tujuan penelitian dengan menghasilkan pengukuran yang akurat dan dapat diandalkan. Penelitian ini menunjukkan bahwa penggunaan sensor MAX30100 dalam alat ukur berbasis Arduino efektif untuk mengukur saturasi oksigen dan detak jantung dengan tingkat akurasi yang tinggi. Alasan Sugiarto dkk memilih menggunakan sensor MAX30100 karena sensor ini memiliki kemampuan untuk mengukur SpO2 dan BPM dengan tingkat akurasi yang lebih tinggi, serta mudah diintegrasikan dengan mikrokontroler seperti Arduino. Penelitian ini menjadi dasar penting untuk pengembangan lebih lanjut, terutama dalam konteks penggunaan teknologi IoT untuk pemantauan kesehatan. Keunggulan sensor MAX30100 dan keberhasilan implementasi dalam alat ukur berbasis Arduino menunjukkan potensi besar untuk aplikasi yang lebih luas dalam bidang medis [31].

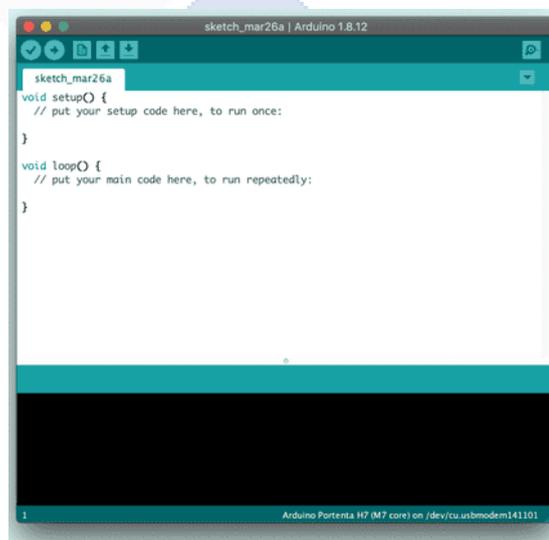
Kemudian yang dilakukan oleh Budi Harianto, Ahmat Hidayat, dan Fitria Nova Hulu yang berjudul “Analisis Penggunaan Sensor MAX30100 pada Sistem Pendeteksi Detak Jantung Berbasis IoT BLYNK”. menggunakan metode rancang bangun. Penelitian ini diawali dengan studi literatur, dilanjutkan dengan perancangan perangkat keras (*hardware*) dan perangkat lunak (*software*), pengujian alat, dan analisis akurasi sensor. Kajian pustaka dalam penelitian ini mencakup berbagai topik terkait detak jantung, saturasi oksigen, tekanan darah, teknik *photoplethysmography*, dan sensor MAX30100. Hasil penelitian menunjukkan bahwa alat pengukur detak jantung berbasis IoT Blynk berhasil dibuat dan diuji. Pengujian alat dilakukan dengan membandingkan hasil pembacaan sensor MAX30100 dengan oximeter pulse fingertip tipe JZK-302 sebagai alat referensi. Hasil pengujian menunjukkan bahwa alat ini memiliki tingkat keakuratan 96.2% untuk detak jantung dan 98.43% untuk saturasi oksigen dalam darah. Penelitian ini melibatkan analisis tingkat akurasi sistem yang dirancang dalam mendeteksi detak jantung dan saturasi oksigen dalam darah. Hasil pengujian menunjukkan bahwa alat dirancang mampu memberikan informasi kondisi kesehatan pengguna secara real-time dan dapat diakses jarak jauh melalui aplikasi Blynk di smartphone. Tingkat akurasi alat cukup tinggi menunjukkan bahwa sensor MAX30100 dapat diandalkan dalam mengukur detak jantung dan saturasi oksigen [32].

Sensor detak jantung dapat memberikan informasi tambahan mengenai kondisi fisik lansia saat terjadi jatuh. Sensor detak jantung merupakan komponen penting dalam sistem pendeteksi jatuh pada lansia. Sensor ini tidak hanya berfungsi untuk memonitor detak jantung, tetapi juga memberikan data tambahan yang berguna untuk menilai kondisi fisik lansia saat terjadi kejadian. Integrasi sensor detak jantung dengan mikrokontroler ESP32 dan sensor MPU6050 menciptakan sistem yang lebih komprehensif dalam memantau dan menjaga keselamatan lansia.

1.3 Software Arduino IDE (*Integrated Development Environment*)

Arduino merupakan program IDE (*Integrated Development Environment*) yang memungkinkan Anda mengembangkan aplikasi mikrokontroler dengan mudah, mulai dari penulisan program, kompilasi, hingga pengujian hasil kompilasi melalui terminal[33].

Arduino IDE akan digunakan untuk membuat dan menyimpan segala program yang dibutuhkan dalam proses pengerjaan alat pendeteksi jatuh. Segala pengaturan dan logika pemrosesan data dari sensor MPU6050, yang mendeteksi percepatan dan rotasi tubuh, akan ditulis dan disimpan dalam Arduino IDE. Selain itu, program di dalam Arduino IDE juga mengatur komunikasi dengan modul *Bluetooth* atau *WiFi* untuk mengirimkan data kejadian jatuh ke aplikasi smartphone yang telah dirancang menggunakan *Mitt App Inventor*. Program ini juga akan mengatur tampilan notifikasi pada layar LCD untuk memberikan informasi langsung mengenai kondisi lansia, sehingga memudahkan pemantauan secara *real-time*. Program Untuk tampilan *Software Arduino IDE* dapat dilihat pada gambar 2.8 dibawah ini:



Gambar 2 8 Software Arduino IDE

Gambar 2.8 Software Arduino IDE alat yang sangat efektif dan serbaguna untuk pengembangan proyek mikrokontroler, menawarkan antarmuka yang

intuitif, alat debugging yang kuat, dan dukungan komunitas yang luas, menjadikannya pilihan ideal bagi pemula dan profesional.

1.4 Mitt App Inventor

Aplikasi Inventor merupakan salah satu perangkat pengembang aplikasi android. Perangkat lunak ini menggunakan pendekatan blok untuk membentuk aplikasi sehingga sangat mudah digunakan. Aplikasi inventor ini tidak menggunakan bahasa pemrograman seperti java dan sejenisnya. aplikasi ini bisa digunakan tanpa koneksi internet, dengan kata lain aplikasi ini bisa dibuat dengan keadaan offline. Penggunaan aplikasi ini dalam sistem deteksi jatuh untuk menghubungkan sistem keseluruhan agar sistem dapat terhubung satu sama. Dalam penelitian ini app inventor digunakan sebagai perangkat untuk mengolah data yang dikirim dari alat deteksi jatuh untuk dikirimkan informasi ke kerabat dari lansia apabila lansia sudah berada dalam posisi terjatuh [34].

Penelitian yang dilakukan oleh Priyatna Budiman, Wahyu Andhyka Kusuma, dan Syaifuddin yang berjudul “Monitoring Jatuh Dengan Menggunakan Sensor *Accelerometer* dan *Gyroscope*” Metode penelitian yang digunakan meliputi studi pustaka, pengumpulan data, analisis dan desain sistem, perancangan perangkat lunak, serta pengujian. Studi pustaka dilakukan untuk mengumpulkan informasi terkait penggunaan sensor *accelerometer* dan *gyroscope* dalam mendeteksi jatuh, sedangkan pengumpulan data dilakukan dengan memanfaatkan sensor yang ada di *smartphone* untuk merekam data gerakan pada sumbu x, y, dan z.

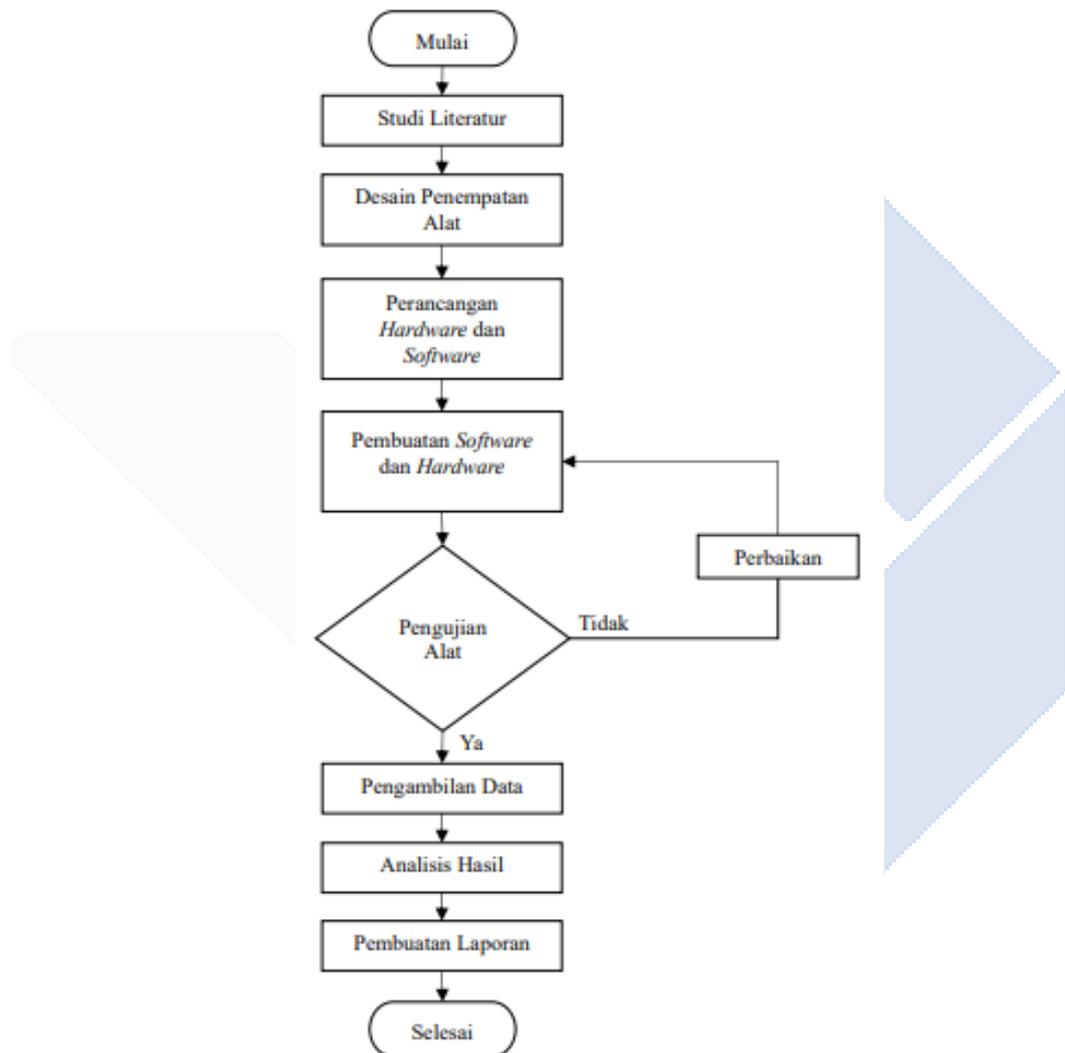
Analisis dan desain sistem melibatkan identifikasi kebutuhan perangkat keras dan perangkat lunak, serta merancang alur aplikasi dan fitur-fitur yang akan dikembangkan menggunakan MIT App Inventor. MIT App Inventor dipilih karena kemudahannya dalam membuat aplikasi Android melalui antarmuka *drag-and-drop*, yang sangat membantu dalam mempercepat proses pengembangan dan mengurangi kesalahan pemrograman. Aplikasi yang dibuat bertujuan untuk memonitor pergerakan sumbu x, y, dan z dengan menggunakan sensor

accelerometer dan *gyroscope* pada *smartphone*. Hasil penelitian menunjukkan bahwa sensor *accelerometer* memiliki tingkat akurasi 79,17% dalam mendeteksi jatuh, sedangkan akurasi sensor *gyroscope* bervariasi tergantung pada jenis gerakan yang diuji. Pembahasan penelitian ini menyoroti efektivitas sensor *accelerometer* dan *gyroscope* pada *smartphone* dalam memonitoring gerakan jatuh, dengan sensor *accelerometer* menunjukkan tingkat akurasi yang cukup tinggi. *True Positive* (TP) dan *False Negative* (FN) digunakan untuk mengukur kinerja sensor dalam mendeteksi jatuh, menunjukkan perbedaan akurasi dan sensitivitas untuk setiap jenis gerakan jatuh.

