

COVER

RANCANG BANGUN SISTEM INFORMASI ABSENSI

BERBASIS ESP8266

TUGAS AKHIR

*Diajukan Sebagai Salah Satu Persyaratan Untuk Memperoleh Gelar Sarjana Di
Departemen Teknik Elektronika Fakultas Teknik Universitas Negeri Padang*



Oleh :

YUDA PRATAMA

20065066/2020

PROGRAM STUDI PENDIDIKAN TEKNIK ELEKTRONIKA

DEPARTEMEN TEKNIK ELEKTRONIKA

FAKULTAS TEKNIK

UNIVERSITAS NEGERI PADANG

2024


HALAMAN PERSETUJUAN TUGAS AKHIR

RANCANG BANGUN SISTEM INFORMASI ABSENSI BERBASIS ESP8266

Nama : Yuda Pratama
NIM/TM : 20065066/2020
Program Studi : Pendidikan Teknik elektronika
Jurusan : Teknik Elektronika
Fakultas : Teknik

Padang, 19 Agustus 2024

Disetujui Oleh,
Pembimbing,


Ilmiyati Rahmy Jasril, S.Pd., M.Pd.T
NIP. 198810072015042003

Kepala Departemen Teknik Elektronika FT-UNP


Dr. Hendra Hidayat, S.Pd., M.Pd.
NIP. 198703052020121012

HALAMAN PENGESAHAN TUGAS AKHIR

Dinyatakan Lulus Setelah Mempertahankan Tugas Akhir didepan Tim Penguji

Program Studi Pendidikan Teknik Elektronika

Departemen Teknik Elektronika

Fakultas Teknik

Universitas Negeri Padang

Judul:

RANCANG BANGUN SISTEM INFORMASI ABSENSI BERBASIS ESP8266

Nama : Yuda Pratama
NIM/TM : 20065066/2020
Program Studi : Pendidikan Teknik Elektronika
Jurusan : Teknik Elektronika
Fakultas : Teknik




Padang, 19 Agustus 2024

Tim Penguji

Nama

Tanda Tangan

- 1. Ketua Penguji : Delsina Faiza, ST., MT**
- 2. Anggota Penguji : Ilmiyati Rahmy Jasril, S.Pd., M.Pd.T**
- 3. Anggota Penguji : Thamrin, S.Pd., M.T**

1. 
2. 
3. 

SURAT PERNYATAAN TIDAK PLAGIAT

Saya yang bertanda tangan dibawah ini:

Nama : Yuda Pratama
NIM/TM : 20065066/2020
Program Studi : Pendidikan Teknik Elektronika
Jurusan : Teknik Elektronika
Fakultas : Teknik

Dengan ini saya menyatakan bahwa tugas akhir saya dengan judul: **RANCANG BANGUN SISTEM INFORMASI ABSENSI BERBASIS ESP8266**. Merupakan karya saya sendiri. Sepanjang pengetahuan saya tidak terdapat karya atau pendapat orang lain kecuali sebagai acuan atau kutipan dengan mengikuti tata penulisan karya ilmiah yang lazim.

Padang, 19 Agustus 2024

Yang menyatakan,

Yuda Pratama

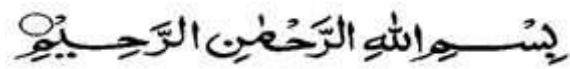
ABSTRAK

Yuda Pratama : Rancang Bangun Sistem Informasi Absensi Berbasis ESP8266

Seiring dengan perkembangan teknologi, manajemen absensi di berbagai institusi menjadi semakin penting untuk meningkatkan efisiensi dan akurasi data. Penelitian ini bertujuan untuk merancang dan membangun sebuah sistem informasi absensi berbasis ESP8266 yang dapat memudahkan proses pencatatan kehadiran secara otomatis dan *real-time*. Sistem ini memanfaatkan modul ESP8266 sebagai komponen utama untuk komunikasi *nirkabel*, yang terhubung dengan *server* melalui jaringan *Wi-Fi*. Proses absensi dilakukan menggunakan RFID (*Radio Frequency Identification*) yang terintegrasi dengan ESP8266, sehingga data kehadiran dapat langsung dikirim dan disimpan dalam *database* secara *real-time*. Metode yang digunakan dalam penelitian ini yaitu metode *Research and Development* (Penelitian dan Pengembangan) dengan desain pengembangan 4D (*Define, Design, Develop and Dessimate*). Sistem ini juga dilengkapi dengan antarmuka pengguna berbasis *web* yang memungkinkan pemantauan dan pengelolaan data absensi dengan mudah dan akurat serta mudah untuk diolah. Hasil dari implementasi sistem ini menunjukkan bahwa sistem informasi absensi berbasis ESP8266 dapat meningkatkan efisiensi pengelolaan absensi, mengurangi kesalahan manual, dan mempermudah akses data absensi dengan cepat dan akurat. Sistem ini diharapkan dapat diterapkan di berbagai lingkungan dalam setiap institusi untuk meningkatkan sistem manajemen kehadiran secara keseluruhan yang baik dan sesuai dengan kebutuhan.

Kata Kunci: Sistem Informasi Absensi, RFID, *Database*, ESP8266, *Internet of Things*.

KATA PENGANTAR



Puji syukur penulis ucapkan kepada Allah Subhanahu Wata'ala yang telah melimpahkan rahmat dan karunia-Nya juga telah memberikan kelancaran dan kemudahan sehingga penulis dapat menyelesaikan tugas akhir yang berjudul **“Rancang Bangun Sistem Informasi Absensi Berbasis ESP8266”** ini dengan baik. Shalawat beserta salam selalu tercurahkan kepada Nabi Muhammad Shallallahu ‘Alaihi Wasallam yang telah membawa risalah kebenaran tauhid kepada umat manusia dengan ilmu pengetahuna yang canggih dan modern seperti yang kita rasakan disaat sekarang ini.

Selama penulisan tugas akhir ini penulis banyak memperoleh bimbingan, saran, motivasi dan bantuan dari berbagai pihak baik secara langsung maupun tidak langsung. Untuk itu, penulis ingin mengucapkan terima kasih kepada:

1. Kedua orang tua yang telah membesarkan, mendidik, memberi perhatian, beserta keluarga besar yang selalu memberikan dorongan dan do'a.
2. Bapak Dr. Ir. Krismadinata, S.T.,M.T selaku Rektor Universitas Negeri Padang.
3. Bapak Dr. Muhammad Anwar, S.Pd, MT selaku Dekan Fakultas Teknik Universitas Negeri Padang.
4. Bapak Dr. Hendra Hidayat, S.Pd., M.Pd., selaku Ketua Departemen Teknik Elektronika Fakultas Teknik UNP, selaku Ketua Program Studi Pendidikan Teknik Elektronika serta selaku dosen penguji yang telah meluangkan waktu dan memberikan arahan dalam pembuatan tugas akhir.

5. Ibu Ilmiyati Rahmy Jasril, S.Pd.,M.Pd.T., selaku Dosen pembimbing akademik serta selaku dosen pembimbing tugas akhir yang telah membimbing dalam perencanaan, pelaksanaan dan pelaporan tugas akhir ini.
6. Ibu Delsina Faiza, S.T, MT selaku Dosen penguji yang telah memberi arahan dan membimbing dalam pembuatan Tugas Akhir.
7. Bapak Thamrin, S.Pd., MT selaku Dosen penguji yang telah memberi arahan dan membimbing dalam pembuatan Tugas Akhir.
8. Seluruh teman PTE20, serta circle Enjoyers dan kita-kita yang telah mensupport penulis dan memberikan motivasi kepada penulis.
9. Teruntuk kakak dan abang tingkat yang selalu support dan bantu informasi serta memberikan semangat dan dukungan sehingga tugas akhir ini dapat terselesaikan.
10. Teruntuk diri sendiri yang sudah sampai ketahap sekarang dan mewujudkan keinginan kedua orang tua serta di usia ke 22 tahun yang akan beranjak ke 23 tahun pada bulan oktober mendatang memberikan hadiah spesial untuk diri sendiri.

Semoga kebaikan yang diberikan oleh semua pihak kepada penulis menjadi amal shaleh yang senantiasa mendapat balasan dan kebaikan yang berlipat ganda dari Allah SWT. Akhir kata, penulis menyadari bahwa masih terdapat kekurangan dalam penulisan Tugas Akhir ini, untuk itu saran dan kritik yang sifatnya membangun sangat penulis harapkan.

Padang, Agustus 2024
Penulis

DAFTAR ISI

COVER	I
HALAMAN PERSETUJUAN TUGAS AKHIR	II
HALAMAN PENGESAHAN TUGAS AKHIR.....	III
SURAT PERNYATAAN TIDAK PLAGIAT.....	IV
ABSTRAK	V
KATA PENGANTAR.....	VI
DAFTAR ISI.....	VIII
DAFTAR TABEL.....	X
DAFTAR GAMBAR.....	XI
DAFTAR LAMPIRAN	XIII
BAB 1 PENDAHULUAN	1
A. Latar Belakang	1
B. Identifikasi Masalah.....	4
C. Batasan Masalah	4
D. Rumusan Masalah.....	4
E. Tujuan Tugas Akhir	5
F. Manfaat Tugas Akhir	5
BAB II LANDASAN TEORI	6
A. Sistem Informasi	6
B. Teknologi <i>Radio Frequency Identification</i> (RFID)	8
C. <i>Database</i>	10
D. <i>My Structured Query Language</i> (MySQL).....	12

E.	Perangkat Keras	13
F.	Perangkat Lunak	22
G.	Kajian Relevan.....	25
BAB III PERANCANGAN DAN PEMBUATAN ALAT		28
A.	Pendefinisian (<i>Define</i>)	29
B.	Perancangan (<i>Design</i>)	30
C.	Pengembangan (<i>Develop</i>)	38
D.	Penyebaran (<i>Disseminate</i>)	39
BAB IV HASIL DAN PEMBAHASAN.....		41
A.	Hasil Pembuatan Alat	41
B.	Pengujian Perangkat Keras dan Perangkat Lunak	44
C.	Pengujian Pengiriman Data Ke <i>WebServer</i>	49
D.	Pengujian Keseluruhan Sistem	50
BAB V PENUTUP.....		55
A.	Kesimpulan	55
B.	Saran	55
DAFTAR PUSTAKA		56
LAMPIRAN.....		59

DAFTAR TABEL

Tabel 1. Perbandingan RFID dan <i>Barcode</i>	9
Tabel 2. Kelebihan MySQL	12
Tabel 3. Spesifikasi NodeMCU ESP8266	14
Tabel 4. Fungsi <i>pin</i> NodeMCU ESP8266	14
Tabel 5. Spesifikasi RFID RC522.....	16
Tabel 6. <i>Memory Map Mifare Classic 1K</i>	18
Tabel 7. Spesifikasi LCD 16x2	20

DAFTAR GAMBAR

Gambar 1. NodeMCU ESP8266	13
Gambar 2. Bentuk dari RFID-RC522	16
Gambar 3. RFID Card Mifare 1K.....	17
Gambar 4. Bentuk Esp32 Cam.....	19
Gambar 5. Bentuk LCD 16x2	20
Gambar 6. Bentuk <i>Buzzer</i>	21
Gambar 7. Bentuk <i>Jumper</i> / Kabel Penghubung.....	22
Gambar 8. Cover Aplikasi Arduino IDE.....	23
Gambar 9. Logo Aplikasi <i>Notepad ++</i>	24
Gambar 10. Logo Aplikasi XAMPP	25
Gambar 11. Tahap Pengembangan Alat.....	28
Gambar 12. Blok Diagram	31
Gambar 13. <i>Flowchart</i> Proses Pada NodeMCU	34
Gambar 14. Bentuk Desain <i>Casing Hardware</i>	35
Gambar 15. Bentuk Desain Skema Rangkaian	35
Gambar 16. Proses Perancangan <i>Website</i>	37
Gambar 17. Gambar Penempatan Alat.....	40
Gambar 18. Tampilan <i>Box</i> dari Depan.....	42
Gambar 19. Posisi Letak RFID-RC522	42
Gambar 20. Posisi Letak NodeMCU Esp8266	43
Gambar 21. Posisi Letak LCD dalam <i>Box</i>	43
Gambar 22. Proses Pengujian <i>Camera ESP32 Cam</i>	45

Gambar 23. Hasil gambar dari <i>Web Server ESP32 Cam</i>	45
Gambar 24. Pengujian <i>RFID Tag</i> dengan <i>RFID Reader</i>	46
Gambar 25. Pengujian <i>RFID Card</i> dengan <i>RFID Reader</i>	46
Gambar 26. Menampilkan Teks Pada LCD	47
Gambar 27. Pengujian <i>Database Absensi</i>	47
Gambar 28. Pengujian <i>Database Camera</i>	48
Gambar 29. Pengujian IP Perangkat	49
Gambar 30. Pengujian Koneksi Wi-Fi	49
Gambar 31. Pengujian Data Terkirim	50

DAFTAR LAMPIRAN

Lampiran 1. Program Keseluruhan Esp8266	59
Lampiran 2. Program Keseluruhan Esp32Cam.....	64
Lampiran 3. Program <i>Database</i> Absensi	70
Lampiran 4. Program <i>Database</i> Kamera	75

BAB 1

PENDAHULUAN

A. Latar Belakang

Revolusi industri 4.0 ditandai dengan perkembangan teknologi, salah satu contoh perkembangan teknologi ini adalah *Internet of Things* (IoT). IoT adalah sebuah konsep di mana suatu peralatan diintegrasikan dengan teknologi seperti sensor dan *software* yang bertujuan untuk sistem komunikasi, sistem kendali, penghubung, dan pertukaran data melalui perangkat lain selama masih terkoneksi ke jaringan internet. IoT berkaitan erat dengan istilah *machine to machine* atau M2M. Setiap perangkat yang memiliki kemampuan berkomunikasi dengan metode M2M ini sering disebut dengan *smart devices* atau perangkat pintar. Teknologi ini menggunakan seperangkat komputer untuk mengolah data, sistem jaringan untuk mengkoneksikan satu komputer dengan komputer lainnya sehingga dapat memenuhi kebutuhan, dan teknologi telekomunikasi digunakan untuk tujuan supaya data dapat disebar dan diakses secara global. (Natakusumah, E.K, 2002).

Internet of Things (IoT) merupakan suatu sistem dimana suatu benda atau objek yang telah diterapkan teknologi seperti sensor dan *software* dengan tujuan dapat berkomunikasi, mengendalikan, menghubungkan, dan bertukar data melalui perangkat lain melalui jaringan internet. *Internet of Things* (IoT) merupakan salah satu perkembangan baru dalam dunia teknologi yang

kemungkinan besar akan menjadi teknologi di masa depan. Kemajuan pada perangkat mikrokontroller berbasis IoT salah satunya yaitu teknologi informasi yang dapat mengendalikan suatu alat elektronika diberbagai tempat tetapi harus tetap terhubung dengan jaringan internet dan dapat memberikan informasi secara *real time* mengenai perpindahan atau pergerakan sesuatu dari lokasi asal ke lokasi tujuan.

Salah satu bentuk perkembangan teknologi IoT adalah RFID (*Radio-frequency Identification*) yang merupakan teknologi yang dapat memudahkan identifikasi dimana teknologi ini dilengkapi dengan *chip* dan *tag* yang memiliki fungsi yang sama dengan *barcode*. Selanjutnya teknologi ini dapat dikembangkan sebagai kartu absensi dengan menggabungkannya dengan NodeMcu yang berfungsi sebagai mikrokontroler. Sistem informasi absensi pada suatu instansi merupakan salah satu hal yang sangat penting dilakukan untuk mendata kehadiran yang ada pada instansi tersebut. Suatu fasilitas atau sarana sangat diperlukan untuk menunjang dan membantu melaksanakan pengolahan data yang tepat. Dengan demikian penggunaan komputer dalam menghasilkan informasi sangat dibutuhkan dalam mendukung sistem pengambilan keputusan.

