

## WEBSITE MONITORING KEHADIRAN PEGAWAI ROTI ENAK MENGUNAKAN SENSOR *FINGERPRINT* BERBASIS *IOT*

Nika Rani Nur Shafa Lubis<sup>1</sup>, Dhiyaurrahman Hamizan Haikal Putra<sup>2</sup>, Vidia Tamie Honesta<sup>3</sup>, Rachel Aureliya Yahya<sup>4</sup>, Ramzy Atchallah Putra<sup>5</sup>

Program Studi Teknologi Rekayasa Perangkat Lunak, Sekolah Vokasi IPB University

Jl. Kumbang No. 14, Kota Bogor, Jawa Barat, Indonesia

E-mail: nikarani@apps.ipb.ac.id<sup>1</sup>, dhamizanhaikalputra@apps.ipb.ac.id<sup>2</sup>, \*honestavidiatamie@apps.ipb.ac.id<sup>3</sup>, rachelyahya@apps.ipb.ac.id<sup>4</sup>, ramzyatchallah@apps.ipb.ac.id<sup>5</sup>

**Abstrak** - *Monitoring* kehadiran pegawai merupakan aspek penting dalam manajemen sumber daya manusia perusahaan untuk menjaga produktivitas kerja dan efisiensi pegawai. Saat ini, perusahaan Roti Enak masih menggunakan sistem absensi manual yang rentan terhadap kesalahan pencatatan dan kehilangan data. Penelitian ini mengusulkan perancangan dan pengembangan sistem monitoring pegawai pada perusahaan Roti Enak menggunakan sensor sidik jari berbasis *Internet of Things* untuk menggantikan sistem manual yang kurang efektif. Sistem ini dibangun menggunakan mikrokontroler *ESP8266* untuk membaca data sidik jari dan mengirimkannya ke server berbasis *Laravel* melalui koneksi Wi-Fi. Data kehadiran yang tercatat akan ditampilkan secara *real-time* melalui antarmuka pengguna berbasis *Vue.js*. Hasil penelitian menunjukkan bahwa sistem dapat mencatat dan menyimpan data kehadiran dengan lebih cepat, akurat, dan aman dibandingkan metode manual. Dengan adanya sistem ini, diharapkan dapat mempermudah perusahaan dalam memonitoring kehadiran pegawai.

**Kata Kunci:** *Internet of Things*, Monitoring Kehadiran, Sensor Sidik Jari

### I. PENDAHULUAN

Absensi adalah kegiatan rutin yang dilakukan oleh suatu perusahaan atau instansi yang berfungsi untuk memantau kehadiran setiap individu dalam menjalankan tugasnya. Sistem absensi membantu dalam proses pemantauan kehadiran, mengidentifikasi masalah terkait keterlambatan atau ketidakhadiran, meningkatkan kedisiplinan pegawai, serta mendukung perencanaan sumber daya manusia (Hidayat et al., 2020).

Saat ini, masih terdapat perusahaan yang menggunakan sistem absensi secara manual dengan *WhatsApp*, salah satunya adalah perusahaan Roti Enak. Perusahaan Roti Enak saat ini mempunyai sekitar 15 orang pegawai yang bekerja secara harian. Selama ini, proses pencatatan kehadiran masih dilakukan secara manual. Sistem seperti ini dapat menimbulkan berbagai kendala, seperti kurang efisiennya dalam merekapitulasi data kehadiran bulanan serta potensi kesalahan dalam pencatatan. Dengan demikian, sistem saat ini dinilai kurang efektif untuk digunakan dalam sebuah perusahaan karena dapat mengakibatkan kekeliruan dalam proses absensi dan menghambat efektivitas manajemen kehadiran pegawai yang dapat berdampak langsung terhadap proses penggajian serta evaluasi kinerja.

Seiring perkembangan teknologi, salah satu solusi untuk meningkatkan efisiensi dan akurasi dalam absensi adalah dengan penggunaan perangkat berbasis *Internet of Things* (IoT). Salah satu teknologi *IoT* yang dapat diterapkan dalam sistem absensi adalah sensor sidik jari yang dapat

mencatat kehadiran secara otomatis dan mengirimkan data secara *real-time*. Studi oleh Hidayat et al. (2020) menyebutkan bahwa perusahaan yang menerapkan sistem absensi digital mengalami peningkatan efisiensi kerja hingga 20% karena pencatatan absensi menjadi lebih akurat dan dapat dipantau secara *real-time*.

Berdasarkan permasalahan yang telah diuraikan, dibutuhkan suatu sistem absensi yang lebih modern dan terintegrasi untuk menggantikan metode manual yang masih digunakan di perusahaan Roti Enak. Oleh karena itu, penelitian ini bertujuan untuk merancang dan mengembangkan sistem absensi berbasis *Internet of Things* menggunakan sensor sidik jari. Sistem yang dibangun diharapkan tidak hanya mampu meningkatkan efisiensi dalam pencatatan kehadiran, tetapi juga memberikan kemudahan bagi pihak yang bersangkutan untuk memantau kehadiran pegawai secara langsung, mempercepat proses rekapitulasi, serta meningkatkan akurasi data guna mendukung pengambilan keputusan yang lebih baik dalam manajemen sumber daya manusia.

### II. TINJAUAN PUSTAKA

Penelitian mengenai sistem absensi berbasis teknologi telah banyak dilakukan seiring berkembangnya zaman dan kebutuhan akan sistem informasi yang lebih efisien dan akurat dalam lingkungan kerja. Salah satu teknologi yang umum digunakan dalam sistem absensi digital adalah sensor biometrik seperti *fingerprint scanner* yang dapat mengidentifikasi sidik jari tiap individu.