Kesimpulan dari penelitian ini menegaskan bahwa sensor *accelerometer* dan *gyroscope* pada *smartphone* efektif digunakan untuk monitoring gerakan jatuh. Aplikasi yang dikembangkan dengan MIT App Inventor menunjukkan kinerja yang cukup baik dalam mendeteksi jatuh dan memberikan notifikasi. Pengembangan lebih lanjut disarankan untuk meningkatkan kinerja sistem dan menambahkan fitur-fitur canggih lainnya guna meningkatkan akurasi dan efektivitas monitoring gerakan jatuh. Dengan demikian, penggunaan MIT App Inventor dalam penelitian ini memberikan kontribusi signifikan dalam memudahkan dan mempercepat proses pengembangan aplikasi deteksi jatuh berbasis IoT [35].

BAB III METODE PENELITIAN

Untuk mempermudah proses dalam pembuatan proyek akhir, dibuat beberapa tahapan dalam bentuk *flowchart* yang ditunjukkan dibawah ini



Gambar 3. 1 FlowChart Proses Pelaksanaan Proyek Akhir

Flowchart untuk pelaksanaan digunakan selain mempermudah membuat alat juga mempermudah dalam mengingat tahap-tahap mengerjakan alat secara *hardware* maupun *software*.

3.1 Studi Literatur

Penelitian dilakukan pada sistem pemantauan kebisingan telah banyak dilakukan oleh peneliti sebelumnya. Berikut tabel perbedaan dari penelitian sebelumnya dapat dilihat pada Tabel 3.1 di bawah ini.

Tabel 3 1 Penelitian Sebelumnya

Judul Penelitian	Tahun	Nama Penulis
Sistem Deteksi Jatuh untuk Manusia Lanjut Usia Berbasis Arduino	2018	Fajar Akbar Maulana
Sistem Pendeteksi Jatuh Wearable untuk Lanjut Usia Menggunakan <i>Accelerometer</i> dan <i>Gyroscope</i>	2019	Syifa Dzikri Tsani dan Indra Hardian Mulyadi
Perancangan Prototipe Pendeteksi Gerakan Jatuh Pada lansia Menggunakan Sensor <i>Accelererometer</i> Berbasis IoT	2020	Ryan Ady Putera Effendy

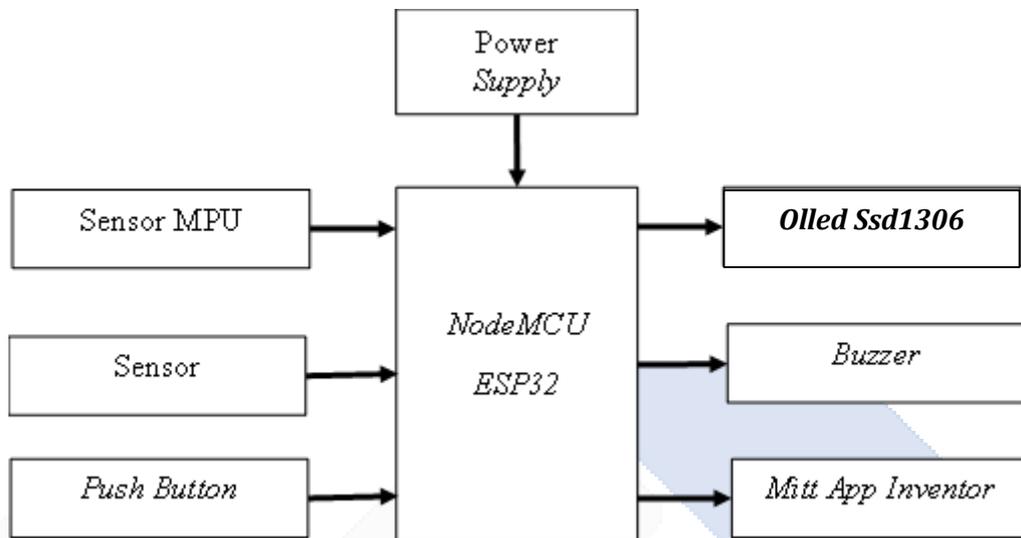
Tabel di atas menyajikan beberapa kajian bibliografi yang diambil dari sebelumnya yang merupakan sumber informasi dan referensi penelitian ini, selain itu juga terdapat kajian-kajian lain yang terkait dan sependapat dengan dan komponen yang digunakan.

3.2 Perancangan *Hardware* dan *Software*

3.2.1 Sistem Kerja Alat

Sistem Kerja alat pendeteksi jatuh pada lansia dalam keadaan rawat jalan berbasis *internet of things* dapat dijelaskan melalui blok diagram. Blok diagram ini menunjukkan hubungan antara berbagai komponen utama, termasuk sensor, mikrokontroler, dan perangkat Output, serta alur data dan proses yang terjadi dalam sistem.

Berikut merupakan blok diagram sistem kerja alat pada Gambar 3.2 dibawah ini.



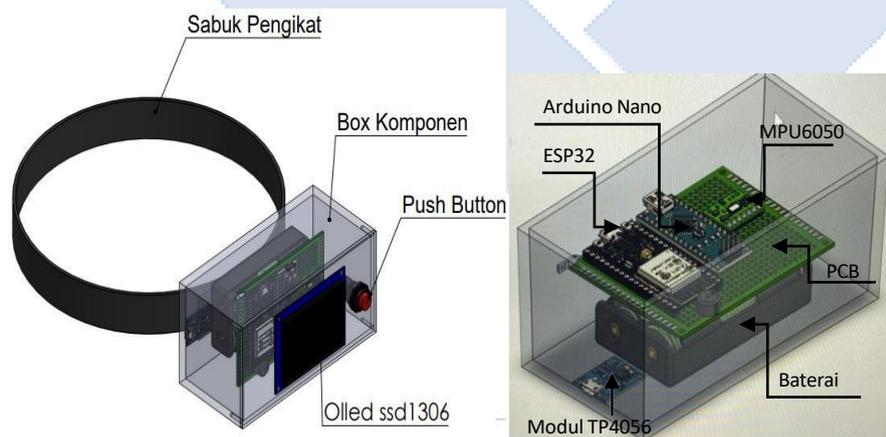
Gambar 3 2 Blok Diagram Sistem Kerja Alat

Sistem kerja alat pendeteksi jatuh pada Gambar 3.2 melibatkan beberapa komponen utama yang terhubung ke mikrokontroler ESP32. Sensor MPU6050 dan MAX30100 serta *pushbutton* berfungsi sebagai masukan yang akan mendeteksi gerakan, rotasi, detak jantung, dan saturasi oksigen dalam darah pada pengguna. Semua data yang dikumpulkan oleh sensor ini akan diproses oleh ESP32, yang juga terhubung ke sumber daya baterai. Sebagai keluaran, ESP32 akan mengirimkan informasi ke layar Oled *ssd1306* untuk menampilkan status secara langsung, mengirimkan notifikasi ke aplikasi *Mitt App Inventor* di *smartphone* melalui koneksi nirkabel, dan mengaktifkan buzzer sebagai alarm suara.

3.2.2 Perancangan *Hardware*

Perancangan *Hardware* di bawah ini merupakan perancangan desain 3D dari alat/box yang akan dibuat. Pada tahapan ini menggunakan aplikasi *FreeCAD*, proses mendesain dilakukan dengan menyusun dan menata komponen-komponen yang digunakan untuk membuat alat pendeteksi jatuh dalam *projectbox*. Komponen yang terdapat pada *project box* adalah ESP32, sensor MPU6050,

sensor MAX30100, baterai, Oled ssd1306, buzzer, push button dan kabel jumper penghubung antar komponen. ESP32 berfungsi sebagai mikrokontroler utama yang mengendalikan seluruh sistem, menghubungkan berbagai sensor dan komponen lainnya melalui Wi-Fi untuk pengolahan data secara real-time. Sensor MPU6050 digunakan untuk mendeteksi gerakan dan orientasi dengan mengukur percepatan dan rotasi, menjadikannya penting untuk mendeteksi jatuh atau perubahan posisi. Sensor MAX30100 berfungsi untuk memantau detak jantung dan kadar oksigen dalam darah, memberikan informasi vital mengenai kondisi kesehatan pengguna. Baterai menyediakan sumber daya portabel bagi sistem, memungkinkan operasi tanpa perlu sambungan listrik langsung. OLED SSD1306 adalah layar kecil yang digunakan untuk menampilkan informasi secara langsung kepada pengguna, seperti data sensor atau status sistem. Buzzer digunakan sebagai alarm untuk memberikan peringatan audio, misalnya ketika terdeteksi jatuh atau anomali kesehatan. Push button berfungsi sebagai input manual dari pengguna, memungkinkan mereka untuk mengaktifkan atau mematikan fungsi tertentu dari sistem. Kabel jumper menghubungkan seluruh komponen secara fisik, memastikan aliran data dan daya antar komponen sesuai kebutuhan. Berikut pada gambar 3.3 perancangan desain box pada alat pendeteksi jatuh.



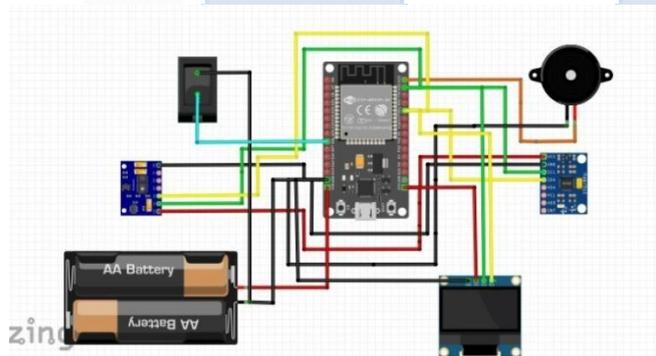
Gambar 3 3 Desain Alat

Perancangan *hardware* merupakan tahapan menentukan sistem kontrol yang akan digunakan pada alat. Adapun komponen-komponen yang akan digunakan seperti pada tabel 3.2 komponen yang Digunakan dibawah ini:

Tabel 3 2 Komponen yang Digunakan

Jenis Komponen yang Digunakan	Jumlah
Sensor MPU6050	1
Sensor MAX30100	1
Oled ssd1306	1
NodeMCU ESP32	1
Buzzer	1
Pushbutton	1
Modul TP40556	1
Baterai Lithium18650	1
Arduino Nano	1

Berikut merupakan wiring diagram pada gambar 3.4 dibawah ini



Gambar 3 4 Wiring Diagram

Wiring diagram dalam proyek ini menggambarkan hubungan antara beberapa komponen utama yang digunakan untuk sistem pendeteksi jatuh bagi lansia berbasis *internet of things(IoT)*. Komenen yang digunakan meliputi sensor

MPU6050, Mikrokontroler ESP32, Sensor denyut jantung MAX30100, Olled ssd1306 display, Buzzer, Pushbutton, dan Modul baterai TP4059.

