



INDONESIA
SEKOLAH TINGGI MANAJEMEN INFORMATIKA & KOMPUTER

**RANCANG BANGUN ALAT PENDETEKSI SUHU
TUBUH DAN PENGENALAN IDENTITAS DIRI
MELALUI E-KTP DAN FOTO BERBASIS IOT**

**Gede Agus Eka Dharma
NIM. 17102009**

**Dosen Pembimbing I
I Gusti Made Ngurah Desnanjaya, M.T.**

**Dosen Pembimbing II
Wayan Sudiarsa, ST., M.Kom**

**PROGRAM STUDI SISTEM KOMPUTER
STMIK STIKOM INDONESIA
DENPASAR 2021**

HALAMAN INI SENGAJA DIKOSONGKAN

HALAMAN PERSETUJUAN

Judul : **RANCANG BANGUN ALAT PENDETEKSI
SUHU TUBUH DAN PENGENALAN
IDENTITAS DIRI MELALUI E-KTP DAN
FOTO BERBASIS IOT**

Nama : **Gede Agus Eka Dharma**

NIM : **17102009**

Program Studi : **Sistem Komputer**

Telah diperiksa dan disetujui sebagai Tugas Akhir mahasiswa
Program
Studi Sistem Komputer STMIK STIKOM Indonesia

Menyetujui:

Pembimbing I



I Gusti Made Ngurah Desnanjaya, M.T. Wayan Sudiarsa, ST., M.Kom
NIDN. 0820128802

Pembimbing II



NIDN. 0812028003

Mengetahui:

Kepala Program Studi Sistem Komputer



I Nyoman Buda Hartawan, M.Kom
NIDN. 0809078602

HALAMAN INI SENGAJA DIKOSONGKAN

HALAMAN PENGESAHAN

Judul : **RANCANG BANGUN ALAT PENDETEKSI
SUHU TUBUH DAN PENGENALAN
IDENTITAS DIRI MELALUI E-KTP DAN
FOTO BERBASIS IOT**

Nama : **Gede Agus Eka Dharma**

NIM : **17102009**


Program Studi : **Sistem Komputer**

Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Komputer pada Program Studi Sistem Komputer, STMIK STIKOM Indonesia.


Ditetapkan di : Denpasar
Tanggal : 30 Juni 2021

DEWAN PENGUJI


Pembimbing I :

I Gusti Made Ngurah Desnanjaya, M.T ()

Penguji : I

Anak Agung Gede Bagus Ariana, S.T., M.T. ()

Penguji : II

Anak Agung Gde Ekayana, S.Pd., M.Pd. ()

HALAMAN INI SENGAJA DIKOSONGKAN

HALAMAN PERNYATAAN ORISINALITAS

Judul : RANCANG BANGUN ALAT PENDETEKSI
SUHU TUBUH DAN PENGENALAN
IDENTITAS DIRI MELALUI E-KTP DAN
FOTO BERBASIS IOT
Nama : Gede Agus Eka Dharma
NIM : 17102009
Program Studi : Sistem Komputer

Dengan ini saya menyatakan bahwa Tugas Akhir ini merupakan hasil karya saya sendiri, semua sumber baik yang dikutip maupun dirujuk telah saya cantumkan dengan benar dan sepanjang pengetahuan saya juga tidak terdapat karya yang sama pernah ditulis atau diterbitkan orang lain. Apabila terdapat permasalahan terhadap isi dari Tugas Akhir ini maka akan menjadi tanggung jawab saya sepenuhnya.

Denpasar, 30 Juni 2021



(Gede Agus Eka Dharma)

HALAMAN INI SENGAJA DIKOSONGKAN

HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI

Sebagai civitas akademik STMIK STIKOM Indonesia, saya yang bertanda tangan di bawah ini:

Nama : Gede Agus Eka Dharma
NIM : 17102009
Program Studi : Sistem Komputer
Jenis Karya : Tugas Akhir

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada STMIK STIKOM Indonesia **Hak Bebas Royalti Noneksklusif** (*Non-exclusive Royalty-Free Right*) atas tugas akhir saya yang berjudul:

RANCANG BANGUN ALAT PENDETEKSI SUHU TUBUH DAN PENGENALAN IDENTITAS DIRI MELALUI E-KTP DAN FOTO BERBASIS IOT

Beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Non-ekskutif ini STMIK STIKOM Indonesia berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan mempublikasikan Tugas Akhir saya tanpa meminta izin dari saya selama tetap mencantumkan saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta. Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Denpasar
Pada tanggal : 30 Juni 2021

Yang menyatakan



(Gede Agus Eka Dharma)

HALAMAN INI SENGAJA DIKOSONGKAN

ABSTRAK

RANCANG BANGUN ALAT PENDETEKSI SUHU TUBUH DAN PENGENALAN IDENTITAS DIRI MELALUI E-KTP DAN FOTO BERBASIS IOT

Nama Mahasiswa : Gede Agus Eka Dharma
NIM : 17102009
Program Studi : Sistem Komputer
Dosen Pembimbing I : I Gusti Made Ngurah Desnanjaya, M.T.
Dosen Pembimbing II : I Wayan Sudiarsa, ST., M.Kom.

Abstrak

Salah satu upaya penegakan *protocol* kesehatan adalah penggunaan alat pemeriksa suhu tubuh pada setiap tempat umum. Ditempat umum pengecekan suhu tubuh menggunakan *thermometer gun*. Masalah yang di hadapi saat melakukan pengecekan sistem protokol kesehatan di antaranya adalah pendataan suhu dan identitas pengunjung yang memasuki Kampus STMIK STIKOM Indonesia agar riwayat data pengunjung setiap hari dapat dilihat kembali dan tersimpan secara otomatis, Realialisasi pemeriksaan suhu tubuh dapat menggunakan sensor MLX90614 yang memiliki kelebihan dapat membaca suhu tubuh tanpa memerlukan kontak langsung antara tubuh dengan sensor dan untuk memantau pengunjung menggunakan RFID yang menggunakan E-KTP sebagai tag identitas serta ESP32CAM untuk mengambil gambar wajah pengunjung untuk didata dan mengirimkan data ke internet. Tujuan melakukan penelitian ini adalah merancang bangun alat pendeteksi suhu tubuh dan pengenalan identitas diri melalui E-KTP dan Foto berbasis IoT, menjelaskan sistem kerja dan kinerja dari alat pendeteksi suhu tubuh dan pengenalan identitas diri melalui E-KTP dan Foto berbasis IoT. Dari hasil pengujian pada alat pendeteksi suhu tubuh dan pengenalan identitas diri melalui E-KTP dan Foto berbasis IoT, didapatkan hasil bahwa sistem ini mampu mengambil data suhu, ID E-KTP, dan foto Wajah. Total dari perbandingan antara sensor suhu MLX90614 dengan thermogun pabrikan mendapatkan mendapatkan nilai rata-rata sebesar 0,2 derajat celcius.

Kata Kunci : *ESP32CAM, RFID, Absensi*

HALAMAN INI SENGAJA DIKOSONGKAN

ABSTRACT

RANCANG BANGUN ALAT PENDETEKSI SUHU TUBUH DAN PENGENALAN IDENTITAS DIRI MELALUI E-KTP DAN FOTO BERBASIS IOT

Name : Gede Agus Eka Dharma
StudentID : 17102009
Program Study : Computer System
Supervisor : I Gusti Made Ngurah Desnanjaya, M.T.
: I Wayan Sudiarsa, ST., M.Kom.

Abstract

One of the efforts to enforce the health protocol is the use of body temperature checkers in every public place. In public places, check body temperature using a thermometer gun. The problems faced when checking the health protocol system include collecting temperature data and the identity of visitors who enter the STMIK STIKOM Indonesia Campus so that the history of visitor data can be viewed and stored automatically every day. Realization of body temperature checks can use the MLX90614 sensor which has the advantage of being able to reading body temperature without requiring direct contact between the body and sensors and to monitor visitors using RFID which uses E-KTP as an identity tag and ESP32CAM to take pictures of visitors' faces to be recorded and send data to the internet. The purpose of this research is to design a body temperature detection device and identify self-identity through IoT-based E-KTP and photos, explain the work system and performance of body temperature detectors and identify self-identity through IoT-based E-KTP and photos. From the test results on body temperature detectors and identification of self through E-KTP and IoT-based photos, the results show that this system is able to retrieve temperature data, E-KTP ID, and facial photos. The total comparison between the MLX90614 temperature sensor and the manufacturer's thermogun gets an average value of 0.2 degrees Celsius.

Keywords: ESP32CAM, RFID, Attendance

HALAMAN INI SENGAJA DIKOSONGKAN

KATA PENGANTAR

Puji syukur penulis panjatkan kepada Tuhan Yang Maha Esa, karena berkat rahmat-Nya Seminar dengan judul Rancang Bangun Alat Pengisian dan Pengurasan Kolam Air Serta Pengaturan Lampu Pemanas Pada Kandang Entok Secara Otomatis Berbasis Arduino Uno dengan format Capitalize Each Word dapat diselesaikan.

Penulis menyadari dalam menyelesaikan laporan Tugas Akhir ini tidak terlepas dari berbagai pihak yang telah membantu dalam membimbing dan membantu baik secara moral, materi, dan ilmu pengetahuan. Untuk itu penulis menyampaikan terimakasih kepada :

1. Bapak I Dewa Made Krishna Muku, M.T., sebagai Ketua STMIK STIKOM Indonesia
2. Bapak I Nyoman Buda Hartawan, M.Kom., sebagai Kepala Program Studi Sistem Komputer STMIK STIKOM INDONESIA.
3. Bapak I Gusti Made Ngurah Desnanjaya, M.T., sebagai dosen pembimbing I
4. Bapak I Wayan Sudiarsa, S.T., M.Kom., sebagai dosen pembimbing II
5. Kepada keluarga dan teman-teman yang telah banyak memberikan bantuan, baik berupa doa, moral dan material.

Untuk itu kritik dan saran dari semua pihak untuk penyempurnaan sangat penulis hargai.

Denpasar, 30 Juni 2021

Penulis

HALAMAN INI SENGAJA DIKOSONGKAN

DAFTAR ISI

HALAMAN PERSETUJUAN	iii
HALAMAN PENGESAHAN	v
HALAMAN PERNYATAAN ORISINALITAS	vii
HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI.....	ix
ABSTRAK	xi
ABSTRACT	xiii
KATA PENGANTAR.....	xv
DAFTAR ISI	xvii
DAFTAR TABEL	xxi
DAFTAR GAMBAR.....	xxi
DAFTAR LAMPIRAN	xxiii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Batasan Masalah	2
1.4 Tujuan Penelitian	2
1.5 Manfaat Penelitian	2
1.6 Sistematika Penulisan	3
BAB II LANDASAN TEORI.....	5
2.1 Penelitian Terdahulu	5
2.2 Mikrokontroler.....	8
2.2.1 Arduino.....	9
2.2.2 ESP32 CAM	12
2.3 RFID	13
2.4 LCD 16x2	14
2.5 Sensor MLX90614.....	15
2.6 Buzzer	16
2.7 Program Arduino IDE.....	16
2.8 Website	17
2.9 DataBase	17

2.10	MySQL.....	18
2.11	XAMPP	18
2.12	Google Sketup	19
2.13	Bahasa Pemograman	19
2.14	EAGLE.....	20
2.15	Flowchart.....	20
BAB III METODOLOGI PENELITIAN		23
3.1	Metode Pengumpulan Data	23
3.1.1	Metode Pengumpulan Data Pengujian Lab	23
3.2	Analisis Kebutuhan Sistem.....	26
3.3	Diagram Blok Sistem	28
3.4	Perancangan <i>Box</i> Dan Penempatan alat.....	29
3.5	Flowchart.....	30
3.6	Perancangan Website.....	31
3.6.1	Perancangan Halaman Login	31
3.6.2	Perancangan Home Page.....	32
3.6.3	Halaman Daftar User	33
3.6.4	Halaman Data Alat.....	33
3.6.5	Halaman Data RFID.....	34
3.6.6	Halaman Absensi	35
3.6.7	Halaman History Alat	36
3.6.8	Halaman Setting.....	36
3.7	Skematik Sistem.....	37
3.8	Skenario Pengujian Sistem	37
3.9	Jadwal Penelitian	38
BAB IV IMPLEMENTASI DAN PEMBAHASAN.....		39
4. 1	Implmentasi.....	39
4. 2	Pemasangan Komponen	39
4.2. 1	Tampilan Alat.....	39
4. 3	Implementasi Halaman Website.....	41
4.3. 1	Halaman Login.....	41
4.3. 2	Halaman Home Page.....	41
4.3. 3	Halaman Daftar User.....	42
4.3. 4	Halaman Data Alat.....	43

4.3. 5 Halaman Data RFID	43
4.3.6 Halaman Absensi	44
4.3. 7 Halaman Histori alat	44
4.3. 8 Halaman Setting	45
4. 4 Pengujian Koneksi ESP32Cam.....	45
4.4.1Tampilan Serial Monitor Pada Saat Mendapatkan Data ..	46
4.4.2Implementasi Pengambilan Gambar Pada ESP32Cam	47
4. 5 Pengujian Black Box Website	48
4. 6 Pengujian Suhu Tubuh.....	52
 BAB V PENUTUP	 54
5.1 Kesimpulan	54
5.2 Saran	54
DAFTAR PUSTAKA	
DAFTAR LAMPIRAN	
BIODATA	

DAFTAR TABEL

Tabel 2. 1 Penelitian Terdahulu	5
Tabel 2. 2 Spesifikasi <i>ESP32 Cam</i>	13
Tabel 3. 1 Pengujian Sensor MLX90614	23
Tabel 3. 2 Kebutuhan Perangkat keras	26
Tabel 3. 3 Perangkat Lunak	28
Tabel 4. 1 Pemasangan PIN RFID Reader Dengan Arduino Uno	40
Tabel 4. 2 Pemasangan PIN LCD 16x2 I2C Dengan Arduino Uno	41
Tabel 4. 3 Pemasangan PIN Sensor MLX90614 Dengan Arduino Uno	41
Tabel 4. 4 Pemasangan PIN ESP32Cam Dengan Arduino Uno.....	41

HALAMAN INI SENGAJA DIKOSONGKAN

DAFTAR GAMBAR

Gambar 2.1 Arduino Uno R3	10
Gambar 2.2 Diagram Pinout Arduino Uno R3	11
Gambar 2.3 ESP32 Cam	12
Gambar 3.1 Dokumentasi Pada Saat Pengecekan Suhu	24
Gambar 3.2 Dokumentasi Pengecekan Suhu Pada Noval	24
Gambar 3.3 Pengecekan Pengecekan Suhu Pada Uzman	25
Gambar 3.4 Pengecekan Pengecekan Suhu Pada Rahimin	25
Gambar 3.5 Pengecekan Pengecekan Suhu Pada Agus	26
Gambar 3.6 Diagram Blok Perancangan Sistem	29
Gambar 3.7 Bagian Dalam Box	29
Gambar 3.8 Bagian Depan Box	30
Gambar 3.9 Halaman Login	32
Gambar 3.10 Halaman Home Page	32
Gambar 3.11 Halaman Daftar User	33
Gambar 3.12 Halaman Info Pengunjung	33
Gambar 3.13 Halaman Kartu RFID Baru	34
Gambar 3.14 Halaman Data RFID	34
Gambar 3.15 Halaman Absensi	35
Gambar 3.16 Halaman <i>History</i> Alat	36
Gambar 3.17 Halaman Setting	36
Gambar 3.18 Skematik Rangkain	37
Gambar 4.1 Box Alat	39
Gambar 4.2 pemasangan komponen kedalam box	40
Gambar 4.3 Code Inisialisasi variable dan library	43
Gambar 4.4 Code Inisialisasi Notifikasi	43
Gambar 4.5 Code Transfer data ESP32Cam ke Arduino Uno	44
Gambar 4. 6 Code Transfer data ESP32Cam ke Arduino Uno	46
Gambar 4.7 Code Notifikasi Buzzer Saat Tap E-KTP	47
Gambar 4.8 Code Inisialisasi Variable dan Library ESP32Cam	48
Gambar 4.9 Code Koneksi dan Sinkronisasi Data dari ESP32Cam Ke Web	50
Gambar 4.10 Code Pengaturan Ukuran Gambar dari ESP32Cam	52
Gambar 4.11 Code Untuk penggantian Mode Baca dan Menambah	52
Gambar 4.12 Proses Koneksi ESP32Cam Dengan Wifi Server	58

Gambar 4.13 Proses Pengambilan Data E-KTP dan Suhu Badan.....59

Gambar 4.14 Proses Pengambilan Gambar59

DAFTAR LAMPIRAN

1. Lampiran Coding
2. Lampiran Dokumentasi Pengambilan Data
3. Lampiran Laporan Cek Plagiarisme

HALAMAN INI SENGAJA DIKOSONGKAN

BAB I

PENDAHULUAN

1.1 Latar Belakang

Virus Corona dapat menyebar melalui media langsung, udara, air, benda dan beberapa faktor penyakit. Salah satu penyakit menular di seluruh dunia adalah Corona virus Disease 19 (COVID-19). COVID-19 adalah virus jenis baru yang belum teridentifikasi sebelumnya terhadap manusia. Virus ini biasanya terdeteksi dalam 2 sampai 14 hari setelah terpapar. Presiden Joko Widodo belum lama ini telah menandatangani Instruksi Presiden (Inpres) Nomor 6 tahun 2020 tentang Peningkatan Disiplin dan Penegakan Hukum Protokol Kesehatan dalam Pencegahan dan Pengendalian Covid-19. Salah satu upaya penegakan protokol kesehatan adalah penggunaan alat pemeriksa suhu tubuh pada setiap tempat umum. Ditempat umum pengecekan suhu tubuh menggunakan *thermometer gun* dan pengecekan data pengunjung dalam ruangan dihitung manual. Pada saat pengecekan suhu dan pengecekan data.