Menurut Fitri (2017), sistem absensi merupakan komponen penting dalam pengelolaan sumber daya manusia karena membantu memantau kedisiplinan dan kinerja pegawai. Sistem manual dengan *WhatsApp* memiliki beberapa kekurangan, seperti resiko kehilangan data, adanya pencatatan ganda, dan potensi terjadinya kecurangan. Kekurangan ini menyebabkan banyak perusahaan mulai beralih ke sistem digital yang lebih aman dan andal.

Hidayat et al. (2020) menunjukkan bahwa penerapan sistem absensi digital mampu meningkatkan efisiensi kerja perusahaan hingga 20% khususnya karena pencatatan yang lebih akurat dan dapat dipantau secara *real-time*. Selain itu, penerapan digitalisasi absensi juga dapat mempercepat proses rekapitulasi kehadiran yang berkaitan langsung dengan penggajian dan evaluasi kerja.

Salah satu teknologi sistem informasi yang menjadi fokus saat ini adalah *Internet of Things*. Menurut Putra et al. (2020), penggunaan mikrokontroler seperti *ESP8266* memungkinkan pengiriman data dari perangkat sensor secara langsung ke server tanpa campur tangan manual. Sensor sidik jari yang terhubung ke jaringan Wi-Fi dapat mengirimkan data kehadiran ke sistem *backend* secara otomatis dan *real-time*, meningkatkan keakuratan dan efisiensi.

Di sisi lain, penerapan *IoT* dalam sistem absensi juga memberikan keuntungan tambahan, seperti kemampuan integrasi dengan sistem penggajian dan penyimpanan berbasis *cloud*. Namun, seperti yang disampaikan oleh Prasetyo et al. (2021), sistem ini juga memiliki beberapa kekurangan, antara lain kebutuhan akan jaringan internet yang stabil dan perlunya keamanan biometrik yang tinggi.

Berdasarkan literatur yang telah ditinjau, sebagian besar sistem absensi yang dikembangkan hanya menampilkan proses digitalisasi absensi tanpa mengintegrasikan sistem secara menyeluruh dari perangkat *IoT* hingga dashboard monitoring berbasis web. Belum banyak sistem yang mengintegrasikan sensor sidik jari dengan *backend Laravel* dan *frontend* berbasis *Vue.js* dalam skala UKM. Oleh karena itu, penelitian ini memiliki nilai kebaruan dalam hal penerapan sistem absensi otomatis pada sektor usaha kecil, yaitu toko Roti Enak, dengan menggabungkan teknologi sensor sidik jari berbasis *IoT*, komunikasi data *real-time*, serta tampilan antarmuka berbasis web yang *user-friendly*.

### III. METODE PENELITIAN

Perancangan dan pengembangan sistem ini dimulai dengan menentukan *timeline* pengerjaan

sistem, terhitung dimulai tanggal 20 Januari 2025 hingga 27 Mei 2025 atau terhitung 14 minggu.

Pengerjaan sistem ini dibagi ke dalam beberapa tahapan kerja.



Gambar 1. *Work Breakdown Structure*

1. Analisis kebutuhan dan penentuan jadwal: Tahap ini dilakukan pada minggu ke-1 hingga minggu ke-2. Tim melakukan pencarian mitra dan diskusi awal dengan pihak pengguna untuk memahami kebutuhan sistem. Pada tahap ini juga dilakukan penyusunan jadwal proyek (*timeline*) dan identifikasi fitur utama yang akan dikembangkan.
2. Analisis dan perancangan sistem: Berlangsung pada minggu ke-3 hingga minggu ke-7. Kegiatan pada tahap ini mencakup analisis kebutuhan fungsional dan non-fungsional, serta perancangan sistem menggunakan berbagai diagram seperti *usecase*, *activity diagram*, dan *flowchart*. Selain itu, dilakukan desain antarmuka pengguna, struktur *database*, dan arsitektur sistem.
3. Pengembangan sistem: Dilaksanakan pada minggu ke-8 hingga minggu ke-12. Tahap ini mencakup proses implementasi kode program berdasarkan desain sistem yang telah disusun. Bahasa pemrograman dan *tools* yang digunakan meliputi *Arduino*, *MySQL*, *HTML*, *CSS*, *JavaScript*.
4. Pengujian dan *Debugging*: Dikerjakan pada minggu ke-13. Tim melakukan pengujian sistem secara menyeluruh untuk memastikan sistem berjalan dengan baik.
5. Implementasi dan Evaluasi: Minggu ke-14 digunakan untuk melakukan peluncuran sistem di lingkungan perusahaan Roti Enak secara internal. Evaluasi dilakukan untuk meninjau performa sistem dan merencanakan perawatan berkala jika dibutuhkan.

Dalam pengembangan sistem ini, metode yang digunakan adalah *Agile SDLC*. Menurut Adani (2020), *Agile Development* adalah model pengembangan perangkat lunak dalam jangka pendek, untuk kemudian diadaptasi secara cepat dalam mengatasi setiap perubahan. Nilai terpenting dari *Agile Development* ini adalah memungkinkan sebuah tim dalam mengambil keputusan dengan

cepat, kualitas dan prediksi yang baik. Sedangkan *System Development Life Cycle* atau *SDLC* adalah proses yang digunakan oleh tim pengembangan untuk mendesain dan membangun perangkat lunak berkualitas tinggi dengan cara yang hemat biaya dan waktu.

Metode Agile *SDLC* digunakan pada pengembangan sistem ini karena proyek membutuhkan tingkat fleksibilitas yang tinggi dan kemampuan untuk merespons perubahan dengan cepat. Sistem yang dikembangkan ditujukan untuk memenuhi kebutuhan mitra yang dinamis, sehingga diperlukan pendekatan yang iteratif memungkinkan pengujian dan evaluasi dilakukan secara berkelanjutan di setiap tahap siklus pengembangan. Dengan menggunakan metode ini, tim dapat melakukan penyesuaian secara *real-time* berdasarkan masukan dari mitra, tanpa harus menunggu hingga seluruh sistem berhasil dibangun. Hal ini memungkinkan peningkatan kualitas produk secara bertahap, meminimalisasi resiko kegagalan, dan memastikan bahwa sistem yang dikembangkan benar-benar sesuai dengan kebutuhan pengguna akhir.