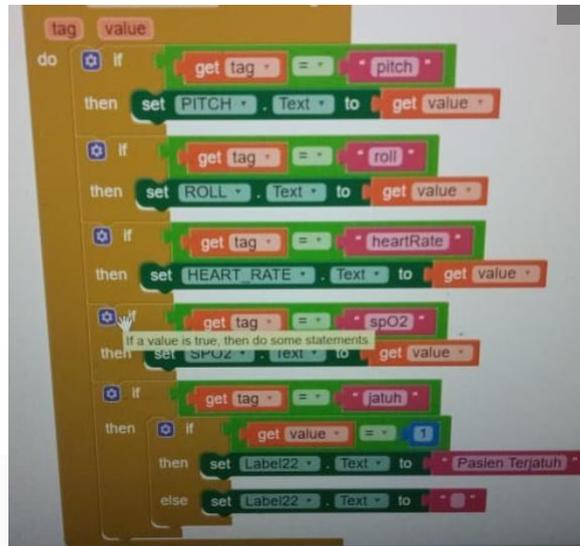
1. Sensor MPU6050: Sensor *accelerometer* dan *gyroscope*.
 - Koneksi:
 - SDA (Serial Data) dari MPU6050 terhubung ke pin SDA pada ESP32.
 - SCL (*Serial Clock*) dari MPU6050 terhubung ke pin SCL pada ESP32.
 - VCC dari MPU6050 terhubung ke pin 3.3V pada ESP32.
2. Sensor Denyut Jantung Max30100: Sensor ini digunakan untuk memonitor denyut jantung.
 - Koneksi:
 - SDA dari Max30100 terhadap ke pin SDA pada ESP32.
 - SCL dari Max30100 terhubung ke pin SCL pada ESP32.
 - VCC dari Max30100 terhubung ke pin 3.3V pada ESP32.
 - GND dari Max30100 terhubung ke GND pada ESP32.
3. Mikrokontroler ESP32: Mikrokontroler ini mengelola data dari sensor dan mengendalikan komponen lainnya.
 - Koneksi
 - Pin SDA dan SCL untuk komunikasi 12C dengan sensor MPU6050.
 - Pin digital untuk kontrol LCD display, buzzer, dan pushbutton.
 - Pin VCC dan GND untuk daya dari modul baterai TP4059.
4. LCD Display: Digunakan untuk menampilkan informasi status dan data yang diproses.
 - Koneksi:
 - Pin data (D4, D5, D6, D7) dari LCD terhubung ke pin digital pada ESP32.

- Pin kontrol (RS, E) terhubung ke pin digital pada ESP32.
 - VCC terhubung ke pin 5V pada ESP32.
 - GND terhubung ke GND pada ESP32.
5. Buzzer : Digunakan sebagai alarm untuk memberi peringatan jika terdeteksi jatuh.
- Koneksi:
 - Pin positif dari buzzer terhubung ke pin digital pada ESP32.
 - Pin negatif dari buzzer terhubung ke GND pada ESP32
6. Pushbutton : Digunakan untuk memberikan input manual ke sistem.
- Koneksi:
 - Satu sisi pushbutton terhubung ke pin digital pada ESP32.
 - Sisi lain terhubung ke GND.
7. Modul Baterai TP4059 : Digunakan untuk mengelola daya dan pengisian baterai.
- Koneksi:
 - Pin VCC dari modul terhubung ke pin 3.7V baterai.
 - Pin GND dari modul terhubung ke GND baterai.
 - Pin OUT dari modul terhubung ke pin 5V pada ESP32 untuk suplay daya.
 - Pin GND dari modul terhubung ke GND ESP32.

Dengan wiring diagram ini, setiap komponen menerima daya yang cukup dan dapat berkomunikasi satu sama lain dengan efektif. Penempatan dan koneksi yang benar antara komponen ini sangat penting untuk memastikan sistem dapat berfungsi dengan optimal dalam mendeteksi dan memberikan peringatan terkait jatuhnya seorang lansia.

3.2.3 Perancangan *software*

Perancangan *Software* dibawah ini merupakan Gambar 3.5 dari Block Diagram Aplikasi Mitt App Inventor.



Gambar 3 5 Block Diagram Pada Aplikasi Mitt App Inventor

Berikut adalah penjelasan singkat terkait diagram blok pada aplikasi Mitt App Inventor yang ditampilkan di gambar:

1. Inisialisasi dan pengambilan Data Sensor:

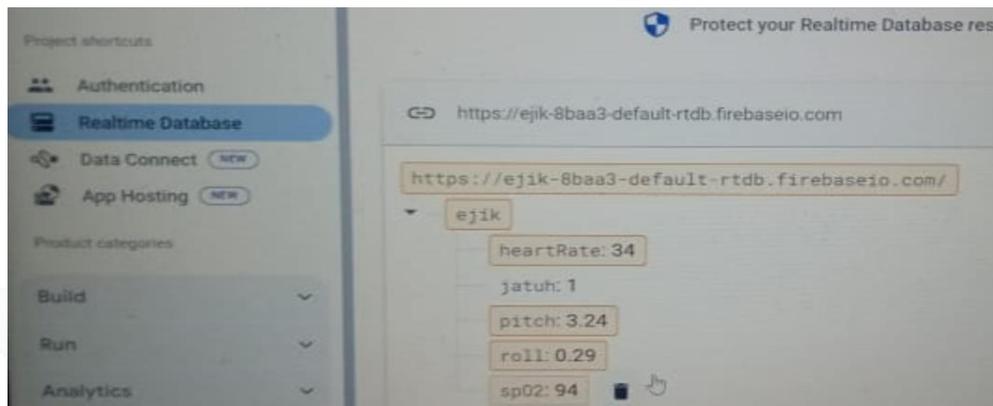
- Blok pertama (' get tag "pitch" ') mengambil nilai pitch dari sensor dan menampilkannya di label teks PITCH.
- Blok kedua (' get tag "roll" ') mengambil nilai roll dari sensor dan menampilkannya di label teks ROLL
- Blok ketiga(' get tag "heartRate" ') mengambil nilai detak jantung dari sensor dan menampilkannya di label teks HEART_RATE.
- Blok keempat (' get tag " SpO2" ') mengambil nilai SpO2 (kadar oksigen dalam darah) dari sensor dan menampilkannya di label teks SpO2.

2. Deteksi Jatuh:

- Blok terakhir (' get tag " jatuh" ') memeriksa apakah kondisi jatuh terdeteksi.

- Jika kondisi jatuh terdeteksi ('value = 1'), maka label teks label22 akan diubah menjadi "Pasien Terjatuh".
- Jika kondisi jatuh tidak terdeteksi ('value = 0'), maka label teks label22 akan diubah menjadi "0".

Berikut adalah Database yang digunakan untuk menyimpan dan mengelola data sensor dari perangkat IoT.



Gambar 3 6 Firebase

Penjelasan terkait *firebase* pada gambar 3.6 sebagai berikut:

- URL yang terlihat di bagian atas (' <https://ejik-8baa3-default-rtdb.firebaseio.com/> ') adalah endpoint dari firebase realtime database tempat data disimpan.
- 'ejik' adalah nama utama dari node atau cabang utama dalam database ini.
- heartRate : Menyimpan nilai detak jantung, dalam contoh ini bernilai '34'.
- jatuh : Menyimpan status jatuh, di sini bernilai '1' yang menunjukkan bahwa kejadian jatuh terdeteksi.
- pitch : Menyimpan nilai pitch dari sensor akselerometer, bernilai '3.24'.
- roll : Menyimpan nilai roll dari sensor akselerometer, bernilai '0.29'.

- SpO2 : Menyimpan nilai SpO2 (kadar oksigen dalam darah), bernilai '94'.

Firebase digunakan untuk menyimpan data secara *real-time* yang memungkinkan aplikasi ini untuk mendapatkan data terbaru dari sensor dengan cepat dan efisien. *Firebase* bertindak sebagai penyimpanan data pusat yang dapat diakses oleh aplikasi *smartphone* untuk menampilkan informasi terkini dan memberikan notifikasi jika terjadi kejadian jatuh.

3.3 Pembuatan Alat

Tahapan pembuatan alat ini dilakukan setelah melakukan perancangan sistem yang dilakukan kemudian akan diimplementasikan pada alat. Pada tahapan pertama pembuatan alat ini, membuat box menggunakan 3D Printing dengan ukuran 45 x 90 x 50 mm.



Gambar 3 7 Pembuatan Box Menggunakan 3D Printing

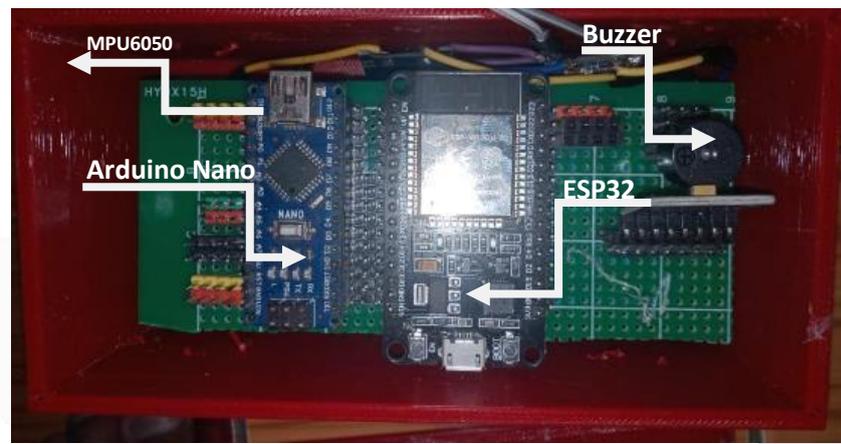
Pada Gambar 3.7 menunjukkan proses pembuatan box perangkat menggunakan teknologi 3D printing. Box ini dicetak dengan bahan plastik PLA (*Polylactic Acid*) yang memiliki kekuatan dan ketahanan yang baik, namun tetap ringan. Proses percetakan dimulai dengan desain model 3D yang dibuat menggunakan perangkat lunak CAD (*Computer-Aided Design*). Setelah desain

selesai, model tersebut diunggah ke mesin 3D printer yang kemudian mencetak box secara lapis demi lapis sesuai dengan spesifikasi desain.

Kemudian penyusunan fisik komponen dilakukan dengan tujuan untuk memastikan bahwa alat pendeteksi jatuh dapat berfungsi dengan optimal dan nyaman digunakan oleh pengguna. Berikut adalah uraian detail mengenai penyusunan fisik komponen:

1. Mikrokontroler ESP32 : Mikrokontroler ini tetap menjadi pusat pengolahan data. Dengan tambahan LCD dan buzzer, program di ESP32 akan diperbarui untuk menampilkan informasi di LCD dan mengontrol buzzer
2. Sensor MPU6050 : Sensor ini ditempatkan pada posisi yang optimal untuk mendeteksi perubahan akselerasi dan orientasi. Data dari sensor ini akan digunakan untuk mendeteksi jatuh.
3. Sensor MAX30100 : Sensor ini ditempatkan di lokasi yang mudah diakses untuk memantau detak jantung dan kadar oksigen dalam darah. Data dari sensor ini akan ditampilkan di LCD untuk memberikan informasi tambahan mengenai kondisi fisik pengguna.
4. Olled Ssd1306 : Olled Ssd1306 ditempatkan di bagian depan alat untuk memberikan visualisasi data secara *real-time*. Olled Ssd1306 ini akan menampilkan informasi seperti detak jantung, kadar oksigen, status sensor jatuh.
5. Buzzer : Buzzer ditempatkan di dalam box tetapi dengan lubang kecil atau grill untuk memastikan suara dapat keluar dengan jelas. Buzzer ini akan berbunyi sebagai alarm saat terjadi jatuh atau saat deteksi kondisi kritis dari sensor MAX30100.
6. Baterai Lithium 18650 : Baterai ini tetap menyediakan daya untuk seluruh sistem. Penempatan baterai dan modul TP4056 untuk pengisian daya tetap berada di dalam box, di lokasi yang mudah diakses untuk penggantian atau pengisian ulang.

7. Modul TP4056 : Modul ini tetap digunakan untuk mengisi daya baterai dan ditempatkan agar mudah diakses.
8. Box dan Sabuk : Box dirancang untuk melindungi seluruh komponen, termasuk LCD dan buzzer, serta menjaga alat tetap nyaman digunakan. Sabuk tetap digunakan untuk memasang alat pada tubuh pengguna, memastikan alat tetap pada posisi yang optimal. Berikut dibawah ini Gambar 3.8 penyusunan komponen-komponen dalam box



Gambar 3 8 Penyusunan Komponen-komponen dalam Box

Dengan tambahan LCD dan buzzer, alat ini tidak hanya memberikan notifikasi visual dan suara saat terjadi jatuh, tetapi juga menampilkan data vital secara *real-time* yang membantu dalam pemantauan kondisi pengguna. Buzzer memberikan peringatan audio yang efektif, sementara LCD memastikan informasi mudah dibaca dan dipahami oleh pengguna dan orang-orang di sekitarnya. Integrasi komponen tambahan ini meningkatkan keamanan dan kenyamanan penggunaan alat pendeteksi jatuh.

3.4 Pengujian Alat

Pada tahapan ini pengujian alat dilakukan untuk mengetahui apakah sistem alat pendeteksi yang dibuat sudah bekerja sesuai dengan diinginkan.

