

**APLIKASI MANAJEMEN PRESENSI GURU TK DARMA
BAKTI DENGAN TEKNOLOGI RFID**

PROYEK AKHIR

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



Disusun Oleh :

Devid Wijaya

NPM 1062138

**POLITEKNIK MANUFAKTUR NEGERI
BANGKA BELITUNG
TAHUN 2024**

LEMBAR PENGESAHAN

**APLIKASI MANAJEMEN PRESENSI GURU TK
DARMA BAKTI DENGAN TEKNOLOGI RFID**

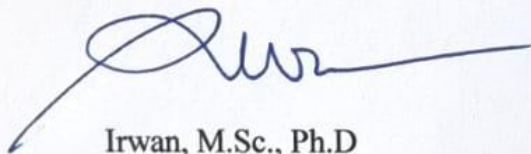
Oleh :

DEVID WIJAYA /1062138

Laporan akhir ini telah disetujui dan disahkan sebagai salah satu
syarat kelulusan Program Sarjana Terapan Politeknik Manufaktur
Negeri Bangka Belitung

Menyetujui,

Pembimbing 1



Irwan, M.Sc., Ph.D

NIP. 197604182014041001

Pembimbing 2



Muhammad Setya Pratama, M.Si

NIP.199208212019031021

Penguji 1

Penguji 2

PERNYATAAN BUKAN PLAGIAT

Yang bertanda tangan di bawah ini :

Nama Mahasiswa : Devid Wijaya NIM : 1062138

Dengan Judul : :APLIKASI MANAJEMEN PRESENSI GURU TK
DARMA BAKTI DENGAN TEKNOLOGI RFID

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

Nama Mahasiswa

Sungailiat, 24 Juli 2024

Tanda Tangan

1. Devid Wijaya



.....

ABSTRAK

Teknologi RFID dan pengenalan wajah semakin populer dalam meningkatkan efisiensi dan akurasi sistem absensi di berbagai institusi pendidikan. Di TK Darma Bakti, sistem absensi manual yang ada sering kali mengalami ketidakakuratan data dan inefisiensi operasional, yang menghambat kelancaran administrasi sekolah. Penelitian ini bertujuan untuk mengembangkan dan menguji aplikasi absensi dan manajemen presensi untuk TK Darma Bakti menggunakan teknologi RFID dan pengenalan wajah. Aplikasi ini dikembangkan dengan metode prototipe, yang melibatkan tahapan komunikasi, perencanaan cepat, desain cepat, konstruksi prototipe, serta pengiriman dan umpan balik dari pengguna. Aplikasi yang dihasilkan menyediakan fitur absensi otomatis menggunakan RFID dan verifikasi identitas melalui pengenalan wajah, yang bertujuan untuk mengatasi masalah ketidakakuratan dan inefisiensi pada sistem absensi manual yang ada. Selain itu, aplikasi ini juga dilengkapi dengan sistem pemantauan riwayat presensi dan pengelolaan izin atau cuti secara efisien. Uji validasi aplikasi dilakukan melalui pengujian User Acceptance Testing (UAT) yang melibatkan tiga puluh tiga responden. Hasil pengujian menunjukkan total skor 1489 dengan persentase 90,24%, yang mengindikasikan bahwa aplikasi ini sangat efektif dan sesuai dengan kebutuhan pengguna.

Kata Kunci : Absensi, Manajemen Presensi, Pengenalan Wajah, RFID

ABSTRACT

RFID and facial recognition technologies are increasingly popular in improving the efficiency and accuracy of attendance systems in various educational institutions. At Darma Bakti Kindergarten, the existing manual attendance system often experiences data inaccuracies and operational inefficiencies, which hamper the smooth running of school administration. This research aims to develop and test an attendance and attendance management application for Darma Bakti Kindergarten using RFID and face recognition technology. The application was developed using the prototyping method, which involves the stages of communication, rapid planning, rapid design, prototype construction, and delivery and feedback from users. The resulting application provides automatic attendance features using RFID and identity verification through facial recognition, which aims to overcome the problems of inaccuracy and inefficiency in the existing manual attendance system. In addition, the application is also equipped with an efficient attendance history monitoring system and permit or leave management. The application validation test was conducted through User Acceptance Testing (UAT) involving thirty-three respondents. The test results showed a total score of 1489 with a percentage of 90.24%, which indicates that this application is very effective and in accordance with user needs.

Keywords: Attendance, Presence Management, RFID, Face Recognition

KATA PENGANTAR

Puji dan syukur penulis panjatkan kepada Tuhan Yang Maha Esa yang selalu memberika rahmat, karunia, nikmat dan hidayah-Nya, sehingga penulis dapat menyelesaikan proyek akhir dan penyusunan laporan proyek akhir dengan judul “Aplikasi Manajemen Presensi Guru TK Darma Bakti dengan Teknologi RFID”.