Gambar 2. Ilustrasi Tahapan *SDLC*

Metode *SDLC* umumnya dibagi menjadi beberapa tahapan. Tahapan tersebut menurut Pressman (2010) adalah:

1. *Planning* (Perencanaan): Pada tahap ini dilakukan identifikasi kebutuhan proyek, penentuan tujuan, serta estimasi biaya dan sumber daya. Perencanaan bertujuan untuk memahami ruang lingkup sistem dan mendefinisikan pendekatan manajemen proyek yang sesuai.
2. *Analysis* (Analisis): Tahapan ini fokus pada pemahaman kebutuhan pengguna. Aktivitas utama meliputi pengumpulan data, identifikasi masalah, dan dokumentasi kebutuhan perangkat lunak. Hasilnya berupa spesifikasi kebutuhan perangkat lunak (SRS).
3. *Design* (Perancangan): Desain sistem dilakukan berdasarkan hasil analisis. Fokusnya adalah menyusun arsitektur sistem, desain antarmuka, model data, dan algoritma. Tujuan dari tahap ini adalah untuk memastikan sistem dapat dikembangkan secara efisien dan mudah dipelihara.

4. *Implementation* (Implementasi): Desain yang telah disusun diubah menjadi kode program dalam tahap implementasi. Pengembang mulai membangun sistem berdasarkan rancangan yang telah dibuat sebelumnya, baik dari sisi *frontend*, *backend*, maupun *database*.
5. *Testing & Integration* (Pengujian dan Integrasi): Sistem diuji untuk memastikan bahwa semua fitur berjalan sesuai kebutuhan dan bebas dari kesalahan. Pengujian meliputi unit testing, integrasi modul, serta validasi terhadap kebutuhan pengguna.
6. *Maintenance* (Pemeliharaan): Setelah sistem diterapkan, dilakukan pemeliharaan yang mencakup koreksi kesalahan (*bug*), peningkatan performa, dan penyesuaian terhadap kebutuhan baru. Tahap ini sangat penting karena perangkat lunak akan digunakan dalam jangka waktu yang lama.



Gambar 3. Ilustrasi Tahapan *Agile*

Salah satu metode *SDLC* adalah Agile. Agile merupakan metode iteratif dan inkremental, di mana setiap siklus pengembangan (*sprint*) mencakup beberapa tahap berulang. Setiap iterasi menghasilkan versi perangkat lunak yang bisa digunakan. Tahapan *SDLC* Agile menurut Sommerville (2016) adalah:

1. *Plan* (Perencanaan): Pada tahap ini, tim dan pemangku kepentingan (*stakeholder*) mendiskusikan tujuan jangka pendek, menetapkan *backlog* produk (daftar kebutuhan), serta merencanakan apa saja yang akan dicapai dalam iterasi tertentu.
2. *Design* (Perancangan): Desain awal dilakukan secara ringan (*lightweight design*), fokus pada fungsionalitas yang akan dikerjakan dalam iterasi tersebut. Dokumentasi desain minimal, namun cukup untuk mendukung implementasi.
3. *Develop* (Pengembangan): Pengembang menulis kode berdasarkan desain dan *backlog*. Tim bekerja secara kolaboratif dalam waktu pendek (biasanya 1-4 minggu).
4. *Test* (Pengujian): Pengujian dilakukan secara terus-menerus (*continuous testing*) selama iterasi. Metode yang umum digunakan adalah *test-driven development* (TDD), di mana pengujian dibuat sebelum pengkodean.
5. *Deploy* (Penerapan): Versi yang telah selesai diuji dideploy ke lingkungan *staging* atau

bahkan produksi. Karena Agile bersifat inkremental, tiap deploy membawa tambahan fitur.

6. *Review* (Tinjauan): Setelah deploy, dilakukan sesi *review* atau *sprint review meeting* dengan seluruh tim dan *stakeholder*. Diperiksa apakah kebutuhan telah terpenuhi dan *feedback* dikumpulkan.

Setelah menentukan metode yang cocok untuk pengembangan perangkat lunak, tahapan selanjutnya yang dilakukan adalah memilih alat dan teknologi yang sesuai dengan kebutuhan sistem. Pemilihan dilakukan berdasarkan kriteria seperti efisiensi, keandalan, dan kemudahan integrasi.

Pada bagian perangkat keras, pengembangan ini menggunakan mikrokontroler *ESP8266* dan sensor *fingerprint DY50*. *ESP8266* adalah modul mikrokontroler berbasis Wi-Fi yang memungkinkan perangkat terhubung ke jaringan internet. Menurut Raza et al. (2017), *ESP8266* banyak digunakan dalam proyek *Internet of Things* karena hemat biaya, konsumsi daya rendah, dan kompatibel dengan berbagai bahasa pemrograman seperti *Arduino IDE*.



Gambar 4. NodeMCU ESP8266

Untuk membaca dan mengenali sidik jari pengguna, pengembangan ini menggunakan perangkat keras modul *fingerprint DY50*. Menurut Wulandari & Yuliana (2020), sensor ini cukup akurat dalam mengidentifikasi identitas biometrik dan banyak digunakan dalam sistem keamanan atau sistem absensi. Kelebihan dari sensor *DY50* adalah kemampuannya dalam menyimpan data sidik jari pengguna secara internal serta kecepatannya dalam merespon proses identifikasi.