Kampus STMIK STIKOM Indonesia yang beralamatkan di Jalan Tkad Pakerisan no 97 Denpasar yang bergerak dalam bidang pendidikan, juga menerapkan sistem protokol kesehatan yang dianjurkan oleh pemerintah. Menurut Bapak Agung Putra selaku petugas keamanan Kampus STMIK STIKOM Indonesia ada salah satu masalah yang di hadapi saat melakukan pengecekan sistem protokol kesehatan di antaranya adalah bagaimana pendataan suhu dan identitas pengunjung yang memasuki Kampus STMIK STIKOM Indonesia agar riwayat data dari pengunjung setiap hari dapat dilihat kembali dan tersimpan secara otomatis, karena pada saat ini pendataan pengunjung belum diterapkan pada saat pengecekan suhu. Berdasarkan hal tersebut dibutuhkan sistem yang bekerja secara otomatis untuk menggantikan dan mempermudah petugas keamanan untuk memeriksa suhu badan dan data diri orang yang masuk ke dalam gedung.

Maka disini penulis ingin menciptakan sebuah alat yang mampu mendeteksi suhu dan data diri secara otomatis serta digabungkan dengan sistem *Internet of Things* (IoT) agar data bisa dipantau dimana saja. *Realialisasi* pemeriksaan suhu tubuh dapat menggunakan sensor MLX90614 yang memiliki kelebihan dapat membaca suhu tubuh tanpa memerlukan kontak langsung antara tubuh dengan sensor dan untuk memantau pengunjung menggunakan RFID yang menggunakan E-KTP sebagai tag identitas serta ESP32CAM untuk mengambil gambar wajah pengunjung untuk didata dan mengirimkan data ke internet. Dari permasalahan diatas maka penulis menyimpulkan untuk mengangkat judul “Rancang bangun alat pendeteksi suhu tubuh dan pengenalan identitas

diri melalui E-KTP dan Foto berbasis IoT”. Untuk mendukung sistem pengecekan suhu tubuh dan pendataan pengunjung yang saat ini masih menggunakan cara manual.

1.2 Rumusan Masalah

Berdasarkan latar belakang di atas penulis mampu merumuskan masalah didalam penelitian yaitu :

- 1) Bagaimana cara merancang bangun alat pendeteksi suhu tubuh dan pengenalan identitas diri melalui E-KTP dan Foto berbasis IoT?
- 2) Bagaimana sistem kerja dari alat pendeteksi suhu tubuh dan pengenalan identitas diri melalui E-KTP dan Foto berbasis IoT?

1.3 Batasan Masalah

Dalam pembuatan alat ini penulis menemukan batasan masalah yaitu :

- 1) Alat ini mampu bekerja pada area yang memiliki akses Wi-Fi yang terhubung pada jaringan internet yang terdapat pada area STMIK STIKOM Indonesia.
- 2) Alat ini mampu mengukur suhu tubuh dengan maksimal jarak 15 Cm dan untuk mengenali identitas menggunakan E-KTP data harus diinputkan manual terlebih dahulu jika belum terdaftar di *database*.
- 3) Hasil dari data pengukuran yang terkirim ke *website* harus terhubung dengan jaringan internet agar bisa diakses.

1.4 Tujuan Penelitian

Adapun tujuan penulis untuk melakukan penelitian ini agar dapat mengetahui tentang proses perancangan alat pendeteksi suhu tubuh dan pengenalan identitas diri melalui E-KTP dan Foto berbasis IoT.

- 1) Rancang bangun alat pendeteksi suhu tubuh dan pengenalan identitas diri melalui E-KTP dan Foto berbasis IoT.
- 2) Menjelaskan sitem kerja dan kinerja dari alat pendeteksi suhu tubuh dan pengenalan identitas diri melalui E-KTP dan Foto berbasis IoT.

1.5 Manfaat Penelitian

Adapun manfaat dari penelitian ini yaitu :

- 1) Bagi Masyarakat
- 2) Dapat membantu mahasiswa, dosen, dan staff STMIK STIKOM Indonesia dalam mengukur dan *memonitoring* suhu tubuh secara otomatis.
- 3) Bagi Instansi
Mampu dijadikan media untuk belajar bagi para mahasiswa dimasa pembelajaran berikutnya.
- 4) Bagi penulis
Penulis mendapatkan pengalaman serta wawasan dan mampu menerapkan hasil pembelajaran dari penelitian di dalam masyarakat.

1.6 Sistematika Penulisan

Penulisan laporan penelitian ini dibagi menjadi lima bab, yaitu:

BAB I PENDAHULUAN

Mengulas tentang latar belakang, masalah, batasan masalah, tujuan, manfaat, dan sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Pada bab ini mengulas tentang penelitian serupa yang pernah dilakukan serta teori-teori terkait dan perancangan sistem.

BAB III ANALISIS DAN PERANCANGAN

Pada bab ini mengulas mengenai metode yang dilakukan selama proses penelitian, mulai dari pengumpulan data, perancangan.

BAB IV IMPLEMENTASI DAN PEMBAHASAN

Bab ini membahas implementasi rancangan ulang ke bentuk jadi dan analisis pengujian alat.

BAB V PENUTUP

Bab ini berisi simpulan dan saran selama penelitian dilakukan.

HALAMAN INI SENGAJA DIKOSONGKAN

BAB II

LANDASAN TEORI

2.1 Penelitian Terdahulu

Penelitian (Saputra, Karmel, and Zainal 2020) Perancangan dan Implementasi *Rapid Temperature Screening Contactless* dan Jumlah Orang Berbasis IoT dengan Protokol MQTT. Sistem terdiri dari tiga bagian yaitu sensor, prosesor, dan transduser. Pada bagian sensor terdiri dari sensor ultrasonik, sensor suhu MLX90614. Prosesor yang digunakan adalah NodeMCU untuk memproses hasil data dari sensor. Transduser terdiri dari Buzzer yang berfungsi sebagai notifikasi suara dan LCD 16x2 berfungsi sebagai penampil hasil dari keluaran sensor. Perancangan dan Implementasi *Rapid Temperature Screening Contactless* dan Jumlah Orang Berbasis IoT dengan Protokol MQTT menggunakan sistem kontrol otomatis, saat sistem otomatis aktif saat sensor ultrasonik mendeteksi seseorang dan sensor suhu otomatis membaca suhu tubuh dan mengirimkan data-data jumlah waktu pengiriman data dari sistem pemeriksa suhu dan jumlah orang menggunakan webserver Thingsboard.

Menurut penelitian yang dilakukan oleh (Perdana dkk. 2019) Alat Pemantau Kondisi Seorang *Gamer*. Alat ini terdiri dari beberapa bagian yaitu berupa, Arduino Atmega 328, LCD, I2C Module, Infrared Thermometer MLX90614, Pulse Sensor SEN-115-74, DF Player Mini, SIM900A dan RTC DS3231. Penelitian ini dilakukan untuk kondisi suhu badan dan detak jantung seorang gamer. Yang dimana apabila suhu badan dan kondisi detak jantungnya sudah tidak normal maka akan dikirimkan notifikasi berupa SMS kepada orang tua mereka, sehingga bisa mengingatkan mereka untuk segera istirahat.

Tabel 2. 1 Penelitian Terdahulu

Keterangan	Penelitian 1	Penelitian 2
Nama peneliliti	(Saputra, Karmel dan Zainal)	(Perdana dkk)
Tahun Penelitan	2020	2019
Judul Penelitian	Perancangan dan Implementasi Rapid Temperature Screening Contactless dan	Alat Pemantau Kondisi Seorang Gamer

	Jumlah Orang Berbasis IoT dengan Protokol MQTT	
Deskripsi	Diterapkan protocol MQTT untuk mengumpulkan data, serta menggunakan sensor MLX90614 untuk bisa membaca suhu tubuh tanpa melakukan kontak dengan orangnya langsung, sementara sensor ultrasonic digunakan untuk menghitung berapa jumlah orang yang masuk kedalam ruangan tersebut. Dan digunakan NodeMCU ESP8266 untuk bisa diintegrasikan dengan mikrokontroller dan terkoneksi dengan internet.	Penelitian ini dilakukan untuk mengontrol suhu tubuh seorang gamer. Yang dimana alat ini akan mengontrol suhu badan dan kondisi detak jantung seorang gamer dan apabila kondisi tubuhnya sudah tidak normal maka sistem akan otomatis mengirimkan SMS kepada orang tuanya yang bertujuan untuk mengingatkannya untuk beristirahat.
Metode Penelitian	Proses penelitian dimulai dengan melakukan studi literatur mengenai	Proses penelitian ini dimulai dengan menguji semua alat yang akan

	<p>sistem yang akan dirancang.</p> <p>Selanjutnya dilakukan analisa kebutuhan. Analisa kebutuhan terdiri atas analisa kebutuhan perangkat keras dan analisa kebutuhan perangkat lunak yang digunakan untuk</p>	<p>digunakan untuk merancang sistem.</p> <p>Selanjutnya dilakukan pengujian kepada seorang pria berusia 23 tahun.</p>
--	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------

Hasil Penelitian	<p>Berdasarkan hasil pengujian sensor MLX 90614 bisa digunakan untuk alat ini dan sensor ultrasonic dapat menghitung jumlah orang dapat dilihat di LCD dan webserver thingsboard.</p> <p>Sedangkan untuk komunikasi device dengan UI protocol MQTT berhasil direalisasikan MQTT broker HiveMQ dengan menentukan device sebagai publisher ke broker HiveMQ dan smarthphone atau PC sebagai subscriber.</p>	<p>Berdasarkan hasil penelitian. Alat yang dibuat telah berhasil karena dapat membaca suhu tubuh, detak jantung dan menampilkan lama waktu bermain dan memberikan peringatan berupa SMS dan Suara ketika kondisi suhu tubuh atau detak jantung dalam kondisi tidak normal atau terlalu lama bermain game.</p>
------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

2.2 Mikrokontroler

Menurut (Sabran and Djawad 2018) Mikrokontroler adalah sebuah sistem komputer lengkap dalam satu chip. Mikrokontroler lebih dari sekedar sebuah mikroprosesor karena sudah terdapat atau berisikan ROM (Read-Only Memory), RAM (Read-Write Memory), beberapa port masukan maupun keluaran, dan beberapa peripheral seperti pencacah/pewaktu, ADC (Analog to Digital converter), DAC (Digital to Analog converter) dan serial komunikasi. Salah satu mikrokontroler yang banyak digunakan saat ini yaitu mikrokontroler AVR. AVR adalah mikrokontroler RISC (Reduce Instruction Set Compute) 8 bit berdasarkan

arsitektur Harvard. Secara umum mikrokontroler AVR dapat dikelompokkan menjadi 3 kelompok, yaitu keluarga AT90Sxx, ATmega dan ATtiny. Pada dasarnya yang membedakan masing-masing kelas adalah memori, peripheral, dan fiturnya. Seperti mikroprosesor pada umumnya, secara internal mikrokontroler ATmega16 terdiri atas unit-unit fungsionalnya Arithmetic and Logical.

mikrokontroler ATmega16 terdiri dari: 1. Arsitektur RISC dengan throughput mencapai 16 MIPS pada frekuensi 16Mhz..

2. Memiliki kapasitas Flash memori 16Kbyte, EEPROM 512 Byte, dan SRAM 1Kbyte.

3. Saluran I/O 32 buah, yaitu Port A, Port B, Port C, dan Port D.

4. CPU yang terdiri dari 32 buah register. 5. User interrupt internal dan eksternal. 6. Port antarmuka SPI dan Port USART sebagai komunikasi serial

7. Fitur *Peripheral*: (a) Dua buah 8-bit timer/counter dengan *prescaler* terpisah dan mode *compare*, (b) Satu buah 16-bit *timer/counter* dengan *prescaler* terpisah, mode *compare*, dan mode *capture*. (c) Real time counter dengan osilator tersendiri, (d) Empat kanal PWM dan Antarmuka komparator analog. (e) 8 kanal, 10 bit ADC, (f) *Byte-oriented Two-wire Serial Interfac*, (g) *Watchdog timer dengan osilator internal*.

2.2.1 Arduino

menurut (Bate, Sartika Wiguna, and Aditya Nugraha 2020) Arduino merupakan system elektronik yang berbasis *open-source* yang fleksibel dan lebih mudah untuk dipergunakan baik itu dari segi perangkat keras maupun perangkat lunak. Oleh karena itu, untuk menguatkan arduino adalah jumlah pemakai yang sangat banyak hingga menyediakan tempat kode program yang akan koding dengan sesama perangkat keras ataupun modulasi yang mendukung (*hardware support modules*) dengan jumlah yang cukup banyak. Tentu saja hal ini untuk dapat memudahkan semua orang untuk mengenal lebih dekat tentang dunia mikrokontroler.



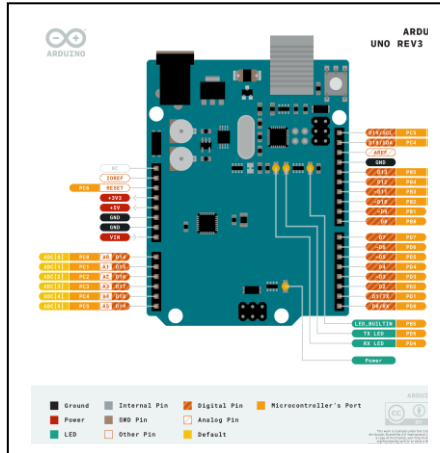
Gambar 2. 1 Arduino Uno R3

<https://store.arduino.cc/usa/arduino-uno-smd-rev3>

Arduino Uno R3 memiliki spesifikasi sebagai berikut :

Tabel 2.1 Spesifikasi ATmega328

Mikrokontroler	ATmega328
Operasi tegangan	5Volt
Input tegangan	disarankan 7-12Volt
Input tegangan batas	6-20Volt
Pin I/O digital	14 (6 bisa untuk PWM)
Pin Analog	6
Arus DC tiap pin I/O	40mA
Arus DC ketika 3.3V	50mA
Memori flash	32 KB (ATmega328) dan 0,5 KB digunakan oleh ootloader
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Kecepatan clock	16 MHz



Gambar 2. 2 Diagram Pinout Arduino Uno R3
<https://store.arduino.cc/usa/arduino-uno-smd-rev3>

menurut (Firmansyah and Bagaskara 2018) Masing-masing dari pin digital arduino uno dapat digunakan sebagai masukan atau keluaran menggunakan fungsi `pinMode()`, `digitalWrite()` dan `digitalRead()`. Setiap pin beroperasi pada tegangan 5 volt. Setiap pin mampu menerima atau menghasilkan arus maksimum sebesar 40 mA dan memiliki 10 resistor pull-up internal (diputus secara default) sebesar 20-30 K Ω . Sebagai tambahan, beberapa pin masukan digital memiliki kegunaan khusus yaitu :

1. Komunikasi serial: pin 0 (RX) dan pin 1 (TX), digunakan untuk menerima(RX) dan mengirim(TX) data secara serial.
2. External Interrupt: pin 2 dan pin 3, pin ini dapat dikonfigurasi untuk memicu sebuah interrupt pada nilai rendah, sisi naik atau turun, atau pada saat terjadi perubahan nilai.
3. Pulse-width modulation (PWM): pin 3,5,6,9,10 dan 11, menyediakan keluaran PWM 8-bit dengan menggunakan fungsi `analogWrite()`.
4. Serial Peripheral Interface (SPI): pin 10 (SS), 11 (MOSI), 12 (MISO) dan 13 (SCK), pin ini mendukung komunikasi SPI dengan menggunakan SPI library.
5. LED: pin 13, terdapat built-in LED yang terhubung ke pin digital 13. Ketika pin bernilai HIGH maka LED menyala, sebaliknya ketika pin bernilai LOW maka LED akan padam. Arduino Uno memiliki 6 masukan analog yang diberi label A0 sampai A5, setiap pin menyediakan resolusi sebanyak 10 bit (1024 nilai yang berbeda). Secara default pin mengukur

nilai tegangan dari ground (0V) hingga 5V, walaupun begitu dimungkinkan untuk mengganti nilai batas atas dengan menggunakan pin AREF dan fungsi *analogReference()*. Sebagai tambahan beberapa pin masukan analog memiliki fungsi khusus yaitu pin A4 (SDA) dan pin A5 (SCL) yang digunakan untuk komunikasi Two Wire Interface (TWI) atau Inter Integrated Circuit (I2C) dengan menggunakan Wire library.