Selama ini proses absensi dicatat secara manual, pencatatan dilakukan di buku dimana perlu mendata secara manual dengan paraf atau *ceklist* yang hadir. Dengan sistem absensi ini dilakukannya pendataan secara satu per satu yang juga memakan banyak waktu untuk digunakan dalam proses absensi. Namun dengan menggunakan sistem absensi manual menimbulkan berbagai masalah,

seperti rusaknya kertas absensi karena sobek, basah dan berbagai masalah lainnya. Pada sistem absensi manual pemantauan waktu kehadiran tidak dapat dilakukan. Sedangkan dengan memantau waktu kehadiran sangat bermanfaat untuk mengamati kebiasaan hadir. Waktu juga merupakan salah satu faktor dalam penilaian. Sistem absensi digital yang akan di buat dalam kasus ini yaitu menggunakan teknologi RFID yang memiliki kepraktisan dan bisa ditambahkan kemampuan untuk memantau ketepatan waktu kehadiran. Mengacu pada permasalahan, tujuan yang hendak di capai dalam penelitian ini adalah merancang suatu sistem absensi dengan menggunakan teknologi RFID yang dapat membantu instansi untuk dapat meningkatkan efektifitas dalam melakukan pengolahan data absensi dengan memperkecil kemungkinan-kemungkinan kesalahan yang akan terjadi dan agar informasi keamanan dapat terjamin. Selain itu hal ini dapat menjadi ikon dari suatu instansi untuk menunjukkan perkembangan teknologi yang semakin maju dan semakin berkembang dengan pesat, hal ini juga menjadi landasan bahwa instansi memerlukan teknologi terbaru yang dapat berjalan tanpa menimbulkan resiko yang besar.

Dari fenomena perkembangan teknologi sistem informasi tersebut maka munculah keinginan untuk membuat sebuah alat yang berjudul ***“Rancang Bangun Sistem Informasi Absensi Berbasis ESP8266”***. Pembuatan alat ini diharapkan dapat memudahkan proses Absensi. Hal ini berdasarkan beberapa artikel sebelumnya yang menjadi informasi serta menjadi bahan penelitian.

B. Identifikasi Masalah

Berdasarkan latar belakang tersebut, dapat diidentifikasi beberapa masalah diantaranya:

1. Beberapa Metode sebelumnya memiliki pencatatan kehadiran yang kurang cepat untuk melakukan rekapitulasi data kehadiran.
2. Beberapa sistem sebelumnya memiliki tingkat keamanan dan keakuratan data yang kurang baik.

C. Batasan Masalah

Berdasarkan identifikasi masalah tersebut, terdapat beberapa batasan masalah diantaranya:

1. Membatasi masalah sampai pada tahap pencatatan absensi yang cepat dalam melakukan rekapitulasi.
2. Membatasi masalah sampai pada tingkat keamanan dan keakuratan data yang maksimal dalam sistem informasi absensi.

D. Rumusan Masalah

Berdasarkan identifikasi masalah dan batasan masalah, maka rumusan masalah yang akan dibahas yaitu:

1. Bagaimana efektivitas sistem informasi absensi berbasis ESP8266 dalam meningkatkan efisiensi pencatatan kehadiran dibandingkan dengan metode konvensional?
2. Bagaimana mengintegrasikan Sistem informasi Absensi berbasis ESP8266 dengan sistem yang sudah ada?

E. Tujuan Tugas Akhir

Berdasarkan batasan dan rumusan masalah di atas, maka tujuan tugas akhir ini adalah:

1. Menciptakan sistem informasi absensi yang dapat meningkatkan efisiensi dan akurasi pencatatan kehadiran dibandingkan sebelumnya.
2. Mengatasi dan memperbaiki kendala-kendala dalam pemantauan kehadiran yang lebih baik.

F. Manfaat Tugas Akhir

Manfaat yang dapat diperoleh dari perancangan tugas akhir ini adalah:

1. Tugas akhir ini diharapkan memberikan kontribusi nyata terhadap efisiensi melalui implementasi sistem informasi absensi yang lebih canggih dan otomatis.
2. Dengan memanfaatkan teknologi ESP8266, tugas akhir ini diharapkan dapat menyediakan data kehadiran yang lebih akurat dan *real-time*.

BAB II

LANDASAN TEORI

A. Sistem Informasi

1. Sistem

Sistem adalah sekelompok dari dua atau lebih komponen atau subsistem yang berhubungan untuk melayani tujuan umum (*James A Hall dalam Ardana, I Cenik, Lukman, 2016*). Sistem adalah kumpulan / grup dari subsistem / bagian / komponen apapun, baik fisik ataupun nonfisik yang saling berhubungan satu sama lain dan bekerja sama secara harmonis untuk mencapai satu tujuan tertentu (*Djahir dan Pratita, 2015*).

Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan kegiatan atau untuk melakukan sasaran yang tertentu (*Hutahaean, 2015*). Sistem adalah suatu kumpulan atau himpunan dari kegiatan, komponen, unsur, elemen atau variabel yang terorganisir, saling berinteraksi dan saling berhubungan satu sama lain melakukan kerjasama dengan cara-cara tertentu secara harmonis sehingga membentuk kesatuan untuk melaksanakan suatu fungsi untuk mencapai tujuan yang telah ditetapkan.

2. Informasi

Informasi adalah data yang telah disusun dan diproses sehingga bermanfaat bagi proses pengambilan keputusan. Semakin banyak dan semakin berkualitas informasi yang tersedia, maka pengambilan keputusan semakin baik (*Dany, 2017*).

Informasi adalah data yang telah diproses ke dalam suatu bentuk yang mempunyai arti bagi penerimanya dan mempunyai nilai nyata dan terasa bagi keputusan saat itu dan mendatang (*Sutabri, 2005*). Informasi adalah suatu rekaman fenomena yang diamati, atau bisa juga berupa putusan-putusan yang dibuat oleh seseorang (*Estabrook dalam Yusup, 2009 : 11*). Informasi adalah data yang telah diproses ke dalam suatu bentuk, baik berupa pesan atau keterangan seperti putusan seseorang, ataupun objek fisik yang disampaikan oleh pengirim kepada penerima, yang berguna untuk penyusunan keterangan, pembuatan kesimpulan, dan dasar bagi pengambilan keputusan baik sekarang maupun di masa yang akan datang.

3. Sistem Informasi

Sistem informasi adalah suatu kegiatan dari prosedur-prosedur yang diorganisasikan, bilamana dieksekusi akan menyediakan informasi untuk mendukung pengambilan keputusan dan pengendalian didalam organisasi. dalam arti yang sangat luas, istilah sistem informasi yang sering digunakan merujuk kepada interaksi antara orang, proses, algoritmik, data, dan teknologi (*McLeod & Schell, 2007*). Sistem informasi dapat didefinisikan secara teknis sebagai suatu rangkaian yang komponen- komponen saling terkait yang mengumpulkan (dan mengambil kembali), memproses, menyimpan dan mendistribusikan informasi untuk mendukung pengambilan keputusan dan mengendalikan perusahaan (*Laudon dalam Ardana, I Cenik, Lukman, 2016*).

Sistem informasi adalah sekumpulan komponen berupa manusia, teknologi dan prosedur untuk memproses dan menghasilkan informasi untuk mencapai suatu tujuan atau sasaran dan mengambil suatu keputusan. Manfaat dari sistem informasi itu sendiri sangatlah banyak, berikut adalah beberapa manfaat dari sistem informasi: (a) Memberikan informasi yang sudah terjamin kebenarannya, (b) Lebih efisien, (c) Meningkatkan kemampuan dalam mengambil keputusan, (d) Meningkatkan kualitas informasi, (e) Lebih terjamin kemanannya (*McLeod & Schell, 2007*).

Sebuah sistem informasi memiliki 3 (tiga) aktivitas atau kegiatan utama, yaitu menerima data sebagai masukan (*input*) selanjutnya memproses data masukan dengan melakukan perhitungan, klasifikasi data dan lain-lain, kemudian selanjutnya menghasilkan informasi yang tepat sebagai keluaran (*output*) untuk orang yang akan menggunakannya ataupun aktivitas lain yang membutuhkan informasi tersebut.

B. Teknologi *Radio Frequency Identification* (RFID)

RFID adalah singkatan dari *Radio Frequency Identification*, yang merupakan teknologi komunikasi *nirkabel* yang digunakan untuk melakukan identifikasi secara unik dari suatu objek yang memiliki *tag* atau label (*Brown, Patadia, & Dua, 2007*). RFID termasuk dalam sebuah teknologi yang disebut dengan nama *Auto identification* (Auto-ID). Teknologi Auto-ID adalah semua hal yang mengumpulkan data dari suatu objek, kemudian memasukkan data tersebut ke dalam sebuah data penyimpanan atau *database* tanpa campur tangan manusia (*Brown et al., 2007*). RFID termasuk dalam teknologi yang

menggunakan gelombang radio elektromagnetik untuk mengidentifikasi suatu objek, tempat, hewan, atau manusia (*Brown et al., 2007*). Nomor unik yang digunakan untuk proses indentifikasi objek tersimpan dalam *Integrated Circuit* (IC) yang tertera pada sebuah antena (*Brown et al., 2007*).

Antena beserta IC tersebut disebut dengan nama *RFID tag*. *Tag* tertera pada sebuah objek untuk diidentifikasi. Alat berupa *reader* berkomunikasi dengan *tag* tersebut yang kemudian digunakan untuk membaca nomor pada *tag* kemudian memproses nya pada sebuah sistem untuk menyimpan atau mencari nomor *tag* tersebut pada *database*. Sebagai perbandingan, barcode juga termasuk dalam teknologi *Auto Identification*. Pada sebuah *barcode* terdapat dua hal yang nantinya akan diidentifikasi oleh *scanner* yaitu sebagai berikut: 1. *Barcode linear* yang menggunakan garis berwarna hitam dan putih dengan ukuran lebar yang berbeda sebagai penyandian nomor. 2. *Barcode matriks* yang menggunakan potongan *array* dua dimensi berwarna hitam dan putih sebagai penyandian informasi. Meskipun sama-sama termasuk dalam teknologi *Auto Identification*, RFID berbeda dengan barcode.

Perbandingan tentang kelebihan dan kekurangan antara RFID dan *barcode* dapat dilihat pada **Table.1** berikut:

Tabel 1. Perbandingan RFID dan *Barcode*

RFID	<i>Barcode</i>
Dapat membaca suatu <i>tag</i> walaupun <i>tag</i> tersebut tidak terlihat atau tertutup oleh suatu objek tertentu.	<i>Scanner barcode</i> harus “melihat” <i>barcode</i> untuk membacanya.

Secara unik mengidentifikasi setiap objek yang berbeda	Hanya mengidentifikasi objek dalam kategori tertentu
Posisi <i>reader</i> dan <i>tag</i> tidak dipermasalahkan ketika proses identifikasi	Posisi <i>reader</i> dan barcode harus tepat ketika proses identifikasi
Dapat mengidentifikasi lebih dari satu objek pada satu waktu	Hanya mengidentifikasi satu objek pada satu waktu
Mempunyai kemampuan membaca atau menulis secara dinamis	Hanya dapat membaca, informasi bersifat statis
Dapat digunakan dalam lingkungan yang lebih keras	Label <i>barcode</i> yang rusak akan sulit untuk dibaca oleh <i>Scanner</i> .
Memiliki data penyimpanan yang lebih besar	Data penyimpanan terbatas

C. Database

Database merupakan kumpulan dari file yang saling berkaitan dan berinteraksi dengan relasi yang ditunjukkan menggunakan kunci dari tiap-tiap file yang ada. Satu *database* menunjukkan suatu kumpulan data yang dipakai dalam suatu lingkup perusahaan atau instansi. Pengolahan *database* merupakan suatu cara yang dilakukan terhadap file-file yang berada di suatu instansi yang mana file tersebut dapat disusun, diurut, diambil sewaktu-waktu serta dapat ditampilkan dalam suatu bentuk laporan sehingga data-data tadi dapat disajikan menjadi suatu informasi yang rapi.

Proses memasukkan dan mengambil data dari media penyimpanan data memerlukan perangkat lunak yang disebut dengan sistem manajemen basis data (*database management system*) atau DBMS. DBMS merupakan sistem

perangkat lunak yang memungkinkan pengguna basis data (*database user*) untuk memelihara, mengontrol dan mengakses data secara praktis dan efisien. Dengan kata lain, semua akses ke basis data akan ditangani oleh DBMS. DBMS ini menjadi lapisan yang menghubungkan basis data dengan program aplikasi untuk memastikan bahwa basis data tetap terorganisasi secara konsisten dan dapat diakses dengan mudah.

Manfaat menggunakan *database* adalah satu *database* bisa digunakan bersama-sama dalam suatu instansi sehingga satu *database* bisa menjadi kontrol atau terpusat. *Database* juga membuat kemudahan dan kecepatan dalam mengakses dan memperoleh data. Berdasarkan jenisnya, basis data dibagi menjadi *database flat-file* dan basis data relasional. Basis data *flat-file* ideal untuk data berukuran kecil dan dapat dirubah dengan mudah. Sedangkan basis data relasional mempunyai struktur yang lebih logis terkait cara penyimpanan. Kata relasional berasal dari kenyataan bahwa tabel-tabel yang ada di basis data relasional dihubungkan satu dengan lainnya. Basis data relasional menggunakan sekumpulan tabel dua dimensi yang masing masing tabel tersusun atas baris (*tupel*) dan kolom (*atribut*).

Untuk membuat hubungan antara dua atau lebih tabel, digunakan *key* (atribut kunci) yaitu *primary key* di salah satu tabel dan *foreign key* di tabel yang lain. Saat ini, basis data relasional menjadi pilihan utama karena keunggulannya. Beberapa contoh basis data relasional diantaranya *oracle*, *Microsoft SQL Server*, *MariaDB*, dan *Mysql*.

D. *My Structured Query Language (MySQL)*

MySQL merupakan turunan salah satu konsep utama dalam *database* yang telah ada sebelumnya, yaitu SQL (*Structured Query Language*) MySQL sendiri merupakan implementasi dari sistem manajemen *database* relasional (RDBMS) yang didistribusikan secara gratis dibawah lisensi GPL (*General Public License*). Setiap pengguna dapat secara bebas menggunakan *MySQL*, namun produk turunan dari perangkat lunak tersebut tidak boleh bersifat komersial. MySQL adalah salah satu jenis *database server* yang sangat terkenal dan banyak digunakan untuk membangun aplikasi *web* yang *database* sebagai sumber dan pengelolaan datanya. Kepopuleran MySQL antara lain karena MySQL menggunakan SQL sebagai bahasa dasar untuk mengakses *database*-nya sehingga mudah untuk digunakan. MySQL juga bersifat *open source* dan *free* pada berbagai platform kecuali pada *windows* yang bersifat *shareware* seperti pada Tabel 2.

Tabel 2. Kelebihan MySQL

1	<i>Source</i> MySQL dapat diperoleh dengan mudah dan gratis.
2	Sintaksnya lebih mudah dipahami dan tidak rumit.
3	Pengaksesan <i>database</i> dapat dilakukan dengan mudah.
4	MySQL merupakan program yang <i>multi threaded</i> , sehingga dapat dipasang pada <i>server</i> yang memiliki <i>multiCPU</i> .
5	Didukung programprogram umum seperti C, C++, Java, Perl, PHP, Python, dsb.
6	Bekerja pada berbagai platform.

7	Memiliki jenis kolom yang cukup banyak sehingga memudahkan konfigurasi <i>system database</i> .
8	Memiliki sistem <i>security</i> yang cukup baik dengan verifikasi <i>host</i> .
9	Mendukung ODBC untuk sistem operasi <i>Windows</i> .
10	Mendukung <i>record</i> yang memiliki kolom dengan panjang tetap atau panjang bervariasi.

E. Perangkat Keras



Gambar 1. NodeMCU ESP8266

Perangkat keras adalah segala sesuatu yang ada pada komputer yang dapat dilihat dan dirasakan, dan secara umum dapat diklasifikasikan sebagai bagian dari sistem komputer. Perangkat keras berfungsi untuk melakukan *input*, *process*, dan *output* data yang dikirimkan dalam bentuk informasi.