Gambar 5. Modul *Fingerprint* DY50

Untuk pengembangan bagian website, pada sisi server digunakan *framework Laravel* berbasis bahasa pemrograman *PHP*. Menurut Rahman & Al-Dabbagh (2020), *Laravel* adalah salah satu *framework PHP* yang populer karena arsitekturnya yang bersih, tersusun rapi, serta mendukung pengembangan aplikasi berbasis *API*. Pengembangan website menggunakan *Laravel* karena sistem *routing* yang efisien, serta adanya fitur *built-in* untuk mengelola autentikasi dan *middleware*.

Pada bagian antarmuka pengguna, digunakan *Vue.js* sebagai *framework Javascript*. Menurut You (2021), *Vue.js* adalah *framework* progresif yang dirancang untuk membangun antarmuka pengguna yang interaktif dan responsif. *Vue.js* banyak digunakan sebagai *framework* karena kemudahan dalam integrasi proyek yang sudah ada, dokumentasi yang lengkap, serta ukuran file yang ringan namun tetap *powerful*.

*Database Management System* (DBMS) adalah sistem perangkat lunak yang memungkinkan pengguna membuat, memelihara, dan mengelola basis data secara efisien. *DBMS* memfasilitasi berbagai operasi seperti mendefinisikan struktur data, membangun isi *database*, memanipulasi data, dan berbagi akses data dengan aman kepada banyak pengguna (Elmasri & Navathe., 2015). Sistem basis data yang digunakan pada pengembangan ini adalah *MySQL*. Menurut Coronel dan Morris (2020), *MySQL* adalah sistem manajemen basis data relasional yang bersifat *open source*, banyak digunakan terutama untuk aplikasi berbasis web karena keandalannya, fleksibilitas, dan kemudahan dalam penggunaannya.

#### IV. HASIL DAN PEMBAHASAN

##### Analisis Sistem

1. Kebutuhan Fungsional

Kebutuhan fungsional mencakup fungsi-fungsi utama yang harus dimiliki sistem agar sistem dapat berjalan sesuai dengan tujuan yang diharapkan. Berikut tabel rincian kebutuhan fungsional:

Tabel 1. Kebutuhan Fungsional

No	Parameter	Deskripsi
1	Autentikasi pengguna	Terdapat masuk ke sistem, keluar dari sistem, dan atur ulang kata sandi.
2	Manajemen profil pengguna	Melihat dan menyunting profil pengguna
3	Absensi	Terdapat pendaftaran sidik jari untuk absensi, melakukan absensi, dan melihat riwayat absensi.
4	Pengajuan dan pengelolaan cuti	Terdapat mengajukan cuti, melihat status

		pengajuan cuti, menerima atau menolak pengajuan cuti
5	Informasi gaji	Melihat data gaji
6	Manajemen gerai	Menambah, melihat dan menyunting data gerai
7	Manajemen pegawai	Menambah, melihat dan menyunting data pegawai

2. Kebutuhan Non-fungsional

Selain kebutuhan fungsional, sistem juga memiliki kebutuhan non-fungsional yang berperan untuk memastikan kualitas, performa, keamanan, serta kemudahan dalam penggunaan sistem. Kebutuhan non-fungsional mencakup aspek-aspek yang tidak secara langsung berhubungan dengan fungsi sistem, namun berkontribusi terhadap kenyamanan dan keandalan sistem secara keseluruhan. Berikut tabel rincian kebutuhan non-fungsional:

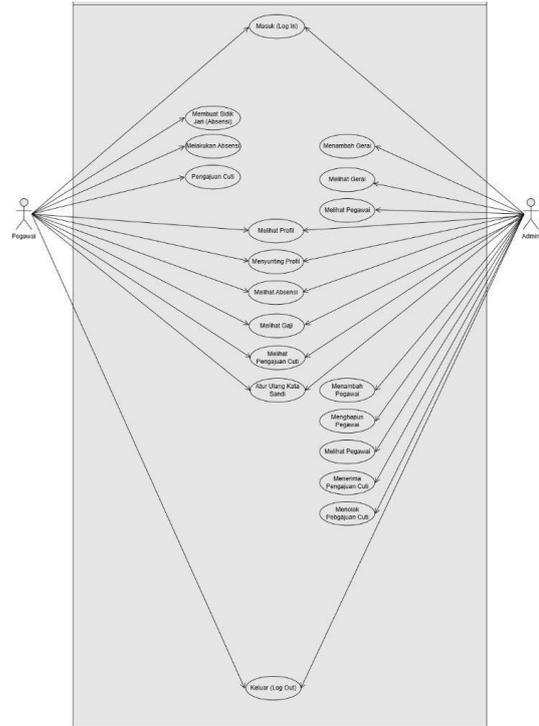
Tabel 2. Kebutuhan Non-fungsional

No	Parameter	Deskripsi
1	Keamanan	Terdapat autentikasi menggunakan <i>username</i> dan <i>password</i> .
2	Kemudahan pengguna	Sistem dapat diakses dari berbagai perangkat.
3	Ketersediaan	Sistem dapat diakses setiap saat (24/7), dan data tersimpan dengan <i>backup</i> berkala.

Perancangan Sistem

1. Use Case Diagram

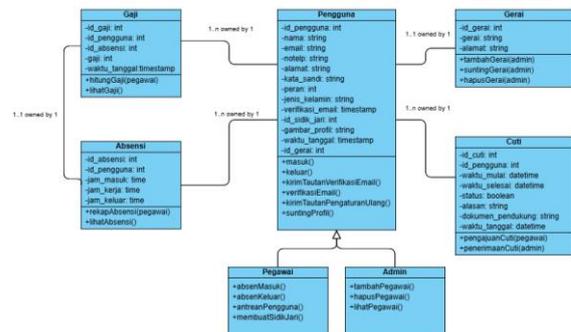
Use case diagram merupakan bagian dari pemodelan perilaku dalam *Unified Modeling Language (UML)* yang menggambarkan interaksi antar aktor dan sistem yang dikembangkan untuk mencapai suatu tujuan tertentu (Pressman., 2010). Use Case Diagram dibuat untuk menggambarkan hubungan antara pengguna dan fungsi-fungsi yang disediakan oleh sistem. Pada gambar 6, user pegawai memiliki 11 fungsi dan admin memiliki 17 fungsi.