2.2.2 ESP32 CAM

Menurut (Firmansyah and Bagaskara 2018) ESP32 merupakan suksesor atau penerus dari ESP8266 yang memiliki banyak fitur tambahan dan keunggulan dibandingkan generasi sebelumnya. Pada ESP32 terdapat inti CPU serta Wi-Fi yang lebih cepat, GPIO yang lebih banyak, dan dukungan terhadap Bluetooth 4.2, serta konsumsi daya yang rendah. Berikut perbandingan antara mikrokontroler ESP32 dan ESP8266.

Menurut (Saleha 2020) ESP32-CAM merupakan mikrokontroler yang dapat diprogram dengan built-in. WiFi dan *Bluetooth*, dengan tambahan 4MB RAM eksternal. ESP32-CAM memiliki modul camera ukuran kecil yang sangat kompetitif yang dapat beroperasi secara *independent*. ESP32-CAM dapat digunakan secara luas di berbagai aplikasi IoT. sangat cocok untuk *home smart devices*, *industrial wireless control*, *wireless monitoring*, *QR wireless identification*, *wireless positioning system signals* dan aplikasi IoT lainnya[12]. ESP32-CAM mengadopsi *DIP package* dan dapat langsung dimasukkan ke dalam *backplane* untuk mewujudkan produk yang cepat, mode koneksi dengan keandalan tinggi.



Gambar 2. 3 ESP32 Cam

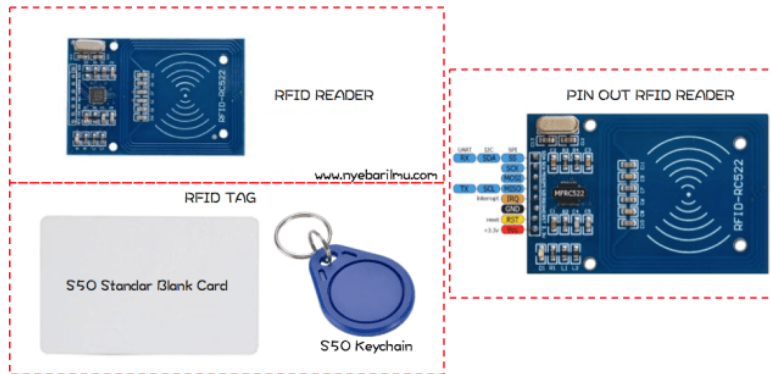
Tabel 2. 2 Spesifikasi ESP32 Cam

Mikrokontroler	Spesifikasi
Operating Voltage Camera	5V, 180 mA
Camera	OV2640 dan OV7670
Image Output Format	2MP JPEG, BMP, Grayscale
Buttons	1
GPIO	8
SPI Flash	Default 32 Mbit
RAM	Internal 520KB + Eksternal 4M PSRAM
WiFi	802.11 b/g/n/e/i
Bluetooth	4.2 BR/EDR dan BLE
Support Interface	UART, SPI, I2C, PWM
Support TF card	4G
UART Baudrate	115200 bps
Dimensi	27 * 40,5 * 4,5 (\pm 0,2) mm
Security	WPA / WPA2 / WPA2-Enterprise / WPS 2.2.4

2.3 RFID

Menurut (Kusumah and Pradana 2019) MFRC522 adalah IC *reader/writer* yang sangat terintegrasi untuk komunikasi *contactless* di 13.56 MHz. Transmitter internal pada MFRC522 mampu mengendalikan antena *reader/writer* yang dirancang untuk berkomunikasi dengan kartu ISO/IEC 14443 A/MIFARE dan *transponder* tanpa sirkuit aktif tambahan [8]. Modul RFID ini bekerja pada tegangan 3.3 volt dan berkomunikasi dengan mikrokontroler melalui protokol interface SPI yang dapat menangani kecepatan data hingga 10 Mbit/s. Untuk kartu atau tag RFID

umumnya bernama PICC yang merupakan singkatan *Proximity Integrated Circuit Tag*. Kartu ini dapat menyimpan data hingga 1K byte dan memiliki *range* 1 cm atau 2cm.



Gambar 2. 4 RFID

2.4 LCD 16x2

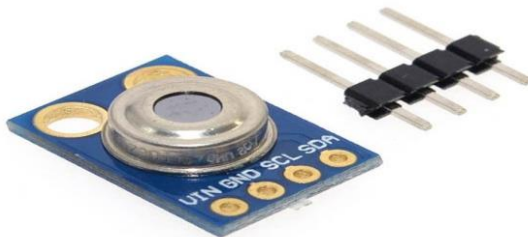
Menurut (Budiyanto n.d.) LCD (Liquid Cristal Display) adalah salah satu jenis display elektronik yang dibuat dengan teknologi CMOS logic yang bekerja dengan tidak menghasilkan cahaya tetapi memantulkan cahaya yang ada di sekelilingnya terhadap front-lit atau mentransmisikan cahaya dari back-lit. LCD berfungsi sebagai penampil data baik dalam bentuk karakter, huruf, angka ataupun grafik. LCD adalah lapisan dari campuran organik antara lapisan kaca bening dengan elektroda transparan indium oksida dalam bentuk tampilan seven-segment dan lapisan elektroda pada kaca belakang. Ketika elektroda diaktifkan dengan medan listrik (tegangan), molekul organik yang panjang dan silindris menyesuaikan diri dengan elektroda dari segmen. Lapisan sandwich memiliki polarizer cahaya vertikal depan dan polarizer cahaya horisontal belakang yang diikuti dengan lapisan reflektor. Cahaya yang dipantulkan tidak dapat melewati molekul-molekul yang telah menyesuaikan diri dan segmen yang diaktifkan terlihat menjadi gelap dan membentuk karakter data yang ingin ditampilkan.



Gambar 2. 5 LCD 16x2

2.5 Sensor MLX90614

Menurut Sensor MLX90614 merupakan sensor yang digunakan untuk mengukur suhu dengan memanfaatkan radiasi gelombang inframerah. Sensor MLX90614 didesain khusus untuk mendeteksi energi radiasi inframerah dan secara otomatis telah didesain sehingga dapat mengkalibrasikan energi radiasi inframerah menjadi skala temperatur. MLX90614 terdiri dari detektor *thermopile* inframerah MLX81101 dan signal conditioning ASSP MLX90302 yang digunakan untuk memproses keluaran dari sensor inframerah. Sensor MLX90614 merupakan sensor suhu *contactless*, artinya untuk mengukur temperatur, sensor ini tidak perlu bersentuhan langsung dengan objek tersebut.



Gambar 2. 6 Sensor MLX90614

2.6 Buzzer

Buzzer merupakan keluarga dari transduser yang sistem kerjanya mengubah sinyal listrik menjadi getaran suara. Transduser sering juga disebut sebagai beeper.



Gambar 2. 7 Buzzer

2.7 Program Arduino IDE

Menurut (Arifin, Zulita, and Hermawansyah 2016) Arduino diciptakan untuk para pemula bahkan yang tidak memiliki basic bahasa pemrograman sama sekali karena menggunakan bahasa C++ yang telah dipermudah melalui library. Arduino menggunakan *Software Processing* yang digunakan untuk menulis program kedalam Arduino. *Processing* sendiri merupakan penggabungan antara bahasa C++ dan Java. *Software* Arduino ini dapat di-install di berbagai *operating system* (OS) seperti LINUX, Mac OS, Windows. Arduino tidak hanya sekedar sebuah alat pengembangan, tetapi kombinasi dari *hardware*, bahasa pemrograman dan *Integrated Development Environment* (IDE) yang canggih. IDE adalah sebuah *software* yang sangat berperan untuk menulis program, meng-compile menjadi kode biner dan meng-upload ke dalam memory mikrokontroler. *Software* IDE Arduino terdiri dari 3 (tiga) bagian:

- a) *Editor* program, untuk menulis dan mengedit program dalam bahasa *processing*. Listing program pada Arduino disebut *sketch*.
- b) *Compiler*, modul yang berfungsi mengubah bahasa *processing* (kode program) kedalam kode biner karena kode biner adalah satu-satunya bahasa program yang dipahami oleh *mikrocontroller*.

c) *Uploader*, modul yang berfungsi memasukkan kode biner kedalam memori mikrokontroller.



Gambar 2. 8 Logo Arduino IDE

2.8 Website

Menurut (Trimarsiah and Arafat 2017) an webadalah dokumen yang ditulis dalam format HTML (*Hyper Text Markup Language*), yang hampir selalu bisa diakses melalui HTTP, yaitu protokol yang menyampaikan informasi dari server website untuk ditampilkan kepada para pemakai melalui web browser. Semua publikasi dari website-website tersebut dapat membentuk sebuah jaringan informasi yang sangat besar. Halaman-halaman dari *website* akan bisa diakses melalui sebuah URL yang biasa disebut *Homepage*. URL ini mengatur halaman-halaman situs untuk menjadi sebuah hirarki, meskipun, *hyperlink-hyperlink* yang ada di halaman tersebut mengatur para pembaca dan memberitahu mereka susunan keseluruhan dan bagaimana arus informasi ini berjalan. Beberapa *website* membutuhkan subskripsi (data masukan) agar para user bisa mengakses sebagian atau keseluruhan isi *website* tersebut (Javacreativity, 2014).

2.9 DataBase

Menurut (yolan dan mansuri 2015) Basis Data adalah suatu susunan / kumpulan data operasional lengkap dari suatu organisasi / perusahaan yang dikelola dan disimpan secara terintegritas dengan

menggunakan metode tertentu menggunakan komputer sehingga mampu menyediakan informasi optimal yang diperlukan pemakainya.

2.10 MySQL

Menurut (yolan dan mansuri 2015) MYSQL merupakan salah satu jenis program database, yaitu sebuah program yang berfungsi untuk mengolah, menyimpan data dan memanipulasi data di server.



Gambar 2. 9 MySQL

2.11 XAMPP

Menurut (yolan dan mansuri 2015) XAMPP adalah sebuah aplikasi web server instan dan lengkap dikarenakan segala yang anda butuhkan untuk membuat sebuah situs web dengan Content Management System (Joomla) bisa dicoba di dalam aplikasi ini. XAMPP adalah sebuah paket installer AMP (Apache, MySQL, dan Php) yang sangat mudah untuk diaplikasikan dalam komputer anda yang belum memiliki server untuk dapat melihat situs yang anda buat menggunakan bahasa server dan *database server* tersebut.



Gambar 2. 10 XAMPP

2.12 Google Sketup

Menurut (KRISDIANTO 2018) SketchUp merupakan sebuah program grafis yang diproduksi oleh google. Program ini memberikan hasil utama yang berupa gambar sketsa grafik 3 dimensi. Sesuai namanya perangkat ini lebih luwes untuk digunakan dalam pra-desain karena memang dimaksudkan untuk membuat objek 3 dimensi dengan perbandingan panjang, lebar maupun tinggi tanpa ukuran yang pasti. Pengeditanya lebih mudah dibanding bila menggunakan perangkat lunak grafis lain dapat menghasilkan gambar yang cukup baik untuk keperluan presentasi sehingga dapat membantu siswa memikirkan informasi baru berupa visualisasi gambar 3D.



Gambar 2. 11 Sketchup

2.13 Bahasa Pemrograman

Menurut (Utara, Wirastuti, and Setiawan 2020) Pengertian luas yang meliputi jumlah kegiatan-kegiatan yang ada pada program termasuk pemrograman, sering juga diistilahkan dengan pmograman computer bahasa, adalah standar intruksi untuk memerintahkan komputer. bahasa pemrograman ini memiliki suatu himpunan dari aturan sintaks dan yang di pakai semantik untuk mendefinisikan program komputer di bawah ini adalah aturan sintak dalam bahasa C. untuk format dalam penulisan.

1. `//` (komentar satu baris) fungsinya untuk member catatan dengan arti tulisan dari kode-kode di belakang Syntax `//` yang\$diabaikan oleh program.
2. `/* */` (komentar banyak baris) jika terdapat banyak catatan syntax komentar semua dapat di buatkan baris yang di abaikan oleh program adalah hal yang terletak di sebuah symbol.
3. `{ }` (kurung kurawal) dipakai untuk mendefinisikan kapan blok program mulai dan kapan berakhir. Kurung kurawal dapat digunakan fungsi dan pengulanganya.

2.14 EAGLE

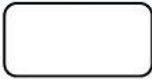

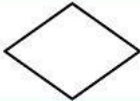

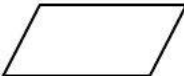
Menurut (Goeritno et al. 2018) Aplikasi EAGLE merupakan salah satu dari sekian banyak aplikasi untuk rancangan dalam pembuatan keping papan tercetak (printed circuit board, PCB) berbantuan komputer (computer aided design, CAD), sehingga dikatakan sebagai editor tata letak yang mudah digunakan bantuan rancangan papan rangkaian tercetak (PCB). Aplikasi EAGLE terdiri atas tiga modul utama, yaitu Editor Tata Letak, Editor Schematic, dan Autorouter yang disematkan dalam satu antarmuka pengguna. Ketiganya dapat diakses dari EAGLE's User Interface (UI) atau antarmuka.

pengguna pada EAGLE, sehingga tidak perlu pengubahan netlists antara skema dan tata letak. Fitur utama aplikasi dicakup juga anotasi maju dan mundur online, fungsi bantuan sensitif untuk konteks, beberapa jendela untuk papan, skematik dan kepustakaan, bahasa pengguna yang powerful, editor teks terpadu, dan tidak terdapat perlindungan salinan perangkat keras.

2.15 Flowchart

Menurut (Santoso and Nurmalina 2017) Flowchart adalah representasi secara simbolik dari suatu algoritma atau prosedur untuk menyelesaikan suatu masalah, dengan menggunakan flowchart akan memudahkan pengguna melakukan pengecekan bagian-bagian yang terlupakan dalam analisis masalah, disamping itu flowchart juga berguna sebagai fasilitas untuk berkomunikasi antara pemrogram yang bekerja dalam tim suatu proyek.

Simbol-simbol Flowchart (Diagram Alir)

	Mulai (Awal) / Akhir [Start / End]
	Proses / Kegiatan [Process / Activity]
	Kondisional / Keputusan [Conditional / Decision]
	Arah Aliran [Flow Direction]
	Masukan / Keluaran [Input / Output]

Gambar 2. 12 Flowchart

HALAMAN INI SENGAJA DIKOSONGKAN

BAB III

METODOLOGI PENELITIAN

3.1 Metode Pengumpulan Data

Metode pengumpulan data merupakan sebuah teknik yang digunakan oleh peneliti untuk mengumpulkan data-data yang valid. Maka harus mengetahui bagaimana cara pengumpulan data di dalam penelitian, sehingga data yang diperoleh bisa mendukung konsep dari rancangan yang akan dibuat.

3.1.1 Metode Pengumpulan Data Pengujian Lab

Pada metode pengumpulan data menggunakan pengujian lab, pengujian dilakukan menggunakan sensor MLX90614 dan juga thermometer digital yang dilakukan pengujian sebanyak 10 kali, suhu diukur menggunakan telapak tangan dengan bantuan relawan yang akan diuji suhunya atas nama :

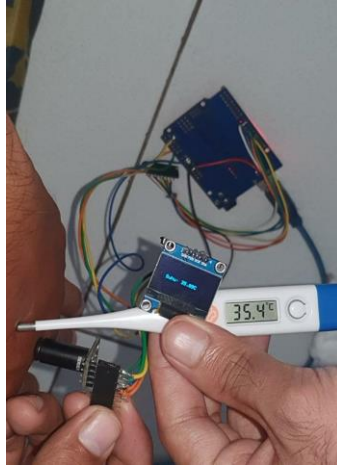
4. Gede Agus Eka Dharma
5. Rahimin
6. Marzuki Uzman
7. Noval Dwi Arianto
- 8.