1. NodeMCU ESP8266

NodeMCU adalah sebuah *board* elektronik yang berbasis *chip* ESP8266 dengan kemampuan menjalankan fungsi mikrokontroler dan juga koneksi internet (Wi-Fi) seperti pada Gambar 1. Terdapat beberapa *pin* I/O

sehingga dapat dikembangkan menjadi sebuah aplikasi *monitoring* maupun *controlling* pada proyek IoT, NodeMCU ESP8266 dapat diprogram dengan *compiler*-nya Arduino, menggunakan Arduino IDE. Bentuk fisik dari NodeMCU ESP8266, terdapat *port* USB (*miniUSB*) sehingga akan memudahkan dalam pemrogramannya seperti Tabel 3 dan Tabel 4 berikut.

Tabel 3. Spesifikasi NodeMCU *ESP8266*

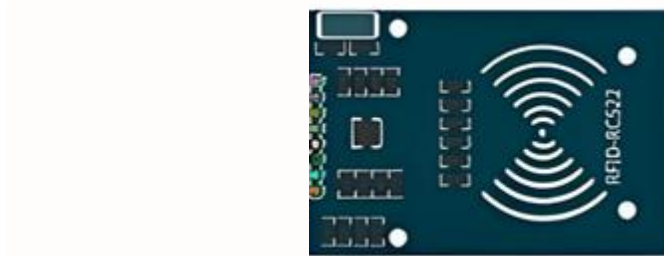
Fitur	Spesifikasi
Mikrokontroler	ESP8266-12E
Ukuran Papan	57 mm x 30 mm
Tegangan Input	3.3V – 5V
GPIO	13 pin
Kanal PWM	10 kanal
10 bit ADC pin	1 pin
<i>Flash Memory</i>	4 MB
<i>Clock Speed</i>	40/26/24 MHz
Wifi	IEEE 802.11b/g/n
Frekuensi	2.4 GHz – 2.5 GHz
USB <i>port</i>	Micro USB
USB to <i>Serial Converter</i>	CH340G
<i>Card Reader</i>	-

Tabel 4. Fungsi *pin* NodeMCU ESP8266

Pin	Fungsi
RST	<i>Chip enable, active high</i>
IO16	GPIO6. Untuk membangun <i>chipset</i> dari <i>mode deep sleep</i>
IO14	GPIO; HSPI_CLK
IO12	GPIOI3; HSPI_MISO
IO13	GPIO3; HSPI_MOSI; UART0_CTS
VCC	Catu daya 3.3V (VDD)
CS0	<i>Chip selection</i>
MISO	<i>Slave Output, main output</i>
IO9	GPIO9
IO10	GPIO10
MOSI	<i>Main output slave input</i>
SCLK	<i>Clock</i>
GND	<i>Ground</i>
IO15	GPIO5;MTDO; HSPICS; UART0_RTS
IO2	GPIO2;UART1_TXD
IO0	GPIO0
IO4	GPIO4
IO5	GPIO5
RXD	UART0_RXD:GPIO3
TXD	UART0_TXD:GPIO1

2. RFID MRC522

RFID *reader* adalah sebuah perangkat / modul untuk membaca ataupun menuliskan data ke kartu RFID seperti pada Gambar 2 Ada banyak variasi dari RFID.



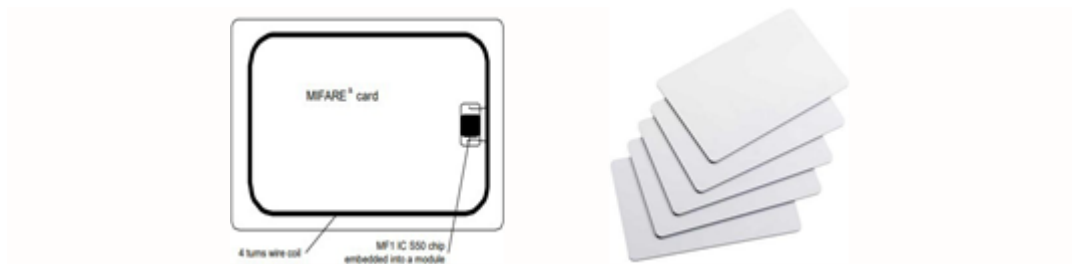
Gambar 2. Bentuk dari RFID-RC522

Berikut adalah tabel pada RFID yang mendetailkan spesifikasi terkait RFID itu sendiri seperti pada Tabel 5 berikut.

Tabel 5. Spesifikasi RFID RC522

Tegangan untuk beroperasi	3.3V
Arus	Idle: 10~13mA Sleep : <80uA Peak : <30mA
Ukuran Papan	40mm x 60mm
Frekuensi	13.56MHz
<i>Supported Cards</i>	<ul style="list-style-type: none"> • <i>Mifare 1 S50 (1K)</i> • <i>Mifare 1 S70 (4K)</i> • <i>Mifare UltraLight</i> • <i>Mifare Pro</i> • <i>Mifare Desfire</i>

3. Kartu RFID



Gambar 3. RFID Card Mifare 1K

Reader / writer seperti pada Gambar 3, diantaranya adalah modul MFRC522 untuk mikrokontroler seperti Arduino. Modul RFID RC522 adalah sebuah *low-cost RFID module* dengan IC tertanam yang mampu melakukan *read / write* dengan teknologi komunikasi *contactless* dengan frekuensi 13.56 MHz. Komunikasi ini dapat berlangsung ketika suatu kartu RFID berada pada jangkauan perangkat RFID RW. Modul ini memiliki spesifikasi yang dapat dilihat pada tabel. Proses komunikasi pada RFID tidak semata-mata ketika menempel langsung terjadi komunikasi. Proses komunikasi ini memiliki beberapa tahapan sebagai berikut:

a. Memilih *sector / block* yang ingin dibaca

Pemilihan *sector / block* untuk penulisan atau proses membaca data bertujuan agar lebih spesifik dan menghemat waktu. Pembacaan data pada satu *block* tentu lebih efisien dibandingkan membaca data dari keseluruhan *block* yang ada pada kartu RFID.

b. Autentikasi

Proses autentikasi adalah proses untuk mencocokkan *key* yang ada pada *trailer block* (*block* untuk menyimpan kunci untuk proses enkripsi

atau dekripsi) dengan kunci yang di-*load* pada RC522. Pemberian kunci pada kartu *RFID* bertujuan agar tidak mudah untuk dibaca dengan

RFID RW lainnya, karena jika kunci yang ada pada kartu tidak sama dengan yang diatur pada RC522, maka kartu tersebut tidak dapat diidentifikasi.

- c. Membaca / menulis data pada *sector* / *block* yang dipilih.

Proses menulis atau membaca kartu *RFID* dilakukan terhadap *block* tertentu. Akan tetapi, jika pada tahap autentikasi gagal, maka proses membaca atau menulis pada kartu *RFID* juga akan gagal.

Tabel 6. *Memory Map Mifare Classic 1K*

		Byte Number within a Block																Description
Sector	Block	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
15	3	Key A				Access Bits				Key B								Sector Trailer 15
	2																	Data
	1																	Data
14	3	Key A				Access Bits				Key B								Sector Trailer 14
	2																	Data
	1																	Data
13	3	Key A				Access Bits				Key B								
	2																	
	1																	
12	3	Key A				Access Bits				Key B								
	2																	
	1																	
11	3	Key A				Access Bits				Key B								
	2																	
	1																	
10	3	Key A				Access Bits				Key B								
	2																	
	1																	
9	3	Key A				Access Bits				Key B								
	2																	
	1																	
8	3	Key A				Access Bits				Key B								
	2																	
	1																	
7	3	Key A				Access Bits				Key B								
	2																	
	1																	
6	3	Key A				Access Bits				Key B								
	2																	
	1																	
5	3	Key A				Access Bits				Key B								
	2																	
	1																	
4	3	Key A				Access Bits				Key B								
	2																	
	1																	
3	3	Key A				Access Bits				Key B								
	2																	
	1																	
2	3	Key A				Access Bits				Key B								
	2																	
	1																	
1	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	
	1																	
0	3	Key A				Access Bits				Key B								
	2																	

Secara keseluruhan, kartu RFID (*Mifare 1K*) tersebut memiliki 16 *sector* dimana pada masing-masing *sector* terdiri dari 4 *block* dan pada setiap *block* mampu menyimpan karakter sebanyak 16 bytes. Kartu RFID memiliki satu *manufacturer block* yang terletak pada *block 0*. *Block* ini berisi informasi mengenai ID kartu (UID). UID (Unique Identifier) ini tidak dapat diubah, karena sudah didesain oleh pabrik agar tidak bisa diubah, sehingga pengisian data dilakukan pada *block* data dengan panjang 16 bytes.

Setiap *sector* terdapat satu *block* yang berfungsi untuk menyimpan *key* untuk sistem keamanan pada kartu RFID yang dinamakan *Trailer Block*. *Block* ini berfungsi untuk mengamankan setiap *sector* agar tidak bisa dibaca oleh sembarang perangkat RFID *Reader*.

4. ESP32 CAM



Gambar 4. Bentuk Esp32 Cam

ESP32-Cam adalah sebuah modul pengembangan yang menggabungkan *chip* ESP32-S, kamera OV2640, dan *slot* kartu *microSD* seperti pada Gambar 4. Modul ini memiliki banyak potensi dan dapat digunakan untuk berbagai proyek. Papan pengembangan Wi-Fi dan *Bluetooth* dengan mikrokontroler Esp32 dan kamera. Modul ini

menyediakan fitur yang dapat digunakan siapa saja, atau bisa dikatakan *open source*, salah satu fiturnya yaitu digunakan untuk mengambil gambar, pengenalan wajah dan deteksi wajah. Modul *peripheral* tersebut dapat digunakan menggunakan *editor* Arduino IDE untuk memanfaatkan *library* atau fitur yang sudah disediakan.

5. LCD 16x2



Gambar 5. Bentuk LCD 16x2

Liquid Crystal Display (LCD) adalah sebuah peraga kristal cair seperti pada Gambar 5. Prinsip kerja LCD adalah mengatur cahaya yang ada, atau nyala LED (*light-emitting diode*). Dibandingkan dengan *seven segment*, banyak orang yang lebih suka memakai LCD karena pemakaian daya yang sangat rendah.

Tabel 7. Spesifikasi LCD 16x2

Pin	Diskripsi
1	<i>Ground (-)</i>
2	Vcc (+)
3	Mengatur Kontras atau pencahayaan
4	<i>Register Select</i>
5	<i>Read/Write LCD Register</i>

6	<i>Enable</i>
7-14	Data I/O (<i>input output</i>)
15	VCC (+) LED
16	<i>Ground</i> (-) LED

Dapat diperlihatkan konfigurasi penyemat LCD yang terdiri dari 16 penyemat seperti pada Tabel 7, yang masing-masing penyemat mempunyai fungsi yang berbeda-beda. LCD 16x2 terdiri dari dua bagian utama. Bagian pertama merupakan panel LCD sebagai media penampil informasi dalam bentuk huruf/angka dua baris, masing-masing baris bisa menampung 16 huruf/angka.

Bagian kedua merupakan sebuah sistem yang dibentuk dengan mikrokontroler yang ditempelkan dibalik panel LCD, yang berfungsi mengatur tampilan informasi dan mengatur komunikasi LCD dengan mikrokontroler diatas dijelaskan bahwa data *input* LCD yang berupa 8 bit.

6. Buzzer



Gambar 6. Bentuk Buzzer

Buzzer merupakan sebuah modul komponen elektronika kategori transduser, yang bekerja dengan cara mengubah sinyal elektrik menjadi sebuah gelombang suara seperti pada Gambar 6. *Buzzer* biasa difungsikan

sebagai alarm sinyal. Biasa di implementasikan pada project penelitian sebagai sebuah indikator terhadap suatu kondisi.

7. Jumper



Gambar 7. Bentuk Jumper / Kabel Penghubung

Digunakan untuk menyalurkan energi listrik. Sebuah kabel listrik terdiri dari *isolator* dan *konduktor* seperti pada Gambar 7. *Isolator* adalah bahan pembungkus kabel yang biasa terbuat dari plastik atau karet, sedangkan *konduktor* terbuat dari tembaga.

F. Perangkat Lunak

Perangkat lunak (*Software*) adalah kumpulan perintah yang dijalankan oleh komputer untuk menyelesaikan pekerjaan. *Software* mencakup instruksi, program, dan data yang diperlukan untuk mengoperasikan sistem komputer.

1. Arduino IDE

IDE merupakan singkatan dari *Integrated Development Environment*. IDE merupakan program yang digunakan untuk membuat program pada NodeMCU Esp8266. Program yang ditulis dengan menggunakan *Software* Arduino IDE disebut sebagai *sketch* seperti pada Gambar 8. *Sketch* ditulis dalam suatu *editor* teks dan disimpan dalam file dengan ekstensi .ino. Pada *Software* Arduino IDE, terdapat semacam *message box* berwarna hitam yang berfungsi menampilkan status, seperti pesan *error*, *compile*, dan

upload program. Di bagian bawah paling kanan *Software Arduino IDE*, menunjukkan *board* yang terkonfigurasi beserta *COM ports* yang digunakan.

- a. *Verify/Compile*, berfungsi untuk mengecek apakah *sketch* yang dibuat ada kekeliruan dari segi *sintaks* atau tidak. Jika tidak ada kesalahan, maka *sintaks* yang dibuat akan di *compile* kedalam bahasa mesin.
- b. *Upload*, berfungsi mengirimkan program yang sudah dikompilasi ke *Arduino Board*.



Gambar 8. Cover Aplikasi Arduino IDE

<https://funkynotes.blogspot.com/>

IDE merupakan perangkat lunak yang memainkan peran yang sangat penting dalam pemrograman, kompilasi *biner*, dan unduhan memori mikrokontroler. Selain banyak modul pendukung (sensor, monitor, pembaca, dll.) Arduino telah menjadi platform karena telah menjadi pilihan bagi banyak profesional. Salah satu alasan Arduino memikat banyak orang adalah karena sifatnya yang *open source*, baik *hardware* maupun *software*. Skema Arduino gratis untuk semua orang. Anda bebas mengunduh gambar, membeli komponen, membuat PCB, dan merakit sendiri tanpa membayar

pembuat Arduino. Demikian pula, Arduino IDE dapat diunduh secara gratis dan diinstal di komputer Anda. Kami perlu berterima kasih kepada tim Arduino karena begitu murah hati dalam berbagi kemewahan kerja keras dengan semua orang. Secara pribadi, saya sangat terkejut dengan kualitas tinggi dan desain canggih dari perangkat keras Arduino, bahasa pemrograman, dan IDE.

2. *Notepad++* (Text Editor)

Notepad++ adalah sebuah aplikasi penyunting teks dan penyunting kode sumber yang berjalan di sistem operasi *Windows* seperti pada Gambar 9. *Notepad++* menggunakan komponen *Scintilla* untuk dapat menampilkan dan menyuntingan teks dan berkas kode sumber berbagai bahasa pemrograman, Kaciak (2016)



Gambar 9. Logo Aplikasi Notepad ++

<https://notepad-plus-plus.org/>

3. XAMPP

XAMPP adalah aplikasi *webserver* instan yang dibutuhkan untuk membangun aplikasi berbasis web seperti pada Gambar 10. Fungsi *XAMPP* adalah sebagai server yang berdiri sendiri (*localhost*) yang terdiri dari atas program *Apache HTTP Server*, *MySQL database*, dan penterjemah bahasa

yang ditulis dengan bahasa pemrograman *PHP dan Perl*. Nama *XAMPP* merupakan singkatan dari *x* (*X= Cross platform*), *Apache*, *MySQL*, *PHP* dan *Perl*. Program ini tersedia dalam lisensi GNU (*General Public License*) dan Gratis, Gunawan (2010:17).