1. Menguji ESP32 dengan sensor MPU6050 melibatkan koneksi respons sensor terhadap gerakan, memastikan data percepatan dan rotasi yang dihasilkan dapat diterima dan diolah oleh ESP32 secara akurat. Pengujian

dilakukan dengan jatuh ke depan dan belakang dengan kemiringan 45°, jatuh kedepan dan belakang dengan kemiringan 90°, jatuh ke kiri serta jatuh ke kanan.

2. Menguji ESP32 dengan sensor MAX30100 melibatkan penghubungan sensor detak jantung ke ESP32 untuk memastikan bahwa data detak jantung (BPM) dan data saturasi oksigen dalam darah (SpO2) dapat dikirim dan diproses secara akurat oleh mikrokontroler.
3. Menguji tampilan Olled ssd 1306 yang akan memonitor berupa pitch, roll, heart rate dan SpO2.
4. Menguji buzzer dan pushbutton

3.5 Perbaikan

Proses ini dilakukan apabila pada saat proses uji coba proyek akhir mengalami kegagalan atau belum memenuhi keinginan sehingga harus melakukan perbaikan baik secara hardware maupun *software*, sesuai dengan hasil dan analisa penyebab kegagalan. Setelah proses perbaikan telah selesai dan tidak mengalami kegagalan atau telah memenuhi keinginan maka dilakukan proses uji coba ulang.

3.6 Pengambilan Data

Data yang didapatkan dari hasil pengujian berupa sensor MPU6050 dan sensor MAX30100 yang mana pada hasil pengujian sensor, sensor tersebut dapat mendeteksi kejadian jatuh dan dapat menampilkan detak jantung pada lansia. Data yang diambil berupa jatuh ke depan, jatuh ke belakang, jatuh ke samping kanan, dan jatuh ke samping kiri, posisi normal, posisi jongkok, BPM serta SpO2. Pengambilan data ini penting untuk memastikan bahwa sistem deteksi jatuh dapat mengenali berbagai jenis gerakan jatuh yang mungkin terjadi dalam situasi nyata dan memberikan respon yang tepat sesuai dengan kondisi kesehatan pengguna.

3.7 Analisis Hasil

Tahap selanjutnya adalah menganalisis hasil dari pengujian yang telah dilakukan. Analisis hasil ini berupa kemiringan yang ditangkap oleh sensor, serta data detak jantung yang diukur oleh sensor MAX30100. Data ini dianalisis untuk

menentukan akurasi sistem dalam mendeteksi berbagai jenis jatuh dan memantau kondisi kesehatan pengguna.

3.8 Pembuatan Laporan

Proses pembuatan laporan merupakan tahap akhir dalam proyek akhir yang bertujuan untuk menyimpulkan hasil keseluruhan secara komprehensif mengenai alat yang telah dikembangkan. Laporan ini disusun untuk memberikan gambaran menyeluruh tentang setiap aspek proyek, dimulai dari latar belakang yang menjelaskan pentingnya alat pendeteksi jatuh pada lansia dalam konteks perawatan kesehatan. Selain itu, laporan ini mencakup rumusan masalah yang diidentifikasi, serta batasan masalah yang ditetapkan untuk menjaga fokus dan ruang lingkup penelitian.

Selanjutnya, laporan menyajikan landasan teori yang relevan, memberikan dasar pengetahuan yang mendukung pengembangan alat ini, serta menguraikan metode pelaksanaan yang telah diikuti, termasuk detail tentang pembuatan *hardware*, *software*, dan prosedur uji coba. Pembahasan dalam laporan ini berfokus pada analisis hasil uji coba yang dilakukan, menyoroti kinerja alat dalam mendeteksi jatuh, mengukur parameter vital, dan memberikan notifikasi darurat.

Di akhir laporan, kesimpulan diambil berdasarkan temuan selama proses pengembangan dan pengujian, sementara saran diberikan untuk pengembangan lebih lanjut, termasuk potensi perbaikan atau penambahan fitur di masa depan. Laporan ini juga dilengkapi dengan lampiran yang relevan, seperti diagram, *flowchart*, dan data pengujian, yang mendukung keseluruhan analisis dan penjelasan yang disampaikan

BAB IV

HASIL DAN PEMBAHASAN

Pada bab ini akan membahas mengenai data yang didapatkan dari hasil pengujian yang telah dilakukan pada pembuatan proyek akhir.

4.1 Bentuk Alat

Adapun bentuk fisik alat yang telah dibuat merupakan hasil dari perakitan berbagai komponen elektronik yang disusun secara sistematis. Bentuk alat ini dirancang agar portabel dan mudah digunakan oleh lansia dalam keadaan rawat jalan dan memastikan alat ini tidak hanya berfungsi dengan baik, tetapi juga nyaman dan praktis saat digunakan. Berikut dibawah ini Gambar 4.1 Bentuk Fisik Alat.



Gambar 4 1 Bentuk Fisik Alat

Pada Gambar 4.1 Menunjukkan bentuk fisik alat yang dikemas dalam sebuah box. Box ini dirancang untuk melindungi komponen elektronik yang ada di dalamnya, seperti sensor MPU6050, sensor MAX30100, NodeMCU ESP32, Arduino Nano, dan baterai. Pada bagian depan box, terdapat layar LCD yang menampilkan informasi status alat dan terdapat tombol pushbutton untuk menyalakan buzzer.

Berikut dibawah ini penggunaan pada alat terletak pada pinggang pengguna.
Gambar 4.2 penggunaan alat pada pengguna.



Gambar 4 2 Penggunaan Alat Pada Pengguna

Gambar 4.2 menunjukkan cara penggunaan alat deteksi jatuh yang terpasang pada pinggang pengguna. Dalam gambar ini, alat diposisikan dengan tepat di area pinggang untuk memastikan sensor accelerometer dapat mendeteksi gerakan dengan akurat. Pemasangan yang benar pada posisi ini memungkinkan alat untuk mengidentifikasi perubahan posisi tubuh yang mendalam, yang penting dalam mendeteksi potensi jatuh. Posisi alat yang stabil dan nyaman pada pinggang juga memastikan bahwa sensor tetap berada di tempatnya, sehingga memberikan hasil deteksi yang konsisten dan dapat diandalkan. tahapan selanjutnya berupa pengujian sensor.

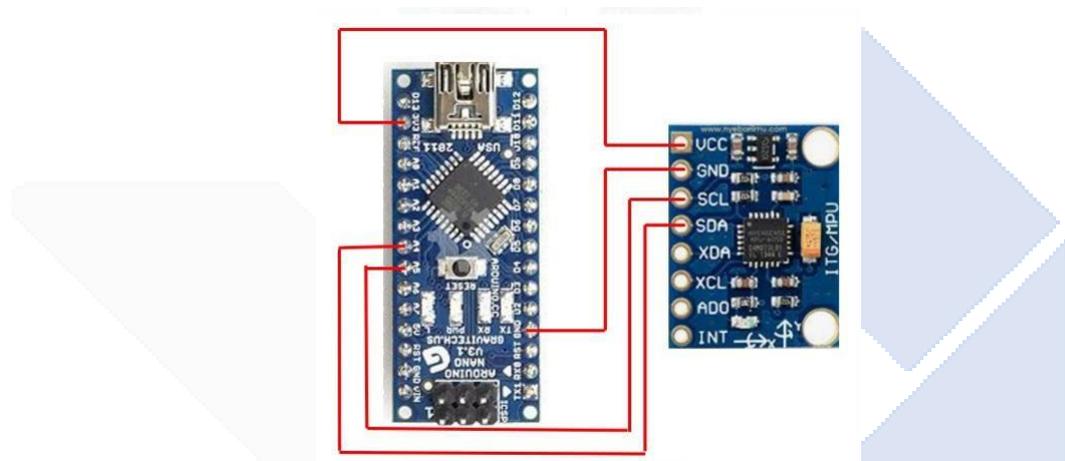
4.2 Pegujian Sensor

Pada tahap ini Pengujian dilakukan untuk memastikan bahwa sensor MPU6050 dan sensor MAX30100 tersebut dapat berfungsi dengan baik. Langkah-langkah pengujian pada sensor MPU6050 meliputi pengujian deteksi jatuh ke depan dan belakang dengan kemiringan 45° dan 90° , deteksi jatuh dengan posisi

miring ke kiri dan kanan. Langkah-langkah pengujian pada sensor MAX30100 untuk mengukur nilai laju detak jantung per menit (BPM) dan nilai saturasi oksigen dalam darah SpO2

4.2.1 Pengujian Sensor MPU6050

Pada pengujian sensor MPU6050 dilakukan pengujian dengan 5 kali pengujian sensor meliputi pengujian jatuh ke depan dan belakang dengan kemiringan 45° dan 90°, deteksi jatuh dengan posisi miring ke kiri dan kanan. Berikut dibawah ini tabel hasil pengujian pada tiap-tiap percobaan.



Gambar 4 3 Wiring Diagram Sensor MPU6050 dengan Arduino Nano

Gambar wiring diagram di atas menunjukkan koneksi antara modul sensor IMU (Inertial Measurement Unit) ITG/MPU dan papan mikrokontroler Arduino Nano. Berikut adalah penjelasan singkat terkait koneksi tersebut.:

1. VCC ke 5V:
 - Konektor VCC pada modul IMU dihubungkan ke pin 5V pada Arduino Nano. Ini menyediakan daya 5V untuk modul sensor dari papan Arduino.
2. GND ke GND:
 - Konektor GND pada modul IMU dihubungkan ke pin GND pada Arduino Nano. Ini memberikan jalur bersama antara kedua perangkat.

3. SCL ke A5:

- Konektor SCL (Serial Clock Line) pada modul IMU dihubungkan ke pin A5 pada Arduino Nano, Pin A5 pada Arduino Nano berfungsi sebagai pin SCL dalam komunikasi I2C.

4. SDA ke A4:

- Konektor SDA (Serial Data Line) pada modul IMU dihubungkan ke pin A4 pada Arduino Nano, Pin A4 pada Arduino Nano berfungsi sebagai pin SDA dalam komunikasi I2C.

5. INT ke D2:

- Konektor INT (Interrupt) pada modul dihubungkan ke pin D2 pada Arduino Nano, Pin ini digunakan untuk menerima sinyal interrupt dari modul IMU yang dapat memberitahu Arduino saat terjadi perubahan data penting.

Dengan konfigurasi ini, Arduino Nano dapat berkomunikasi dengan modul IMU melalui protokol I2C untuk membaca data sensor seperti akselerasi dan kecepatan sudut. Data ini dapat digunakan untuk berbagai aplikasi seperti stabilisasi, navigasi, atau deteksi gerakan.