Tabel 3. 1 Pengujian Sensor MLX90614

No	Nama Relawan	Sensor MLX90614 (°C)	Thermometer Digital (°C)	Selisih Suhu (°C)
1	Uzman	35,1	35,2	0,1
2	Uzman	34,9	35,0	0,1
3	Uzman	35,1	35,4	0,3
4	Rahimin	35,6	35,8	0,2
5	Rahimin	34,1	34,4	0,3
6	Rahimin	35,1	35,3	0,2
7	Agus	34,9	35,1	0,2
8	Agus	34,5	34,7	0,2
9	Noval	35,0	35,4	0,4
10	Noval	35,2	35,5	0,3
Rata-Rata Nilai Pengujian				0,23

1. Dokumentasi

Metode ini dilakukan untuk mendokumentasikan pengujian suhu pada sensor MLX90614.

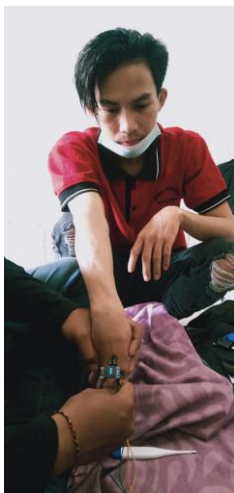


Gambar 3. 1 Dokumentasi Pada Saat Pengecekan Suhu

Pada gambar 3.1 menunjukkan gambar pada saat melakukan pengecekan suhu pada tangan menggunakan sensor MLX90614 dan termometer digital pada saat bersamaan.



Gambar 3. 2 Dokumentasi Pengecekan Suhu Pada Noval



Gambar 3. 3 Pengecekan Pengecekan Suhu Pada Uzman



Gambar 3. 4 Pengecekan Pengecekan Suhu Pada Rahimin



Gambar 3. 5 Pengecekan Pengecekan Suhu Pada Agus

3.2 Analisis Kebutuhan Sistem

Maka disini diperlukan sebuah alat untuk mendeteksi suhu badan dengan akurat serta mengetahui identitas diri dengan mudah melalui KTP dan hasil foto yang ditangkap sesuai dengan wajah pemilik KTP atau bukan secara otomatis tanpa diperlukan pengawasan untuk memantau orang-orang yang memasuki kawasan STMIK STIKOM Indonesia. Adapun terdapat analisis kebutuhan sistem, dalam analisis kebutuhan sistem Memiliki proses identifikasi kebutuhan yang diperlukan dalam menciptakan sebuah sistem baru dan melakukan evaluasi terhadap permasalahan-permasalahan yang diperlukan. dalam menciptakan sistem ini diperlukan beberapa perangkat-perangkat keras dalam pengimplementasiannya. Berikut merupakan beberapa perangkat keras yang dibutuhkan dalam penerapan ini meliputi:

Tabel 3. 2 Kebutuhan Perangkat keras

Alat Dan Bahan	Kegunaan
Arduino Uno R3	Sebagai Pengendali dan Pengolala daya

ESP32Cam	Untuk Mengambil Gambar dan Mengirim Data ke <i>Website</i>
RFID-RC522	Untuk membaca kartu identitas KTP
Sensor MLX90614	Untuk membaca suhu tubuh
LCD 16x2	Untuk menampilkan pemberitahuan notifikasi hasil dari pengecekan suhu dan data KTP
Adaptor 5V 2A	Sebagai catu daya alat yang digunakan
Kabel jumper	Untuk menghubungkan komponen
Box Akrilik	Sebagai tempat Rangkaian yang sudah Dirakit

Spesifikasi alat ini adalah sebagai berikut ;

- 1 Sumber tegangan 5V yang digunakan untuk Arduino Uno.
- 2 Menggunakan RFID-RC522 untuk mengidentifikasi identitas pengguna melalui KTP
- 3 Menggunakan sensor MLX90614 untuk membaca keadaan suhu tubuh.
- 4 Menggunakan OLED Display untuk menampilkan Hasil data suhu yang sudah dibaca.
- 5 Menggunakan ESP32Cam untuk menangkap gambar pada pengguna dan mengirimkan data hasil pengecekan suhu tubuh ke website untuk pencatatan data.

Spesifikasi sistem alat yang dibuat oleh sebagai berikut :

sistem ini memiliki 1 buah kamera yang dimana fungsinya untuk mengambil gambar pada saat melakukan scan KTP menggunakan RFID serta memiliki satu sensor untuk mendeteksi suhu badan tanpa harus menyentuh kan tubuh pada sensor tersebut. pada saat

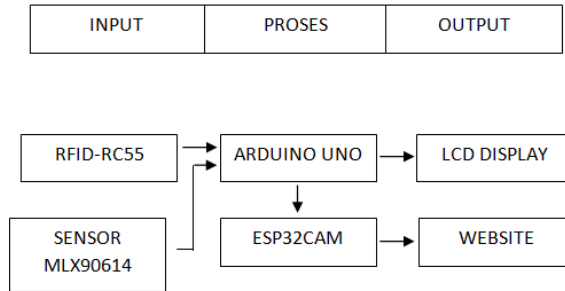
pemrosesan suhu tubuh dan KTP serta pengambilan gambar maka Arduino akan memproses dan anne-marie kan hasil datanya ke LCD serta mengirimkan data kata-kata modul ESP32Cam yang berfungsi untuk mengirimkan data ke dalam website agar data bisa dipantau langsung menggunakan website dan disimpan ke dalam database. selain menggunakan perangkat keras diperlukan juga perangkat lunak termasuk bahasa pemrograman yang fungsinya untuk melakukan implementasi. perangkat lunak yang digunakan adalah *software* Arduino yang berfungsi untuk mengubah nilai Input ke bentuk besaran digital yang diolah oleh mikrokontroler.

Tabel 3. 3 Perangkat Lunak

Alat Dan Bahan	Kegunaan
Arduino IDE	Sebagai pemrograman Arduino dan ESP32Cam
Mysql	Sebagai pemograman database
Server Hosting	Sebagai tempat penyimpanan data yang dikirim dari perangkat keras

3.3 Diagram Blok Sistem

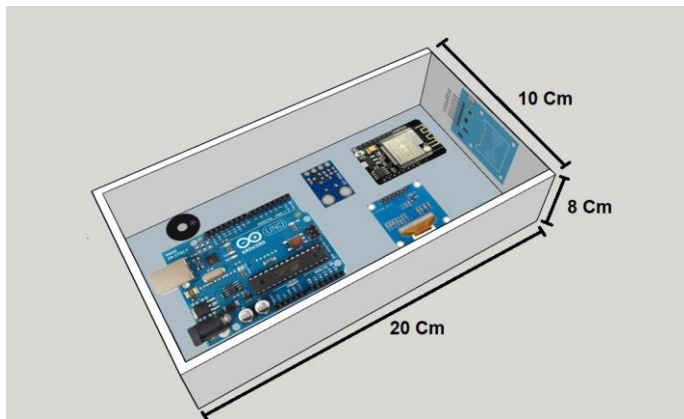
Diagram blok sistem memiliki peranan penting dalam perancangan dan pembuatan alat ini, dalam diagram blok dapat diketahui prinsip kerja dari keseluruhan rangkaian alat tersebut. tujuan lain dari diagram blok adalah untuk mempermudah proses perancangan dan pembuatan an terhadap masing-masing bagian sehingga akan membentuk suatu sistem yang sesuai dengan rancangan sebelumnya.



Gambar 3. 6 Diagram Blok Perancangan Sistem

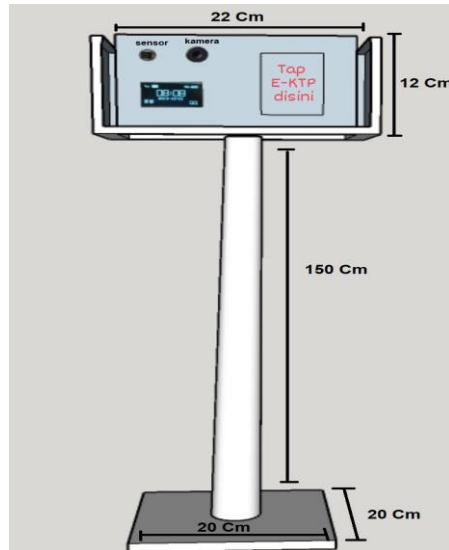
Gambar 3.6 diatas merupakan blok diagram dari rancangan alat pendeteksi suhu tubuh dan pengenalan identitas diri melalui E-KTP dan Foto berbasis IoT dengan rangkain RFID dan sensor suhu sebagai input, kemudian diproses

3.4 Perancangan *Box* Dan Penempatan alat



Gambar 3. 7 Bagian Dalam Box

Pada gambar 3.7 terlihat box untuk penempatan Arduino Uno, module RFID, sensor MLX90614, OLED Display, dan ESP32CAM. Bahan untuk box menggunakan bahan dari akrilik.

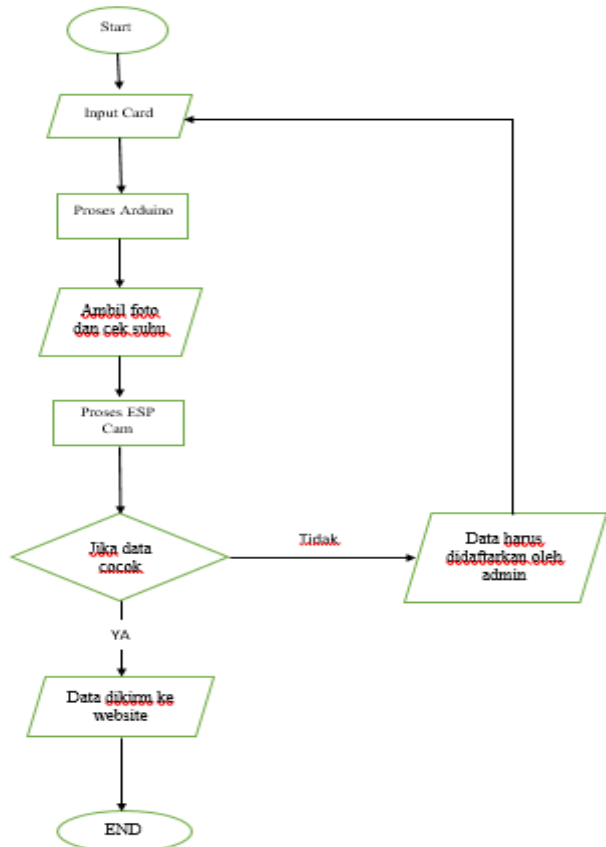


Gambar 3. 8 Bagian Depan Box

Pada gambar 3.8 terlihat bagian depan box yang berisikan lubang, lubang tersebut berfungsi untuk memperlihatkan tampilan dari OLED Display, sensor MLX90614, dan kamera ESP32CAM.

3.5 Flowchart

Pada *flowchart* ini menjelaskan mengenai alur proses sistem agar mudah dimengerti dan mudah dipahami. Pertama *user* melakukan *input data* kartu dengan cara melakukan *tap card* pada alat, kemudian data yang masuk akan diproses pada arduino untuk memberikan perintah mengambil suhu dan gambar. Setelah itu ESP32cam mengirim dan mencocokkan data yang ada di website, jika data tidak cocok maka data harus ditambahkan terlebih dahulu oleh admin dan diarahkan untuk melakukan tap kartu ulang sehingga data baru akan tersimpan di *website*, apabila data cocok akan diteruskan ke *website*.



Gambar 3. 9 Flowchart

3.6 Perancangan Website

Pada perancangan ini penulis akan membuat halaman login, home page, data pengunjung, dan info pengunjung.

3.6.1 Perancangan Halaman Login

Pada gambar 3.10 merupakan tampilan dari halaman login untuk mengakses website. Pada halaman login admin harus memasukkan *username* dan *password* yang sudah didaftarkan agar bisa mengakses website tersebut.



Gambar 3. 10 Halaman Login

3.6.2 Perancangan Home Page

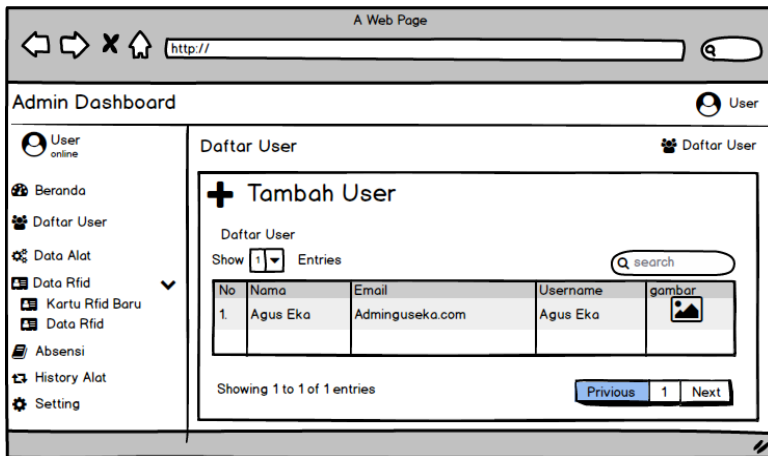


Gambar 3. 11 Halaman Home Page

Pada gambar 3.11 menampilkan halaman home page yang menampilkan *user admin*, beranda, daftar *user*, data alat, data RFID yang berisikan data kartu RFID baru dan data RFID, absensi, *history* alat dan *setting*. Tampilan pada dashboard beranda menampilkan tulisan selamat

datang di BILKESTI, tampilan data RFID yang sudah tersimpan, data alat yang terkoneksi, data pengunjung yang masuk dan keluar.

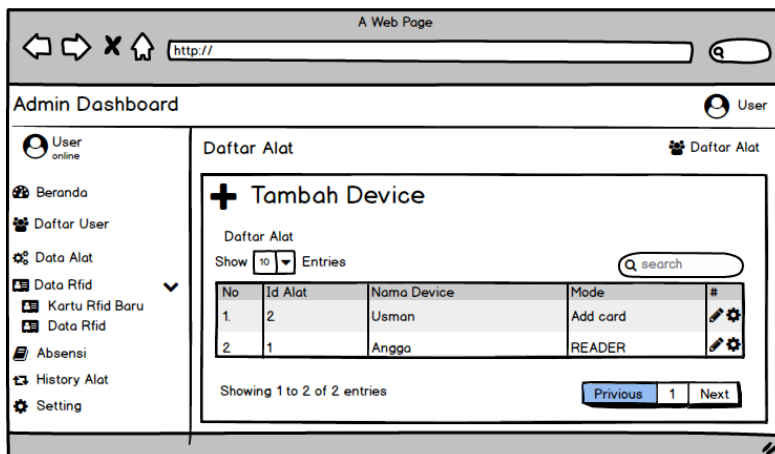
3.6.3 Halaman Daftar User



Gambar 3. 12 Halaman Daftar User

Gambar 3.12 menunjukkan halaman daftar user admin yang sudah terdaftar.