Gambar 10. Logo Aplikasi XAMPP

<https://itbox.id/>

G. Kajian Relevan

1. S.Sunardi, Amirah, Salman, Santi. (2024) dengan judul " *Perancangan Dan Implementasi Sistem Absensi Karyawan Berbasis RFID dan Web Server*". Hasil akhir dari penelitian menunjukkan bahwa Sistem kendali kehadiran berupa teknologi *website* dapat di integrasikan dengan *hardware microcontroller* dan *reader* RFID serta dapat mengoptimalkan proses presensi dengan mengubah sistem awal menjadi sistem komputerisasi.
2. Ryan Hidayat, F. Yudi Limpraptono, Michael Ardita. (2022) dengan judul "*Rancang Bangun Alat Absensi Karyawan Menggunakan RFID dan ESP32CAM Berbasis Internet Of Things*". Hasil akhir dari penelitian menunjukkan bahwa Alat absensi yang telah dibangun dengan

menggunakan sensor RFID dan ESP32-Cam dapat berfungsi dengan baik sesuai perencanaan. Adapun untuk sensor dapat mendeteksi RFID *card* dengan baik sedangkan hasil foto dari ESP32-Cam tersimpan pada *database* dengan konfigurasi ini dapat diperhatikan pada *inspect* foto hasil absensi. Rancangan *interface* yang sesuai untuk komunikasi antara NodeMCU dengan *Web Server* juga berhasil dilakukan pada NodeMCU. Kemudian dari data hasil absensi yang ditampilkan pada *Web Server* berisi beberapa informasi.

3. Ery Setyawan, Djameludin, Silvia Ayunda Murad. (2021) Dengan Judul "*Sistem Alat Absensi Menggunakan RFID Dan Camera Berbasis Internet Of Things*". Hasil akhir dari penelitian menunjukkan bahwa hasil percobaan, bahwa NodeMCU dan RFID dapat digunakan untuk perancangan sistem absensi berbasis IoT, ESP32-Cam dapat digunakan sebagai sensor untuk pengambilan gambar wajah. Selain itu sistem absensi dapat merekam kehadiran per-hari dengan merekam *id* dari RFID tersebut dan menyimpan gambar ke dalam *database* sebagai laporan dan validasi data absensi.
4. Kuku Prasetiyo Aji, Ucu Darusalam, Novi Dian Nathasia. (2020) dengan judul "*Perancangan Sistem Presensi Untuk Pegawai Dengan RFID Berbasis IoT Menggunakan NodeMCU ESP8266*". Hasil akhir dari penelitian menunjukkan bahwa Berdasarkan hasil pengujian yang telah dilakukan pembacaan RFID berhasil dilakukan dengan jarak 3-4 cm. Banyak jenis kartu yang bisa terbaca oleh RFID *Reader*. Data yang

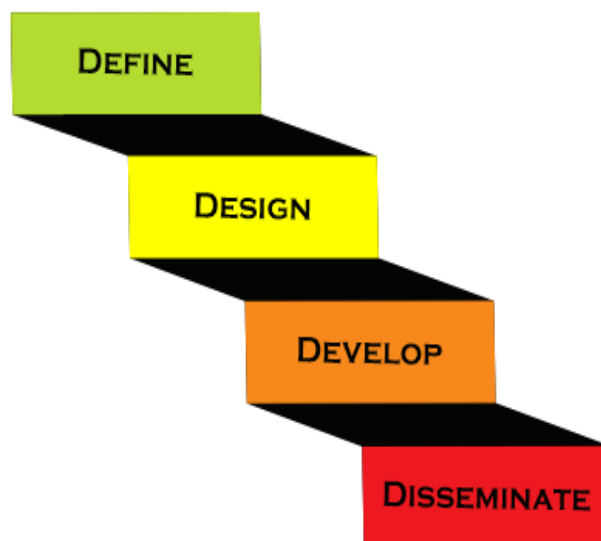
diperoleh dari pembacaan RFID yang berupa kode *Tag* berhasil tersimpan pada *database*. Untuk meningkatkan kinerja, adapun saran dari penulis untuk lebih meningkatkan tampilan antarmuka *website* sistem presensi dan disarankan untuk memperluas jangkauan jarak baca dari RFID *Reader*.

5. Dian Ariyanto. (2024) dengan judul “*Rancang Bangun Sistem Pencatat Kehadiran Di Laboratorium Menggunakan RFID Dan ESP32-Cam Berbasis IoT*” Hasil akhir dari penelitian ialah diperoleh jarak maksimal kartu RFID *Tag* terhadap modul MFRC522 adalah 4.0 cm jika jarak lebih dari 4.0 cm maka data tidak dapat terbaca. Hasil foto dari ESP32-Cam yang tersimpan di *google drive* mempunyai format jpeg dengan kualitas gambar VGA (*Video Graphic Adaptor*) yang mempunyai resolusi 640x480 *pixel* dengan ukuran file berkisar di 19KB-21KB.

BAB III

PERANCANGAN DAN PEMBUATAN ALAT

Metode pengembangan media pembelajaran ini menggunakan pengembangan model 4-D sebagai metodologinya. Penelitian ini mengacu pada model pengembangan 4-D (*Four-D*) yang dikembangkan oleh Thiagarajan, Semmel, dan Semmel (1974). Langkah-langkah model pengembangan 4-D menurut Thiagarajan, dkk (dalam Trianto, 2012:232) terdiri dari empat tahap pengembangan, yaitu *define*, *design*, *develop*, dan *disseminate*, atau diadaptasikan menjadi model 4-P seperti pada Gambar 11, yaitu pendefinisian, perancangan, pengembangan, dan penyebaran. Tujuan penelitian ini secara umum ialah untuk mengembangkan suatu strategi pembelajaran inovatif.



Gambar 11. Tahap Pengembangan Alat

A. Pendefinisian (*Define*)

Tahap pendefinisian dilakukan untuk menetapkan kebutuhan dan komponen yang digunakan dalam pembuatan sistem absensi berbasis ESP8266. Tujuan dari tahap ini adalah memastikan semua aspek sistem telah direncanakan dengan baik sebelum memasuki proses perancangan.

1. Analisis Kebutuhan Sistem

Analisis kebutuhan dilakukan untuk mengidentifikasi perangkat keras dan perangkat lunak yang diperlukan:

a. Perangkat Keras

1. *RFID Reader* MFRC522 berfungsi membaca data kartu RFID.
2. NodeMCU ESP8266 berfungsi sebagai mikrokontroler dan penghubung dengan *server* melalui Wi-Fi.
3. ESP32 CAM mengambil gambar untuk keperluan identifikasi tambahan.
4. LCD 16x2 menampilkan informasi absensi seperti nama pengguna atau status kehadiran.
5. *Buzzer* memberikan notifikasi berupa suara.
6. *Power Supply* memberikan daya untuk seluruh komponen.
7. Kabel *Jumper* menghubungkan komponen elektronik.

b. Perangkat Lunak

1. Arduino IDE untuk memprogram NodeMCU dan ESP32 CAM.
2. *Notepad++* sebagai *editor script*.

3. XAMPP digunakan untuk membangun *server* lokal dengan *database* MySQL.

2. Blok Diagram Sistem

Sistem dirancang dengan memanfaatkan tiga elemen utama yaitu *input*, *process*, dan *output*. Diagram blok sistem menunjukkan:

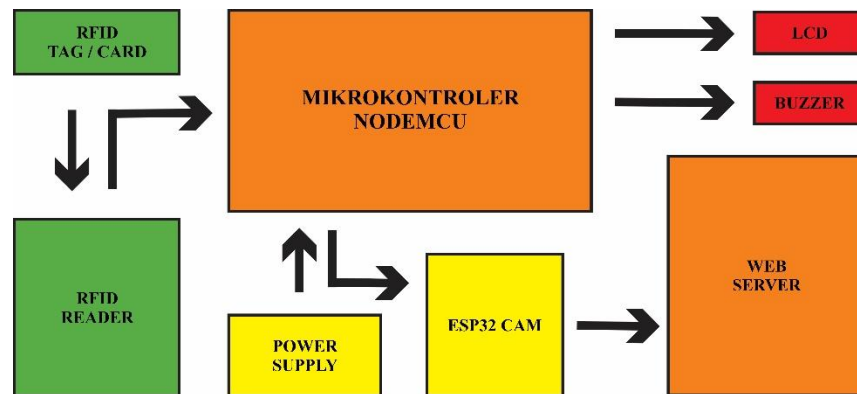
- a. *Input* berisi Kartu RFID yang dibaca oleh RFID Reader.
- b. *Process* berisi Data dari RFID Reader diproses oleh NodeMCU dan dikirim ke *server* melalui koneksi Wi-Fi.
- c. *Output* berisi data kehadiran ditampilkan pada LCD dan disimpan di *database server*.

B. Perancangan (*Design*)

Tujuannya untuk menyiapkan prototipe perangkat. Selanjutnya, Mulyatiningsih (2014:197) menjelaskan pada tahap perancangan peneliti sudah membuat produk awal (*prototype*) atau rancangan produk sehingga dapat dilanjutkan dalam uji coba.

a. Blok Diagram

Perancangan dan pembuatan sistem diawali dengan membuat blok diagram untuk memperjelas konsep sistem secara keseluruhan. Blok diagram tersebut ditunjukkan pada gambar berikut. Berdasarkan blok diagram yang telah dipaparkan dapat dijelaskan bahwasannya ada beberapa yang dibagi diantaranya yaitu *input*, *process* dan *output* seperti pada Gambar 12.



Gambar 12. Blok Diagram

1) *Input*

RFID *Reader* dan RFID *Tag/Card* adalah dua komponen utama dalam sistem RFID (*Radio Frequency Identification*). RFID *Reader* adalah perangkat yang mendekode informasi yang terkandung dalam *tag*. *Reader* memodulasinya dengan data, dan mengirim energi pada frekuensi tertentu. Pembaca RFID menghasilkan frekuensi radio tertentu. Saat *tag* RFID yang kompatibel memasuki jangkauan deteksi pembaca, *tag* mengirimkan sinyal balik. Informasi ini biasanya disimpan sebagai *string* data 96-bit yang disebut kode produk elektronik *engineering procurement construction (EPC)*.

Pembaca menggunakan algoritma kode koreksi kesalahan untuk menentukan akurasi EPC. Pada titik ini, pembaca meneruskan identifikasi *tag* ke pengguna sistem, *server* atau *database*, yang memperbarui data *tag* RFID sesuai kebutuhan. RFID *Tag* adalah semacam *barcode* pintar yang melekat pada barang untuk mengidentifikasinya dengan mudah setiap *tag* RFID memiliki antena

yang terhubung dengan IC RFID, sehingga dapat menerima, menyimpan, dan mengirimkan data melalui gelombang radio. *Tag* RFID biasanya menggunakan teknologi RFID pasif tanpa sumber daya *internal*. *Tag* RFID ditenagai untuk bekerja oleh energi elektromagnetik yang diterima dari pembaca RFID.

2) *Process*

Mikrokontroler merupakan sebuah sistem komputer yang seluruh atau sebagian besar elemennya dikemas dalam satu *chip* IC, sehingga sering disebut *single chip microcomputer*. Mikrokontroler merupakan sistem komputer yang mempunyai satu atau beberapa tugas yang sangat spesifik, berbeda dengan PC (*Personal Computer*) yang memiliki beragam fungsi. Perbedaan lainnya adalah perbandingan RAM (*Random Access Memory*) dan ROM (*Read Only Memory*) yang sangat berbeda antara komputer dengan mikrokontroler. Mikrokontroler adalah sebuah *System Microprocessor* dimana didalamnya sudah terdapat CPU, ROM, RAM, I/O, *Clock* dan peralatan *internal* lainnya yang sudah saling terhubung dan terorganisasi (teralamati) dengan baik oleh pabrik pembuatnya dan dikemas dalam satu *chip* yang siap pakai.

NodeMCU ESP8266 adalah sebuah platform IoT (*Internet of Things*) *open-source* dan juga berfungsi sebagai Mikrokontroler. NodeMCU ESP8266 merupakan Mikrokontroler yang telah dilengkapi dengan modul Wi-Fi ESP8266. Hal ini digunakan sebagai

pemroses dan pengiriman sinyal data yang diterima dari input yang telah dikonfigurasi.

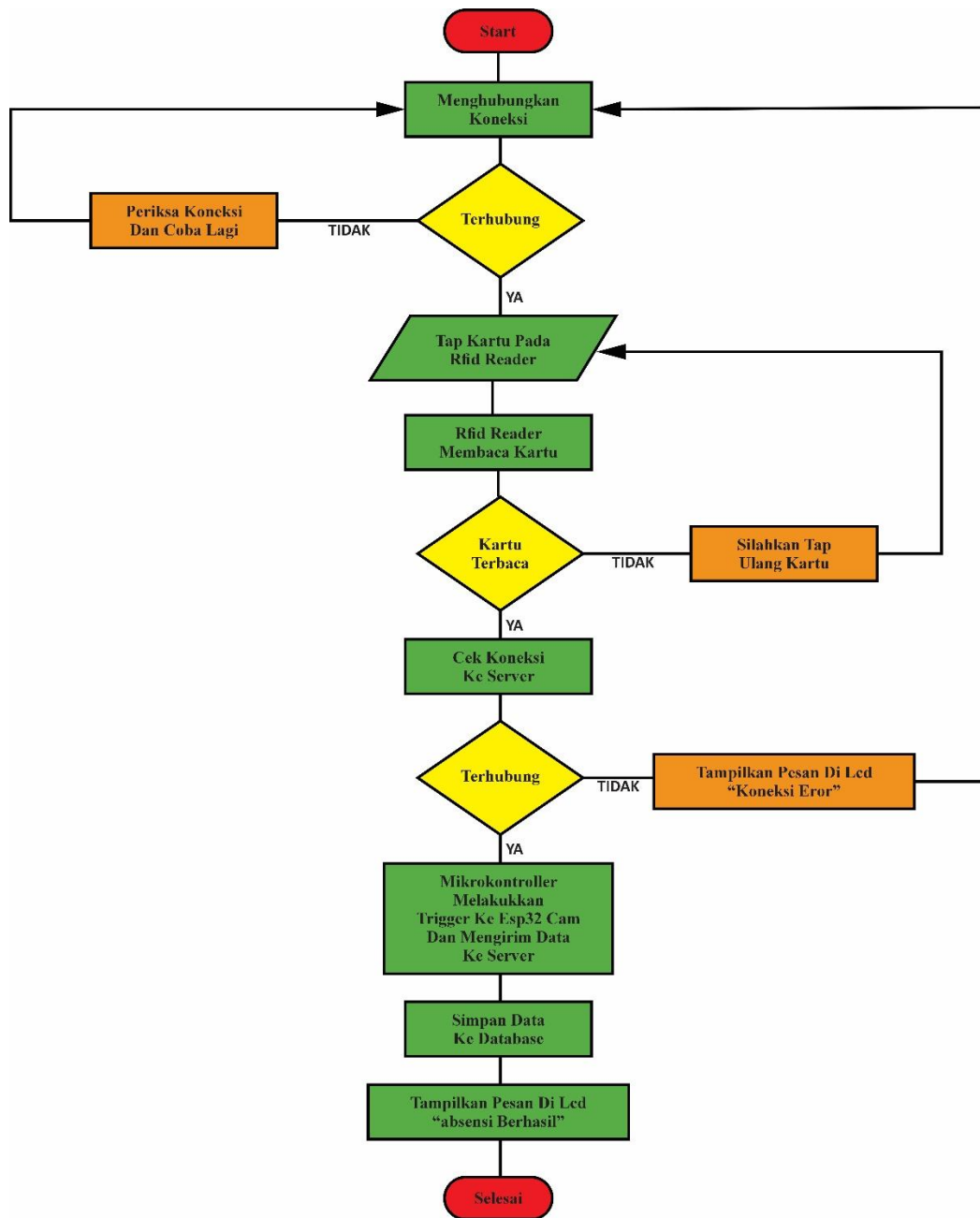
3) *Output*

PC Interface adalah batas yang dibagi antara dua atau lebih komponen terpisah dari sistem komputer untuk bertukar informasi, Sedangkan antarmuka perangkat lunak adalah program yang memungkinkan pengguna berinteraksi dengan komputer langsung. Sedangkan antarmuka perangkat lunak adalah program yang memungkinkan pengguna berinteraksi dengan komputer secara langsung atau melalui jaringan, MySQL digunakan dalam berbagai aplikasi, mulai dari pengelolaan hingga pengembangan aplikasi.