Penyusunan proyek akhir ini merupakan salah satu persyaratan untuk menyelesaikan pendidikan Diploma IV di Politeknik Manufaktur Negeri Bangka Belitung. Selama proses penyusunan, banyak pihak yang memberikan bantuan sehingga proyek akhir ini dapat berjalan dengan lancar. Pada kesempatan ini, penulis ingin menyampaikan rasa terima kasih yang sebesar-besarnya kepada semua pihak yang telah membantu penulis dari tahap awal hingga tahap selesai. Ucapan terima kasih juga ditujukan kepada rekan-rekan atas dukungan, bantuan moril dan material sehingga penulisan makalah ini dapat diselesaikan dengan baik. Oleh karena nya penulis ingin mengucapkan terima kasih kepada :

1. Ucapan syukur kepada Tuhan Yang Maha Esa.
2. Ucapan terkhusus untuk kedua Orang Tua serta Keluarga yang selalu senantiasa memberikan perhatian serta doa yang penuh dengan keikhlasan serta dukungan baik secara moril ataupun materil.
3. Bapak Irwan, M.Sc. P.hD selaku pembimbing I
4. Bapak Muhammad Setya Pratama, M.Si. selaku pembimbing II
5. Teman seperjuangan DIV Teknologi Rekayasa Perangkat Lunak yang telah bersedia membantu baik suka dan duka bersama-sama dalam pengerjaan proyek akhir.
6. Semua pihak yang telah berkenan untuk membantu penulis yang tidak dapat disebutkan satu persatu, baik itu secara langsung ataupun bantuan tidak langsung.

Dalam penyusunan laporan proyek akhir ini, penulis menyadari bahwa masih banyak sekali kekurangan serta kesalahan, maka penulis mohon maaf yang sebesar-besarnya. Hal ini dikarenakan keterbatasan, kemampuanm dan juga pemahaman

yang kurang. Penulis juga sangat antusias dan terima baik atas kritik dan saran apabila terdapat kesalahan penulisan pada laporan proyek akhir. Hal ini demi perbaikan yang bersifat membangun serta memberikan pemahaman yang lebih baik yang belum penulis begitu mengerti dan pahami.

Demikian laporan yang penulis buat, penulis mengucapkan terima kasih dan berharap semoga dengan adanya proyek akhir yang penulis buat dapat menambah wawasan untuk kita semua dan teruntuk para pembaca.

Sungailiat, 24 Juli 2024



Penulis

DAFTAR ISI

PERNYATAAN BUKAN PLAGIAT	iii
ABSTRAK	iv
ABSTRACT	v
KATA PENGANTAR	vi
DAFTAR ISI	viii
DAFTAR TABEL	xi
DAFTAR GAMBAR	xii
DAFTAR LAMPIRAN	xiii
BAB I PENDAHULUAN	1
1.1. Latar Belakang.....	1
1.2. Perumusan Masalah	2
1.3. Tujuan	2
BAB II LANDASAN TEORI	3
2.1. Tinjauan Pustaka.....	3
2.2 Sistem Absensi	5
2.3 Manajemen Presensi.....	6
2.4 Teknologi RFID (Radio Frequency Identification).....	6
Tabel 2.2 Perbandingan Teknologi RFID dengan Teknologi Identifikasi Lainnya	7
2.4.1 RFID TAG (TRANSPONDER).....	8
Gambar 2.1 RFID Tag.....	9
2.4.2 RFID READER.....	10
Gambar 2.2 Blok Diagram Reader [17]	10
2.4 Internet of Things (IoT)	10
BAB III METODE PELAKSANAAN	13
3.1 Metode Pengembangan	13
3.1.1 Communication	13
3.1.2 Quick Plan.....	14
3.1.3 Modeling Quick Design	15
Gambar 3.4 Mockup Dashboard Admin	20
Gambar 3.5 Mockup Halaman Perizinan	20