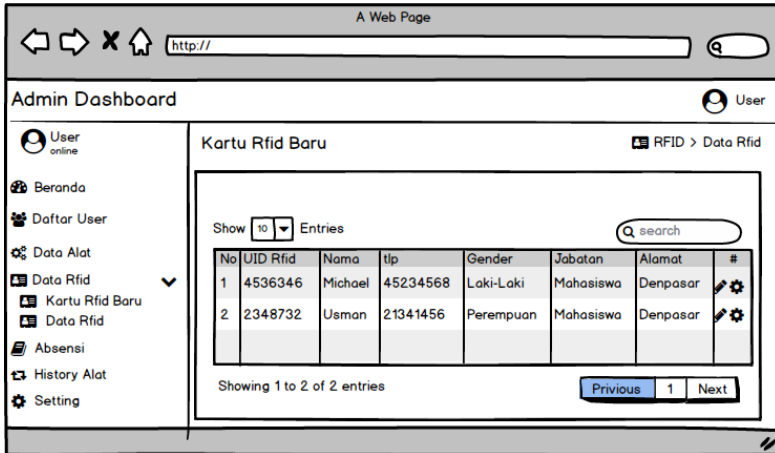
3.6.4 Halaman Data Alat



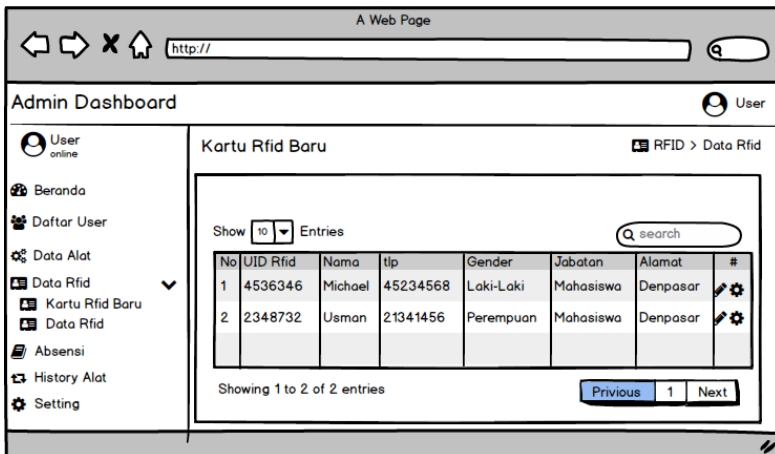
Gambar 3. 13 Halaman Info Pengunjung

Gambar 3.13 menunjukkan halaman data alat card reader RFID yang terhubung pada web.

3.6.5 Halaman Data RFID



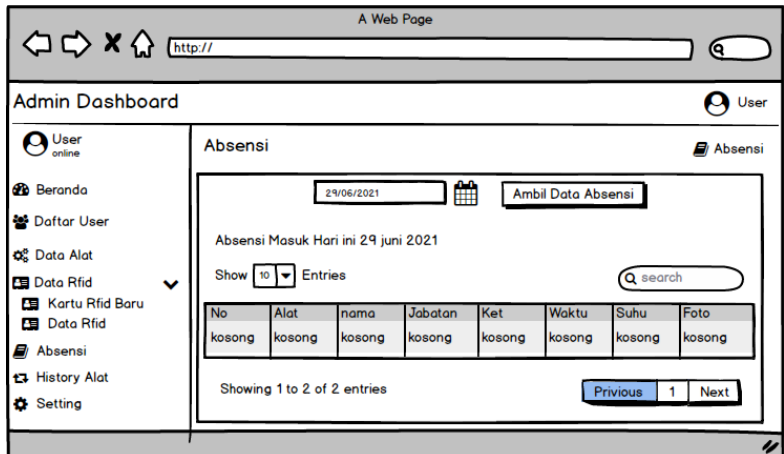
Gambar 3. 14 Halaman Kartu RFID Baru



Gambar 3. 15 Halaman Data RFID

Gambar 3.14 menunjukkan halaman data kartu RFID baru yang akan didaftarkan dan gambar 3.15 menunjukkan halaman data RFID yang sudah didaftarkan.

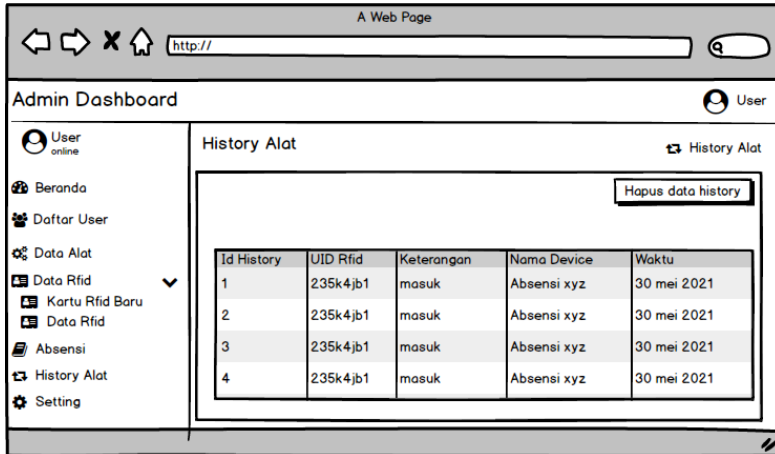
3.6.6 Halaman Absensi



Gambar 3. 16 Halaman Absensi

Gambar 3.16 menunjukkan halaman yang menampilkan daftar pengunjung yang melakukan absensi, serta pencarian data tanggal pengunjung yang melakukan absensi, dan tombol ambil data absensi yang di ubah ke file Microsoft Exel.

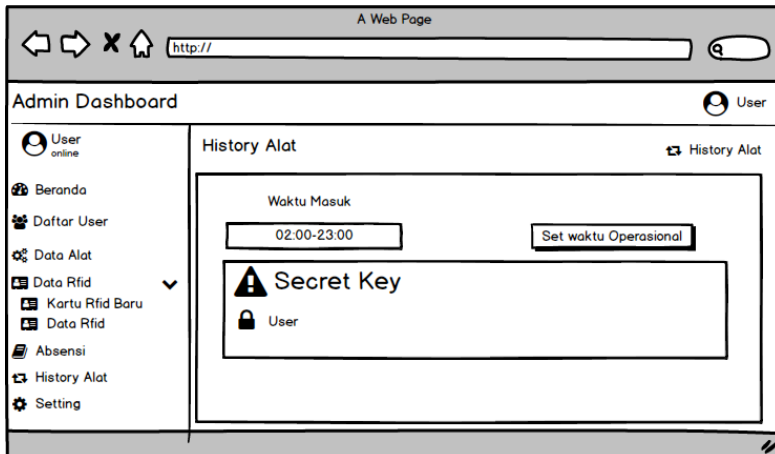
3.6.7 Halaman History Alat



Gambar 3. 17 Halaman *History* Alat

Gambar 3.17 menunjukkan halaman *History* penggunaan alat *card reader* sebagai *add card* atau *reader*.

3.6.8 Halaman Setting

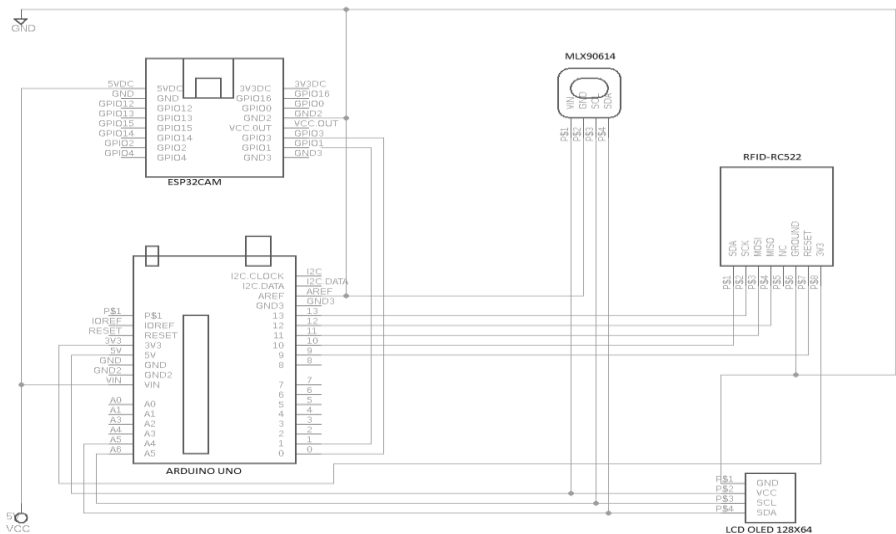


Gambar 3. 18 Halaman Setting

Gambar 3.18 menunjukkan halaman setting yang berfungsi untuk mengatur jadwal operasional pengunjung.

3.7 Skematik Sistem

Skematik sistem adalah sebuah rangkaian yang berfungsi untuk menggambarkan komponen-komponen yang disatukan menjadi sebuah rangkaian. skema ini Ini fungsinya untuk memperjelas rangkain yang terhubung pada komponen-komponen yang sudah dirangkai menjadi satu. berikut ini merupakan gambar rangkaian dasar yang akan dirangkai menjadi sebuah sistem :



Gambar 3. 19 Skematik Rangkaian

Pada gambar 3.19 merupakan sebuah rangkaian sirkuit yang dibangun dalam perancangan alat.

3.8 Skenario Pengujian Sistem

Tujuan pengujian sistem adalah untuk menguji sistem yang sudah dirancang. Pengujian dilakukan untuk menyesuaikan hasil yang diharapkan dari analisa dan perancangan yang telah dibuat.

Tabel 3.4 Skenario Pengujian Sistem

No	Skenarion Pengujian	Hasil yang diharapkan
1	Menguji sensor MLX90614	Sebagai inputan, sensor dapat mendeteksi suhu tubuh tanpa bersentuhan.
2	Menguji sensor RFID	Sebagai inputan, sensor dapat mendeteksi E-KTP sebagai pengenalan identitas.
3	Menguji sensor kamera ESP32CAM	Sebagai inputan, sensor dapat mengambil gambar <i>user</i> .
4	Menguji kinerja sistem keseluruhan	Diharapkan alat dapat melakukan pengambilan suhu badan, data E-KTP, dan gambar secara otomatis.

3.9 Jadwal Penelitian

Jadwal penelitian digunakan untuk mengukur waktu yang diperlukan dalam melakukan penelitian. Berikut merupakan tabel yang berisi jadwal penelitian yang dilakukan.

Tabel 3.5 Jadwal Penelitian

kegiatan	November 2020				Desember 2020			
	1	2	3	4	1	2	3	4
Pengajuan usulan judul profosal seminar								
Bimbingan BAB I								
Melakukan Pengumpulan Data								
Bimbingan BAB I, II, dan III								

BAB IV

IMPLEMENTASI DAN PEMBAHASAN

4. 1 Implmentasi

Implementasi merupakan penerapan sistem alat yang akan dioperasikan. Tahapan ini akan menjelaskan langkah-langkah dalam perakitan rangkaian dan komponen-komponen yang digunakan pada sistem alat pendeteksi suhu tubuh dan pengenalan identitas diri melalui E-KTP dan Foto berbasis IoT. Selanjutnya akan dilakukan proses pengujian pada sistem tersebut.

4. 2 Pemasangan Komponen

Pada langkah ini dilakukan proses pemasangan komponen-komponen dan box alat pendeteksi suhu tubuh dan pengenalan identitas diri melalui E-KTP dan Foto berbasis IoT.

4.2. 1 Tampilan Alat



Gambar 4. 1 Tampilan Depan Box Alat dan Komponen



Gambar 4. 2 Tampilan samping Kiri Box Alat dan Komponen



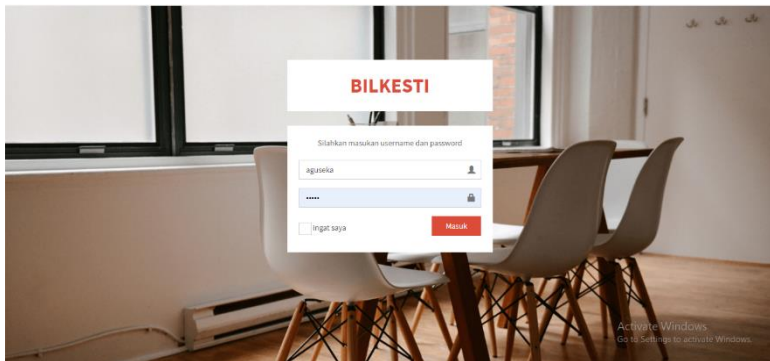
Gambar 4. 3 Tampilan Atas Box Alat Dan Komponen

4. 3 Implementasi Halaman Website

Pada tahap ini menampilkan tampilan *website* yang sudah dirancang sesuai dengan mockup yang sudah dijelaskan pada 3.6 perancangan *website*. Berikut merupakan halaman tampilan keseluruhan *website*.

4.3. 1 Halaman Login

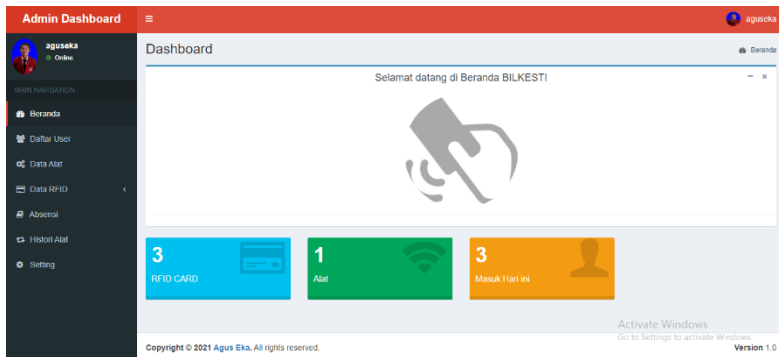
Pada halaman *login* berfungsi untuk admin masuk pada sistem website agar bisa mengontrol website tersebut. Adapun gambar *login* yang dapat dilihat pada gambar berikut.



Gambar 4. 4 Halaman Login

4.3. 2 Halaman Home Page

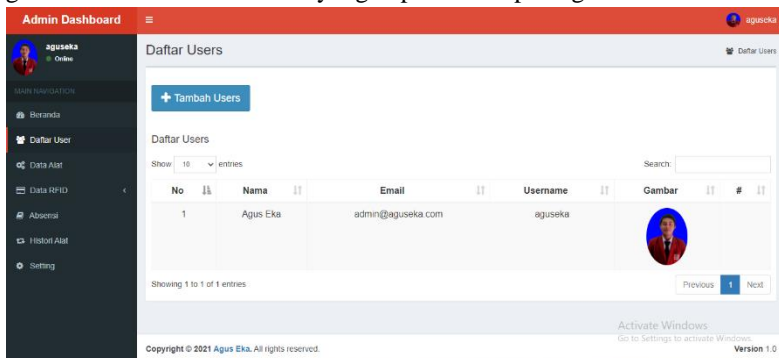
Pada halaman *home page* yang admin dapat mengakses beranda, daftar *user*, data alat, data RFID yang berisikan data kartu RFID baru dan data RFID, abesensi, *history* alat dan *setting*. Adapun gambar Halaman *home page* yang dapat dilihat pada gambar berikut.



Gambar 4. 5 Halaman Home Page

4.3. 3 Halaman Daftar User

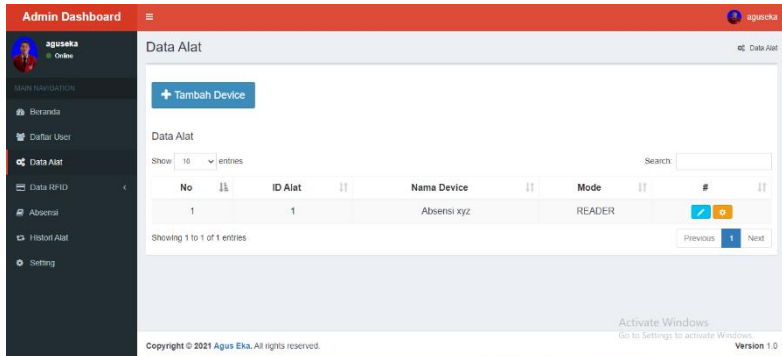
Pada halaman daftar *user* menampilkan *user* admin yang terdaftar pada *website* dan tombol untuk menambah *user* admin. Adapun gambar Halaman daftar *user* yang dapat dilihat pada gambar berikut.



Gambar 4. 6 Halaman Daftar User

4.3. 4 Halaman Data Alat

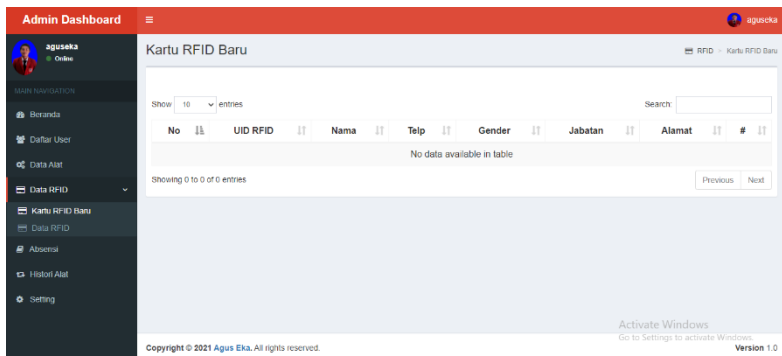
Pada halaman data alat admin bisa menambah *device* dan mengubah *mode add user* dan *mode reader*. Adapun gambar Halaman data alat yang dapat dilihat pada gambar berikut.