Hardware seperti LCD dan *Buzzer* penampil dan juga sebagai indikator notifikasi dari alat yang digunakan, tentunya dapat mempermudah dan dapat memberi informasi juga ada notifikasinya apabila terjadi data yang tidak sesuai dengan yang diharapkan.

b. *Flowchart*

Dalam merancang sistem, akan dijelaskan bagaimana cara kerja dari sistem alat ini. Dalam perancangan tersebut dijelaskan bagaimana NoceMCU ESP8266 dapat memproses data dari mana sistem *input* itu bekerja hingga *output*-nya tampil pada *database*. Untuk merancang sistem tersebut, terlebih dahulu akan dibuat *flowchart* seperti pada Gambar 13 untuk menggambarkan proses yang berlangsung pada sistem.



Gambar 13. *Flowchart* Proses Pada NodeMCU

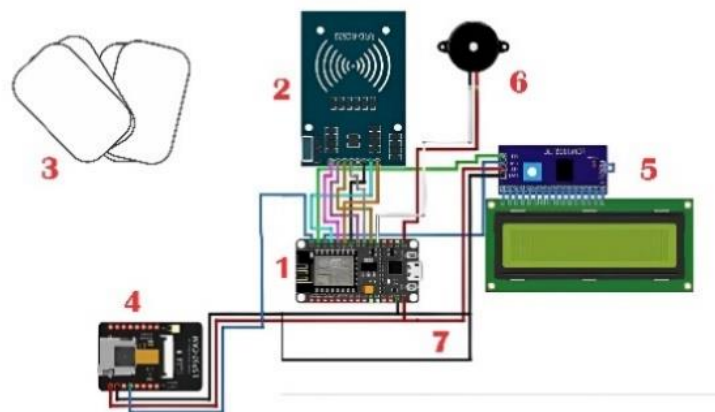
c. Perancangan Perangkat Keras

Perancangan perangkat keras sangat diperlukan untuk mengupayakan hasil yang maksimal dalam pembuatan alat, hal ini dapat dilakukan dengan memulai merancang *casing* perangkat keras seperti pada Gambar 14.

selain itu gambar skema alat juga sangat diperlukan ketika ingin menempatkan dan menyusun komponen sehingga menjadi satu seperti pada Gambar 15.



Gambar 14. Bentuk Desain *Casing Hardware*



Gambar 15. Bentuk Desain Skema Rangkaian

Berikut detail nama nama komponen yang digunakan seperti pada gambar diatas:

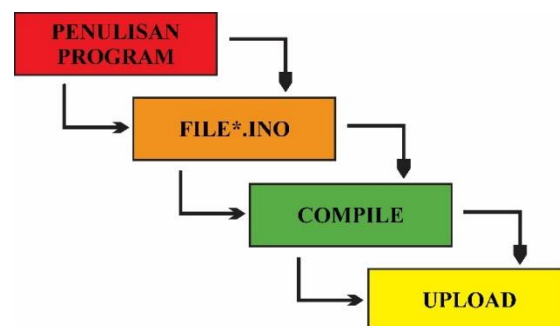
1. Mikrokontroler ini akan menjadi otak dari sistem. ESP8266

memiliki kemampuan Wi-Fi yang memungkinkan komunikasi dengan *server* atau *database*.

2. Sensor RFID MFRC522 akan digunakan untuk membaca data melalui kartu yang menggunakan teknologi RFID.
3. RFID Card ini dirancang untuk menyimpan dan mentransmisikan data secara *nirkabel*, menjadikannya alat yang nyaman dan efisien untuk berbagai aplikasi.
4. ESP32 CAM ialah mikrokontroler ESP32-S dan kamera OV2640, yang memungkinkan pengguna untuk mengambil gambar dan video, berfungsi sebagai *server web* mandiri, mendukung konektivitas Wi-Fi dan Bluetooth, memiliki slot *MicroSD*, dan dapat digunakan untuk implementasi pengenalan wajah
5. *Module I2C* digunakan untuk menghubungkan LCD 16x2 dengan Mikrokontroler, LCD ini akan menampilkan kondisi alat, baik sebelum melakukan *scan* kartu maupun setelahnya.
6. *Buzzer* akan memberikan indikator suara bahwa telah selesai melakukan *scan* kartu dan datanya telah dipublish ke *database* PHP MySQL.
7. *Jumper* atau kabel penghubung memiliki fungsi yaitu menghubungkan dua titik dalam rangkaian listrik atau elektronik.
8. *Power Supply* ini akan memberikan tegangan ke mikrokontroler ESP8266 dan semua komponen yang membutuhkannya.

d. Perancangan *Website*

Program merupakan hal terpenting dalam pembuatan sebuah sistem kendali yang menggunakan mikrokontroler. Proses perancangan program pada mikrokontroler. Penulisan program bahasa C/C++ dilakukan menggunakan aplikasi Arduino IDE. *Listing* program yang telah buat nantinya akan tersimpan dalam format *.ino. Selanjutnya lakukan proses kompilasi program untuk mengecek kesalahan pada baris program. Kemudian apabila tidak mendapati kesalahan dalam penulisan program maka program dapat di *upload* sesuai *board* mikrokontroler yang digunakan. Setelah pembuatan program selesai dilanjutkan pembuatan *website* sebagai *database* pada sistem informasi absensi tersebut.



Gambar 16. Proses Perancangan *Website*

Website selanjutnya dirancang dengan baik seperti alur pada Gambar 16 sehingga dapat mempermudah dalam melakukan pengaksesan dan lainnya, sistem *website* sendiri berada dalam satu kesatuan yang utuh, sehingga dapat menjadikan sebuah kesatuan yang baik dan juga bagus, perancangan ini memerlukan beberapa bagian diantaranya:

- 1) Desain antarmuka kita perlu merancang tata letak dan desain antarmuka *website*. Ini termasuk pemilihan warna, *font*, dan elemen-

elemen *visual* lainnya. Desain yang baik akan memastikan pengguna dapat dengan mudah berinteraksi dengan sistem.

- 2) Pemrograman *Front-End*, kita perlu mengimplementasikan antarmuka menggunakan bahasa pemrograman *web* seperti HTML, CSS, dan *JavaScript*. Ini akan mencakup pembuatan halaman-halaman yang menampilkan informasi kehadiran, tombol absensi, dan fitur-fitur lainnya.
- 3) Pengembangan *Back-End*, kita perlu mengembangkan logika bisnis dan menghubungkan *website* dengan *database*. ESP8266 akan berkomunikasi dengan *server* melalui API atau protokol lainnya. Data akan disimpan dan dikelola di *database*.
- 4) *Website* harus dapat berkomunikasi dengan ESP8266 melalui protokol HTTP atau MQTT. Ketika melakukan absensi, ESP8266 akan mengirimkan data kehadiran ke *server* melalui *website*.
- 5) Keamanan data harus menjadi prioritas utama. Kita perlu mengimplementasikan *enkripsi* data dan mengamankan akses ke *website* dan *database*.

C. Pengembangan (*Develop*)

Tujuan tahap ini yaitu untuk menghasilkan proyek yang sudah direvisi berdasarkan masukan dari para pakar. Tahap pengembangan bertujuan untuk menghasilkan prototipe sistem yang dapat diuji dan disempurnakan.

1. Simulasi Sistem

- a. Komponen diuji satu per satu untuk memastikan setiap bagian berfungsi dengan baik.
- b. RFID *Reader* diuji untuk membaca berbagai jenis kartu RFID.
- c. NodeMCU diuji untuk memastikan koneksi dengan Wi-Fi dan pengiriman data ke *server*.

2. Uji Coba Terbatas

- a. Sistem diuji dengan skenario terbatas untuk memastikan integrasi perangkat keras dan perangkat lunak berjalan lancar.
- b. Pengujian meliputi:
 - 1) Pembacaan kartu RFID dan pengiriman data ke *server*.
 - 2) Tampilan data pada LCD.
 - 3) Penyimpanan data absensi di *database*.

D. Penyebaran (*Disseminate*)

Tahap ini melibatkan pengujian dan implementasi sistem dalam skala yang lebih luas.

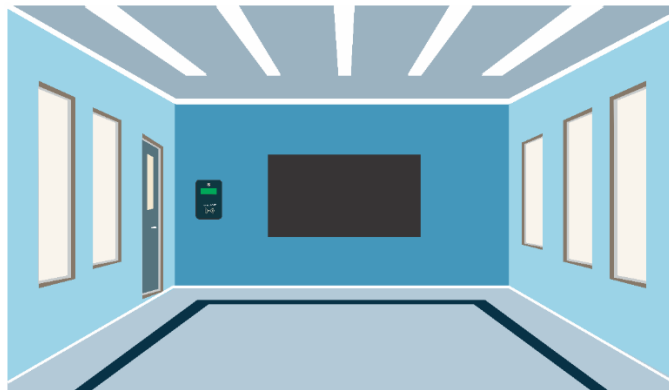
1. Validasi Sistem

- a. Sistem diuji dalam berbagai kondisi jaringan Wi-Fi untuk memastikan koneksi tetap stabil.
- b. Data absensi diuji untuk memastikan akurasi dan kecepatan pengiriman ke *server*.

2. Uji Keandalan

- a. Sistem diuji dengan beban kerja untuk mengevaluasi kinerjanya.

- b. Pengujian dilakukan untuk memastikan sistem dapat menangani banyak pengguna secara simultan tanpa masalah.
- 3. Dokumentasi, Penyebaran dan Penempatan
 - a. Dokumentasi sistem disiapkan untuk memudahkan pengguna memahami cara kerja dan pemeliharaan sistem.
 - b. Sistem diterapkan di lokasi yang membutuhkan absensi otomatis, seperti sekolah, kantor, atau institusi lainnya.
 - c. Penempatan alat sangat mempengaruhi keindahan ruang sehingga diperlukan bagian untuk menempatkan alat yang mudah di akses dan stabil tentunya seperti pada Gambar 17.



Gambar 17. Gambar Penempatan Alat

BAB IV

HASIL DAN PEMBAHASAN

Setelah alat selesai dibuat, alat yang dirancang diuji baik dari segi *Software* maupun *Hardware* untuk menentukan keberhasilan alat yang dirancang serta membandingkan dengan spesifikasi yang diinginkan. Selama proses pembuatan rangkaian elektronik, semua perangkat keras yang digunakan perlu diuji terlebih dahulu sebelum dapat dihubungkan untuk membentuk suatu sistem yang utuh.

Pengujian perangkat keras dilakukan untuk memastikan bahwa perangkat keras tersebut berfungsi dengan baik. Sensor RFID diuji, Koneksi NodeMCU ESP8266 diuji sebagai bagian dari pengujian ini. keseluruhan dalam hal kinerja dan hasil. Alat dibuat sesuai dengan perancangan sistem. Desain pembuatan alat diatur sedemikian rupa sehingga tidak terjadi kesalahan pada saat mengukurnya.

A. Hasil Pembuatan Alat

1. Bentuk *Box* Alat

Adalah bagian terluar dari alat yang dibuat, bagian ini terdiri dari bahan akrilik yang sudah didesain dan dibuat sedemikian rupa, sehingga bentuk yang dihasilkan sesuai dengan kegunaan, selain itu bagian ini juga dilapisi stiker agar dapat lebih tahan lama dari paparan sinar matahari atau jamur dan lainnya, seperti pada Gambar 18 dibawah ini.



Gambar 18. Tampilan *Box* dari Depan

2. Posisi RFID-RC522

Posisi RFID ini menggambarkan bentuk dari RFID tersebut, hal ini bertujuan agar komponen dapat disusun secara teratur dan rapi, hal ini membuat tentu menjadi nilai tambah dalam melakukan efisiensi. Seperti pada Gambar 19 dibawah ini.



Gambar 19. Posisi Letak RFID-RC522

3. Posisi Mikrokontroler NodeMCU ESP8266

Mikrokontroler adalah bagian paling penting dalam hal ini sehingga posisi yang ditentukan juga harus sesuai karena hal ini dapat mempengaruhi bagaimana efisiensi ruang dapat tersusun dan terisi dengan baik seperti pada Gambar 20 dibawah ini.



Gambar 20. Posisi Letak NodeMCU Esp8266

4. Posisi LCD 16x2

LCD merupakan bagian paling baik dalam memberikan informasi karena sumber teks yang dihasilkan dapat memberitahukan bahwasannya informasi seperti apa yang sedang digunakan, sehingga posisi penempatan LCD harus baik dan tepat sehingga informasi yang dihasilkan dapat dibaca dengan baik seperti pada Gambar 21 dibawah ini.



Gambar 21. Posisi Letak LCD dalam *Box*

B. Pengujian Perangkat Keras dan Perangkat Lunak

Pengujian Perangkat ini dilakukan untuk melihat hasil dari beberapa tes yang dilakukan untuk memastikan komponen serta bahan yang digunakan berjalan dengan baik, setelah itu dilakukannya penyesuaian sesuai kebutuhan perangkat yang akan digunakan. Hal ini bertujuan untuk menguji apakah perangkat yang digunakan telah sesuai yang dibutuhkan dalam pengoperasian alat. Selain itu, pengujian ini juga bertujuan untuk memastikan apakah perangkat bekerja dengan baik serta tidak terdapat kerusakan atau eror yang terjadi saat perangkat itu digunakan nantinya.

Untuk pengujian perangkat keras, dilakukan untuk memastikan bahwa perangkat keras yang akan digunakan pada sistem berkerja dengan baik.

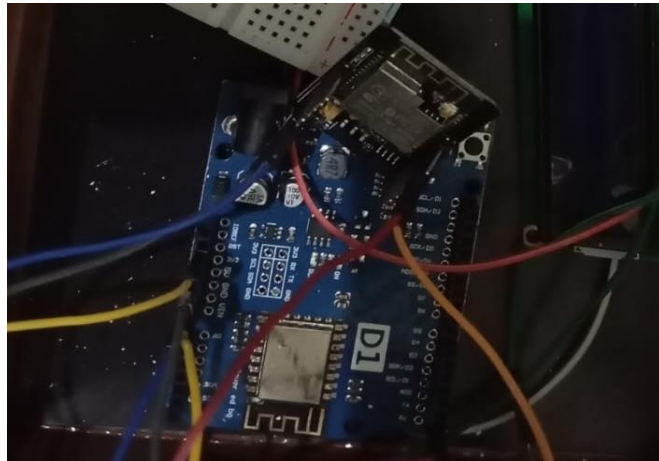
1. Pengujian Perangkat Keras

a. Pengujian ESP32 *Cam*

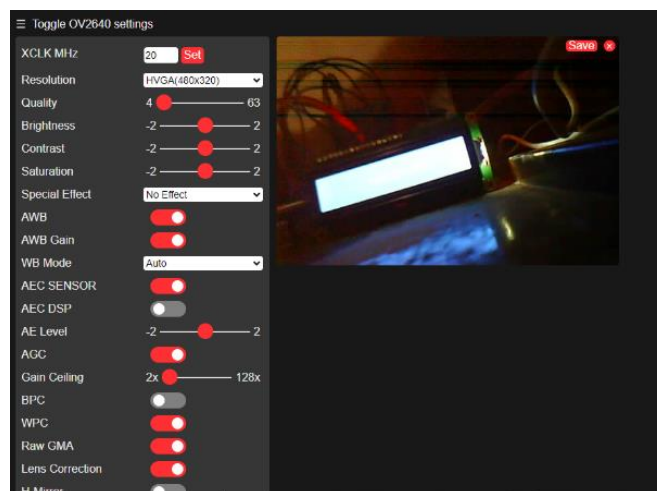
Pengujian ESP32-*Cam* bertujuan untuk memastikan bahwa modul tersebut berfungsi dengan baik dan dapat memenuhi kebutuhan aplikasi yang direncanakan. Proses pengujian ini mencakup berbagai aspek, termasuk verifikasi konektivitas jaringan, kualitas gambar yang dihasilkan, responsivitas kamera terhadap perintah, serta kestabilan kinerja dalam kondisi operasi yang berbeda. Selain itu, pengujian juga dilakukan untuk memastikan bahwa modul ESP32-*Cam* dapat berintegrasi dengan komponen lain dalam sistem secara efektif.

Proses pengujiannya yaitu dengan menghubungkan dengan mikrokontroler seperti pada Gambar 22 dibawah ini dah hasilnya

dengan memanfaatkan IP Address sehingga gambar dapat dimunculkan pada monitor seperti pada Gambar 23 untuk memastikan apakah sistem bekerja dengan baik atau tidak.