Berikut dibawah ini tampilan program pada Arduino Nano dengan MPU6050 Pada Gambar 4.4.

```
DOIT ESP32 DEVKIT V1
jul29b.ino
#include <Wire.h>
#include <MPU6050.h>

MPU6050 mpu;

void setup() {
  Serial.begin(9600);
  Wire.begin();

  // Inisialisasi MPU6050
  mpu.initialize();
  if (mpu.testConnection()) {
    Serial.println("MPU6050 connection successful");
  } else {
    Serial.println("MPU6050 connection failed");
    while (1);
  }
}

void loop() {
  // Membaca data akselerometer dan gyroscope
  int16_t ax, ay, az;
  int16_t gx, gy, gz;

  mpu.getMotion6(&ax, &ay, &az, &gx, &gy, &gz);

  // Konversi ke G-force (9.81 m/s^2)
  float ax_g = ax / 16384.0;
  float ay_g = ay / 16384.0;
}

}
}

void loop() {
  // Membaca data akselerometer dan gyroscope
  int16_t ax, ay, az;
  int16_t gx, gy, gz;

  mpu.getMotion6(&ax, &ay, &az, &gx, &gy, &gz);

  // Konversi ke G-force (9.81 m/s^2)
  float ax_g = ax / 16384.0;
  float ay_g = ay / 16384.0;
  float az_g = az / 16384.0;

  // Menghitung kemiringan dalam derajat
  float roll = atan2(ay_g, az_g) * 180.0 / PI;
  float pitch = atan2(-ax_g, sqrt(ay_g * ay_g + az_g * az_g)) * 180.0 / PI;

  // Tampilkan hasil di Serial Monitor
  Serial.print("Roll: ");
  Serial.print(roll);
  Serial.print(" degrees, Pitch: ");
  Serial.print(pitch);
  Serial.println(" degrees");

  delay(500);
}
//Untuk program arduino nano dan sensor mpu6050
```

Gambar 4 4 Program MPU6050 dengan Arduino Nano

Berikut adalah penjelasan singkat dari kode yang ditampilkan dalam gambar:

1. Inisialisasi dan Setup:

- `#include <Wire.h>` dan `#include <MPU6050.h>` : Menyertakan pustaka yang diperlukan untuk komunikasi I2C dan penggunaan sensor MPU6050.
- `MPU6050 mpu;` : membuat objek untuk sensor MPU6050.
- `Void setup ()` : fungsi setup yang dijalankan sekali saat program dimulai.
- `Serial.begin (9600)` ; : memulai komunikasi serial dengan baud rate 9600.
- `Wire.begin ()` ; : memulai komunikasi I2C.
- `mpu.initialize ()` ; : menginisialisasi sensor MPU6050.
- `if (mpu.testConnection ())` : memeriksa apakah koneksi ke sensor MPU6050 berhasil.
- `Serial.println(“ MPU6050 connection failed”)` ; : menampilkan pesan sukses jika koneksi berhasil
- `serial.println (“MPU6050 connection failed”)` ; : menampilkan program dengan `while (1)`;

2. LOOP UTAMA:

- `void loop ()` ; fungsi loop yang dijalankan berulang kali selama program berjalan.
- `mpu.getMotion6 (&ax, &ay, &az, &gx, &gy, &gz)` ; : membaca data akselerometer dan gyroscope dari sensor MPU6050.
- Konversi data akselerometer ke nilai G-force:
 - `float ax_g = ax / 16384.0;`
 - `float ay_g = ay / 16384.0;`
 - `float az_g = az / 16384.0 ;`
- menghitung kemiringan dalam derajat:

- `float roll = atan2 (ay_g, az_g) * 180.0 / pi;`
- `float pitch = atan2(-ax_g, sqrt (ay_g * ay_g + az_g)) * 180.0 / PI;`
- Menampilkan hasil perhitungan kemiringan pada serial monitor:
 - `Serial.print("Roll: ");`
 - `Serial.print(roll);`
 - `Serial.print (pitch);`
 - `Serial.println ("degrees");`
 - `delay (500);` : menunda eksekusi selama 50 milidetik sebelum memulai loop lagi.

Penjelasan ini mencakup seluruh langkah penting yang ditampilkan dalam kode pada gambar tersebut.

Pengujian dilakukan dengan posisi pada pengguna. Berikut dibawah ini Gambar 4.5 Posisi Jatuh.



Gambar 4 5 Posisi Pengujian Alat Pada Pengguna

Dalam proses pengujian alat, perangkat *fall detection* ditempatkan pada bagian pinggang pengguna dengan menggunakan sabuk elastis. Posisi ini dipilih untuk memastikan bahwa sensor MPU6050 dapat mendeteksi perubahan akselerasi dan orientasi tubuh secara akurat saat terjadi pergerakan yang mendadak atau jatuh. Pengujian dilakukan dengan berbagai skenario, seperti simulasi, untuk memastikan bahwa perangkat dapat mendeteksi jatuh dengan tepat dari berbagai posisi.

Pengujian pertama menggunakan sensor MPU6050. Pengujian ini bertujuan untuk mengukur dan mencatat respon sensor terhadap gerakan jatuh terdapat 7 kali percobaan. Berikut dibawah ini penjelasan pada Tabel 4.1.

Tabel 4 1 Pengujian Deteksi Jatuh Pertama

No	Posisi Pengujian					
	jatuh ke depan (45°)	jatuh ke depan (90°)	jatuh ke belakang (45°)	jatuh ke belakang (90°)	jatuh kekiri	jatuh Kekanan
1	45.55	88.23	47.36	-80.27	82.87	-87.02
2	45.85	88.45	46.72	-80.39	82.28	-87.09
3	45.65	87.47	47.13	-80.24	83.14	-86.00
4	46.03	88.09	46.82	-79.89	83.38	-86.06
5	45.68	87.49	46.66	-80.01	82.93	-85.01
6	45.87	87.73	46.68	-80.43	82.88	-84.45
7	46.62	87.41	46.68	-80.47	82.81	-86.84
Rata- rata	45.89	87.84	46.86	-80.24	82.90	-86.07

Pada tabel 4.1 Ketika posisi jatuh berada pada sudut 45°, nilai yang dihasilkan lebih rendah karena sudut ini mencerminkan posisi tubuh yang berada di antara posisi tegak (90°) dan posisi horizontal (0°). Oleh karena itu, nilai percepatan atau orientasi yang terdeteksi oleh sensor menunjukkan angka yang lebih kecil karena perubahan posisi tubuh yang tidak sepenuhnya vertikal.

Ketika posisi jatuh berada pada sudut 90° , nilai yang dihasilkan lebih besar karena sudut ini mencerminkan posisi tubuh yang sepenuhnya vertikal. Sensor mendeteksi perubahan posisi tubuh dari keadaan tegak lurus yang menghasilkan nilai percepatan atau orientasi yang lebih tinggi karena perubahan drastis dalam posisi tubuh.

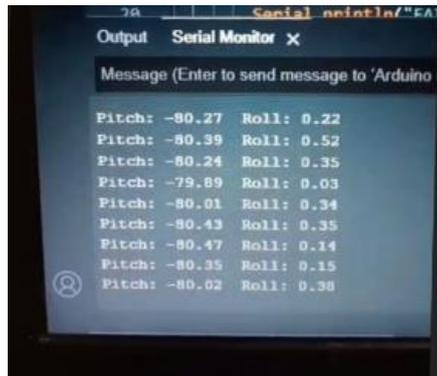
Saat terjadi jatuh ke kiri, sensor menunjukkan nilai positif. Ini karena sumbu koordinat sensor yang digunakan menganggap arah ke kiri sebagai arah positif pada sumbu tertentu. Orientasi ini menyebabkan pembacaan nilai positif ketika tubuh miring atau jatuh ke kiri.

Sebaliknya, saat terjadi jatuh ke kanan, sensor menunjukkan nilai negatif. Ini disebabkan oleh orientasi sumbu koordinat sensor yang menganggap arah ke kanan sebagai arah negatif pada sumbu tersebut. Karena itulah, saat tubuh miring atau jatuh ke kanan, sensor memberikan nilai negatif.

data menunjukkan hasil rata-rata dari setiap posisi jatuh sebagai berikut:

- Jatuh ke depan dengan sudut 45° : Rata-rata 45.89
- Jatuh ke depan dengan sudut 90° : Rata-rata 87.84
- Jatuh ke belakang dengan sudut 45° : Rata-rata 46.86
- Jatuh ke belakang dengan sudut 90° : Rata-rata -80.24
- Jatuh ke kiri : Rata-rata 82.90
- Jatuh ke kanan : Rata-rata : -86.07

Hasil pengujian ini menunjukkan bahwa sensor MPU6050 dapat mendeteksi variasi sudut jatuh dengan cukup baik pada pengujian pertama. Berikut di bawah ini tampilan serial monitor pada Gambar 4.6



Gambar 4 6 Serial Monitor Pada Pengujian Pertama

Gambar 4.6 menunjukkan tampilan serial monitor selama pengujian pertama. Gambar ini memperlihatkan data dari sensor MPU6050 ketika pengujian pertama melakukan berbagai gerakan jatuh. Data tersebut mencakup informasi mengenai sudut dan akselerasi yang digunakan untuk menganalisis respons sensor terhadap setiap jenis jatuh yang di uji.

Pengujian kedua menggunakan sensor MPU6050, pengujian ini bertujuan untuk mengukur dan mencatat respons sensor terhadap berbagai gerakan jatuh. Sama seperti pengujian pertama, pengujian ini juga dilakukan sebanyak tujuh kali untuk setiap jenis gerakan jatuh. Berikut adalah hasil dari pengujian yang tercantum pada tabel 4.2

Tabel 4 2 Pengujian Deteksi Jatuh Pada Pengujian Ke Dua

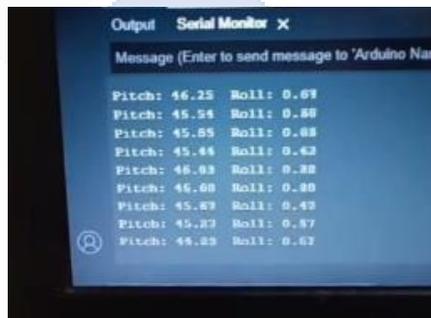
No	Posisi Pengujian					
	jatuh kedepan (45°)	jatuh kedepan (90°)	jatuh kebelakang (45°)	jatuh kebelakang (90°)	jatuh kekiri	jatuh Kekanan
1	46.08	88.85	-48.03	-83.92	77.48	-85.92
2	45.79	88.75	-47.59	-83.79	76.75	-86.11
3	46.10	89.47	-48.06	-83.97	76.23	-85.69
4	46.13	88.81	-47.67	-83.92	76.95	-85.79
5	45.72	89.34	-48.20	-83.37	76.97	-85.42

6	45.42	88.92	-47.61	-84.11	76.40	-85.87
7	45.70	88.39	-47.60	-84.05	76.96	-85.29
Rata-rata	45.85	88.93	-47.82	-83.88	76.82	-85.73

Pada tabel 4.2, data menunjukkan hasil rata-rata dari setiap posisi jatuh sebagai berikut:

- Jatuh ke depan dengan sudut 45° : Rata-rata 45.85
- Jatuh ke depan dengan sudut 90° : Rata-rata 88.93
- Jatuh ke belakang dengan sudut 45° : Rata-rata -47.82
- Jatuh ke belakang dengan sudut 90° : Rata-rata -83.88
- Jatuh ke kiri : Rata-rata 76.82
- Jatuh ke kanan : Rata-rata : -85.73

Data di atas menunjukkan bahwa sensor MPU6050 dapat mendeteksi variasi sudut jatuh dengan akurat, dengan rata-rata yang konsisten untuk setiap posisi jatuh yang di uji. Berikut di bawah ini tampilan serial monitor pada Gambar 4.7

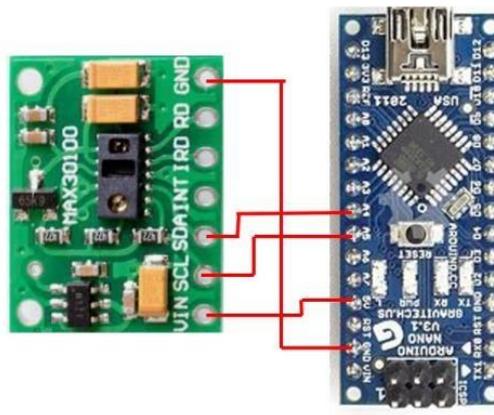


Gambar 4 7 Tampilan Serial Monitor Pengujian Ke Dua

Gambar 4.7 menunjukkan tampilan serial monitor selama pengujian pada penelitian kedua, Gambar ini menampilkan data dari sensor MPU6050 yang mencakup informasi mengenai sudut dan akselerasi, digunakan untuk menganalisis respons sensor terhadap berbagai gerakan jatuh yang dilakukan pada pengujian kedua.

4.2.2 Pengujian Sensor MAX30100

Pengambilan data sensor MAX30100 menggunakan jari telunjuk, jarak jari dan sensor yakni $\pm 2-3$ cm dengan delay yakni (800) atau 0,8 detik. Berikut di bawah ini wiring diagram dari pengujian sensor MAX30100.



Gambar 4.8 Wiring Diagram Sensor MAX30100 dengan Arduino Nano

Gambar 4.8 wiring diagram di atas menunjukkan koneksi antara modul sensor MAX30100 dan peran mikrokontroler Arduino Nano. Berikut adalah penjelasan terkait koneksi tersebut.