Gambar 3.6 Mockup Halaman Histori Absensi	21
3.1.4 Construction of Prototype	21
3.1.5 <i>Deployment Delivery and Feedback</i>	22
3.2 Evaluasi dan Perbaikan	25
BAB IV PEMBAHASAN	26
4.1 Analisa Kebutuhan Sistem	26
Gambar 4.1 Use Case Diagram	27
4.2 Implementasi Sistem Manajemen Presensi	27
4.2.1 Pengembangan Konstruksi RFID	27
4.2.2 Pengembangan Sistem Kontrol	28
Tabel 4.1 Pengujian <i>Reader</i> dan Tag RFID	31
Gambar 4.5 Rancangan Monitoring Sistem Manajemen Presensi	32
4.2.3 Sistem Informasi Manajemen Presensi	32
Gambar 4.6 Entity Relationship Diagram	33
Gambar 4.8 Tabel Data Izin	34
Gambar 4.9 Tabel Data Pegawai	34
Gambar 4.10 Tabel Database <i>get_uid</i>	35
Gambar 4.11 Login Panel Admin	36
Gambar 4.12 Halaman Data Guru	37
Gambar 4.13 Form Pengisian Data Guru Baru	38
Gambar 4.15 Menu Perizinan Tidak Disetujui	39
Gambar 4.16 Menu Perizinan Menunggu Persetujuan	40
Gambar 4.17 Menu Histori Presensi Harian	40
Gambar 4.18 Menu Historu Presensi Bulanan	41
4.2.3.4 Implementasi Interface Aplikasi Android	41
Gambar 4.19 Landing Page	42
Gambar 4.22 Riwayat Absensi	45
4.2.3.5 Pengujian Sistem Informasi dengan Metode Black Box	51
Tabel 4.2 Pengujian Black Box	52
Tabel 4.3 Pengujian Black Box	52

4.2.3.6 Pengolahan Image atau Gambar	53
Gambar 4.29 Tahapan Pengolahan Gambar.....	54
4.2.3.7 Pengujian Pengolahan Gambar.....	54
Tabel 4.3 Hasil Pengujian Pengambilan Gambar ESP32-CAM	55
4.2.3.8 Sistem Keseluruhan	55
4.2.3.9 Pengujian Sistem Keseluruhan	55
4.2.3.9.1 Pengujian RFID Sistem Manajemen Presensi	56
Gambar 4.30 Data Guru TK Darma Bakti	56
4.2.3.10 Pengujian Sistem	62
Tabel 4.5 Jumlah Jawaban Kuesioner	62
5.1 Kesimpulan.....	64
5.2 Saran.....	65
Daftar Pustaka	66
LAMPIRAN 1	70
DAFTAR RIWAYAT HIDUP.....	70
LAMPIRAN 2	76
KODE PROGRAM.....	76
LAMPIRAN 3	216
DOKUMENTASI KEGIATAN.....	216

DAFTAR TABEL

Tabel 2.1 Perbandingan Teknologi RFID dengan Teknologi Identifikasi Lainnya	7
Tabel 3.1 Skala Likert.....	23
Tabel 3.2 Pernyataan Persentase Skala Likert.....	24
Tabel 3.3 Pertanyaan Skala Penelitian.....	24
Tabel 4.1 Pengujian <i>Reader</i> & Tag RFID.....	31
Tabel 4.2 Pengujian Black Box.....	52
Tabel 4.3 Hasil Pengujian Pengambilan Gambar ESP32-CAM.....	55
Tabel 4.4 Pengujian ID RFID	56
Tabel 4.5 Jumlah Jawaban Kuesioner.....	62

DAFTAR GAMBAR

Gambar 2.1 RFID Tag.....	9
Gambar 2.2 Blok Diagram Reader [17]	10
Gambar 3.1 Tahap Pelaksanaan Metode Prototype.....	13
Gambar 3.2 Blok Diagram Sistem Manajemen Presensi.....	16
Gambar 3.3 Blok Diagram Perancangan Hardware.....	17
Gambar 3.4 Mockup Dashboard Admin	20
Gambar 3.5 Mockup Halaman Perizinan	20
Gambar 3.6 Mockup Halaman Histori Absensi	21
Gambar 4.1 Use Case Diagram.....	27
Gambar 4.5 Rancangan Monitoring Sistem Manajemen Presensi.....	32
Gambar 4.6 Entity Relationship Diagram.....	33
Gambar 4.8 Tabel Data Izin	34
Gambar 4.9 Tabel Data Pegawai.....	34
Gambar 4.10 Tabel Database get_uid	35
Gambar 4.11 Login Panel Admin	36
Gambar 4.12 Halaman Data Guru.....	37
Gambar 4.13 Form Pengisian Data Guru Baru	38
Gambar 4.15 Menu Perizinan Tidak Disetujui	39
Gambar 4.16 Menu Perizinan Menunggu Persetujuan	40
Gambar 4.17 Menu Histori Presensi Harian	40
Gambar 4.18 Menu Histori Presensi Bulanan.....	41
Gambar 4.19 Landing Page.....	42
Gambar 4.22 Riwayat Absensi.....	45
Gambar 4.29 Tahapan Pengolahan Gambar.....	54
Gambar 4.30 Data Guru TK Darma Bakti	56