Gambar 4. 7 Halaman Data Alat

4.3. 5 Halaman Data RFID

Pada halaman ini admin akan bisa melihat data RFID baru yang masuk serta melakukan input data *user* dan dapat melihat data *user* yang sudah terdaftar. Adapun gambar Halaman kartu RFID baru dan data RFID yang dapat dilihat pada gambar berikut.



Gambar 4. 8 Halaman Kartu RFID

No	UID RFID	Nama	Telp	Gender	Jabatan	Alamat	#
1	88043C82	michael gusli ernanda lonteng	081339970513	pria	mahasiswa	Denpasar	
2	88042973	Rahmin	087041349794	pria	mahasiswa	Sumbawa	
3	077BF0D8	Gede Agus Eka Dharma	08970807150	pria	mahasiswa	Denpasar	

Gambar 4. 9 Data RFID

4.3. 6 Halaman Absensi

Pada halaman ini admin bisa melihat hasil *user* yang melakukan absensi dan dapat *mendownload* data absensi dengan cara menekan tombol ambil data absensi. Adapun gambar Halaman absensi yang dapat dilihat pada gambar berikut.

; 2. Alat Absensi xyz (1), Name Rahmin, Jabatan/Kelas mahasiswa, Keterangan masuk, Waktu 13:37:26 - 28 Jun 2021, Suhu 37.05, Foto ; 3. Alat Absensi xyz, Name michael gusli ernanda, Jabatan/Kelas mahasiswa, Keterangan masuk, Waktu 13:40:23 - 28 Jun 2021, Suhu 36.15, Foto . The page also shows a sidebar with navigation options like Beranda, Daftar User, Data Alat, Data RFID, Kartu RFID Rami, Data RFID, Absensi, Histori Alat, and Setting. The footer includes copyright information for Agus Eka, 2021, and a version number 1.0."/>

No	Alat	Nama	Jabatan/Kelas	Keterangan	Waktu	Suhu	Foto
1	Absensi xyz (1)	Gede Agus Eka Dharma	mahasiswa	masuk	13:21:30 - 28 Jun 2021	34.97	
2	Absensi xyz (1)	Rahmin	mahasiswa	masuk	13:37:26 - 28 Jun 2021	37.05	
3	Absensi xyz	michael gusli ernanda	mahasiswa	masuk	13:40:23 - 28 Jun 2021	36.15	

Gambar 4. 10 Halaman absensi

4.3. 7 Halaman Histori alat

Pada halaman ini bisa melihat hasil histori dari keseluruhan data user yang melakukan tap pada alat. Adapun gambar Halaman histori alat yang dapat dilihat pada gambar berikut.

ID Histori	UID RFID	Keterangan	Name Device	Waktu
44	88043C52	masuk	Absensi xyz (1)	28 Jun 2021, 13:40:23
43	88042973	masuk	Absensi xyz (1)	28 Jun 2021, 13:37:28
42	88043C52	masuk	Absensi xyz (1)	28 Jun 2021, 13:34:38
41	88042973	masuk	Absensi xyz (1)	28 Jun 2021, 13:26:11
40	88042973	masuk	Absensi xyz (1)	28 Jun 2021, 13:25:40
39	88042973	masuk	Absensi xyz (1)	28 Jun 2021, 13:22:31
38	88042973	masuk	Absensi xyz (1)	28 Jun 2021, 13:21:47
37	077BF008	masuk	Absensi xyz (1)	28 Jun 2021, 13:21:30
36	077BF008	masuk	Absensi xyz (1)	28 Jun 2021, 13:18:59
35	88042973	masuk	Absensi xyz (1)	28 Jun 2021, 13:17:50

Gambar 4. 11 Halaman Histori Alat

4.3. 8 Halaman Setting

Pada halaman ini admin bisa melakukan setting waktu sesuai jadwal yang diinginkan dengan cara menekan tombol set waktu operasional. Adapun gambar Halaman *setting* yang dapat dilihat pada gambar berikut.

Waktu Masuk
07:00:00

set waktu operasional

SECRET KEY
AgusEka2021

Activate Windows
Go to Settings to activate Windows.

Copyright © 2021 Agus Eka. All rights reserved. Version 1.0

Gambar 4. 12 Halaman Setting.

4. 4 Pengujian Koneksi ESP32Cam

Pada tahapan ini dilakukan proses pengujian koneksi dengan menyambungkan ESP3Cam dengan WiFi yang sudah disetting sebelumnya.

```
COM5

Check ModeRequest Link:http://192.168.43.217:8080/ekaproject/api/getmoderfidcam
ets Jun  8 2016 00:22:57

rst:0x1 (POWERON_RESET),boot:0x13 (SPI_FAST_FLASH_BOOT)
configsip: 0, SPIWP:0xee
clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:0x00,wp_drv:0x00
mode:DIO, clock div:1
load:0x3fff0018,len:4
load:0x3fff001c,len:1216
ho 0 tail 12 room 4
load:0x40078000,len:10944
load:0x40080400,len:6388
entry 0x400806b4
F55j5RFID CAM
MAC Address
3C:61:05:31:AA:58
ESP Board MAC Address: 3C:61:05:31:AA:58
ID DEVICE 1
URL SERVER
http://192.168.43.217:8080/ekaproject
http://192.168.43.217:8080/ekaproject/api/getmoderfidcam
http://192.168.43.217:8080/ekaproject/api/absensirfidcam
http://192.168.43.217:8080/ekaproject/api/addcardrfidcam
ABSENST RFID CAM
Stiki.Gedung5
connect ke gedung5

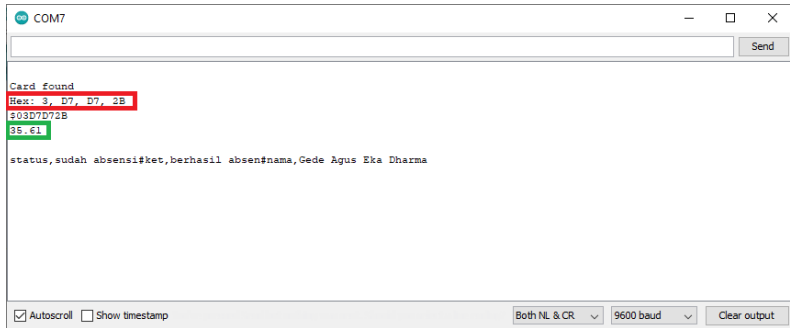
Connecting to Stiki.Gedung5
connect ke gedung5
Connecting to Stiki.Gedung5
...
WiFi connected
IP address:
192.168.43.27
Check ModeRequest Link:http://192.168.43.217:8080/ekaproject/api/getmoderfidcam

☒ Autoscrol ☐ Show timestamp
```

Gambar 4. 13 Proses Koneksi ESP32Cam Dengan Wifi Server

4.4. 1 Tampilan Serial Monitor Pada Saat Mendapatkan Data

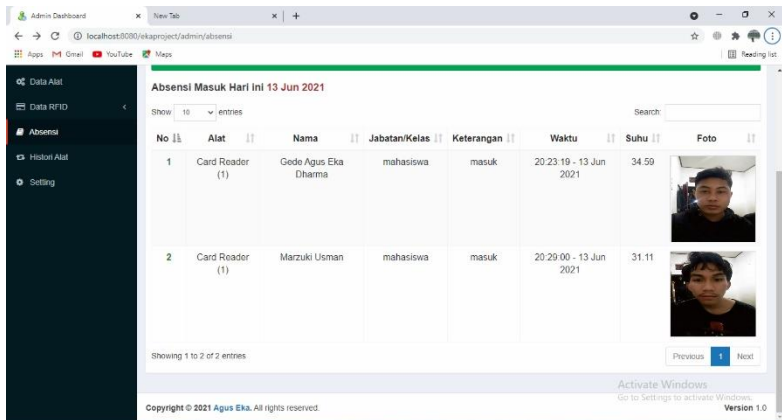
Pada tahap ini ESP32Cam berhasil mendapatkan suhu dan ID E-KTP yang ditampilkan pada *serial monitor* aplikasi Arduino IDE. Pada saat menempelkan kartu E-KTP pada alat pendeteksi suhu tubuh dan pengenalan identitas diri melalui E-KTP dan Foto berbasis IoT. Pada tampilan *serial monitor* kotak berwarna merah menunjukkan RFID *reader* berhasil membaca E-KTP dan kotak berwarna hijau pada *serial monitor* sensor MLX90614 berhasil membaca suhu tubuh.



Gambar 4. 14 Proses Pengambilan Data E-KTP dan Suhu Badan

4.4. 2 Implementasi Pengambilan Gambar Pada ESP32Cam

Pada tahap ini ESP32Cam melakukan pengambilan gambar, saat berhasil dalam pengambilan foto Wajah maka otomatis akan diteruskan pada Web.



Gambar 4. 15 Proses Pengambilan Gambar

4. 5 Pengujian Black Box Website

Pada tahanan ini dilakukan pengujian pada website menggunakan pengujian *black box*.

Tabel 4. 1 Pengujian Website Menggunakan Black Box

Skenario Pengujian	Kasus Pengujian	Hasil Yang Diharapkan	Hasil Pengujian	Kesimpulan
Masuk	Setelah memasukan username dan password kemudian klik ‘tombol’ masuk	Berhasil masuk pada tampilan beranda website	Sesuai	Berhasil
Daftar User	Saat ingin menuju tampilan menu daftar user klik ‘tombol’ daftar user	Berhasil masuk pada tampilan menu daftar user	Sesuai	Berhasil
Tambah User	Ingin menambahkan admin baru pada menu daftar user lalu klik ‘tombol’ tambah user	Berhasil masuk pada tampilan menu tambah user	Sesuai	Berhasil

Simpan	Setelah memasukkan data admin yang baru pada tampilan menu tambah user lalu klik ‘tombol’ simpan	Berhasil menyimpan data admin yang baru	Sesuai	Berhasil
Data Alat	Saat ingin menuju tampilan menu data alat klik ‘tombol’ data alat	Berhasil masuk pada tampilan menu data alat	Sesuai	Berhasil
Tambah Device	Saat ingin menambahkan device baru pada menu data alat lalu klik ‘tombol’ tambah device	Berhasil masuk pada tampilan menu tambah device	Sesuai	Berhasil
Simpan	Setelah memasukkan data nama device yang baru pada tampilan menu tambah device lalu klik ‘tombol’ simpan	Berhasil menyimpan data device baru	Sesuai	Berhasil

Rubah Nama	Saat ingin mengubah nama device pada menu data alat lalu klik 'tombol' rubah nama	Berhasil masuk pada tampilan menu rubah nama	Sesuai	Berhasil
Simpan	Setelah mengubah data nama device pada tampilan menu rubah nama lalu klik 'tombol' simpan	Berhasil menyimpan data device yang diubah	Sesuai	Berhasil
Data RFID	Saat ingin menuju tampilan menu data RFID klik 'tombol' data RFID	Berhasil masuk pada tampilan menu data RFID	Sesuai	Berhasil
Edit	Saat ingin mengubah nama user RFID pada menu data RFID klik 'tombol' edit	Berhasil masuk pada tampilan menu edit data RFID	Sesuai	Berhasil

Simpan	Setelah mengubah data nama user RFID pada tampilan menu edit data RFID lalu klik ‘tombol’ simpan	Berhasil menyimpan data user RFID yang diubah	Sesuai	Berhasil
Hapus	Saat ingin menghapus nama user RFID pada menu data RFID lalu klik ‘tombol’ hapus	Berhasil menghapus user RFID pada tampilan menu data RFID	Sesuai	Berhasil
Absensi	Saat ingin menuju tampilan menu absensi klik ‘tombol’ absensi	Berhasil masuk pada tampilan absensi	Sesuai	Berhasil
Setting	Saat ingin menuju tampilan menu setting klik ‘tombol’ setting	Berhasil masuk pada tampilan absensi	Sesuai	Berhasil

4. 6 Pengujian Suhu Tubuh

Pada tahap ini dilakukan pengujian pada sensor suhu MLX90614 dengan sensor suhu *thermogun* buatan pabrik untuk mendapatkan selisih error pada suhu. Disini penulis melakukan sepuluh kali pengujian pada masing-masing relawan.



Gambar 4. 16 Thermogun Digital Pabrikan

Tabel 4. 2 Pengujian Perbandingan Sensor Suhu MLX90614

No	Nama	Hasil Pengecekan Suhu Menggunakan Sensor MLX90614 (°C)									
1	Marzuki	34.1	33.9	34.1	34.0	34.1	34.2	34.1	34.1	34.3	34.2
2	Rahimin	34.5	34.3	34.4	34.5	34.3	34.4	34.5	34.6	34.5	34.5
3	Angga	34.8	34.6	34.7	34.8	34.8	34.7	34.8	34.8	34.7	34.8
4	Michael	34.8	34.9	34.8	34.9	34.9	34.7	34.8	34.8	34.9	34.9
5	Ratna	34.6	34.7	34.7	34.8	34.7	34.7	34.7	34.6	34.7	34.7
6	Noval	35.2	35.2	35.4	35.3	35.4	35.4	35.3	35.4	35.3	35.3
7	Hendra	35	35.1	35.1	35	35.1	35.2	35.1	35.1	35.2	35.2
8	Arya	34.9	35	34.9	35.1	35	35	35.1	35.1	35.1	35.1
9	Agus	35.3	35.4	35.4	35.4	35.4	35.3	35.4	35.4	35.3	35.4
10	Eka	35.1	35.1	35.2	35.2	35.2	35.2	35.1	35.2	35.2	35.3

Tabel 4. 3 Pengujian Perbandingan Thermogun Pabrikan

No	Nama	Pengujian Perbandingan Thermogun Pabrik (°C)									
1	Marzuki	34.3	34.2	34.2	34.1	34.2	34.2	34.1	34.3	34.4	34.4
2	Rahimin	34.6	34.5	34.6	34.6	34.4	34.6	34.7	34.7	34.7	34.6
3	Angga	34.9	34.8	34.9	35	34.5	34.8	34.9	35	34.9	34.9
4	Michael	35	35.1	35	35.1	35.1	34.8	34.9	35.1	35.1	35.1
5	Ratna	34.9	34.9	34.9	35	34.8	34.8	34.8	34.7	34.9	34.8
6	Noval	35.4	35.4	35.5	35.5	35.6	35.6	35.5	35.5	35.4	35.5
7	Hendra	35.3	35.3	35.3	35.2	35.4	35.4	35.4	35.4	35.5	35.5
8	Arya	35.2	35.3	34.2	35.2	35.2	35.3	35.4	35.4	35.4	35.4
9	Agus	35.5	35.7	35.6	35.6	35.6	35.5	35.6	35.6	35.6	35.6
10	Eka	35.3	35.3	35.5	35.5	35.5	35.4	35.3	35.5	35.4	35.5

Tabel 4. 4 Hasil Perbandingan Pengujian MLX90614 Dengan Thermogun

No	Nama	Total MLX90614 (°C)	Total Thermogun (°C)	Selisih Suhu (°C)
1	Marzuki	34.1	34.2	0.1
2	Rahimin	34.5	34.6	0.1
3	Angga	34.8	34.9	0.1
4	Michael	34.8	35.0	0.2
5	Ratna	34.7	34.9	0.2
6	Noval	35.3	35.5	0.2
7	Hendra	35.1	35.4	0.3
8	Arya	35.0	35.2	0.2
9	Agus	35.4	35.6	0.2
10	Eka	35.2	35.4	0.2
Rata-Rata total Pengujian Suhu				0,2

BAB V

PENUTUP

5.1 Kesimpulan

Dari rumusan masalah yang telah dibuat, penulis mendapatkan sebuah kesimpulan yaitu :

1. Dalam penelitian ini penulis melakukan rancang bangun sistem pendeteksi suhu tubuh dan pengenalan identitas diri melalui E-KTP dan foto berbasis IoT. Adapun Tahapan dalam perancangan yaitu tahapan dalam merancang sistem, tahapan pembuatan alat, tahapan pembuatan aplikasi, dan tahapan pengujian aplikasi.
2. Sistem kerja dari alat pendeteksi suhu tubuh dan pengenalan identitas diri melalui E-KTP dan foto berbasis IoT memiliki langkah-langkah sebagai berikut. Pertama, pastikan *power supply* sudah terhubung dengan catu daya dan tunggu kurang lebih 5 detik sampai buzzer berbunyi beep satu kali dan LCD menampilkan tulisan terkoneksi WiFi. Kedua, pastikan mode *add* atau *read* dari alat yang akan digunakan sudah dipilih sesuai dengan kondisi yang diinginkan, setelah itu tempelkan kartu pada alat untuk melakukan penyimpanan data. Ketiga, admin masuk kedalam website untuk melakukan pengecekan secara berkala.