Gambar 6. Use Case Diagram

2. Class Diagram

Class Diagram adalah diagram struktural dalam UML yang menggambarkan struktur sistem dengan menunjukkan kelas, atribut, metode, serta hubungan antar kelas. Diagram ini membantu pengembang memahami bagaimana objek-objek dalam sistem saling berinteraksi dan digunakan untuk merancang kerangka dasar sistem perangkat lunak.



Gambar 7. Class Diagram

Implementasi

Setelah tahap analisis sistem dan perancangan sistem telah dilakukan, maka tahap selanjutnya adalah mengimplementasikan sistem. Tahap ini adalah proses penerapan hasil perancangan kedalam bentuk nyata, baik dalam perangkat keras maupun perangkat lunak. Implementasi bertujuan untuk memastikan bahwa sistem dapat berjalan sesuai kebutuhan dan tujuan yang telah ditetapkan sebelumnya.

1. Perangkat Keras

Dalam mengembangkan sistem ini, terdapat beberapa komponen perangkat keras yang digunakan dan dikoneksikan dengan mikrokontroler NodeMCU ESP8266. Komponen utamanya adalah sensor *fingerprint* dan layar OLED. Untuk memastikan seluruh sistem bekerja dengan baik, maka diperlukan koneksi pin yang tepat antara masing-masing komponen dan NodeMCU. Berikut tabel konfigurasi pin yang digunakan:

a. Koneksi Sensor *Fingerprint*

Sensor *fingerprint* adalah perangkat input biometrik yang berfungsi untuk membaca dan mengenali sidik jari pengguna. Sensor ini berkomunikasi dengan NodeMCU melalui antarmuka *UART*. Berikut konfigurasi pinnya:

Tabel 3. Konfigurasi Pin Sensor *Fingerprint*

No	Sensor <i>Fingerprint</i>	NodeMCU ESP8266
1	GND	3
2	RX	D2
3	TX	D1
4	3V3	3V

- GND disambungkan ke ground NodeMCU untuk menyamakan referensi tegangan.
- RX (Receive) dihubungkan ke pin D2 NodeMCU (TX)
- TX (Transmit) dihubungkan ke pin D1 NodeMCU (RX)
- 3V3 memberikan tegangan suplai 3.3V dari NodeMCU ke sensor

b. Koneksi OLED Display

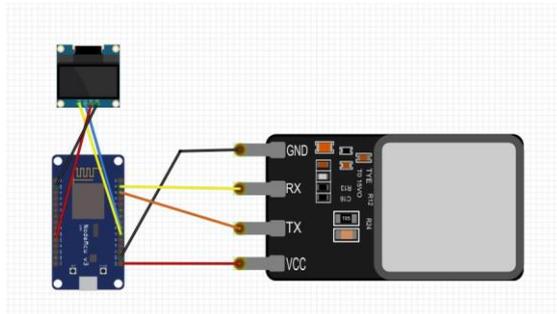
OLED Display digunakan untuk output visual menampilkan status sistem, seperti hasil verifikasi sidik jari pengguna. OLED ini menggunakan protokol komunikasi I2C, yang memerlukan dua pin yaitu SCL dan SDA.

Tabel 4. Konfigurasi Pin OLED Display

No	OLED Display	NodeMCU ESP8266
1	GND	G
2	VCC	3V
3	SCL	D6
4	SDA	D7

- GND dihubungkan ke *ground*
- VCC menerima tegangan suplai 3.3V dari NodeMCU ESP8266
- SCL (Serial Clock Line) dihubungkan ke pin D6
- SDA (Serial Data Line) dihubungkan ke pin D7

Untuk memastikan bahwa pengembangan perangkat keras telah berjalan dengan baik, maka dilakukan serangkaian pengujian terhadap sensor sidik jari dan *OLED display* yang telah dikonfigurasi pada NodeMCU ESP8266. Hasil pengujian menunjukkan bahwa sensor sidik jari berhasil membaca sidik jari pengguna dengan akurat dan stabil, kemudian data dikirimkan ke mikrokontroler melalui komunikasi *UART*.



Gambar 8. Rangkaian Sistem Absensi Menggunakan NodeMCU, Sensor *Fingerprint*, dan *OLED Display*



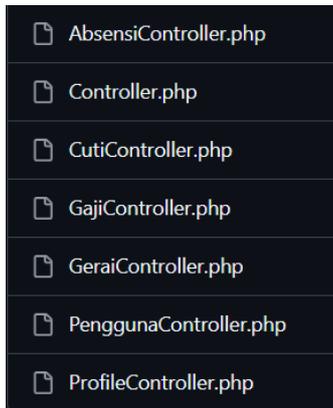
Gambar 9. Konfigurasi Perangkat Keras Sistem Absensi dengan Sensor Sidik Jari

2. Perangkat Lunak

Pada tahap implementasi perangkat lunak mencakup penerapan komponen sistem berdasarkan hasil desain yang telah dibuat pada tahap sebelumnya, baik dari sisi antarmuka pengguna, fungsionalitas sistem, hingga kode untuk mengintegrasikan sistem dengan perangkat keras yaitu sensor sidik jari sehingga hasil identifikasi sidik jari dapat ditampilkan pada website pengguna.