DAFTAR LAMPIRAN

LAMPIRAN 1 DAFTAR RIWAYAT HIDUP.....	70
LAMPIRAN 2 PENGUJIAN BLACKBOX TESTING ADMIN & USER.....	71
LAMPIRAN 3 PENGUJIAN UAT.....	72
LAMPIRAN 4 KODE PROGRAM.....	75
LAMPIRAN 5 DOKUMENTASI KEGIATAN.....	215



BAB I

PENDAHULUAN

1.1. Latar Belakang

Peran utama dalam pembentukan mutu pendidikan terletak pada guru, sehingga mutu pendidikan secara signifikan dipengaruhi oleh kualitas pendidik dalam pelaksanaan tugas dan tanggung jawab mereka dalam masyarakat[1]. Sama seperti guru pada umumnya, guru Taman Kanak-Kanak adalah pendidik profesional dengan tugas utama mendidik, mengajar, membimbing, mengarahkan, melatih, menilai, dan mengevaluasi peserta didik[2]. Namun, masih ada tantangan dalam manajemen kehadiran dan presensi guru TK yang dapat memakan waktu dan berpotensi menimbulkan ketidakakuratan.

Menghadapi hal tersebut maka diperlukan upaya untuk meningkatkan efisiensi dan keamanan manajemen kehadiran, penggunaan teknologi modern dapat menjadi solusi yang efektif[3]. Teknologi Radio-Frequency Identification (RFID) dan pengenalan wajah telah terbukti memberikan solusi canggih untuk mengatasi tantangan ini. Integrasi kedua teknologi ini dapat membentuk dasar untuk aplikasi absensi dan manajemen presensi guru TK yang lebih modern, akurat, dan efisien. Absensi manual guru TK rentan terhadap kesalahan dan memakan waktu, yang dapat menyebabkan catatan kehadiran yang tidak akurat[4]. Selain itu, tidak adanya sistem terpadu untuk melacak dan mengelola presensi guru menyebabkan kesulitan dalam melacak riwayat absensi dan mengelola izin atau cuti.

Dalam rangka mengatasi permasalahan di atas, proyek ini bertujuan untuk mengembangkan "Aplikasi Absensi dan Manajemen Presensi Guru TK Darma Bakti dengan Teknologi RFID dan Pengenalan Wajah" Solusi ini dapat memberikan beberapa keuntungan, antara lain menggunakan RFID untuk absensi otomatis dan akurat, serta pengambilan gambar wajah untuk verifikasi identitas. Aplikasi ini juga mampu menyediakan sistem terpadu untuk melacak dan mengelola presensi guru secara efisien, dengan riwayat absensi yang dapat diakses dengan mudah untuk keperluan pelaporan dan pemantauan. Fitur pengelolaan izin dan cuti akan

diintegrasikan, memungkinkan proses persetujuan oleh admin untuk mengelola izin atau cuti guru secara efisien. Harapan proyek ini dapat meningkatkan keamanan, akurasi, dan efisiensi dalam manajemen presensi[5] guru TK, memberikan kontribusi positif terhadap lingkungan pendidikan anak usia dini.

1.2. Perumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan, terdapat beberapa masalah yang perlu dipecahkan, yaitu:

- a) Bagaimana cara meningkatkan ketepatan dan kecepatan proses absensi guru TK yang saat ini masih bergantung pada pencatatan manual?
- b) Bagaimana membuat perangkat absensi dengan teknologi RFID dan pengambilan gambar wajah?
- c) Bagaimana membuat aplikasi absensi mengelola presensi guru TK dengan pemantauan riwayat presensi dan penanganan izin atau cuti?

1.3. Tujuan

Berdasarkan rumusan masalah yang telah dijelaskan, tujuan dari penelitian ini adalah:

- a) Merancang sistem absensi berbasis RFID dan pengenalan wajah untuk meningkatkan ketepatan dan kecepatan proses absensi guru TK.
- b) Merancang dan mengimplementasikan perangkat absensi yang memanfaatkan teknologi RFID untuk otentikasi kehadiran guru dan pengambilan gambar wajah sebagai validasi identitas.
- c) Membuat fitur pemantauan riwayat presensi guru serta sistem pengelolaan izin atau cuti melalui aplikasi dengan persetujuan oleh admin.

BAB II

LANDASAN TEORI

2.1. Tinjauan Pustaka

Dalam penelitian ini, penulis melakukan tinjauan pustaka dengan mengkaji penelitian-penelitian sebelumnya yang relevan dengan topik penelitian ini. Berikut adalah beberapa hasil temuan dari penelitian sebelumnya, yang akan membantu dalam memberikan pemahaman mengenai topik yang dibahas. Penelitian-penelitian terdahulu ini akan memberikan sudut pandang dan temuan yang selanjutnya dapat digunakan sebagai acuan dan juga memberikan perbandingan dalam penelitian proyek akhir ini.