5.2 Saran

Untuk penelitian selanjutnya pada alat pendeteksi suhu tubuh dan pengenalan identitas diri melalui E-KTP dan Foto berbasis IoT agar berfungsi lebih optimal penulis menyarankan untuk :

1. menggunakan modul kamera dan WiFi secara terpisah untuk mencegah cepatnya kerusakan pada modul yang disebabkan panas.
2. Menggunakan koneksi internet yang stabil agar saat pengiriman data gambar ke Web terkirim dengan cepat dan tidak rusak.

DAFTAR PUSTAKA

- Arifin, Jauhari, Leni Natalia Zulita, and Hermawansyah. 2016. "Perancangan Murottal Otomatis Menggunakan Mikrokontroller Arduino Mega 2560." *Jurnal Media Infotama* 12(1):89–98.
- Bate, Plasidius YM, Anggri Sartika Wiguna, and Danang Aditya Nugraha. 2020. "KURAWAL Jurnal Teknologi, Informasi Dan Industri." 3:81–92.
- Budiyanto, Setiyo. n.d. "Sistem Logger Suhu Dengan Menggunakan Komunikasi Gelombang Radio." 21–27.
- Firmansyah, Rifqi .., and Satria .. Bagaskara. 2018. "Penerapan Modul RF 433 Dalam Pengukuran Intensitas Cahaya Menggunakan Sensor LDR Berbasis Arduino." *INAJEEE: Indonesian Journal of Electrical and Eletronics Engineering* 1(1):1. doi: 10.26740/inajeee.v1n1.p1-6.
- Goeritno, Arief, Dian Anwar, Syaiful Syaiful, Ritzkal Ritzkal, Ahya Ibadina Syahida, and Rezi Setria. 2018. "Implementasi Literasi Terhadap Aplikasi Easily Applicable Graphical Layout Editor Berbasis Pendidikan Islam." *TA'DIBUNA: Jurnal Pendidikan Agama Islam* 1(1):79. doi: 10.30659/jpai.1.1.79-106.
- KRISDIANTO, RENDY. 2018. "Penerapan Media 3D Sketchup Pada Model Pembelajaran Langsung Mata Pelajaran Menggambar Dengan Perangkat Lunak Di Smk Negeri 1 Bendo Magetan." *Jurnal Kajian Pendidikan Teknik Bangunan* 2(2/JKPTB/18).
- Kusumah, Hendra, and Restu Adi Pradana. 2019. "Penerapan Trainer Interfacing Mikrokontroler Dan Internet of Things Berbasis Esp32 Pada Mata Kuliah Interfacing." *Journal CERITA* 5(2):120–34. doi: 10.33050/cerita.v5i2.237.
- Perdana, Adi Wisnu, Mochamad Fajar W, M. Kom, Teknik Komputer, Fakultas Teknik, Universitas Komputer Indonesia, Jl Dipati, and Ukur No. 2019. "Alat Pemantau Kondisi Seorang Gamer Monitoring Condition Device For a Gamer." *Jurnal Teknik Komputer* 1(1):1–5.
- Sabran, and Yasser Abd Djawad. 2018. "Perancangan Modul Pembelajaran Berbasis Projek Pada Matakuliah Dasar Mikrokontroler." *Jurnal MEKOM (Media Komunikasi Pendidikan Kejuruan)* 5(1):23–31.
- Saleha, Relanti. 2020. "Klasifikasi Data Time Series Pola Pergerakan Manusia Di Depan Rumah Menggunakan Sensor Passive Infrared Dan Camera Ov2640 Dengan Metode SVM." 1(1):1–65.
- Santoso, Santoso, and Radna Nurmalina. 2017. "Perencanaan Dan

Pengembangan Aplikasi Absensi Mahasiswa Menggunakan Smart Card Guna Pengembangan Kampus Cerdas (Studi Kasus Politeknik Negeri Tanah Laut).” *Jurnal Integrasi* 9(1):84–91.

Saputra, Dede Irawan, Gian Melky Karmel, and Yuda Bakti Zainal. 2020. “Perancangan Dan Implementasi Rapid Temperature Screening Contactless Dan Jumlah Orang Berbasis IOT Dengan Protokol MQTT.” *Journal Of Energy And Electrical Engineering (JEEE)* 02(01):20–30.

Trimarsiah, Yunita, and Muhajir Arafat. 2017. “Analisis Dan Perancangan Website Sebagai Sarana.” *Jurnal Ilmiah MATRIK* Vol. 19 No:1–10.

Utara, Gede Sastra, N. M. A. E. D. Wirastuti, and Widyadi Setiawan. 2020. “Prototipe Monitoring Suhu Ruangan Dan Detektor Gas Bocor Berbasis Aplikasi Blynk.” *Jurnal SPEKTRUM* 7(2):1–7.

yolan dan mansuri. 2015. “Sistem Informasi Pariwisata Propinsi Nangroe Aceh Darussalam Berbasis Web.” *Jupiter* 1:32–39.

DAFTAR LAMPIRAN

1. Lampiran Koding

Koding Arduino Uno

```
#include <SPI.h>
#include <RFID.h>
#include <SoftwareSerial.h>
#include <Wire.h>
#include <Adafruit_MLX90614.h>
#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27, 16, 2);

Adafruit_MLX90614 mlx = Adafruit_MLX90614();

SoftwareSerial mySerial (2, 3);

#define SS_PIN 10
#define RST_PIN 9
#define Buzzer 4

RFID rfid(SS_PIN, RST_PIN);

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

  lcd.begin();
  lcd.print("ABSENSI RFIDSUHU");

  SPI.begin();
  rfid.init();
  mlx.begin();
```

```
pinMode(Buzzer, OUTPUT);
BEEP(1, 1000, 0);
}

void loop()
{
  String datamasuk = "";
  if (mySerial.available()) {
    datamasuk = mySerial.readStringUntil('%');
    Serial.println(datamasuk);
    String Status = Parsing(datamasuk, '#', 0);
    String Ket = Parsing(datamasuk, '#', 1);

    if (Parsing(Status, ',', 1) == "success") {
      String nama = Parsing(datamasuk, '#', 2);

      lcd.clear();
      lcd.print(nama);
      lcd.setCursor(0, 1);
      lcd.print(Parsing(Ket, ',', 1));
      BEEP(3, 200, 100);
    } else {
      lcd.clear();
      lcd.print("Status ");
      lcd.print(Parsing(Status, ',', 1));

      lcd.setCursor(0, 1);
      lcd.print(Parsing(Ket, ',', 1));
      BEEP(1, 1000, 100);
    }

    delay(5000);
```

```

    lcd.clear();
    lcd.print("ABSENSI RFIDSUHU");
}
if (rfid.isCard()) {
    if (rfid.readCardSerial()) {
        BEEP(2, 200, 100);

        delay(5000);    // delay sebelum pembacaan suhu
        float suhu = 0.0;
        suhu = mlx.readObjectTempC();

        Serial.println(" ");
        Serial.println("Card found");

        Serial.print("Hex: ");
        Serial.print(rfid.serNum[0], HEX);
        Serial.print(", ");
        Serial.print(rfid.serNum[1], HEX);
        Serial.print(", ");
        Serial.print(rfid.serNum[2], HEX);
        Serial.print(", ");
        Serial.print(rfid.serNum[3], HEX);
        Serial.println(" ");

        String rfid0;
        String rfid1;
        String rfid2;
        String rfid3;

        if (rfid.serNum[0] < 0x0f) {
            rfid0 = "0";
            rfid0 += String(rfid.serNum[0], HEX);

```

```
} else {  
    rfid0 = String(rfid.serNum[0], HEX);  
}  
  
if (rfid.serNum[1] < 0x0f) {  
    rfid1 = "0";  
    rfid1 += String(rfid.serNum[1], HEX);  
} else {  
    rfid1 = String(rfid.serNum[1], HEX);  
}  
  
if (rfid.serNum[2] < 0x0f) {  
    rfid2 = "0";  
    rfid2 += String(rfid.serNum[2], HEX);  
} else {  
    rfid2 = String(rfid.serNum[2], HEX);  
}  
  
if (rfid.serNum[3] < 0x0f) {  
    rfid3 = "0";  
    rfid3 += String(rfid.serNum[3], HEX);  
} else {  
    rfid3 = String(rfid.serNum[3], HEX);  
}
```

```
String RFIDcard = "$" + rfid0 + rfid1 + rfid2 + rfid3;  
RFIDcard.toUpperCase();
```

```

    lcd.clear();
    lcd.print(RFIDcard);
    lcd.setCursor(0, 1);
    lcd.print("suhu ");
    lcd.print(suhu);
    Serial.println(RFIDcard);
    Serial.println(suhu);
    mySerial.print(RFIDcard);
    mySerial.print("-");
    mySerial.print(suhu);
    mySerial.print("#");

    delay(2000);
    lcd.clear();
    lcd.print("ABSENSI RFIDSUHU");
  }
}

rfid.halt();
}

void BEEP(byte c, int wait1, int wait2) {
  for (byte b = 0; b < c; b++) {
    digitalWrite(Buzzer, HIGH);
    delay(wait1);
    digitalWrite(Buzzer, LOW);
    delay(wait2);
  }
}

String Parsing(String data, char separator, int index) {

```

```
int found = 0;
int strIndex[] = {0, -1};
int maxIndex = data.length() - 1;
for (int i = 0; i <= maxIndex && found <= index; i++) {
    if (data.charAt(i) == separator || i == maxIndex) {
        found++;
        strIndex[0] = strIndex[1] + 1;
        strIndex[1] = (i == maxIndex) ? i + 1 : i;
    }
}
return found > index ? data.substring(strIndex[0], strIndex[1]) :
"";
}
```


Koding ESP32Cam

```
#include <Arduino.h>
#include <WiFi.h>
#include <HTTPClient.h>
#include "soc/soc.h"
#include "soc/rtc_cntl_reg.h"
#include "esp_camera.h"
#include "RTCLib.h"

RTC_Millis rtc;

String ssidName, pass, SERVER, Key, IDdev;
bool flagDevice = 0;

//Arduino D3 => IO14
#define RXD2 14
//Arduino D2 => IO2
#define TXD2 2
#define btnRst 12
#define btnMode 13
#define btnFlash 4

HardwareSerial hwSerial(1);

String sendThis = "";

String iddev, key, suhu;

String hostMode, hostSCAN, hostAdd, hostDateTime;
String hostModex = "/api/getmoderfidcam";
String hostSCANx = "/api/absensirfidcam";
```

```
String hostAddx = "/api/addcardrfidcam";  
String hostDateTimex = "/api/datetimer";  
String Servers;
```

```
String ModeAlat = "";  
String postData;
```

```
int intervalGetTime = 30000;  
unsigned long GetTime;  
bool updateTime = false;  
String timeFromServer;  
byte reqTime;
```

```
// CAMERA_MODEL_AI_THINKER
```

```
#define PWDN_GPIO_NUM    32  
#define RESET_GPIO_NUM  -1  
#define XCLK_GPIO_NUM    0  
#define SIOD_GPIO_NUM    26  
#define SIOC_GPIO_NUM    27
```

```
#define Y9_GPIO_NUM      35  
#define Y8_GPIO_NUM      34  
#define Y7_GPIO_NUM      39  
#define Y6_GPIO_NUM      36  
#define Y5_GPIO_NUM      21  
#define Y4_GPIO_NUM      19  
#define Y3_GPIO_NUM      18  
#define Y2_GPIO_NUM       5  
#define VSYNC_GPIO_NUM   25  
#define HREF_GPIO_NUM    23  
#define PCLK_GPIO_NUM    22
```

```
void setup() {  
  WRITE_PERI_REG(RTC_CNTL_BROWN_OUT_REG, 0);  
  Serial.begin(115200);  
  hwSerial.begin(9600, SERIAL_8N1, RXD2, TXD2);  
  
  pinMode(btnRst, INPUT_PULLUP);  
  pinMode(btnMode, INPUT_PULLUP);  
  pinMode(btnFlash, OUTPUT);  
  
  delay(1000);  
  
  sendThis = "ABSENSI RFID CAM";  
  Serial.println(sendThis);  
  
  sendThis = "MAC Address";  
  Serial.println(sendThis);  
  
  sendThis = String(WiFi.macAddress());  
  Serial.println(sendThis);  
  
  Serial.print("ESP Board MAC Address: ");  
  Serial.println(WiFi.macAddress());  
  
  flagDevice = 1;  
  iddev = "1";  
  
  if (iddev.toInt() == 0) {  
    sendThis = "ID DEVICE SALAH";  
    Serial.println(sendThis);  
  } else {  
    sendThis = "ID DEVICE ";
```

```
    sendThis += iddev;
    Serial.println(sendThis);
}
delay(1000);

key = "AgusEka2021";

Servers = "http://192.168.43.217:8080/ekaproject";

if (flagDevice) {
    sendThis = "URL SERVER";
    Serial.println(sendThis);
    sendThis = Servers;
    Serial.println(sendThis);

    hostMode = Servers + hostModex;
    hostSCAN = Servers + hostSCANx;
    hostAdd = Servers + hostAddx;
    hostDateTime = Servers + hostDateTimex;

    Serial.println(hostMode);
    Serial.println(hostSCAN);
    Serial.println(hostAdd);

    delay(3000);

    sendThis = "ABSENSI RFID CAM";
    Serial.println(sendThis);
}

if (flagDevice) {
    ClientAP();
```

```
ModeDevice();
```

```
camera_config_t config;  
config.ledc_channel = LEDC_CHANNEL_0;  
config.ledc_timer = LEDC_TIMER_0;  
config.pin_d0 = Y2_GPIO_NUM;  
config.pin_d1 = Y3_GPIO_NUM;  
config.pin_d2 = Y4_GPIO_NUM;  
config.pin_d3 = Y5_GPIO_NUM;  
config.pin_d4 = Y6_GPIO_NUM;  
config.pin_d5 = Y7_GPIO_NUM;  
config.pin_d6 = Y8_GPIO_NUM;  
config.pin_d7 = Y9_GPIO_NUM;  
config.pin_xclk = XCLK_GPIO_NUM;  
config.pin_pclk = PCLK_GPIO_NUM;  
config.pin_vsync = VSYNC_GPIO_NUM;  
config.pin_href = HREF_GPIO_NUM;  
config.pin_sscb_sda = SIOD_GPIO_NUM;  
config.pin_sscb_scl = SIOC_GPIO_NUM;  
config.pin_pwdn = PWDN_GPIO_NUM;  
config.pin_reset = RESET_GPIO_NUM;  
config.xclk_freq_hz = 20000000;  
config.pixel_format = PIXFORMAT_JPEG;
```

```
// init with high specs to pre-allocate larger buffers  
if (psramFound()) {  
    config.frame_size = FRAMESIZE_VGA;  
    config.jpeg_quality = 10; //0-63 lower number means higher  
quality  
    config.fb_count = 2;  
} else {
```

```

    config.frame_size = FRAMESIZE_VGA;
    config.jpeg_quality = 12; //0-63 lower number means higher
quality
    config.fb_count = 1;
}

// camera init
esp_err_t err = esp_camera_init(&config);
if (err != ESP_OK) {
    Serial.printf("Camera init failed with error 0x%x", err);

    delay(5000);
    ESP.restart();
}

sensor_t * s = esp_camera_sensor_get();
s->set_framesize(s, FRAMESIZE_VGA); //
UXGA|SXGA|XGA|SVGA|VGA|CIF|QVGA|HQVGA|QQVGA

rtc.begin(DateTime(F(__DATE__), F(__TIME__)));
getTimeServer();
if (updateTime) {
    DateTime now = rtc.now();

    String HourStr = String(now.hour(), DEC);
    String MinStr = String(now.minute(), DEC);
    if (HourStr.toInt() < 10) {
        HourStr = "0" + HourStr;
    }
    if (MinStr.toInt() < 10) {
        MinStr = "0" + MinStr;
    }
}

```

```

    }

    //String(now.second(), DEC);
} else {
    delay(10);
}
GetTime = millis();
}
}

void loop() {
    //Serial.println(digitalRead(btnMode));
    if (digitalRead(btnMode) == LOW){
        while(digitalRead(btnMode) == LOW){
            Serial.println("button di tekan");
            delay(100);
        }
        ModeDevice();
    }

    bool detectRFID = false;
    String RFIDcard = "";
    String dataMasuk = "";
    if (hwSerial.available() > 0) {
        dataMasuk = hwSerial.readString();

        Serial.println(dataMasuk);

        if (dataMasuk.substring(0, 1) == "$" &&
dataMasuk.substring(15, 16) == "#") {
            dataMasuk.remove(15, 16);