Salah satu tahap dalam mengimplementasikan perangkat lunak adalah dengan membuat *Application Programming Interface* atau yang kita kenal dengan *API*. *API* memiliki peran sebagai jembatan komunikasi antara antarmuka pengguna, server dan perangkat *IoT* yang dibangun. *API* memungkinkan pertukaran data secara *real-time* sehingga tiap komponen yang dikembangkan dapat saling terintegrasi dengan baik. Terdapat endpoint

untuk mengelola data absensi, mengelola pengajuan cuti, mengelola informasi gaji, mengelola informasi gerai, mengelola data pengguna, serta menampilkan dan memperbarui data profil pengguna.

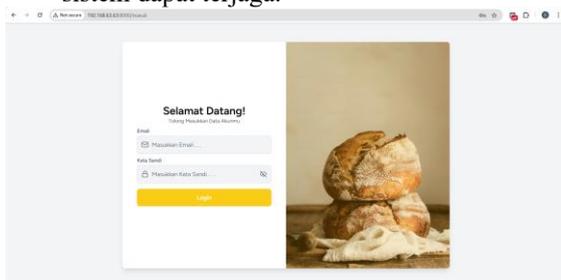


Gambar 10. API

Setelah membuat API untuk menghubungkan sistem, tahap selanjutnya adalah mengembangkan *frontend* sebagai antarmuka pengguna. Antarmuka pengguna adalah komponen yang penting dalam sebuah sistem karena akan menjadi titik interaksi antara pengguna dan aplikasi yang dibuat. Pada bagian ini, akan dijabarkan rancangan serta hasil implementasi antarmuka pengguna yang dikembangkan dalam sistem monitoring pegawai berbasis IoT. Antarmuka dirancang dengan memperhatikan kemudahan penggunaan (*usability*), keterbacaan informasi, serta kesesuaian fungsi dengan kebutuhan pengguna.

a. Tampilan Login

Halaman *Login* adalah halaman awal bagi pengguna sebelum dapat masuk ke dalam sistem. Pada halaman ini, pengguna diminta untuk memasukkan kredensial berupa *username* dan *password* yang sudah terdaftar pada *database*. Fitur ini berfungsi untuk membatasi akses hanya pada pengguna yang memiliki izin, sehingga keamanan data dalam sistem dapat terjaga.

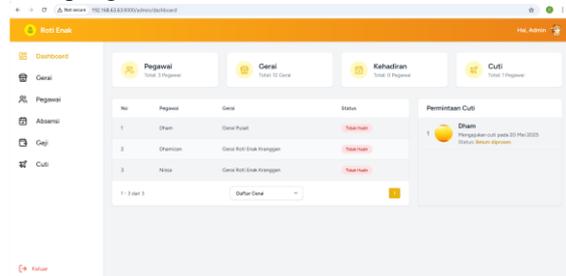


Gambar 11. Tampilan halaman Login

b. Tampilan Dashboard Admin

Halaman *Dashboard* adalah halaman utama yang muncul setelah admin berhasil masuk pada sistem. Halaman ini menyajikan

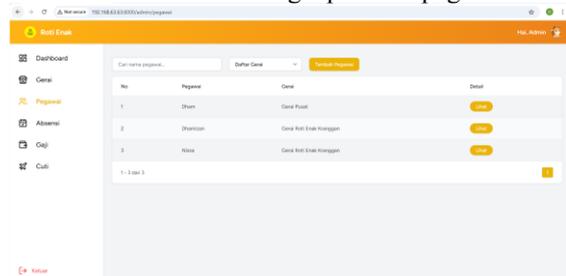
menu navigasi yang memudahkan admin dalam mengelola seluruh sistem. Melalui *dashboard*, admin dapat mengetahui data pegawai, data absensi, data gaji, serta data pengajuan cuti oleh pegawai. Tujuan halaman ini adalah memberikan kemudahan bagi admin dalam menjalankan fungsi pengawasan dan pengelolaan data.



Gambar 12. Tampilan Dashboard admin

c. Tampilan Data Pegawai

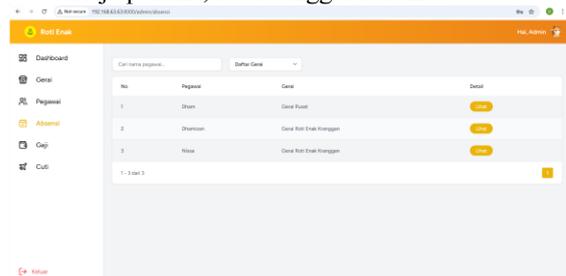
Halaman Data Pegawai adalah halaman yang menampilkan seluruh informasi yang sudah terdaftar pada sistem. Informasi yang ditampilkan berupa nama pegawai, letak penempatan, serta admin dapat mengubah, menambah atau menghapus data pegawai.



Gambar 13. Tampilan Data Pegawai

d. Tampilan Absensi pada Admin

Pada halaman absensi pegawai pada admin, admin dapat memantau dan merekap kehadiran pegawai dalam sistem. Data yang ditampilkan meliputi nama pegawai, letak penempatan, dan detail dari absensi seperti jam masuk pegawai, jam keluar, total jam kerja perhari, serta tanggal absensi.

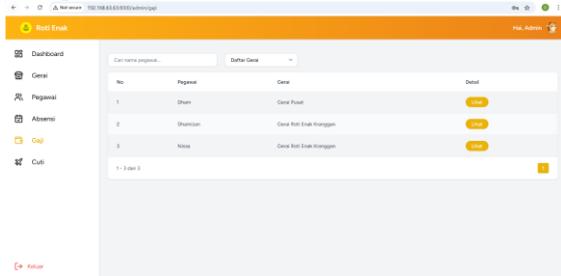


Gambar 14. Tampilan Absensi pada Admin

e. Tampilan Gaji Pegawai

Halaman gaji pegawai pada sisi admin bertujuan untuk menampilkan informasi

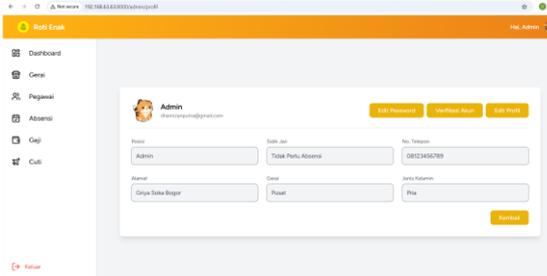
penggajian setiap pegawai. Terdapat nama pegawai, letak penempatan, serta detail gaji. Pada detail gaji terdapat jumlah gaji karyawan perharinya.