Penelitian mengenai teknologi absensi berbasis RFID telah menunjukkan berbagai solusi dan pendekatan untuk meningkatkan efisiensi manajemen kehadiran. Misalnya, penelitian yang berjudul "Desain Sistem Absensi PNLs Berbasis RFID" dilakukan karena sistem absensi manual yang digunakan oleh Badan Kepegawaian Daerah Provinsi Kalimantan Timur masih menggunakan tanda tangan harian. Penelitian ini menggunakan metode penelitian kualitatif dengan teknik pengumpulan data meliputi survei, wawancara, observasi, analisis dokumentasi, dan desain sistem [6].

Selain itu, penelitian "Aplikasi Absensi Penyiar Radio dengan Menggunakan Teknologi RFID Berbasis Web Studi Kasus Radio Polnes FM" mengungkap bahwa sistem manajemen absensi di Radio Polnes FM masih manual, di mana koordinator radio harus melihat berkas absensi yang dicatat dengan tulisan tangan. Penelitian ini menggunakan metode Research and Development (R&D) dengan teknik pengumpulan data seperti sampling, investigasi, wawancara, dan observasi [7]. Penelitian lainnya, "Rancang Bangun Alat Absensi Karyawan Menggunakan RFID dan ESP32CAM Berbasis Internet of Things", dilakukan untuk mempermudah monitoring absensi pekerja perusahaan atau karyawan. Metode penelitian yang digunakan adalah eksperimen dengan teknik pengumpulan data yang mencakup observasi, survei, analisis data, dan pengujian [8].

Pada penelitian berjudul "Desain Sistem Absensi Karyawan dengan Radio Frequency Identification Berbasis Web", ditemukan bahwa proses absensi pada CV Wiragati Media masih menggunakan sistem manual dengan alat cek clock, yang rentan terhadap manipulasi data kehadiran oleh karyawan. Teknik pengumpulan data dalam penelitian ini terdiri dari observasi dan wawancara[9]. Penelitian terakhir, "Perancangan Sistem Kehadiran Pendidik dan Tenaga Kependidikan Berbasis Mikrokontroler ESP32CAM", menyoroti permasalahan sistem kehadiran konvensional yang kurang efektif dan efisien. Penelitian ini berfokus pada perancangan alat dalam sistem kehadiran pendidik dan tenaga kependidikan dengan memanfaatkan pengenalan wajah sebagai data kehadiran. Metode yang digunakan adalah System Development Life Cycle (SDLC) dengan pengumpulan data melalui wawancara dan kajian pustaka[10].

Melalui kajian literatur yang telah penulis kaji, penulis mengeksplorasi berbagai penelitian terkini terkait sistem absensi pegawai dengan menggunakan RFID serta ESP32Cam. Penelitian-penelitian tersebut menunjukkan beberapa keunggulan signifikan, di antaranya pemanfaatan RFID sebagai metode absensi yang cepat dan efisien serta penerapan teknologi IoT untuk ESP32Cam. Meskipun demikian, kami mengidentifikasi beberapa kekurangan yang perlu dipertimbangkan dalam pengembangan sistem absensi berbasis RFID. Kelemahan yang dapat diidentifikasi dari tinjauan pustaka tersebut meliputi: beberapa penelitian masih menggunakan metode yang biasa; hanya mengandalkan absensi dengan RFID yang masih bisa dimanipulasi; beberapa penelitian tidak menambahkan fitur rekapitulasi absensi; beberapa penelitian juga kurang dalam hal kustomisasi atau manajemen data pengguna, yang dapat membatasi fleksibilitas pengguna dalam mengelola dan mengonfigurasi sistem; serta perangkat lunak yang dapat digunakan berbasis website atau spreadsheet.

Sementara itu, penelitian ini memiliki beberapa keunggulan di antaranya adalah menggunakan gabungan antara RFID dan ESP32Cam di mana jika melakukan absensi maka ESP32Cam otomatis mengambil gambar yang melakukan absensi, adanya rekapitulasi, manajemen akun multilevel, serta membuat sendiri perangkat lunak berbasis aplikasi untuk memudahkan pengolahan data di mana saja dan kapan

saja. Sebagai hasilnya, penelitian ini menawarkan solusi yang lebih maju, aman, dan sesuai dengan permintaan TK Darma Bakti.