Gambar 22. Proses Pengujian *Camera ESP32 Cam*

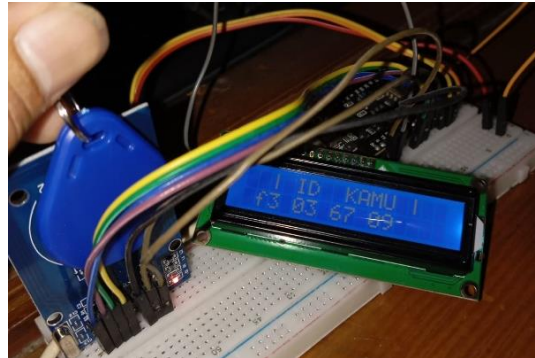


Gambar 23. Hasil gambar dari *Web Server ESP32 Cam*

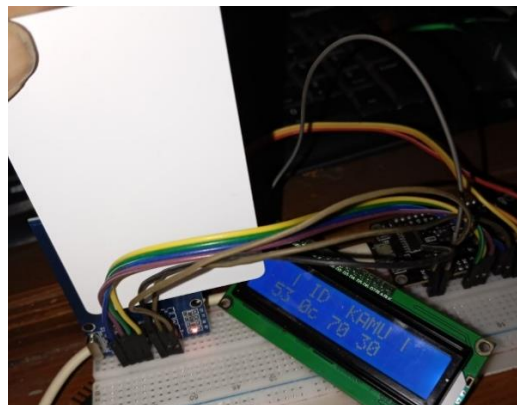
b. Pengujian Sensor RFID-RC522

Tujuan dari pengujian sensor RFID-RC522 adalah untuk menilai kinerja pembacaan kartu secara menyeluruh. Pengujian ini akan memastikan bahwa sensor RFID-RC522 dapat membaca dan mengidentifikasi berbagai jenis kartu RFID dengan akurat dan cepat.

Selain itu, pengujian ini bertujuan untuk mengukur jarak pembacaan optimal di mana sensor masih dapat berfungsi dengan baik. Sebagaimana pada Gambar 24 pengujian dan hasil dari RFID *tag* dan pada Gambar 25 pengujian dan hasil dari RFID *Card*.



Gambar 24. Pengujian RFID *Tag* dengan RFID *Reader*



Gambar 25. Pengujian RFID *Card* dengan RFID *Reader*

c. Pengujian LCD

Tujuan dari pengujian LCD I2C 16x2 adalah untuk memastikan bahwa layar LCD ini berfungsi dengan baik dan dapat digunakan secara optimal dalam berbagai aplikasi. Pengujian ini bertujuan untuk menilai kualitas tampilan karakter pada layar, memastikan bahwa setiap karakter ditampilkan dengan jelas dan tanpa *distorsi*.

Seperti pada Gambar 26 dibawah ini uji coba LCD berhasil yakni dengan tampilnya huruf yang diinputkan sehingga dapat hasil *output* yang sesuai pada LCD.

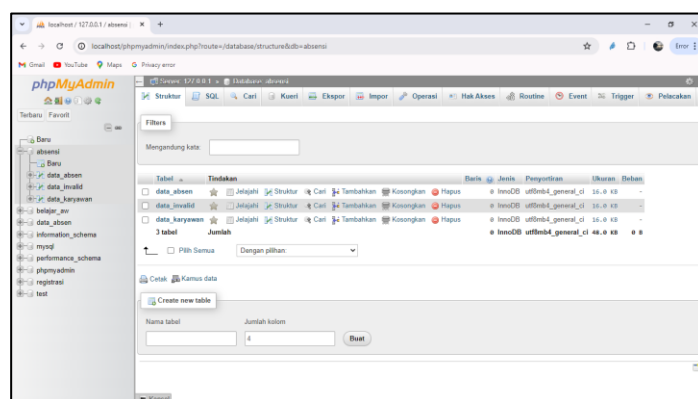


Gambar 26. Menampilkan Teks Pada LCD

2. Pengujian Perangkat Lunak

a. Pengujian *Database* Absensi

Tujuan dari pengujian *database* absensi adalah untuk memastikan bahwa sistem ini dapat menyimpan, mengelola, dan mengakses data absensi dengan akurat dan efisien. Pengujian ini bertujuan untuk menilai integritas data, memastikan bahwa setiap *entri* data absensi disimpan dengan benar dan tidak mengalami korupsi seperti pada Gambar 28.



Gambar 27. Pengujian *Database* Absensi

b. Pengujian *Database* Kamera

Tujuan dari pengujian *database* kamera adalah untuk memastikan bahwa sistem basis data yang digunakan untuk menyimpan dan mengelola data gambar dari kamera berfungsi dengan optimal seperti pada Gambar 29. Pengujian ini bertujuan untuk menilai keandalan dalam menyimpan dan mengakses data gambar atau video, memastikan bahwa tidak ada data yang hilang atau korup selama proses penyimpanan dan pengambilan.



Gambar 28. Pengujian *Database Camera*

c. Pengujian *IP Address*

Tujuan dari pengujian *IP Address* adalah untuk memastikan bahwa sistem jaringan dapat mengelola dan mengalokasikan alamat IP dengan benar dan efisien. Pengujian ini bertujuan untuk menilai keakuratan dalam penetapan alamat IP *statis* dan *dinamis*, memastikan bahwa setiap perangkat yang terhubung ke jaringan menerima alamat IP yang unik dan tidak ada konflik alamat yang terjadi seperti pada Gambar 30 dibawah. Pengujian IP ini dilakukan

menggunakan CMD pada komputer agar IP yang digunakan dapat terhubung secara langsung.

```
Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . : 
Wireless LAN adapter Wi-Fi 3:
Connection-specific DNS Suffix . : 
Link-local IPv6 Address . . . . . : fe80::22ae:fadd:b4ca:7689%9
IPv4 Address. . . . . : 192.168.1.12
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : 192.168.1.1
Ethernet adapter Bluetooth Network Connection 2:
Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . : 
C:\Users\LENOVO>
```

Gambar 29. Pengujian IP Perangkat

C. Pengujian Pengiriman Data Ke *WebServer*

1. Pengujian Koneksi Modul Wi-Fi NodeMCU ESP8266

Tujuan dari pengujian koneksi modul Wi-Fi NodeMCU ESP8266 adalah untuk memastikan bahwa modul ini dapat terhubung ke jaringan Wi-Fi dengan benar dan berfungsi secara optimal dalam berbagai kondisi. Pengujian ini bertujuan untuk menilai kecepatan dan keandalan koneksi Wi-Fi yang dihasilkan, memastikan bahwa NodeMCU ESP8266 dapat melakukan koneksi ke jaringan Wi-Fi secara cepat dan tetap stabil tanpa mengalami putus sambungan seperti pada Gambar 31.

```
//inisialisasi LED
LCD_I2C lcd(0x27, 16, 2);

#define RST_PIN      16 //pin-D0
#define SS_PIN       0  //pin-D3
byte buzzer = 2;      //pin D8
MFRC522 mfrc522(SS_PIN, RST_PIN); // Create MFRC522 instance

const char* ssid = "c"; //Your Wifi SSID
const char* password = "10102001"; //Wifi Password
String server_addr = "192.168.30.128"; //your server address or computer IP

byte readCard[4];
uint8_t successRead;
String UIDCard;
```

Gambar 30. Pengujian Koneksi Wi-Fi

Setelah Program di *Upload* maka akan memunculkan keterangan *Connected* pada LCD, setelah itu koneksi perangkat siap digunakan dan siap untuk melakukan *transfer* data dari *Hardware* ke *Web Server*.

2. Pengujian Data Terkirim ke *Web Server*

Tujuan dari pengujian data yang terkirim ke *web server* adalah untuk memastikan bahwa data dapat dikirim dari perangkat ke *server* dengan benar dan efisien seperti pada Gambar 32. Pengujian ini bertujuan untuk menilai keakuratan data yang dikirim, memastikan bahwa data yang diterima oleh *server* sesuai dengan data yang dikirim oleh perangkat tanpa ada perubahan atau kerusakan.

```
WiFiClient client;
String address, message, first_name;

//equate with your Server address (computer's IP address) and your directory application
address ="http://"+server_addr+"/absensi/webapi/api/create.php?uid="+UIDCard;

HTTPClient http;
//http.begin(address);
http.begin(client,address);
int httpCode = http.GET();           //Send the GET request
String payload;
Serial.print("Response: ");
```

Gambar 31. Pengujian Data Terkirim

Jika sudah melakukan *Upload* Program maka akan keluar data data yang di ujicobakan pada sebuah *database*, seperti halnya pengujian *web server* sebelumnya, maka IP yang telah dimasukkan akan menjadi wadah lokasi pengiriman data tersebut.

D. Pengujian Keseluruhan Sistem

Tujuan dari pengujian keseluruhan sistem adalah untuk memastikan bahwa semua komponen sistem bekerja secara terintegrasi dan berfungsi sesuai dengan spesifikasi yang diharapkan. Pengujian ini mencakup pengujian fungsi

dan kinerja dari seluruh sistem secara menyeluruh, dari awal (*input*) hingga akhir (*output*), termasuk interaksi antara komponen-komponen yang berbeda.

Pengujian ini akan menilai integrasi perangkat keras (*hardware*) dan perangkat lunak (*software*), memastikan bahwa setiap komponen dapat berkomunikasi dan beroperasi bersama tanpa adanya konflik atau ketidakcocokan yang dapat mengganggu kinerja sistem secara keseluruhan. Pengujian akan mencakup pengujian fungsionalitas masing-masing komponen secara individual dan kemudian pengujian integrasi untuk memastikan bahwa semua fungsi tersebut berjalan dengan baik dalam kondisi operasional yang sebenarnya.

Selain itu, pengujian keseluruhan sistem juga akan mengevaluasi keandalan sistem dalam berbagai situasi dan kondisi yang mungkin terjadi, termasuk uji keandalan, ketahanan terhadap beban kerja yang tinggi, serta pengujian kesalahan (*error*). Pengujian performa juga akan dilakukan untuk menilai *respons* sistem terhadap situasi yang mengharuskan pengolahan data dalam waktu nyata, serta kemampuan sistem dalam menangani jumlah pengguna atau permintaan yang tinggi secara simultan.

Pengujian keseluruhan sistem bertujuan untuk memastikan bahwa sistem dapat beroperasi dengan efektif, efisien, dan dapat diandalkan dalam memenuhi tujuan dan kebutuhan yang diinginkan.

E. Pembahasan

1. Desain Perangkat Keras

Tahap pertama dalam implementasi sistem absensi berbasis ESP8266 adalah desain perangkat keras. Di sini, komponen utama sistem terdiri dari beberapa elemen, di mana modul ESP8266 berfungsi sebagai jantung dari sistem. Pemilihan ESP8266 tidak hanya didasarkan pada ketersediaan dan biaya, tetapi juga pada kemampuannya dalam mengelola konektivitas *nirkabel*. S. Sunardi et al. (2024) dalam penelitiannya mengungkapkan bahwa sistem kendali kehadiran dapat dioptimalkan dengan mengintegrasikan teknologi berbasis *web* dengan *microcontroller* dan *reader* RFID. Ini menunjukkan bahwa ESP8266, sebagai modul WiFi, memberikan fleksibilitas yang tinggi untuk komunikasi data ke *server*, mempercepat proses absensi yang sebelumnya manual menjadi otomatis.

Penggunaan RFID reader dalam sistem ini bertujuan untuk membaca kartu identifikasi yang digunakan oleh para pengguna. RFID memiliki keuntungan signifikan dibandingkan metode lain, seperti pembacaan barcode yang memerlukan garis pandang langsung. Sistem RFID yang dirancang memungkinkan pembacaan yang lebih cepat dan efisien di mana saja dalam jarak yang ditentukan, yakni 3-4 cm, sesuai temuan oleh Kukuh Prasetyo Aji et al. (2020). Ini menjadi *essential* dalam lingkungan di mana kecepatan dan akurasi pencatatan kehadiran adalah prioritas.

Rangkaian perangkat keras dirancang sedemikian rupa untuk memastikan keterintegrasian antara komponen. Misalnya, modul ESP8266 dihubungkan ke RFID reader melalui pin GPIO, yang memungkinkan komunikasi antara perangkat. Selain itu, manajemen daya harus

diperhatikan dengan melakukan konversi tegangan dari 5V menjadi 3.3V untuk menjaga kompatibilitas dengan modul ESP8266, yang sangat sensitif terhadap perubahan tegangan. Penambahan indikator LED untuk memberikan umpan balik visual kepada pengguna juga sangat penting, di mana pengguna dapat dengan jelas mengetahui apakah kartu mereka telah terbaca dengan sukses atau tidak.

2. Pengembangan Perangkat Lunak

Setelah fase desain perangkat keras selesai, fokus beralih ke pengembangan perangkat lunak, yang mengontrol keseluruhan operasi sistem. Perangkat lunak ini dikembangkan menggunakan Arduino IDE, *platform* yang khusus untuk pemrograman modul seperti ESP8266. Di sinilah hasil penelitian Ryan Hidayat et al. (2022) menjadi relevan, di mana dalam studi mereka menunjukkan bahwa sistem yang dibangun dapat mendeteksi kartu RFID dengan baik dan menyimpan data yang dikumpulkan ke dalam database, mengoptimalkan proses absensi. Hal ini menggambarkan bahwa perangkat lunak memiliki peran kunci dalam mencapai tujuan sistem, termasuk koneksi ke *server* dan proses pengiriman data.

Fokus utama perangkat lunak adalah memastikan bahwa semua fungsi utama seperti pembacaan data RFID, pengiriman data ke *server* melalui protokol HTTP POST, dan pengelolaan konektivitas *nirkabel* dapat berjalan lancar. Ketika kartu RFID dibaca, data ID disimpan sementara sebelum dikirim ke *server*, sehingga sistem mampu mencatat absensi

secara real-time a feature that set this system apart from traditional methods. Keunggulan ini sejalan dengan temuan Ery Setyawan et al. (2021) yang menekankan bahwa sistem berbasis IoT dapat merekam kehadiran dengan validasi yang lebih baik melalui pengambilan gambar wajah, yang bisa digabungkan dengan pembacaan RFID untuk validasi ganda.

3. Integrasi antara Perangkat Keras dan Perangkat Lunak

Integrasi antara perangkat keras dan perangkat lunak adalah langkah kritical berikutnya untuk memastikan sistem berfungsi secara kolaboratif. Pengujian koneksi ESP8266 dengan *server* melalui jaringan WiFi menjadi prioritas guna memastikan bahwa data yang dikirim dari perangkat ke *server* sampai dengan baik tanpa mengalami gangguan. Penelitian Kukuh Prasetyo Aji et al. (2020) menunjukkan pentingnya keandalan ini, di mana banyak jenis kartu RFID dapat terbaca oleh sistem dengan efisiensi yang baik.

Pengujian juga perlu dilakukan terhadap proses pembacaan RFID untuk memastikan sistem dapat mengenali setiap kartu dengan akurasi tinggi. Sebagaimana yang dibuktikan oleh Dian Ariyanto (2024), hasil pengujian menunjukkan bahwa batas maksimal pembacaan RFID pada jarak 4.0 cm adalah indikasi kinerja yang baik. Dengan pengujian yang melibatkan berbagai kondisi jaringan, menunjukkan bahwa ESP8266 dapat berfungsi dengan baik bahkan pada jarak yang cukup jauh dari *router* WiFi, dengan latensi pengiriman data yang masih dalam batas toleransi.

BAB V

PENUTUP

A. Kesimpulan

Penelitian ini berhasil merancang dan membangun sistem informasi absensi berbasis ESP8266 dengan memanfaatkan teknologi RFID dan IoT (*Internet of Things*). Sistem ini mampu meningkatkan efisiensi dan akurasi pencatatan kehadiran dibandingkan metode manual, yang sering kali memakan waktu dan rawan kesalahan. Dengan integrasi modul ESP8266, RFID Reader, ESP32 CAM, serta antarmuka *web*, sistem ini memungkinkan pengelolaan absensi secara *real-time*, pengambilan data yang akurat, dan pemantauan yang mudah. Hasil pengujian menunjukkan bahwa sistem bekerja secara efektif, mampu membaca kartu RFID dengan cepat, serta menyediakan fitur identifikasi tambahan melalui kamera ESP32 CAM. Keunggulan ini menjadikan sistem absensi berbasis ESP8266 sebagai solusi inovatif dan praktis untuk diterapkan dalam berbagai institusi untuk dikembangkan lebih lanjut guna meningkatkan fungsionalitas dan skalabilitasnya.