Gambar 15. Tampilan Gaji Pegawai

f. Tampilan Profil

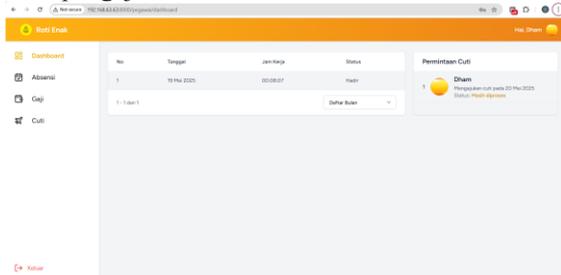
Pada halaman profil, terdapat rincian informasi akun yang sedang login, seperti posisi dalam sistem (admin/pegawai), alamat, nomor telepon, jenis kelamin, penempatan gerai, serta verifikasi sidik jari. Pada halaman ini pengguna juga dapat mengedit informasi mengenai akun yang sedang login.



Gambar 16. Tampilan Profil Admin

g. Tampilan Dashboard Pegawai

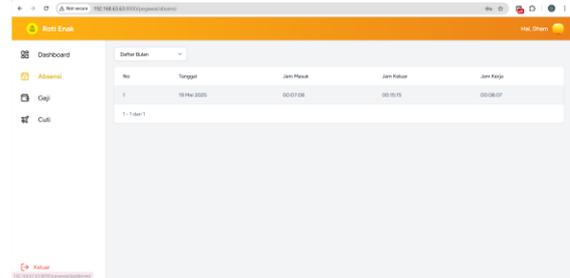
Halaman *dashboard* pada sisi pegawai adalah halaman utama yang muncul setelah pegawai berhasil memasukkan data yang telah terverifikasi pada halaman *login*. Pada halaman ini, pegawai dapat mengetahui status kehadirannya, total jam kerja, serta status pengajuan cuti.



Gambar 17. Tampilan Dashboard Pegawai

h. Tampilan Absensi Pegawai

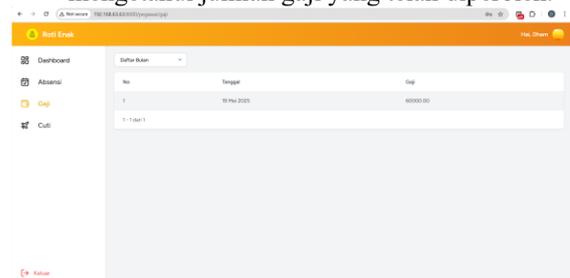
Pada halaman ini, pegawai dapat melihat informasi mengenai absensinya. Terdapat informasi mengenai tanggal absensi, jam masuk, jam keluar, dan total jam kerja pada tanggal tersebut.



Gambar 18. Tampilan Absensi Pegawai

i. Tampilan Gaji Pegawai

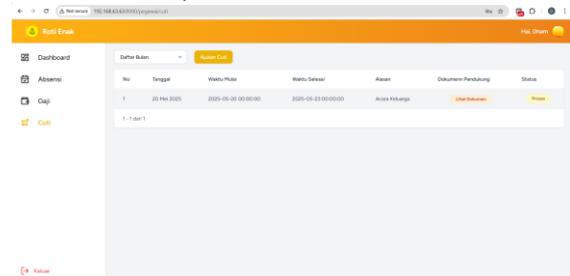
Pada halaman ini, pegawai dapat melihat gaji yang diperolehnya selama sehari. Halaman ini dibuat agar pegawai dapat mengetahui jumlah gaji yang telah diperoleh.



Gambar 19. Tampilan Gaji Pegawai

j. Tampilan Cuti Pegawai

Halaman cuti pegawai menampilkan daftar pengajuan cuti yang telah diajukan. Pegawai dapat melihat pengajuan cuti mereka diterima, ditolak, atau masih dalam proses. Selain itu, pegawai juga dapat mengajukan form cuti dimana pegawai harus mengisi tanggal cuti, alasan cuti, dan bukti cuti.



Gambar 20. Tampilan Cuti Pegawai

Setelah mengembangkan *frontend* sebagai antarmuka pengguna, maka langkah terakhir yang dapat dilakukan adalah menguji kemampuan perangkat lunak dalam menerima data dari perangkat keras melalui *API*. Setelah *fingerp* dikenali perangkat keras, data identitas pengguna secara otomatis dikirim ke server dan ditampilkan di halaman website. Dengan keberhasilan ini, maka menandakan bahwa integrasi sistem antara *firmware ESP8266*, *API Laravel*, dan antarmuka *Vue.js* telah berjalan dengan baik.

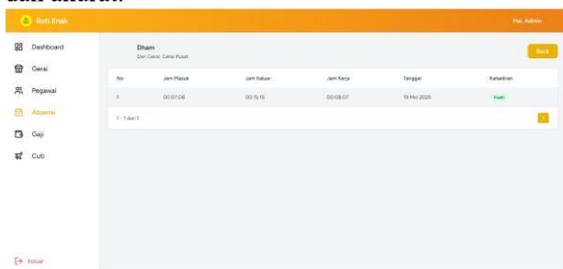
Gambar ini menunjukkan hasil deteksi sidik jari oleh sensor *fingerp* yang telah terhubung ke

NodeMCU *ESP8266* dan dipantau melalui serial monitor pada *Arduino IDE*. Ketika pengguna meletakkan sidik jari pada sensor, maka sistem berhasil mengenali identitas pengguna dan mengirimkan data absensi tersebut ke server melalui koneksi internet. Informasi yang dikirim mencakup nama pegawai, waktu absensi, status kehadiran, serta durasi bekerja. Hal ini menunjukkan bahwa integrasi antara perangkat keras dan perangkat lunak telah berjalan dengan baik.