1. VCC (atau VIN) ke 5V:
 - Konektor VIN pada modul MAX30100 dihubungkan ke pin 5V pada Arduino Nano, ini menyediakan daya 5V untuk modul sensor dari papan Arduino.
2. GND ke GND:
 - Konektor GND pada modul MAX30100 dihubungkan ke pin GND pada Arduino Nano, ini memberikan jalur referensi bersama antara kedua perangkat.
3. SCL ke A5:
 - Konektor SCL (*Serial Clock Line*) pada modul MAX30100 dihubungkan ke pin A5 pada Arduino Nano, pin A5 pada Arduino Nano berfungsi sebagai pin SCL dalam komunikasi I2C.
4. SDA ke A4:

- Konektor SDA (*Serial Data Line*) pada modul MAX30100 dihubungkan ke pin A4 pada Arduino Nano. Pin A4 pada Arduino Nano berfungsi sebagai pin SDA dalam komunikasi I2C.

Komunikasi antara modul Max30100 dan Arduino Nano menggunakan protokol I2C, yang memerlukan dua jalur utama, yaitu SDA untuk data dan SCL untuk clock. Modul MAX30100 adalah sensor oksigen darah dan detak jantung yang populer yang memanfaatkan komunikasi I2C untuk berinteraksi dengan mikrokontroler seperti Arduino. Dengan konfigurasi ini, Arduino Nano dapat membaca data dari modul MAX30100, memprosesnya, dan kemudian menggunakan data tersebut untuk aplikasi seperti pemantauan kesehatan atau proyek IoT terkait kesehatan.

Berikut dibawah ini program antara sensor MAX30100 dengan Arduino Nano

```
1  #include <Wire.h>
2  #include "MAX30100_PulseOximeter.h"
3
4  #define REPORTING_PERIOD_MS 1000
5
6  PulseOximeter pox;
7  uint32_t tsLastReport = 0;
8
9  void onBeatDetected() {
10 |   Serial.println("Beat detected!");
11 | }
12
13 void setup() {
14 |   Serial.begin(9600);
15 |   Serial.print("Initializing pulse oximeter..");
16
17 |   // Inisialisasi sensor
18 |   if (!pox.begin()) {
19 |       Serial.println("FAILED");
20 |       for(;;);
21 |   } else {
22 |       Serial.println("SUCCESS");
23 |   }
24
25 |   // Mengatur callback untuk deteksi detak jantung
26 |   pox.setOnBeatDetectedCallback(onBeatDetected);
27 | }
28
29 void loop() {
```

```

15     Serial.print("Initializing pulse oximeter..");
16
17     // Inisialisasi sensor
18     if (!pox.begin()) {
19         Serial.println("FAILED");
20         for(;;);
21     } else {
22         Serial.println("SUCCESS");
23     }
24
25     // Mengatur callback untuk deteksi detak jantung
26     pox.setOnBeatDetectedCallback(onBeatDetected);
27 }
28
29 void loop() {
30     // Update sensor
31     pox.update();
32
33     // Melaporkan hasil setiap REPORTING_PERIOD_MS
34     if (millis() - tsLastReport > REPORTING_PERIOD_MS) {
35         tsLastReport = millis();
36         Serial.print("Heart rate:");
37         Serial.print(pox.getHeartRate());
38         Serial.print(" bpm / SpO2:");
39         Serial.print(pox.getSpO2());
40         Serial.println(" %");
41     }
42 }

```

Gambar 4.9 Sensor MAX30100 dengan Arduino Nano

- `#include <Wire.h>`: Menyertakan pustaka Wire untuk komunikasi I2C.
- `#include "MAX30100_PulseOximeter.h"`: Menyertakan pustaka untuk interfacing dengan pulse oximeter MAX30100
- `#define REPORTING_PERIOD_MS 1000`: Mendefinisikan konstanta `REPORTING_PERIOD_MS` dengan nilai 1000 milidetik(1 detik).
- `- PulseOximeter pox;`: Membuat instance dari kelas PulseOximeter.
- `- uint32_t tsLastReport = 0;`: Mendeklarasikan variabel untuk melacak waktu laporan terakhir
- `void onBeatDetected() { Serial.println("Beat detected!");` Fungsi ini akan dipanggil setiap kali detak jantung terdeteksi. Fungsi ini mencetak pesan "Beat detected!" ke Serial Monitor.

- - `Serial.begin(9600);`: Memulai komunikasi serial dengan baud rate 9600.
- `Serial.print("Initializing pulse oximeter..");`: Mencetak pesan inisialisasi.
- `if (!pox.begin());`: Mencoba menginisialisasi sensor pulse oximeter. Jika gagal, mencetak "FAILED" dan memasuki loop tanpa akhir.
- `else:` Jika berhasil, mencetak "SUCCESS".
- `pox.setOnBeatDetectedCallback(onBeatDetected);`: Mengatur callback untuk deteksi detak jantung ke fungsi `onBeatDetected`.
- `-pox.update();`: Memperbarui data dari sensor.
- `if (millis() - tsLastReport > REPORTING_PERIOD_MS):` Memeriksa apakah waktu yang telah berlalu lebih dari `REPORTING_PERIOD_MS` (1 detik).
- `tsLastReport = millis();`: Memperbarui waktu laporan terakhir.
- `Serial.print("Heart rate:");`: Mencetak teks "Heart rate:".
- `Serial.print(pox.getHeartRate());`: Mencetak nilai denyut jantung yang terdeteksi oleh sensor.
- `Serial.print(" bpm / SpO2:");`: Mencetak teks " bpm / SpO2:".
- `Serial.print(pox.getSpO2());`: Mencetak nilai SpO2 (saturasi oksigen) yang terdeteksi oleh sensor.
- `Serial.println(" %");`: Mencetak simbol persen dan memulai baris baru di Serial Monitor.

Pengujian dilakukan dengan menggunakan Tensimeter pada responden pada Gambar 4.10.



Gambar 4.10 Pengujian Menggunakan Tensimeter

Pengujian dilakukan dengan menggunakan tensimeter pada responden untuk mendapatkan data detak jantung dan kadar oksigen dalam darah sebagai pembandingan hasil dari tampilan OLED SSD1306 dan MIT App Inventor. Proses ini bertujuan untuk memastikan akurasi dan konsistensi perangkat dalam mendeteksi parameter vital tersebut, sehingga hasil yang diperoleh dapat diandalkan.

Berikut di bawah ini hasil perbandingan pengujian pertama Pada Tampilan Olled Ssd1306, Tampilan Mitt App Inventor, dan Tensimeter.

Tabel 4 3 Perbandingan Pengujian Pertama

No	Tampilan Oled ssd1306		Tampilan Mitt App Inventor		Tensimeter
	<i>Heart Rate</i>	SpO2	<i>Heart Rate</i>	SpO2	HeartRate

1	2 BPM	0%	2 BPM	0 %	2 BPM
2	31 BPM	94 %	34 BPM	94 %	33 BPM
3	72 BPM	97 %	72 BPM	97 %	75 BPM
4	87 BPM	96 %	87 BPM	96 %	87 BPM
5	91 BPM	97 %	91 BPM	97 %	91 BPM
6	90 BPM	97 %	90 BPM	97 %	90 BPM
7	94 BPM	97 %	94 BPM	97 %	95 BPM
8	95 BPM	97 %	95 BPM	97 %	95 BPM
9	94 BPM	96 %	94 BPM	96 %	94 BPM
10	93 BPM	97 %	93 BPM	97 %	93 BPM

BPM (*Beats Per Minute*) adalah satuan yang digunakan untuk mengukur jumlah detak jantung per menit. Normalnya, detak jantung orang dewasa berkisar antara 60-100 BPM saat istirahat. Nilai BPM dapat dipengaruhi oleh berbagai faktor seperti aktivitas fisik, emosi, dan kondisi kesehatan seseorang.

Dalam tabel 4.3, tampak bahwa hasil pengujian pertama menunjukkan variasi BPM yang diukur menggunakan tiga perangkat berbeda: OLED SSD1306, MIT App Inventor, dan Tensimeter. Pada sebagian besar pengujian, nilai BPM yang ditampilkan oleh ketiga perangkat hampir sama, dengan sedikit perbedaan di beberapa titik. Pada baris kedua, OLED SSD1306 menunjukkan 31 BPM sementara MIT App Inventor menunjukkan 34 BPM, dan Tensimeter menunjukkan 33 BPM. Perbedaan kecil ini bisa disebabkan oleh sensitivitas sensor, keterlambatan dalam pembacaan data, atau kalibrasi yang sedikit berbeda antar perangkat.

Secara keseluruhan, hasil pengujian menunjukkan bahwa ketiga perangkat memberikan hasil yang cukup konsisten dan dapat diandalkan untuk memonitor detak jantung. Hasil ini juga menegaskan bahwa perangkat berbasis OLED

SSD1306 dan MIT App Inventor dapat digunakan sebagai alternatif yang cukup akurat dibandingkan dengan Tensimeter konvensional.

Nilai rata-rata Perbandingan Pengujian Pertama yaitu:

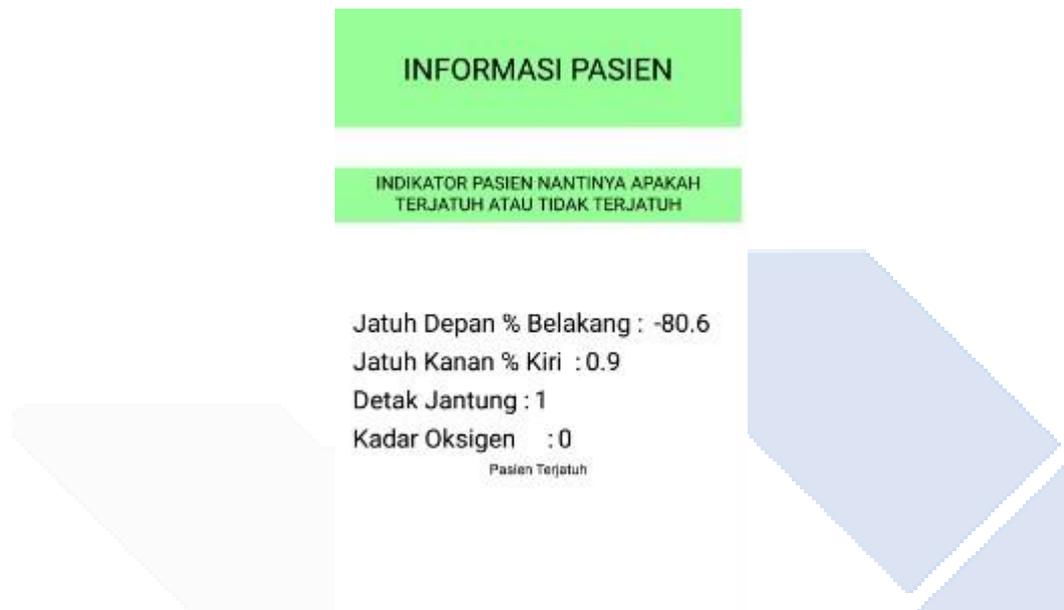
- Tampilan Oled Ssd1306 menunjukkan hasil heart rate dan SpO2 yang hampir konsisten dengan hasil dari tampilan MIT App Inventor dan tensimeter.
- Pada uji pertama, hasil heart rate yang ditampilkan oleh ketiga perangkat adalah 2 BPM.
- Pada uji kedua, hasil heart rate dari Oled Ssd1306 adalah 31 BPM, dari Mitt App Inventor 34 BPM, dan dari tensimeter 33 BPM dengan SpO2 konsisten di 94%
- Pada uji ketiga hasil heart rate dari Oled Ssd1306 dan Mitt App Inventor sama yaitu 72 BPM, sedangkan dari tensimeter 75 BPM dengan SpO2 konsisten di 97%
- Pada uji keempat hasil heart rate dari ketiga perangkat sama yaitu 87 BPM dengan SpO2 96%
- Pada uji kelima hingga uji ke sepuluh, hasil heart rate dari ketiga perangkat berkisar antara 90 BPM hingga 95 BPM dengan SpO2 konsisten di 96% atau 97%.

Dari hasil perbandingan tersebut, dapat disimpulkan bahwa tampilan Oled Ssd1306 dari Mitt App Inventor memberikan hasil sebanding dengan tensimeter, menunjukkan akurasi yang cukup baik dalam pengukuran heart rate dan SpO2.