2.2 Sistem Absensi

Sistem merupakan kumpulan dari elemen yang saling berinteraksi untuk mencapai sebuah tujuan tertentu. Sistem ini merepresentasikan suatu kejadian yang nyata seperti sebuah obyek nyata, tempat, benda serta orang-orang yang ada dan terjadi [11]. Kemudian absensi dapat diartikan sebagai suatu pendataan terhadap kehadiran yang merupakan bagian dari aktivitas pelaporan yang ada pada sebuah institusi [12]. Absensi juga dapat dimaknai sebagai ketidakhadiran, namun absensi juga bisa dikatakan sebagai ketidakhadiran atau kehadiran suatu entitas yang dalam hal ini adalah manusia, dimana manusia tersebut terlibat dalam suatu instansi yang mengharuskan adanya mekanisme pemberitahuan tentang keadaan atau kehadiran atau ketidakhadiran dalam lingkup instansi tersebut [13].

Dengan demikian Sistem Absensi dapat kita maknai sebagai suatu mekanisme atau sistem yang dirancang dengan tujuan untuk mencatat kehadiran individu, baik itu karyawan, siswa atau anggota instansi lainnya. Keberadaan dari sistem absensi merupakan sebuah hal yang penting dalam sebuah instansi, karena dengan adanya sistem absensi yang baik maka dapat membantu dalam mengendalikan proses penyelesaian pekerjaan sehingga didapatkan hasil maksimal dan sesuai dengan tujuan yang telah ditetapkan sebelumnya [14].

Keberadaan sistem absensi juga adalah sebuah hal yang penting mengingat dalam aspek institusi atau organisasi karena absensi dapat menjadi pertimbangan terhadap tindak lanjut serta keputusan bagi kelangsungan institusi atau organisasi terkait. Sistem absensi sendiri terdiri dari tiga jenis yakni : (1) Manual, dalam hal ini menggunakan kertas dan pena untuk mencatat kehadiran, jenis ini biasanya rentan terhadap kesalahan dan manipulasi, (2) Semi-Otomatis, menggunakan perangkat seperti kartu magnetic atau barcode yang dapat dibaca oleh mesin, dan (3) Otomatis, menggunakan teknologi canggih seperti biometric, RFID atau aplikasi berbasis *cloud* untuk pencatatan yang lebih akurat dan efisien [12].

2.3 Manajemen Presensi

Presensi merupakan kegiatan pencatatan terhadap setiap kehadiran pegawai dengan tujuan untuk melakukan pendaataan yang berkaitan dengan kehadiran masuk serta keluar pegawai secara berkala baik secara harian maupun bulanan [15]. Sementara itu manajemen presensi adalah proses yang melibatkan pengelolaan, pemantauan, pencatatan dan analisis kehadiran individu dalam suatu organisasi. Manajemen ini bertujuan untuk memastikan bahwa kehadiran dan ketidakhadiran dicatat dengan akurat dan informasi yang dihasilkan digunakan untuk berbagai keperluan seperti penilaian kinerja, penggajian dan lain sebagainya. Oleh karenanya manajemen presensi adalah komponen yang sangat penting dalam administrasi sebuah organisasi. [16].

Keberadaan manajemen presensi juga dapat memungkinkan organisasi memastikan bahwa setiap tindakan yang diambil dapat menjadi dasar yang kuat untuk pengambilan keputusan yang lebih baik dan menciptakan lingkungan kerja yang lebih produktif dan harmonis

2.4 Teknologi RFID (Radio Frequency Identification)

Radio Frequency Identification atau yang juga dikenal sebagai RFID merupakan sebuah metode untuk mengidentifikasi objek dengan menggunakan frekuensi dari transmisi radio [17]. Identifikasi frekuensi radio pada teknologi RFID berada pada kisaran 125 KHz, 13,65MHz atau 800-900MHz [18]. Pada teknologi RFID proses identifikasi dilakukan oleh RFID Reader dan RFID Transponder (RFID Tag). Mekanisme identifikasi dari teknologi ini sendiri dilakukan oleh RFID Reader dengan RFID Tag yang dilekatkan pada sebuah benda atau objek yang selanjutnya akan diidentifikasi. Pada sistem RFID, setiap tag atau transponder membawa sejumlah informasi unik yang didalamnya memuat serial number, model, warna, tempat perakitan serta data lainnya [19]. Karena serial unik pada setiap tagnya maka bisa dikatakan bahwa tidak ada RFID tag yang mempunyai serial number yang sama.