Gambar 21. Tampilan Serial Monitor pada Arduino IDE setelah Sensor *Fingerprint* Mendeteksi Sidik Jari

Gambar ini adalah tampilan *frontend* pada halaman absensi dibagian admin setelah data absensi yang dikirim dari sensor *fingerprint* berhasil tersimpan di dalam sistem dan ditampilkan ke dalam antarmuka web. Sistem ini menampilkan data nama pegawai, jam masuk, jam keluar, total jam kerja, tanggal absensi, dan status kehadiran. Keberhasilan ini menunjukkan bahwa proses integrasi data dari alat ke server dan dari server ke tampilan pengguna telah berjalan secara *real-time* dan akurat.



Gambar 22. Tampilan Antarmuka Website setelah Data Absensi dari Sidik Jari Terdeteksi dan Tersimpan pada Sistem

## V. KESIMPULAN DAN SARAN

### Kesimpulan

Berdasarkan hasil dari pengembangan sistem yang telah dibuat, kesimpulan yang didapat adalah:

1. Sistem berhasil dikembangkan menggunakan metode pendekatan *Agile Software Development Life Cycle* yang mampu meningkatkan fleksibilitas dan iterasi cepat dalam menyesuaikan kebutuhan mitra.
2. Implementasi sistem absensi dengan sensor sidik jari berbasis *IoT* dan mikrokontroler NodeMCU *ESP8266* mampu mencatat kehadiran secara *real-time* melalui koneksi Wi-Fi ke server *Laravel*.

3. Sistem yang dibangun memiliki kelebihan dalam efisiensi waktu pencatatan absensi, peningkatan keakuratan data, dan kemudahan bagi pusat untuk memonitoring pegawai.

### Saran

Berdasarkan hasil implementasi sistem absensi dengan sensor sidik jari berbasis *IoT*, terdapat beberapa hal yang dapat dikembangkan lebih lanjut. Saran-saran berikut disusun sebagai acuan bagi peneliti selanjutnya untuk meningkatkan efektivitas dan efisiensi dari sistem yang telah dibangun:

1. Sistem dapat dikembangkan dengan fitur notifikasi secara *real-time* melalui *WhatsApp* untuk menginformasikan kehadiran secara langsung kepada admin.
2. Penelitian lanjutan pada penggunaan modul hemat daya pada *ESP8266* untuk mengurangi konsumsi energi.

## DAFTAR PUSTAKA

- Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., ... Thomas, D. (2001). *Manifesto for Agile Software Development*. <https://agilemanifesto.org/>
- Budi, S., & Prasetyo, Y. (2018). Sistem absensi sidik jari menggunakan sensor fingerprint dan *ESP8266* berbasis *IoT*. *Jurnal Ilmiah Teknologi dan Informasi*, 5(2), 120–126.
- Fitri, R. (2017). Sistem informasi absensi pegawai berbasis web. *Jurnal Ilmiah Komputer dan Informatika (KOMPUTA)*, 6(1).
- Hidayat, T., Supriyanto, A., & Maulana, I. (2020). Penerapan *IoT* dalam sistem absensi otomatis menggunakan sidik jari. *Jurnal Teknologi Informasi dan Komunikasi*, 12(2).
- Huda, M., & Azizah, N. (2021). Perancangan sistem absensi menggunakan sensor sidik jari dan mikrokontroler *ESP8266* berbasis *IoT*. *Jurnal Teknologi dan Informasi*, 9(2), 105–112.
- Kusnady, D. (2017). Analisis dan perancangan use case diagram dalam sistem informasi. *Jurnal Ilmiah Komputer dan Informatika (KOMPUTA)*, 6(2), 77–83.
- Maulana, I., & Sari, R. (2019). Analisis penggunaan *Vue.js* dalam pengembangan antarmuka aplikasi web. *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIK)*, 6(2), 205–212.
- Pressman, R. S. (2010). *Software engineering: A practitioner's approach* (7th ed.). New York, NY: McGraw-Hill.
- Putra, A. R., Lestari, D., & Ramadhan, F. (2021). Implementasi *IoT* pada sistem absensi berbasis fingerprint dengan NodeMCU. *Jurnal Teknologi dan Sistem Komputer*, 9(3).

- Rakhman, A., & Syahputra, H. (2020). Perancangan sistem informasi berbasis web menggunakan framework Laravel dan Vue.js. *Jurnal Teknologi Informasi*, 13(1), 33–40.
- Ramadani, R., & Amalia, A. R. (2021). Penerapan metode agile dalam pengembangan perangkat lunak. *Jurnal Teknologi dan Sistem Informasi*, 2(1), 1–8.
- Ramadhan, R., & Sari, D. N. (2020). Perancangan dan implementasi sistem informasi berbasis Laravel. *Jurnal Teknologi dan Sistem Komputer*, 8(3), 209–214.
- Sari, N. R., & Nugroho, D. (2019). Perancangan sistem absensi pegawai berbasis web dengan menggunakan fingerprint. *Jurnal Informatika*, 10(1).
- Sommerville, I. (2011). *Software engineering* (9th ed.). Boston, MA: Addison-Wesley.
- Sudaryono. (2021). *Pemrograman mikrokontroler dengan ESP8266 dan Arduino IDE*. Jakarta, Indonesia: Elex Media Komputindo