4.3 Pengujian aktivasi Sistem melalui *SmartPhone*

Pada Bab 4, pengujian aktivasi sistem melalui *smartphone* diuraikan secara mendetail untuk memastikan setiap tahapan dilakukan dengan benar dan sistem dapat berfungsi sebagaimana mestinya. Langkah pertama yang harus dilakukan adalah memastikan bahwa *smartphone* yang digunakan terhubung dengan koneksi internet yang stabil dan memiliki daya baterai yang cukup untuk menyelesaikan

seluruh proses pengujian. Setelah itu, buka aplikasi sistem yang telah diunduh dan diinstal sebelumnya pada *smartphone*. Masukkan kredensial login yang telah diberikan, seperti *username* dan *password*, untuk mengakses aplikasi. Gambar 4.14 Tampilan Mitt App Inventor.



Gambar 4.11 Tampilan Mitt App Inventor

Setelah berhasil login, pengguna harus menemukan dan memilih menu "Aktivasi Sistem" yang biasanya terletak pada bagian utama atau menu pengaturan aplikasi. Pada halaman aktivasi, akan terdapat tombol "Aktifkan" yang harus ditekan untuk memulai proses aktivasi sistem. Setelah tombol "Aktifkan" ditekan, sistem akan mulai memproses permintaan aktivasi. Pengguna perlu memperhatikan notifikasi atau pesan yang muncul di layar *smartphone* untuk memastikan bahwa proses aktivasi berjalan dengan lancar. Notifikasi ini penting sebagai konfirmasi bahwa sistem telah berhasil diaktifkan.

Jika terdapat pesan kesalahan atau kegagalan dalam proses aktivasi, pengguna harus memeriksa kembali koneksi internet dan mencoba mengulangi langkah-langkah dari awal. Selain itu, pastikan bahwa aplikasi sistem yang digunakan adalah versi terbaru dan tidak terdapat pembaruan yang belum diinstal.

Apabila masalah tetap berlanjut, disarankan untuk menghubungi tim dukungan teknis untuk mendapatkan bantuan lebih lanjut.



BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan rancangan alat pendeteksi jatuh pada lansia dalam keadaan rawat jalan, kesimpulan yang dapat diambil sebagai berikut:

1. Alat pendeteksi jatuh pada lansia telah berhasil dirancang dan diuji menggunakan sensor MPU6050 dan mikrokontroler NodeMCU ESP32. Sensor ini mampu mendeteksi gerakan dan orientasi tubuh, serta membedakan antara gerakan normal dan kejadian jatuh. Hasil pengujian menunjukkan bahwa alat ini memiliki keakuratan tinggi dalam mendeteksi berbagai jenis jatuh, baik ke depan, belakang, samping kiri, maupun samping kanan.
2. Aplikasi Mitt App Inventor telah dikembangkan untuk secara otomatis menerima notifikasi dari alat pendeteksi jatuh ketika terjadi kejadian jatuh. Pengguna hanya perlu mengaktifkan aplikasi ini melalui smartphone, dan sistem akan terus memantau kondisi lansia serta memberikan notifikasi *real-time* kepada keluarga atau pengasuh. Proses aktivasi aplikasi juga telah diuji dan berjalan dengan baik.
3. Sistem monitoring kondisi detak jantung dan saturasi oksigen darah telah berhasil dirancang menggunakan sensor MAX30100 yang terhubung ke mikrokontroler ESP32. Detak jantung dan SpO₂ yang dihasilkan kemudian ditampilkan pada layar Olled Ssd1306 dan juga dikirimkan ke aplikasi Mitt App Inventor untuk pemantauan *real-time*. Pengujian menunjukkan bahwa sistem ini mampu memberikan hasil yang akurat dan konsisten dalam memonitor kondisi lansia.

5.2 Saran

1. Peningkatan Algoritma Deteksi : Algoritma deteksi jatuh dapat terus ditingkatkan untuk mengurangi kesalahan positif dan negatif, serta meningkatkan akurasi deteksi menggunakan teknologi *machine learning*.

2. Miniaturisasi Perangkat : Ukuran perangkat dapat diperkecil untuk meningkatkan kenyamanan pemakaian oleh lansia, dengan penggunaan komponen yang lebih kecil dan desain casing yang ergonomis.
3. Peningkatan Daya Tahan Baterai : Mengembangkan solusi untuk meningkatkan efisiensi energi dan daya tahan baterai agar perangkat dapat digunakan lebih lama tanpa sering diisi ulang.

Dengan mempertimbangkan kesimpulan dan saran di atas, diharapkan pengembangan lebih lanjut dari alat pendeteksi jatuh dan sistem monitoring kondisi kesehatan lansia dapat memberikan kontribusi yang lebih besar dalam meningkatkan kualitas hidup dan keamanan para lansia.



DAFTAR PUSTAKA

1. Herdianty S. *Hubungan Usia dengan Risiko Jatuh pada Lansia*. Skripsi FK-Usakti.2019
2. Kementerian Kesehatan Republik Indonesia. *Laporan Riskesdas Nasional 2018*. Kementerian Kesehatan Republik Indonesia, 2019.
3. Departemen Sosial RI, Direktorat Jendral Pelayanan dan Rehabilitas Sosial, Direktorat Bina Pelayanan Sosial Lanjut Usia. 1999. *Definisi Lansia*. Jakarta.
4. S. M. Kiiik, J. Sahar, and H. Permatasari, “ Peningkatan Kualitas Hidup Lanjut Usia (Lansia) Di Kota Depok Dengan Latihan Keseimbangan, “ *J. Keperawatan Indonesia.*, vol,21, no. 2, pp.109-116,2018.
5. BKKBN, “Info Demografi Vol 1 Tahun 2019,” *Bkkn*, vol.1,pp. 1-16, 2019.
6. S. Djaja, “ PENDUDUK LANJUT USIA DI INDONESIA MENURUT RISET KESEHATAN DASAR 2007 (Analysis Cause of Death and Threat Faced by Elderly Population in Indonesia according to Baseline Health Research 2007),” *Bul. Penelit,Sist, Kesehat.*, vol,15, no.4, pp. 323-330,2012.
7. Kemenkes RI,”Analisis Lansia di Indonesia,” Pus, data dan Inf. Kementrian. Kesehat. RI, pp. 1-2,2017
8. Azizah, Lilik Ma’rifatul. 2011. *Keperawatan Lanjut Usia*. Yogyakarta: Graha Ilmu.
9. A. Bourke, J. O’Brien, and G. Lyons, 2007, Evaluation of a thresholdbased triaxial accelerometer full detection algorithm, *Gait & Posture*, vol. 26, no.2, pp. 194-199.
10. Bandiyah, Siti, 2009. *Lanjut Usia dan Keperawatan Gerontik*. Yogyakarta: Nuha Medika.
11. Liandana, Made. Dkk. 2014. “Pengembangan Sistem Deteksi Jatuh Pada Lanjut Usia Menggunakan Sensor Accelerometer Pada Smartphone Android.

12. Effendy, R. A. P. (2020). *Perancangan prototipe pendeteksi gerakan jatuh pada lansia menggunakan sensor accelerometer berbasis IoT* (Skripsi, Universitas Islam Indonesia).
13. Tsani, S. D., & Mulyadi, I. H. (2019). Sistem pendeteksi jatuh wearable untuk lanjut usia menggunakan accelerometer dan gyroscope. *Journal Of Applied Electrical Engineering*, 3(2).
14. Maulana, F. A. (2018). *Sistem deteksi jatuh untuk manusia lanjut usia berbasis Arduino* (Skripsi, Universitas Islam Negeri Syarif Hidayatullah jakarta).
15. Sasmoko Dani, Arie Mahendra (2017), Rancang Bangun Sistem Pendeteksi Kebakaran Berbasis IoT dan Sms Gateway Meggunakan Arduino.
16. Dwijoseputra. S ,(2019) “*Sistem deteksi jatuh berbasis Internet of things*”
17. Gumilar, G., & Rachmat, H. H. (2018). *Sistem Pendeteksi Jatuh Wireless Berbasis Sensor Accelerometer*. Jurusan Teknik Elektro, Institut Teknologi Nasional Bandung.
18. H. Kusumah dan R. A. Pradana, “Penerapan trainer interfacing mikrokontroler dan internet of things berbasis esp32 pada mata kuliah interfacing.” *journal cerita*, vol.5,no.2, hlm. 120-134, 2019.
19. I. Suharjo,”Prototype Alat Kendali Otomatis Penjemur Pakaian Menggunakan NodeMCU ESP32 Dan Telegram Bot Berbasis Internet of Things(IoT),” *Journal of Information System And Artificial Intelligence*, vol.1, no.1, hlm. 17-24, 2020
20. Joy-it, “NodeMCU Esp32 Microcontroller Development Board,” Joy-it, 2018. https://cdnreichelt.de/documents/datenblatt/A300/sbc-nODEmcpu-Esp32-datasheet_V1.2.pdf (diakses jan 12, 2023)
21. Mikrokontroler ESP32, apa itu? (bagian 1). (2019) #Microcontrollers101 [Online]. Available:<https://timur.ilearning.me/2019/04/19/mikrokontroler-esp32-apa-itu/>

22. Mikrokontroler ESP32, apa itu? (bagian 2). (2019 #Microcontrollers101 [Online]. Available: <https://timur.ilearning.me/2019/04/19/detail-esp32-bagian-2-microcontrollers101/>
23. Firman. B, (2016) “ *Implementasi Sensor IMU MPU6050 Berbasis Serial I2C self-balancing robot*”
24. Darawan.D., Stefani. A., Santoso.D.B.” *Perancangan Sistem Instrumentasi berbasis Internet Of Things pada alat pendeteksi bangkitan kejang pengidap epilepsi*”
25. Firman, B. (2016). Implementasi sensor IMU MPU6050 berbasis serial I2C pada self-balancing robot. **Jurnal Teknologi Technoscientia**, 9(1), 1-6.
26. Sari, D. P., Kartika, W., Safitri, M., & Loniza, E. (2022). Alat bantu salat bagi disabilitas tunarungu berbasis gyroscope MPU6050. **PROtek: Jurnal Ilmiah Teknik Elektro**, 9(1), 1-6.
27. Ivensense. (2013). MPU6050 Datasheed. Mpu6000 And Mpu6050 Product Spesification.
28. Pandelaki, S., Sitanayah, L., & Liem, M. (2023). *Sistem Pendeteksi Jatuh Berbasis Internet of Things*. JEECOM, 5(1), April 2023. Universitas Katolik De La Salle Manado.
29. Marsa, M., & Syaryadi, M. (2019). *Penerapan Wearable Device untuk Mendeteksi Lansia Jatuh pada Rumah Aceh*. KITEKTRO: Jurnal Online Teknik Elektro, 4(3), 12-18. Jurusan Teknik Elektro dan Komputer, Fakultas Teknik, Universitas Syiah Kuala.
30. Harianto, B., Hidayat, A., & Hulu, F. N. (2021, Desember 15). Analisis penggunaan sensor MAX30100 pada sistem pendeteksi detak jantung berbasis IoT Blynk. Dalam **Seminar Nasional Teknologi, Sains dan Humaniora 2021 (SemanTECH 2021)**. Gorontalo. ISBN: 978-623-96213-1-5

31. Maxim Integrated, "Pulse Oximeter and Heart-Rate Sensor IC for Wearable Health," Lect. Notes Energy, vol. 38, pp. 1–29, 2014, [Online]. Available: www.maximintegrated.com
32. Sugiarto, & Herdiansyah, R. (2022). *Perancangan Alat Ukur Saturasi Oksigen dan Detak Jantung Berbasis Arduino dengan Menggunakan Sensor MAX30100 dan LCD*. Scientia Sacra: Jurnal Sains, Teknologi dan Masyarakat, 2(4), 62.
33. Sura Menda Ginting, Amir, H., & Ginting, R. S. (2022). Pengembangan Media Pembelajaran Mit App Inventor Berplatform Android Pada Materi Stoikiometri Di Kelas X Mipa Sman 7 Kota Bengkulu. *Alotrop*, 6(2), 102–109. <https://doi.org/10.33369/alo.v6i2.24345>
34. Cytron T."Product User Manual-HC SR04 Ultrasonic Sensor" Mei,2013
35. Budianto, P., Kusuma, W. A., & Syaifuddin. (2021). *Monitoring Jatuh Dengan Menggunakan Sensor Accelerometer dan Gyroscope*. REPOSITOR, 3(1), 51-64. Universitas Muhammadiyah Malang.