RFID sendiri merupakan salah satu teknologi Automatic Identification (AutoID) [20] sehingga dalam penggunaannya sangat mudah dan cocok untuk dioperasikan secara otomatis. Kombinasi dalam teknologi RFID adalah sebuah keunggulan yang tidak dimiliki oleh teknologi identifikasi yang lain [17]. Keunggulan inilah yang membuat RFID dapat disediakan pada perangkat yang hanya membaca saja (*Read Only*) atau dapat dibaca serta tulis (*Read/Write*), teknologi ini memungkinkan tidak perlu melakukan kontak langsung atau harus bergantung pada cahaya untuk dapat beroperasi, sehingga dapat dioperasikan dalam berbagai kondisi lingkungan dengan tingkat integritas data yang tinggi.

Teknologi pada RFID bekerja dengan menggabungkan fungsi dari kopling elektromagnetik atau elektrosatik pada frekuensi radio yang berasal dari spectrum elektromagnetik untuk mengidentifikasi objek [17]. Hal ini lah yang membuat teknologi RFID banyak digunakan dalam berbagai bidang seperti perusahaan, rumah sakit, hingga pendidikan karena kelebihan dan keunggulan yang ditawarkannya dibandingkan teknologi identifikasi yang lain. Tabel 2.1 menunjukkan keunggulan serta perbandingan dari teknologi RFID dengan teknologi identifikasi lainnya

Tabel 2.1 Perbandingan Teknologi RFID dengan Teknologi Identifikasi Lainnya [22]

System Parameter	OCR	Barcode	Voice Recognition	Smart Card	RFID
Typical Data Quantity (Bytes)	1-100	1-100	-	16-64 k	16-64 k
Data Density	Low	Low	High	Very High	Very High
Machine Readability	Good	Good	Expensive	Good	Good
Readibility by People	Limited	Simple	Difficult	Impossible	Impossible
Influence of Dirt/Damp	Very High	Very High	-	Possible	No Influence

Influence of Covering	Total Failure	Total Failure	Possible	-	No Influence
Influence of Direction and Position	Low	Low	-	Undirectional	No Influence
Degradation/ Wear	Limited	Limited	-	Contacts	No Influence
Purchase Cost	Very Low	Medium	Very High	Low	Medium
Operating Costs	Low	Low	None	Medium	None
Unauthorized Modification	Slight	Slight	Impossible	Impossible	Impossible
Reading Speed	Low ~4s	Low ~3s	Very Low >5-10s	Low ~4s	Very Fast ~0.5s
Maximum Distance	0-50cm	1cm Scanner	Direct Contact	Direct Contact	0-5m

2.4.1 RFID TAG (TRANSPONDER)

Tag merupakan perangkat yang melekat pada objek yang digunakan untuk diidentifikasi oleh perangkat *reader* [16]. Adapun objek yang ingin diidentifikasi oleh sistem RFID membutuhkan tag di dalamnya. Tag RFID didesain sedemikian rupa menggunakan teknologi moderen dan memiliki bentuk geometri terkecil dari proses silicon yang tersedia [17]. Pada Tag RFID biasanya memuat NVM (*non-volatile memory*) yang berfungsi untuk menyimpan data. Susanto (2009) mengklasifikasikan tag menjadi lima kelas yakni :

1. Class 0/I, Read Only, Factory Programmed

Jenis ini merupakan paling sederhana di kelasnya, dimana data akan ditulis sekali saja ketika dimanufaktur. Selanjutnya memori akan di nonaktifkan dari segala macam pembaruan yang ada.

2. Class II, Write Once Read Only, Factory or User Programmed

Tag ini diproduksi tanpa ada data yang tertulis di dalam memorinya. Data ditulis pada saat proses manufaktur tag atau dapat dilakukan juga oleh pengguna untuk satu kali penulisan. Setelah itu tag tidak akan bisa deprogram lagi, tetapi hanya dapat membaca saja.

3. Class III, Read Write

Class III merupakan tag yang fleksibel, karena pengguna memiliki akses terhadap penulisan dan pembacaan data ke dalam memori tag.

4. Class VI, Read Write with on Board Sensors

Tag jenis ini memiliki sensor onboard yang berguna untuk mereka parameter seperti temperature, tekanan udara serta pergerakan yang dapat direkam dengan menuliskan datanya ke dalam tag. Pembacaan parameter pada tag jenis ini dilakukan ketika terhubung dengan reader, tag yang digunakan juga bisa berjenis aktif atau semi-pasif.

5. Class V, Read Write with Integrated Transmitters

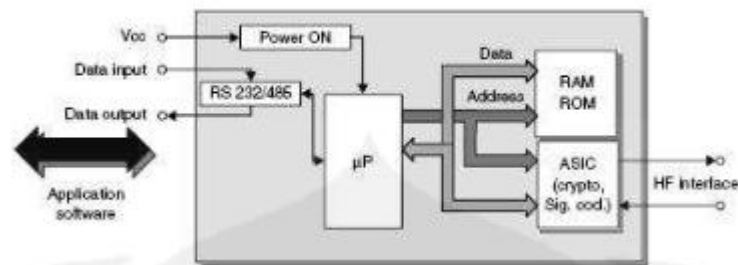
Tag jenis ini dapat berkomunikasi dengan tag dan devices lain tanpa memerlukan reader. Tag ini aktif dengan power dari baterai yang ada pada sistem mereka sendiri.