B. Saran

1. Pastikan jaringan internet yang digunakan stabil dan memiliki jangkauan yang cukup untuk mendukung pengiriman data *real-time*.
2. Lakukan pengujian secara menyeluruh untuk memastikan sistem berfungsi dengan baik dan lakukan pemeliharaan.
3. Lakukan Pembelajaran untuk memahami sistem ini agar dapat bekerja secara optimal

DAFTAR PUSTAKA

- Agrawal, A. &. (2013). nline attendance management system using RFID with object counter. *International Journal of Information and Computation Technology*, 3(3), 131-138.
- Aji, K. P. (2020). perancangan sistem presensi untuk pegawai dengan rfid berbasis IOT menggunakan nodeMCU ESP8266. *JOINTECS (Journal of Information Technology and Computer Science)*, 25.
- Al-Taai, S. H.-D. (2023). The importance of using the internet of things in education. *Int. J. Emerg. Technol. Learn.*, 18(1), 19-39.
- Aminah, S. H. (2021). PPERANCANGAN ALAT ABSENSI MAHASISWA BERDASARKAN MATA KULIAH MENGGUNAKAN E-KTP BERBASIS NODEMCU. *JUTSI: Jurnal Teknologi dan Sistem Informasi*, 103-110.
- Anggraini, A. P. (2022). Rancang Bangun Sistem Absensi Menggunakan Id Card Dengan Pengiriman Data Via Telegram Berbasis Nodemcu Esp8266 Menggunakan Metode Fuzzy Logic. *JTIK (Jurnal Teknik Informatika Kaputama)*, 587-596.
- Anri, I. (2023). Perancangan Alat Absensi Guru dan Doorlock System Menggunakan RFID pada NodeMCU ESP8266 Berbasis Web.
- Ariyanto, D. (2024). RANCANG BANGUN SISTEM PENCATAT KEHADIRAN DI LABORATORIUM MENGGUNAKAN RFID DAN ESP32-CAM BERBASIS IOT. *Technologia: Jurnal Ilmiah*, 205-214.
- Cahyana, A. &. (2023). APLIKASI PRESENSI BERBASIS INTERNET OF THINGS UNTUK MENGAKTIFKAN KAMERA PADA RASPBERRY PI. *Telematika MKOM*, 75-85.
- Daffa, R. &. (2023). Perancangan Sistem Absensi Dengan RFID Berbasis IoT Menggunakan NODEMCU ESP8266 Terhadap PT Halal Fresh Indonesia. *Jurnal Ilmu Komputer, Teknik dan Multimedia*, 1175-1190.
- El Mrabet, H. &. (2020). IoT-school attendance system using RFID technology. *International Association of Online Engineering*.
- Fadilah, G. A. (2023). Rancang Bangun Sistem Absensi Siswa Berbasis Mikrokontroller Esp8266 Dan Web Menggunakan RFID. *Voteteknika (Vocational Teknik Elektronika dan Informatika)*, 11(4).

- Ferdiansyah, P. S. (2022). Rancang Bangun Absensi Mahasiswa Menggunakan RFID dan Sensor DS18B20 Berbasis NodeMCU Di Universitas Darul Ulum. *Jurnal Sains dan Teknologi (JSIT)*, 158-164.
- Gunawan, E. (2022). RANCANG BANGUN ALAT PRESENSI MAHASISWA BIMBINGAN LAPORAN AKHIR DENGAN RADIO FREQUENCY IDENTIFICATION (RFID) DAN KAMERA ESP-32 BERBASIS INTERNET OF THINGS (IOT). *Doctoral dissertation, Politeknik Negeri Sriwijaya*.
- Hidayat, R. (2022). Rancang Bangun Alat Absensi Karyawan menggunakan RFID dan ESP32Cam Berbasis Internet of Things (Doctoral dissertation. *ITN MALANG*.
- HL, N. I. (2023). Developing Teaching Material of Research Methodology and Learning with 4D Model in Facilitating Learning During the Covid-19 Pandemic to Improve Critical Thinking Skill. *Jurnal Kependidikan: Jurnal Hasil Penelitian Dan Kajian Kepustakaan Di Bidang Pendidikan, Pengajaran Dan Pembelajaran*, 9(2), 541-554.
- Huda, Q. F. (2022). PROTOTYPE SISTEM ABSENSI MENGGUNAKAN RFID BERBASIS IOT. *Jurnal Publikasi Ilmu Komputer dan Multimedia*, 87-90.
- Ilham, R. (2023). PERANCANGAN SISTEM ABSENSI SIDIK JARI BERBASIS ARDUINO ESP8266 DAN WEB. *UNDERGRADUATE RESEARCH, FACULTY OF INDUSTRIAL TECHNOLOGY, BUNG HATTA UNIVERSITY*, 32-32.
- Khang, A. A. ((Eds.). (2024)). *Advanced IoT technologies and applications in the industry 4.0 digital economy*. CRC Press.
- Kurnialensya, T. &. (2023). Absensi Sistem Monitoring Kehadiran Siswa Menggunakan Mikrokontroller Berbasis Web. *Jurnal Teknologi dan Sistem Informasi Univrab*, 92-99.
- Maheswari, B. U. (2023). Internet of things and machine learning-integrated smart robotics. *Internet of things and machine learning-integrated smart robotics*, 240-258.
- Nasution, S. (2018). Presensi Online menggunakan RFID pada kartu mahasiswa. *Intecom: Journal of Information Technology and Computer Science*, 1(1), 19-27.
- Oktaviona, R. &. (2023). Pengembangan Media Pembelajaran Menggunakan AR Assemblr Edu Pada Mata Pelajaran Penerapan Rangkaian Elektronika. *Votetknika (Vocational Teknik Elektronika Dan Informatika)*, 11(2), 178-186.

- P., R. H. (2009). Overview of RFID technology and its applications in the food industry. *Journal of food science*, 74(8), R101-R106.
- Parihar, Y. S. (2019). Internet of things and nodemcu. *journal of emerging technologies and innovative research*, 6(6), 1085.
- Patel, R. P. (2012). Online students' attendance monitoring system in classroom using radio frequency identification technology: a proposed system framework. *International journal of emerging technology and advanced engineering*, 2(2), 61-66.
- Permana, B. A. (2023). Penerapan Sistem Absensi Siswa Menggunakan Teknologi Internet Of Things. *Jurnal informatika dan Teknologi*, 170-176.
- Purnomo, M. A. (2022). Implementasi Sistem Absensi Pegawai Menggunakan RFID E-Ktp Dan Kamera Berbasis Website. *Jurnal Teknologi Terkini*, 2(11).
- Putra, F. N. (2021). Rancang Bangun Sistem Presensi Online dengan RFID Berbasis Internet Of Things (IoT) di Universitas Nahdlatul Ulama Blitar. *Jurnal Riset dan Konseptual*, 216-224.
- Setyawan, E. D. (2021). Sistem Alat Absensi Menggunakan RFID dan Kamera Berbasis Internet of Things: Sistem Alat Absensi Menggunakan RFID dan Camera Berbasis Internet of Things. *Jurnal Ilmiah Fakultas Teknik*, 123-129.
- Sukmaulidian, A. S. (2024). EKSPLORASI MATERIAL ALUMUNIUM DAN TEMBAGA UNTUK CASING KOMPUTER MICRO-ATX. *eProceedings of Art & Design*, 11(1).
- Sunardi, S. A. (2024). Perancangan Dan Implementasi Sistem Absensi Karyawan Berbasis RFID dan Web Server. In *SISITI: Seminar Ilmiah Sistem Informasi dan Teknologi Informasi*. 104-110.
- Sutikno, T. P. (2021). Internet of things-based photovoltaics parameter monitoring system using NodeMCU ESP8266. *International Journal of Electrical and Computer Engineering*, 11(6), 5578-5587.
- Tansir, F. A. (2021). Pengembangan Sistem Kehadiran Karyawan Paruh Waktu Berbasis Rfid. *Jurnal Teknik dan Sistem Komputer*, 40-52.
- Tansir, F. A. (2022). Pengembangan Sistem Kehadiran Karyawan Paruh Waktu Berbasis Rfid (Studi Kasus: Pizza Hut Antasari, Lampung). *Jurnal Teknik dan Sistem Komputer*, 2(2), 40-52.

LAMPIRAN

Lampiran 1. Program Keseluruhan Esp8266

```
#include <ESP8266WiFi.h>                String server_addr=
#include                                "192.168.161.128";//your server
<ESP8266HTTPClient.h>                  address or computer IP
#include <SPI.h>                          byte readCard[4];
#include <MFRC522.h>                       uint8_t successRead;
#include <LCD_I2C.h>                       String UIDCard;
#include <ArduinoJson.h>                   void setup() {
//inisialisasi LED                       pinMode(buzzer, OUTPUT);
LCD_I2C lcd(0x27, 16, 2);                 Serial.begin(115200);// Initialize
#define RST_PIN 16//pin-D0                 serial communications with the PC
#define SS_PIN 0 //pin-D3                  SPI.begin();// Init SPI bus
byte buzzer = 2;//pin D8                  mfrc522.PCD_Init();// Init
MFRC522 mfrc522(SS_PIN,                   MFRC522 card
RST_PIN);// Create MFRC522
instance
const char* ssid = "c";//Your
Wifi SSID
const char* password =
"10102001"; //Wifi Password
Serial.println(F("Read Uid data on
a MIFARE PICC:"));
ShowReaderDetails(); lcd.begin();
lcd.backlight();
lcd.print("Assalamu'alaikum");
lcd.setCursor(3, 1);
lcd.print("Hai Semuanya");
delay(2000);
```



```

ConnectWIFI();

delay(2000);

lcd.clear();}

void loop() {

lcd.print(" Tap Your Card!");

lcd.setCursor(0, 1);

lcd.print(" To The Reader!");

delay(1000);

successRead = getID();

lcd.clear();}

uint8_t getID() {

if(!mfrc522.PICC_IsNewCardP

resent()){return 0;}

if

(!mfrc522.PICC_ReadCardSeri

al()){

return 0;}UIDCard ="";

Serial.println(F("Scanned

PICC's UID:"));

for ( uint8_t i = 0; i <

mfrc522.uid.size; i++) {

UIDCard +=

String(mfrc522.uid.uidByte[i],

HEX);}

UIDCard.toUpperCase();

//Capital

Serial.print("UID:");

Serial.println(UIDCard);

Serial.println(F("**End

Reading**"));

digitalWrite(buzzer,

HIGH);delay(200);

digitalWrite(buzzer,

LOW);delay(200);

digitalWrite(buzzer,

HIGH);delay(200);

digitalWrite(buzzer, LOW);

storeData(); //store data to DB

delay(2000);

mfrc522.PICC_HaltA(); // Stop

reading

return 1;}

void storeData(){

```

```

    ConnectWIFI(); //check wifi
connection

WiFiClient client;

String address, message,
first_name;

//equate with your Server
address (computer's IP address)
and your directory application
address="http://" + server_addr +
"/absensi/webapi/api/create.php
?uid="+UIDCard;

HTTPClient http;

//http.begin(address);

http.begin(client,address);

int httpCode = http.GET();

//Send the GET request

String payload;

Serial.print("Response: ");

if (httpCode > 0) { //Check the
returning code

payload = http.getString(); //Get
the request response payload

```

```

payload.trim(); //remove \n
character

if( payload.length() > 0 ){

Serial.println(payload + "\n");

} }

http.end(); //Close connection

const size_t capacity =
JSON_OBJECT_SIZE(4) + 70;

//simulate your JSON data
https://arduinojson.org/v6/assist
ant/

DynamicJsonDocument
doc(capacity);

// Deserialize the JSON
document

DeserializationError error =
deserializeJson(doc, payload);

// Test if parsing succeeds.

if (error) {

Serial.print(F("deserializeJson()
failed: "));

Serial.println(error.c_str());

return; }

```

```

const char* waktu_res =
doc["waktu"];

String nama_res = doc["nama"];

const char* uid_res =
doc["uid"]; String status_res =
doc["status"];

//char * first_name;

for(int i = 0; i <
nama_res.length(); i++){

if(nama_res.charAt(i) == ' '){

first_name =
nama_res.substring(0, i);

break;}}

lcd.clear();

//Print Data

if (status_res == "INVALID"){

message = "?";

lcd.print(message);

lcd.setCursor(3, 0);

lcd.print(uid_res);

lcd.setCursor(1, 1);

lcd.print(status_res);}else{

if (status_res == "IN"){

message = "welcome";}

else{

message = "See you!";}

lcd.print(message);

lcd.setCursor(0, 1);

lcd.print(first_name);

lcd.setCursor(13, 1);

lcd.print(status_res);}

delay(3000);}

void ConnectWIFI(){

if(WiFi.status() !=

WL_CONNECTED){Serial.pri

nt("Attempting to connect to

SSID: ");

Serial.println(ssid);

WiFi.begin(ssid, password);

int i=0;

int a=0;

while(WiFi.status() !=

WL_CONNECTED){

Serial.print(".");

lcd.clear();

lcd.print("Connecting ...");

```

```

delay(1000);

++i;

if (i==30){

i=0;

Serial.println("\n Failed to

Connect.");break; } }

// Get the MFRC522 software

version

byte v =

mfrc522.PCD_ReadRegister(mf

rc522.VersionReg);

Serial.print(F("MFRC522

Software Version: 0x"));

Serial.print(v, HEX);

if (v == 0x91)

Serial.print(F(" = v1.0"));

else if (v == 0x92)

Serial.print(F(" = v2.0"));

else

Serial.print(F("

(unknown),probably a chinese

clone?"));

Serial.println("");

```

```

Serial.println("\n Connected!");

lcd.clear();

lcd.print("Connected!");

delay(2000);} }

void ShowReaderDetails() {

// When 0x00 or 0xFF is

returned, communication

probably failed

if ((v == 0x00) || (v == 0xFF)) {

Serial.println(F("WARNING:

Communication failure, is the

MFRC522 properly

connected?"));

Serial.println(F("SYSTEM

HALTED: Check

connections."));

while (true); // do not go

further} }

```

Lampiran 2. Program Keseluruhan

Esp32Cam

#include <Arduino.h>

#include <WiFi.h>

#include "soc/soc.h"

#include "soc/rtc_cntl_reg.h"

#include "esp_camera.h"

const int pinButton = 15;

const char* ssid = "c";

const char* password =

"10102001";

String serverName =

"192.168.161.128"; //alamat

server

String serverPath =

"/absensi/upload_img.php";

const int serverPort = 80;

WiFiClient client;

//

CAMERA_MODEL_AI_THIN

KER

#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

```

const int Interval = 30000; //
proses pengambilan photo
interval 30 detik
unsigned long previousMillis =
0;

String jsonres;

void setup() {
pinMode(pinButton,
OUTPUT);

WRITE_PERI_REG(RTC_CN
TL_BROWN_OUT_REG, 0);

Serial.begin(115200);

WiFi.mode(WIFI_STA);

Serial.println();

Serial.print("Connecting to ");

Serial.println(ssid);

WiFi.begin(ssid, password);

while (WiFi.status() !=
WL_CONNECTED) {

Serial.print(".");

config.pin_d5 =
Y7_GPIO_NUM;

delay(500);}

Serial.println();

Serial.print("ESP32-CAM IP
Address: ");

Serial.println(WiFi.localIP());

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_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 = 12; //0-
63 lower number means higher
quality
config.fb_count = 2;
} else {
config.frame_size =
FRAMESIZE_SVGA;
config.jpeg_quality = 8; //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(1000);

ESP.restart();}

sensor_t * s =

esp_camera_sensor_get();

s->set_framesize(s,

FRAMESIZE_SVGA);

//UXGA|SXGA|XGA|SVGA|V

GA|CIF|QVGA|HQVGA|QQV

GA}

void loop() {

if (touchRead(T5)< 40) {

Serial.println("Bersiap kirim

foto ke server..");

digitalWrite(pinButton,HIGH);

sendPhoto();

delay(2000);

// Adjust the delay time (in

milliseconds) as needed

} else {

digitalWrite(pinButton,LOW);

delay(2000);

// Adjust the delay time (in

milliseconds) as needed}}