LAMPIRAN 1

DAFTAR RIWAYAT HIDUP

1. Data Pribadi

Nama Lengkap : Muhammad Rezi Madiza Arradea
Tempat/Tanggal Lahir : Sungailiat, 9 Juni 2003
Jenis Kelamin : Laki-Laki
Agama : Islam
Alamat Rumah : Jalan Batin Tikal, Gang Lingga
No. Telpon/Hp : 083169735542
Email : muhammadrezi22oon@gmail.com
Jurusan/Prodi : D-IV Teknik Elektronika
NPM : 1052120

2. Riwayat Pendidikan

SD : SD N 25 SUNGAILIAT
SMP : SMP N 2 SUNGAILIAT
SMA : SMK N 1 SUNGAILIAT

Sungailiat, 24 juli 2024



Muhammad Rezi Madiza Arradea

DAFTAR RIWAYAT HIDUP

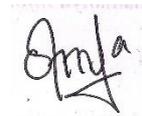
1. Data Pribadi

Nama Lengkap : Septia Eriska
Tempat/Tanggal Lahir : Sungailiat, 10 September 2003
Jenis Kelamin : Perempuan
Agama : Islam
Alamat Rumah : Desa Dalil
No. Telpon/Hp : 083151448185
Email : septiaeriskaa@gmail.com
Jurusan/Prodi : D-IV Teknik Elektro dan Informatika
NPM : 1052125

2. Riwayat Pendidikan

SD : SD N 5 BAKAM
SMP : SMP N 1 BAKAM
SMA : SMA N 1 BAKAM

Sungailiat, 24 Juli 2024



Septia Eriska

LAMPIRAN 2

1. Program Arduino Nano dan MPU6050

```
#include <Wire.h>
#include <MPU6050.h>
#include "MAX30100_PulseOximeter.h"
#include <SoftwareSerial.h>

MPU6050 mpu;
PulseOximeter pox;

// Buat instance SoftwareSerial
SoftwareSerial mySerial(10, 11); // RX, TX

#define REPORTING_PERIOD_MS 1000
uint32_t tsLastReport = 0;

void onBeatDetected() {
  Serial.println("Detak jantung terdeteksi!");
}

void setup() {
  Serial.begin(9600);
  mySerial.begin(9600); // Kecepatan baud untuk SoftwareSerial

  Wire.begin();

  // Inisialisasi MPU6050
  mpu.initialize();
```

```

if (!mpu.testConnection()) {
    Serial.println("MPU6050 tidak terhubung");
    while (1);
}

// Inisialisasi MAX30100
Serial.print("Menginisialisasi pulse oximeter..");
if (!pox.begin()) {
    Serial.println("Gagal memulai MAX30100");
    for (;;)
} else {
    Serial.println("Berhasil memulai MAX30100");
}
pox.setOnBeatDetectedCallback(onBeatDetected);
}

void loop() {
    // Pembacaan sensor MPU6050
    int16_t ax, ay, az;
    int16_t gx, gy, gz;

    // Pembacaan sensor MAX30100
    pox.update();
    if (millis() - tsLastReport > REPORTING_PERIOD_MS) {
        float heartRate = pox.getHeartRate();
        float spO2 = pox.getSpO2();

        Serial.print("Detak jantung: ");
        Serial.print(heartRate);
        Serial.print(" bpm / SpO2: ");

```

```

Serial.print(spO2);
Serial.println(" %");

// Kirim data ke ESP32 melalui SoftwareSerial

mpu.getMotion6(&ax, &ay, &az, &gx, &gy, &gz);

float axg = ax / 16384.0;
float ayg = ay / 16384.0;
float azg = az / 16384.0;

float roll = atan2(ayg, azg) * 57.2957795;
float pitch = atan2(-axg, sqrt(ayg * ayg + azg * azg)) * 57.2957795;

Serial.print("Roll: ");
Serial.print(roll);
Serial.print(" Pitch: ");
Serial.println(pitch);

// Kirim data roll dan pitch ke ESP32 melalui SoftwareSerial
mySerial.println((String)heartRate + "," + spO2 + "," + roll + "," +
    pitch);
Serial.println((String)heartRate + "," + spO2 + "," + roll + "," + pitch);

    tsLastReport = millis();
}
}

//final nano

```

2. Program Arduino Nano dan MAX30100

```

#include <Wire.h>
#include <Adafruit_SSD1306.h>
#include <Adafruit_GFX.h>

// Definisi pin OLED dan ukuran layar
#define SCREEN_WIDTH 128
#define SCREEN_HEIGHT 64
#define OLED_RESET -1
Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT,
    &Wire, OLED_RESET);

#include <HardwareSerial.h>
HardwareSerial mySerial(1); // Pilih serial port yang sesuai

void setup() {
    // Inisialisasi serial
    Serial.begin(115200);
    mySerial.begin(9600, SERIAL_8N1, 16, 17); // Ganti pin RX dan TX
        sesuai kebutuhan

    // Inisialisasi OLED
    if (!display.begin(SSD1306_I2C_ADDRESS, 0x3C)) { // Ganti dengan
        alamat I2C OLED jika berbeda
        Serial.println(F("SSD1306 allocation failed"));
        for (;;) // Loop tidak berakhir
        }
    display.clearDisplay();
    display.setTextSize(1);
    display.setTextColor(SSD1306_WHITE);
    display.setCursor(0,0);

```

```

display.println(F("Starting..."));
display.display();
}

void loop() {
  // Periksa jika ada data yang tersedia untuk dibaca
  if (mySerial.available()) {
    String data = mySerial.readStringUntil('\n'); // Membaca string sampai
    newline

    // Parsing data
    int firstComma = data.indexOf(',');
    int secondComma = data.indexOf(',', firstComma + 1);
    int thirdComma = data.indexOf(',', secondComma + 1);

    if (firstComma > 0 && secondComma > 0 && thirdComma > 0) {
      String heartRateStr = data.substring(0, firstComma);
      String spO2Str = data.substring(firstComma + 1, secondComma);
      String rollStr = data.substring(secondComma + 1, thirdComma);
      String pitchStr = data.substring(thirdComma + 1);

      // Convert to integers or floats as needed
      int heartRate = heartRateStr.toInt();
      int spO2 = spO2Str.toInt();
      float roll = rollStr.toFloat();
      float pitch = pitchStr.toFloat();

      // Tampilkan hasil parsing di serial monitor
      Serial.print("Heart Rate: ");
      Serial.println(heartRate);
    }
  }
}

```

```
Serial.print("SpO2: ");
Serial.println(spO2);
Serial.print("Roll: ");
Serial.println(roll);
Serial.print("Pitch: ");
Serial.println(pitch);

// Tampilkan hasil parsing di OLED
display.clearDisplay();
display.setCursor(0,0);
display.setTextSize(1);
display.print("Heart Rate: ");
display.println(heartRate);
display.print("SpO2: ");
display.println(spO2);
display.print("Roll: ");
display.println(roll);
display.print("Pitch: ");
display.println(pitch);
display.display();
}
}
}
```

3. Program FireBase

```
#include <WiFi.h>
#include <Firebase_ESP_Client.h>
```

```
#include "addons/TokenHelper.h"
#include "addons/RTDBHelper.h"
#include <Adafruit_MPU6050.h>
#include <Adafruit_Sensor.h>
#include <Wire.h>
```

```
Adafruit_MPU6050 mpu;
```

```
#define WIFI_SSID "modal"
#define WIFI_PASSWORD "minimalbiznet"

#define API_KEY "AIzaSyDwCEq4wvvSxIsGE_1iYvnwwoPGD_2vrfQ"
#define DATABASE_URL "https://ejik-8baa3-default-
    rtdb.firebaseio.com/"
#define vcc 19
#define d2 2
#define d3 3
FirebaseData fbdo;
FirebaseAuth auth;
FirebaseConfig config;
bool signupOK = false;
```

```
void setup(){
  Serial.begin(115200);
  WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
  Serial.print("Connecting to Wi-Fi");
  pinMode(d2,INPUT);
  pinMode(d3,INPUT);
  while (WiFi.status() != WL_CONNECTED){
```

```

    Serial.print(".");
    delay(300);
}
Serial.println();
Serial.print("Connected with IP: ");
Serial.println(WiFi.localIP());
Serial.println();
config.api_key = API_KEY;
config.database_url = DATABASE_URL;
if (Firebase.signUp(&config, &auth, "", "")){
    Serial.println("ok");
    signupOK = true;
}
else{
    Serial.printf("%s\n", config.signer.signupError.message.c_str());
}
config.token_status_callback = tokenStatusCallback; //see
    addons/TokenHelper.h

Firebase.begin(&config, &auth);
Firebase.reconnectWiFi(true);
while (!Serial)
    delay(10); // will pause Zero, Leonardo, etc until serial console opens

Serial.println("Adafruit MPU6050 test!");

// Try to initialize!
if (!mpu.begin()) {
    Serial.println("Failed to find MPU6050 chip");
    while (1) {

```

```
    delay(10);
  }
}
Serial.println("MPU6050 Found!");

mpu.setAccelerometerRange(MPU6050_RANGE_8_G);
Serial.print("Accelerometer range set to: ");
switch (mpu.getAccelerometerRange()) {
case MPU6050_RANGE_2_G:
  Serial.println("+2G");
  break;
case MPU6050_RANGE_4_G:
  Serial.println("+4G");
  break;
case MPU6050_RANGE_8_G:
  Serial.println("+8G");
  break;
case MPU6050_RANGE_16_G:
  Serial.println("+16G");
  break;
}
mpu.setGyroRange(MPU6050_RANGE_500_DEG);
Serial.print("Gyro range set to: ");
switch (mpu.getGyroRange()) {
case MPU6050_RANGE_250_DEG:
  Serial.println("+ - 250 deg/s");
  break;
case MPU6050_RANGE_500_DEG:
  Serial.println("+ - 500 deg/s");
  break;
}
```

```
case MPU6050_RANGE_1000_DEG:
    Serial.println("+ - 1000 deg/s");
    break;
case MPU6050_RANGE_2000_DEG:
    Serial.println("+ - 2000 deg/s");
    break;
}
```

```
mpu.setFilterBandwidth(MPU6050_BAND_5_HZ);
Serial.print("Filter bandwidth set to: ");
switch (mpu.getFilterBandwidth()) {
case MPU6050_BAND_260_HZ:
    Serial.println("260 Hz");
    break;
case MPU6050_BAND_184_HZ:
    Serial.println("184 Hz");
    break;
case MPU6050_BAND_94_HZ:
    Serial.println("94 Hz");
    break;
case MPU6050_BAND_44_HZ:
    Serial.println("44 Hz");
    break;
case MPU6050_BAND_21_HZ:
    Serial.println("21 Hz");
    break;
case MPU6050_BAND_10_HZ:
    Serial.println("10 Hz");
    break;
case MPU6050_BAND_5_HZ:
```

```

    Serial.println("5 Hz");
    break;
}

Serial.println("");
delay(100);
}

void loop(){
  if (Firebase.ready() && signupOK ) {
    sensors_event_t a, g, temp;
    mpu.getEvent(&a, &g, &temp);

    /* Print out the values */
    Serial.print("Acceleration X: ");
    Serial.print(a.acceleration.x);
    Serial.print(", Y: ");
    Serial.print(a.acceleration.y);
    Serial.print(", Z: ");
    Serial.print(a.acceleration.z);
    Serial.println(" m/s^2");
    Serial.println("");
    Firebase.RTDB.setFloat(&fbdo, "ejik/SumbuX",a.acceleration.x);
    Firebase.RTDB.setFloat(&fbdo, "ejik/SumbuY",a.acceleration.y);
    if(a.acceleration.x <= -9 || a.acceleration.x >= 9 || a.acceleration.y >= 9 ||
        a.acceleration.y <= -9){
      Firebase.RTDB.setFloat(&fbdo, "ejik/sensor",1);
    }else{
      Firebase.RTDB.setFloat(&fbdo, "ejik/sensor",0);
    }
  }
}

```

```
}  
Serial.println("_____");  
}
```