```

```

dataMasuk.remove(0, 1);

RFIDcard = Parsing(dataMasuk, '-', 0);
suhu = Parsing(dataMasuk, '-', 1);

Serial.println(RFIDcard);
Serial.println(suhu);
detectRFID = true;
} else {
    sendThis = "UID RFID ERROR";
    Serial.println(sendThis);
}
}

if (detectRFID) {
    if (ModeAlat == "ADD") {
        HTTPClient http;

        postData = "key=" + key + "&iddev=" + iddev + "&rfid=" +
RFIDcard;

        Serial.print("Request Link:");
        Serial.println(hostAdd);

        http.begin(hostAdd);
        http.addHeader("Content-Type", "application/x-www-form-
urlencoded");

        int httpCode = http.POST(postData);           //Send the
request
        String payload = http.getString();           //Get the response

```


payload from server

```
Serial.print("Response Code:"); //200 is OK
Serial.println(httpCode);    //Print HTTP return code

Serial.print("Returned data from Server:");
Serial.println(payload);    //Print request response payload

if (httpCode == 200)
{
    hwSerial.flush();
    delay(100);
    hwSerial.print(payload);
    hwSerial.print("%");
    String respon = Parsing (payload, '#', 1);

    sendThis = respon;
    Serial.println(sendThis);

} else {
    Serial.println();
    Serial.println("Koneksi Error");
    hwSerial.flush();
    delay(100);
    hwSerial.print("status,Koneksi Error%");
}

delay(5000);

http.end();
postData = "";
} else if (ModeAlat == "SCAN") {
```

```
String getAll;  
String getBody;  
  
int serverPort = 0;  
String serverName = Parsing(hostSCAN, '/', 2);  
serverName = Parsing(serverName, ':', 0);  
String serverPortStr = Parsing(hostSCAN, ':', 2);  
serverPortStr = Parsing(serverPortStr, '/', 0);  
  
String ServerPath = "";  
if (Parsing(Servers, '/', 3) == "") {  
    ServerPath = hostSCANx;  
} else {  
    ServerPath = "/" + Parsing(hostSCAN, '/', 3) + hostSCANx;  
}  
  
if (serverPortStr.toInt() != 0) {  
    serverPort = serverPortStr.toInt();  
} else {  
    serverPort = 80;  
}  
  
digitalWrite(btnFlash, HIGH);  
  
WiFiClient client;  
  
camera_fb_t * fb = NULL;  
fb = esp_camera_fb_get();  
if (!fb) {  
    Serial.println("Camera capture failed");  
  
    delay(5000);  
}
```

```

        ESP.restart();
    }

    if (client.connect(serverName.c_str(), serverPort)) {
        //Serial.println("Connection successful!");
        String head = "--agusekabali\r\nContent-Disposition: form-
data; name=\"suhu\"\\r\\n\\r\\n\" + suhu + "\\r\\n--
agusekabali\r\nContent-Disposition: form-data;
name=\"iddev\"\\r\\n\\r\\n\" + iddev + "\\r\\n--agusekabali\r\nContent-
Disposition: form-data; name=\"key\"\\r\\n\\r\\n\" + key + "\\r\\n--
agusekabali\r\nContent-Disposition: form-data;
name=\"rfid\"\\r\\n\\r\\n\" + RFIDcard + "\\r\\n--
agusekabali\r\nContent-Disposition: form-data; name=\"foto\";
filename=\"esp32-cam.jpg\"\\r\\nContent-Type:
image/jpeg\\r\\n\\r\\n\";
        String tail = "\\r\\n--agusekabali--\\r\\n\";

        //Serial.println(head);

        uint32_t imageLen = fb->len;
        uint32_t extraLen = head.length() + tail.length();
        uint32_t totalLen = imageLen + extraLen;

        client.println("POST " + ServerPath + " HTTP/1.1");
        client.println("Host: " + serverName);
        client.println("Content-Length: " + String(totalLen));
        client.println("Content-Type: multipart/form-data;
boundary=agusekabali");
        client.println();
        client.print(head);
    }

```

```

uint8_t *fbBuf = fb->buf;
size_t fbLen = fb->len;
for (size_t n = 0; n < fbLen; n = n + 1024) {
    if (n + 1024 < fbLen) {
        client.write(fbBuf, 1024);
        fbBuf += 1024;
    }
    else if (fbLen % 1024 > 0) {
        size_t remainder = fbLen % 1024;
        client.write(fbBuf, remainder);
    }
}
client.print(tail);

esp_camera_fb_return(fb);

digitalWrite(btnFlash, LOW);

int timeoutTimer = 10000;
unsigned long startTimer = millis();
boolean state = false;

while ((startTimer + timeoutTimer) > millis()) {
    //Serial.print(".");
    delay(100);
    while (client.available()) {
        char c = client.read();
        if (c == '\n') {
            if (getAll.length() == 0) {
                state = true;
            }
        }
    }
}

```

```

        getAll = "";
    }
    else if (c != '\r') {
        getAll += String(c);
    }
    if (state == true) {
        getBody += String(c);
    }
    startTimer = millis();
}
if (getBody.length() > 0) {
    break;
}
}
//Serial.println();
client.stop();
Serial.println(getBody);
String Status = Parsing(getBody, ',', 1);
if (Status.substring(0,7) == "success") {
    hwSerial.flush();
    delay(100);
    hwSerial.print(getBody);
    hwSerial.print("%");
    String ket = Parsing(getBody, '#', 1);
    String nama = Parsing(getBody, '#', 2);
    Serial.println(nama);
    Serial.println(ket);
} else if (Status.substring(0,13) == "sudah absensi") {
    hwSerial.flush();
    delay(100);
    hwSerial.print(getBody);

```

```

        hwSerial.print("% ");
        String ket = Parsing(getBody, '#', 1);
        String nama = Parsing(getBody, '#', 2);
        Serial.println(nama);
        Serial.println(ket);
    } else {
        Serial.println(Status);
    }
}
else {
    getBody = "Connection to " + serverName + " failed.";
    Serial.println(getBody);
    hwSerial.flush();
    delay(100);
    hwSerial.print("status,Koneksi Error%");
}

delay(5000);
}
}

if (millis() - GetTime > intervalGetTime) {
    reqTime++;
    if (reqTime >= 10) {
        getTimeServer();
        reqTime = 0;
    }
    if (updateTime) {
        DateTime now = rtc.now();

        sendThis = "tanggal ";
        sendThis += String(now.day(), DEC);
    }
}

```

```

sendThis += ":";
sendThis += String(now.month(), DEC);
sendThis += ":";
sendThis += String(now.year(), DEC);
Serial.println(sendThis);
delay(10);

String HourStr = String(now.hour(), DEC);
String MinStr = String(now.minute(), DEC);
if (HourStr.toInt() < 10) {
    HourStr = "0" + HourStr;
}
if (MinStr.toInt() < 10) {
    MinStr = "0" + MinStr;
}
sendThis = "jam ";
sendThis += HourStr;
sendThis += ":";
sendThis += MinStr;
Serial.println(sendThis);
delay(10);
//String(now.second(), DEC);
} else {
    sendThis = "tanggal - ";
    Serial.println(sendThis);
    sendThis = "jam error";
    Serial.println(sendThis);
}
GetTime = millis();
}
}

```

```

void getTimeServer() {
  HTTPClient http;

  http.begin(hostDateTime);

  int httpCode = http.GET();      //Send the request
  String payload = http.getString(); //Get the response payload
  from server

  if (httpCode == 200)
  {
    updateTime = true;
    timeFromServer = payload;
  } else {
    updateTime = false;
  }

  http.end();

  String thn = Parsing(timeFromServer, ',', 0);
  String bln = Parsing(timeFromServer, ',', 1);
  String tg = Parsing(timeFromServer, ',', 2);
  String jm = Parsing(timeFromServer, ',', 3);
  String mn = Parsing(timeFromServer, ',', 4);
  String dt = Parsing(timeFromServer, ',', 5);
  rtc.adjust(DateTime(thn.toInt(), bln.toInt(), tg.toInt(), jm.toInt(),
mn.toInt(), dt.toInt()));
}

void ModeDevice() {
  Serial.print("Check Mode");

```



```
HTTPClient http;

postData = "key=" + key + "&iddev=" + iddev;

Serial.print("Request Link:");
Serial.println(hostMode);

http.begin(hostMode);
http.addHeader("Content-Type", "application/x-www-form-
urlencoded");

int httpCode = http.POST(postData);          //Send the request
String payload = http.getString();  //Get the response payload
from server

Serial.print("Response Code:"); //200 is OK
Serial.println(httpCode);      //Print HTTP return code

Serial.print("Returned data from Server:");
Serial.println(payload);  //Print request response payload

if (httpCode == 200)
{
    String responMode = Parsing (payload, '*', 1);

    sendThis = "Mode: ";
    sendThis += responMode;
    Serial.println(sendThis);

    if (responMode == "SCAN") {
        ModeAlat = "SCAN";
```

```

    } else if (responMode == "ADD") {
        ModeAlat = "ADD";
    } else {
        ModeAlat = "";
    }
} else {
    Serial.println();
    Serial.println("Koneksi Error");
}
delay(5000);

http.end();
postData = "";
}

String Parsing(String data, char separator, int index) {
    int found = 0;
    int strIndex[] = {0, -1};
    int maxIndex = data.length() - 1;
    for (int i = 0; i <= maxIndex && found <= index; i++) {
        if (data.charAt(i) == separator || i == maxIndex) {
            found++;
            strIndex[0] = strIndex[1] + 1;
            strIndex[1] = (i == maxIndex) ? i + 1 : i;
        }
    }
    return found > index ? data.substring(strIndex[0], strIndex[1]) :
    "";
}

```

Koding Koneksi ke WiFi

```
void ClientAP(){

    WiFi.mode(WIFI_STA);
    delay(10);

    ssidName = "Stiki.Gedung5";
    pass = "connect ke gedung5";
    Serial.println(ssidName);
    Serial.println(pass);

    char charssid[ssidName.length()+1];
    ssidName.toCharArray(charssid, ssidName.length()+1);
    char charpass[pass.length()+1];
    pass.toCharArray(charpass, pass.length()+1);

    Serial.println();
    Serial.print("Connecting to ");
    Serial.println(ssidName);
    Serial.println(pass);

    connectToWiFi();

}

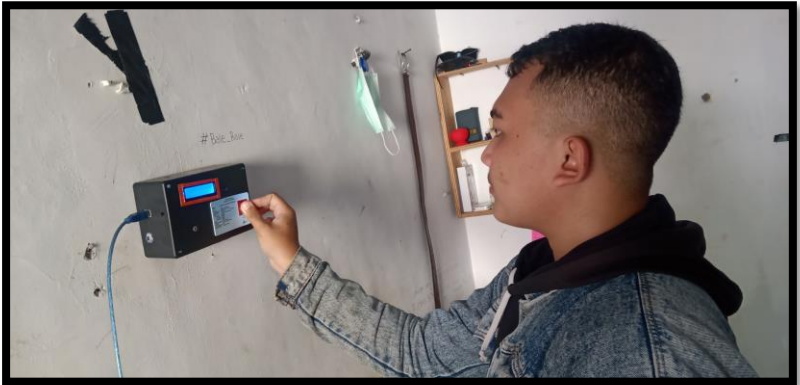
void connectToWiFi(){
    Serial.print("Connecting to ");
    Serial.println(ssidName);
    delay(10);

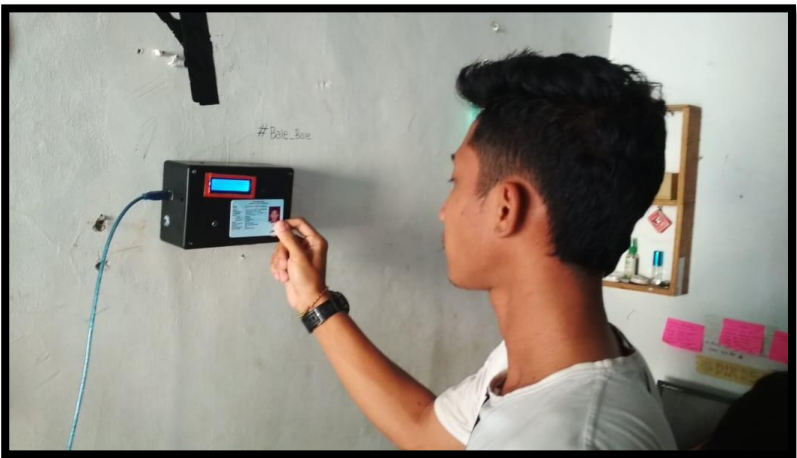
    WiFi.begin(ssidName.c_str(), pass.c_str());
```

2.

```
while (WiFi.status() != WL_CONNECTED) {  
  Serial.print(".");  
  delay(2000);  
}  
  
Serial.println("");  
Serial.println("WiFi connected");  
Serial.println("IP address: ");  
Serial.println(WiFi.localIP()); //You can get IP address  
assigned to ESP  
  
  delay(3000);  
}
```

2. Lampiran Dokumentasi Pengambilan Data











3. Lampiran Laporan Cek Plagiarieme

Page 1

Plagiarism Scan Report



Report Title	Report
Generated Date	29-Jul-2021
Total Words	202
Total Characters	1455
Report Generated By	 Plagiarismchecker.co
Exclude URL	None

Plagiarised	Unique	Total Words Ratio	Spelling Mistake	Grammar Mistake
0	0	100%	139 Error	0 Error

Content Checked For Plagiarism

Salah satu upaya penegakan protocol kesehatan adalah penggunaan alat pemeriksa suhu tubuh pada setiap tempat umum. Ditempat umum pengecekan suhu tubuh menggunakan thermometer gun. Masalah yang di hadapi saat melakukan pengecekan sistem protokol kesehatan di antaranya adalah pendataan suhu dan identitas pengunjung yang memasuki Kampus STMIK STIKOM Indonesia agar riwayat data pengunjung setiap hari dapat dilihat kembali dan tersimpan secara otomatis, Realialisasi pemeriksaan suhu tubuh dapat menggunakan sensor MLX90614 yang memiliki kelebihan dapat membaca suhu tubuh tanpa memerlukan kontak langsung antara tubuh dengan sensor dan untuk memantau pengunjung menggunakan RFID yang menggunakan E-KTP sebagai tag identitas serta ESP32CAM untuk mengambil gambar wajah pengunjung untuk didata dan mengirimkan data ke internet. Tujuan melakukan penelitian ini adalah merancang bangun alat pendeteksi suhu tubuh dan pengenalan identitas diri melalui E-KTP dan Foto berbasis IoT, menjelaskan sistem kerja dan kinerja dari alat pendeteksi suhu tubuh dan pengenalan identitas diri melalui E-KTP dan Foto berbasis IoT. Dari hasil pengujian pada alat pendeteksi suhu tubuh dan pengenalan identitas diri melalui E-KTP dan Foto berbasis IoT, didapatkan hasil bahwa sistem ini mampu mengambil data suhu, ID E-KTP, dan foto Wajah. Total dari perbandingan antara sensor suhu MLX90614 dengan thermogun pabrikan mendapatkan mendapatkan nilai rata-rata sebesar 0,2 (0C)

BIODATA PENULIS



Nama : Gede Agus Eka Dharma
Alamat : Jalan Tukad Batanghari VI B no
5, Denpasar, Bali
TTL : Tamblang, 31 januari 1997
Telepon : 08970907150
Email : gusputtra31@gmail.com
Angkatan : 2017
Moto : Berjuang dulu sebelum menyerah

Latar Belakang Pendidikan :

2003-2009 : SD Negeri 4 Panjer
2009-2012 : SMP Negeri 2 Kubutambahan
2012-2015 : SMK Rekayasa Denpasar
2017-Sekarang : STMIK STIKOM Indonesia

Pengalaman Organisasi :

- Relawan Pencatatan data Pengungsi Korban Bencana Gunung Agung
- Anggota UKM Robotika Periode 2017-2019
- Anggota UKM E-Sport Periode 2018-2019

Pengalaman Kerja :

2015-2016 : Grand Mirah Boutique Hotel
Jabatan : Senior Engineering Staff
2016-2019 : Amaris Hotel Sunset Road
Jabatan : Engineering Staff
2019-November 2019 : Amaris Hotel Teuku Umar
Jabatan : Engineering Staff
2019-2020 : Club Bali Hotel
Jabatan : Engineering Staff
2021-Sekarang : Artotel Sanur Bali Hotel
Jabatan : Engineering Staff