```

```

void sendPhoto() {

String AllData;

String DataBody;

//pre capture for accurate timing

for (int i = 0; i <= 3; i++) {

camera_fb_t * fb = NULL;

fb = esp_camera_fb_get();

if(!fb) {

Serial.println("Camera capture

failed");

delay(1000);

ESP.restart();

return;}

esp_camera_fb_return(fb);

delay(200);}

camera_fb_t * fb = NULL;

fb = esp_camera_fb_get();

Serial.println("Connecting to

server: " + serverName);

if

(client.connect(serverName.c_st

r(), serverPort)) {

```



```

Serial.println("Connection
successful!");

form-data; name="imageFile";
filename="_esp32Photo.jpg"\r
\nContent-Type:
image/jpeg\r\n\r\n";

String head = post_data;

String boundary = "\r\n--
dataMarker--\r\n";

//Serial.println(head);

uint32_t imageLen = fb->len;

uint32_t dataLen =
head.length() +
boundary.length();

uint32_t totalLen = imageLen +
dataLen;

client.println("POST " +
serverPath + " HTTP/1.1");

client.println("Host: " +
serverName);

client.println("Content-Length:
" + String(totalLen));

```

```

String post_data = "--
dataMarker\r\nContent-
Disposition:
client.println("Content-Type:
multipart/form-data;
boundary=dataMarker");

client.println();

client.print(head);

uint8_t *fbBuf = fb->buf;

size_t fbLen = fb->len;

//Serial.println(fbLen);

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(boundary);

esp_camera_fb_return(fb);

```

```

int timeoutTimer = 10000;

long startTimer = millis();

boolean state = false;

Serial.println("Response:");

while ((startTimer +
timeoutTimer) > millis()) {

Serial.print(".");

delay(200);

// Skip HTTP headers

while (client.available()) {

char c = client.read();

if (c == '\n') {

if (AllData.length()==0) {

state=true; }

AllData = "";}

else if (c != '\r') { AllData +=

String(c); }

if (state==true) { DataBody +=

String(c); }

startTimer = millis();}

if (DataBody.length()>0) {

break; }

}

client.stop();

Serial.println(DataBody);

Serial.println("#####

#");

Serial.println();}

else {

DataBody = "Connection to " +

serverName + " failed.";

Serial.println(DataBody);

}}

```

Lampiran 3. Program *Database*

Absensi

a. API

```
<?php
header("Access-Control-Allow-Origin: *");
header("Content-Type: application/json; charset=UTF-8");
include_once
'../config/database.php';
include_once
'../class/absensi.php';
$database = new Database();
$db = $database->getConnection();
} else{
http_response_code(404);
echo json_encode("Failed!");
}?>
```

b. Class

```
<?php
```

```
$item = new Absensi($db);
$item->uid =
isset($_GET['uid']) ?
$_GET['uid'] : die('wrong
structure!');
if($item->createData()){
// create array
$data_arr = array(
"waktu" => $item->waktu,
"nama" => $item->nama,
"uid" => $item->uid,
"status" => $item->status);
http_response_code(200);
echo json_encode($data_arr);

date_default_timezone_set('Asia/Jakarta');
class Absensi{
// Connection
private $conn;
// Table
```

```

private $db_table =
"data_absen";
private $db_table1 =
"data_karyawan";
private $db_table2 =
"data_invalid";
// Columns
public $id;
public $tanggal;
    public $waktu;
    public $uid;
    public $status;
    public $last_status;
    public $nama;
// Db connection
public function
__construct($db){
    $this->conn = $db;}
// CREATE
public function createData(){
//1. Cek user
    $sqlQuery = "SELECT *
FROM ". $this->db_table1 ."
WHERE uid = :uid LIMIT 0,1";
    $stmt = $this->conn-
>prepare($sqlQuery);
    $stmt->bindParam(":uid", $this-
>uid);
    $stmt->execute();
    if($stmt->errorCode() == 0) {
        while(($dataRow = $stmt-
>fetch(PDO::FETCH_ASSOC)
        ) != false) {
            $this->nama =
            $dataRow['nama'];}
        } else {$errors = $stmt-
>errorInfo();
        echo($errors[2]);}
        $itemCount = $stmt-
>rowCount();
        if($itemCount > 0){
            //UID terdaftar -> cek status
            terakhir

```

```

$sqlQuery = "SELECT
data_absen.id, data_absen.uid,
data_absen.status,
data_karyawan.nama
FROM ". $this->db_table .", ".
$this->db_table1 ."
WHERE data_absen.id =
(SELECT MAX(data_absen.id)
FROM ". $this->db_table ."
WHERE data_absen.uid = :uid)
AND data_karyawan.uid=
:uid";

$stmt = $this->conn-
>prepare($sqlQuery);

$stmt->bindParam(":uid", $this-
>uid);

$stmt->execute();

$itemCount = $stmt-
>rowCount();

if($itemCount > 0){
//error handling
if($stmt->errorCode() == 0) {

while(($dataRow = $stmt-
>fetch(PDO::FETCH_ASSOC)
) != false) {
$this->last_status =
$dataRow['status'];

$this->nama =
$dataRow['nama'];

//echo($this->last_status);} }
else {
$errors = $stmt->errorInfo();
echo($errors[2]);} }else{
$this->last_status = "OUT";}

//set status
if ($this->last_status == "IN"){
$this->status = "OUT";}else{
$this->status= "IN";}

//Insert Data to data_absen
$sqlQuery = "INSERT INTO ".
$this->db_table ."
SET waktu = :waktu, uid = :uid,
status = :now_status";

$this->waktu = date("H:i:s");

```

```

$stmt = $this->conn-                                // sanitize
>prepare($sqlQuery);

$this-                                                waktu = :waktu,
>uid=htmlspecialchars(strip_tag                    uid = :uid,
s($this->uid));                                       status = :now_status";

// bind data                                         $this->waktu = date("H:i:s");
$stmt->bindParam(":uid", $this-                       $stmt = $this->conn-
>uid);                                              >prepare($sqlQuery);

$stmt-                                              // sanitize $this-
>bindParam(":now_status",                          >uid=htmlspecialchars(strip_tag
$this->status);                                       s($this->uid));

$stmt->bindParam(":waktu",                            // bind data
$this->waktu);                                       $stmt->bindParam(":uid", $this-
>uid);

if($stmt->execute()){                                $stmt-
    return true;}                                  >bindParam(":now_status",
return false;}else{                                $this->status);
//UID tidak terdaftar                             $stmt->bindParam(":waktu",
$this->status= "INVALID";                           $this->waktu);
$this->nama ="Invalid";
//Insert Data to data_invalid

$sqlQuery = "INSERT INTO
". $this->db_table2 ."SET

```

c. Config

```

<?php

```

```

class Database {

private $host = "localhost";

private $database_name =

"absensi";

private $username = "root";

private $password = "";

public $conn;

public function

getConnection(){

$this->conn = null;try{

$this->conn = new

PDO("mysql:host=" . $this-

>host . ";dbname=" . $this-

>database_name, $this-

>username, $this->password);

$this->conn->exec("set names

utf8");}catch(PDOException

$exception){

echo "Database could not be

connected: " . $exception-

>getMessage();}

return $this->conn;}} ?>

```

Lampiran 4. Program *Database*

Kamera

a. Upload_Image

```
<?php
```

```
//based on PHP File Upload
```

```
basic example
```

```
https://www.w3schools.com/ph
```

```
p/php_file_upload.asp
```

```
date_default_timezone_set('Asia/Jakarta');
```

```
$target_dir =
```

```
"captured_images/"; //folder
```

```
untuk menyimpan gambar
```

```
$date = new DateTime(); //this
```

```
returns the current date time
```

```
$date_string = $date-
```

```
>format('Y-m-d His');
```

```
$target_file = $target_dir .
```

```
$date_string.
```

```
basename($_FILES["imageFile
```

```
"]["name"]);
```

```
$uploadOk = 1;
```

```
$imageFileType =
```

```
strtolower(pathinfo($target_file,
```

```
PATHINFO_EXTENSION));
```

```
$file_name =
```

```
pathinfo($target_file,PATHINF
```

```
O_BASENAME) ;
```

```
// Check if image file is a actual
```

```
image or fake image
```

```
if(isset($_POST["imageFile"]))
```

```
{
```

```
$check =
```

```
getimagesize($_FILES["imageF
```

```
ile"]["tmp_name"]);
```

```
if($check !== false) {
```

```
echo "File is an image - " .
```

```
$check["mime"] . ". ";
```

```
$uploadOk = 1;
```

```
} else {echo "File is not an
```

```
image.";
```

```
$uploadOk = 0;}}
```

```
// Check if file already exists
```

```
if (file_exists($target_file)) {
```



```

echo "Sorry, file already
exists.";

$uploadOk = 0;}

// Check file size

if

($_FILES["imageFile"]["size"]
> 500000) {

echo "Sorry, your file is too
large.";

$uploadOk = 0;}

// Allow certain file formats

if($_imageFileType != "jpg" &&
$imageFileType != "png" &&
$imageFileType != "jpeg" &&
$imageFileType != "gif" ) {

echo "Sorry, only JPG, JPEG,
PNG & GIF files are allowed.";

$uploadOk = 0;}

// Check if $uploadOk is set to 0
by an error

if ($uploadOk == 0) {

echo "Sorry, your file was not
uploaded.";

```

```

// if everything is ok, try to
upload file} else {

if

(move_uploaded_file($_FILES[
"imageFile"]["tmp_name"],
$_target_file)) {echo "Photo
berhasil dipupload di server
dengan nama " .$_file_name;

} else {echo "Sorry, Ada error
dalam proses upload
photo.";}}?>

```

b. Database

```

<!doctype html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta name="viewport"
content="width=device-width,
initial-scale=1">

<title>ESP32CAM Gallery
Demo</title>

<link

href="https://cdn.jsdelivrivr.net/np

```

```

m/bootstrap@5.2.0-
beta1/dist/css/bootstrap.min.css
" rel="stylesheet"
integrity="sha384-
0evHe/X+R7YkIZDRvuzKMR
qM+OrBnVFBL6DOitfPri4tjfH
xaWutUpFmBp4vmVor"
crossorigin="anonymous">
<script
src="https://cdn.jsdelivr.net/np
m/bootstrap@5.2.0-
beta1/dist/js/bootstrap.bundle.m
in.js" integrity="sha384-
pprn3073KE6tl6bjs2QrFaJGz5/
SUsLqktiwsUTF55Jfv3qYSDh
gCecCxMW52nD2"
crossorigin="anonymous"></sc
ript>
<script
src="https://code.jquery.com/jq
uery-3.6.0.min.js"
integrity="sha256-
/xUj+3OJU5yExlq6GSYGSHk

```

```

7tPXikynS7ogEvDej/m4="
crossorigin="anonymous"></sc
ript>
</head>
<body>
<div class="container"
style="padding-top:30px;">
<div class="d-flex justify-
content-center"
><h1>ESP32CAM PHOTO
Gallery</h1></div>
<hr class="mt-2 mb-5">
<?php
// Image extensions
$image_extensions =
array("png","jpg","jpeg","gif");
// Check delete HTTP GET
request - remove images
if(isset($_GET["delete"])){
$imageFileType =
strtolower(pathinfo($_GET["del
ete"],PATHINFO_EXTENSIO
N));

```

```

if (file_exists($_GET["delete"])
& ($imageFileType == "jpg" ||
$imageFileType == "png" ||
$imageFileType == "jpeg")) {
    unlink($_GET["delete"]);
    echo "<script>
$(document).ready(function(){
$('#myModalOK').modal('show'
);
}); </script>"; } else {
    echo 'File not found - <a
href="galeria.php">refresh</a>'
; } }

// Target directory
$dir = 'captured_images/';
if (is_dir($dir)){ ?>
<div class="row text-center
text-lg-start"><?php
$count = 1;
$files = scandir($dir);
rsort($files);
foreach ($files as $file) {

```

```

if ($file != '.' && $file != '..')
{ ?>
<div class="col-lg-3 col-md-4
col-6" style="padding-
bottom:30px;">
<div class="row">
<a href="<?php echo $dir .
$file; ?>"
class="d-block mb-4 h-100">

</a></div>
<div class="row justify-content-
end"><div class="col md-8">
<p><?php echo $file; ?></p>
</div><div class="col md-4">
<a
href="index.php?delete=<?php
echo $dir . $file; ?>" class="btn
btn-danger btn-sm">Delete</a>
</div></div></div><?php
$count++; } }

```

```

if($count==1) { echo "<p>No
images found</p>"; }
}?></div>

=<!-- Modal Delete OK-->

=<div class="modal fade"
id="myModalOK" tabindex="-
1" aria-
labelledby="myModalOKLabel
" aria-hidden="true">

= <div class="modal-dialog
modal-dialog-centered">

=<div class="modal-content">

<div class="modal-header"

style="background-
color:#2cc791;">

<h5 class="modal-title"
id="exampleModalLabel"
style="color:white;">Success</
h5>

<button type="button"
class="btn-close" data-bs-
dismiss="modal" aria-

```

```

label="Close"></button>

</div>

<div class="modal-body">

Successfully delete image file

<b><?php $path_parts =
pathinfo($_GET["delete"],PAT
HINFO_BASENAME) ; echo
$path_parts; ?> </b></div>

<div class="modal-footer">

<a class="btn btn-primary"
href="index.php"
role="button">OK</a>

</div></div> </div>

</div><!-- Modal Delete Not
OK-->

<div class="modal fade"
id="myModalOK" tabindex="-
1" aria-
labelledby="myModalOKLabel
"
aria-hidden="true">

<div class="modal-dialog
modal-dialog-centered">

```

```

<div class="modal-content">
</div></div> </div></div>
<div class="modal-header"
style="background-
color:#fc8403;">
<h5 class="modal-title"
id="exampleModalLabel"
style="color:white;">Success</
h5>
<button type="button"
class="btn-close" data-bs-
dismiss="modal" aria-
label="Close"></button></div>
<div class="modal-body">
Something wrong while
deleting file <?php $path_parts
=
pathinfo($_GET["delete"],PAT
HINFO_BASENAME) ; echo
$path_parts; ?></div><div
class="modal-footer">
<a class="btn btn-secondary"
href="galeria.php"
role="button">OK</a>

```



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET DAN TEKNOLOGI
UNIVERSITAS NEGERI PADANG
FAKULTAS TEKNIK

DEPARTEMEN TEKNIK ELEKTRONIKA

Alamat: Jln. Prof. Dr. Hamka Air Tawar Padang 25131
Telephone: 0751-444614, Website: <http://elektronika.ft.unp.ac.id>

SURAT TUGAS PEMBIMBING SKRIPSI/TUGAS AKHIR

Nomor: 024/UN35.2.2/PD/2025

Berdasarkan Keputusan MENKOWASBANGPAN No. 38/Kep/MK.WASPAN/8/1999 tentang Jabatan Fungsional Dosen dan Keputusan Menteri Pendidikan Nasional No. 36/D/O/2001 tentang Petunjuk Teknis Pelaksanaan Penilaian Angka Kredit Jabatan Dosen, Pimpinan Jurusan menunjuk dan menetapkan:

Nama : Ilmiyati Rahmy Jasril, S.Pd., M.Pd.T
NIP : 19881007 201504 2 003
Pangkat/Gol. : Penata /III/c
Jabatan : Lektor

Sebagai pembimbing proposal skripsi/tugas akhir/proyek akhir mahasiswa

Nama : Yuda Pratama
TM/NIM : 20065066
Program Studi : Pendidikan Teknik Elektronika
Judul : Rancang Bangun Sistem Informasi Absensi Berbasis ESP8266

Surat tugas ini berlaku mulai dari tanggal ditetapkan sampai mahasiswa yang bersangkutan selesai mengikuti seminar.

Demikianlah surat tugas ini dibuat, untuk dapat digunakan sebagaimana mestinya.

Padang, 08 Januari 2024
Kepala Departemen,

Dr. Hendra Hidayat, S.Pd., M.Pd.
NIP. 198703052020121012