Gambar 2.1 RFID Tag

2.4.2 RFID READER

Reader merupakan alat yang berfungsi untuk memindai kode pada perangkat tag [16]. Reader sendiri terbagi menjadi dua jenis yakni pasif dan aktif. Pada reader pasif sistem hanya dapat menerima sinyal radio dari perangkat tag yang aktif sehingga hal ini memungkinkan sistem memiliki perlindungan sekaligus pengawasan terhadap aset. Sementara itu reader aktif sistem hanya dapat memancarkan sinyal interrogator saja menuju perangkat tag dan hanya dapat menerima balasan autentikasi yang berasal dari perangkat yang dituju. Terlepas dari aktif atau pasif reader memiliki dua blok fungsi utama seperti yang ditunjukkan pada Gambar 2.2



Gambar 2.2 Blok Diagram Reader [17]

2.4 Internet of Things (IoT)

Internet of Things yang selanjutnya dikenal dengan IoT merupakan teknologi yang pertama kali ditemukan oleh Kevin Ashton pada tahun 1999 [22]. Internet of Things sendiri merupakan penerapan teknologi elektronika modern yang memungkinkan untuk mengontrol, berkomunikasi dan mengintegrasikan perangkat keras melalui internet [23]. Keberadaan IoT memungkinkan adanya komunikasi dengan koneksi jaringan yang berskala global yang memungkinkan untuk tidak terjadinya interaksi antar individu atau interaksi manusia dengan komputer. Dalam Internet of Things terdapat unsur-unsur yang masuk ke dalamnya yakni koneksi

internet yang berperan sebagai media komunikasi, modul sensor yang berperan sebagai media fisik, kemudian perangkat nirkabel, kode QR yang berperan sebagai media pengumpulan data dan juga terdapat pusat data yang berfungsi sebagai media penyimpanan. Melalui unsur diatas maka Internet of Things dapat digunakan dengan mudah untuk mengidentifikasi objek serta modul sensor dan dapat dimanfaatkan juga sebagai aplikasi yang kolaboratif [24].

Keberadaan IoT yang sangat fungsional ini juga dapat dimanfaatkan untuk melakukan pendataan secara *real time* terhadap absensi suatu instansi. Pendataan absensi suatu instansi disini dapat memanfaatkan IoT yang selanjutnya diintegrasikan dengan teknologi RFID. Sistem absensi pada dasarnya bermula dari manual yang dilakukan pada secarik kertas, namun metode seperti ini sudah tidak efektif mengingat memungkinkan terjadinya manipulasi data serta rentan hilangnya data absensi. Penelitian yang dilakukan oleh Qomarul dkk yang membahas mengenai RFID berbasis IoT dengan judul “Prototype Sistem Absensi Menggunakan RFID Berbasis IoT” Dalam penelitiannya menghasilkan prototype sistem absensi menggunakan RFID berbasis IoT yang dapat mencatat dan mengirimkan data kehadiran secara realtime bagi dosen dan mahasiswa di STMK Lombok ke database server berdasarkan jadwal akademik yang telah dibuat [25].

Penelitian lain terkait penggunaan IoT dalam RFID yang berjudul “Sistem Alat Absensi Menggunakan RFID dan Camera Berbasis Internet of Things” yang membahas mengenai penggunaan kamera dalam RFID yang teritegrasi dengan IoT, penelitian ini menggunakan metode prototype. Penelitian ini menghasilkan bahwa teknologi RFID dapat menggunakan ESP32CAM yang digunakan untuk pengambilan wajah karyawan, kemudian data akan disimpan ke dalam database ssebagai laporan dan validasi data absensi karyawan [26]. Lalu terdapat juga penelitian yang dilakukan oleh Fitriyadi dan Hariono dengan judul “Perancangan Sistem Absensi Perkuliahan dengan Menggunakan RFID, Studi Kasus STMIK Banjarbaru” penelitian ini membahas pengujian teknologi RFID menggunakan 10 kartu mahasiswa. Pendekatan yang digunakan dalam penelitian ini adalah Research and Development [27].

Secara keseluruhan penelitian-penelitian yang telah ada sebelumnya menunjukkan bahwa IoT dan RFID memiliki keunggulan dan potensi yang besar dalam meningkatkan manajemen presensi, meminimalisir manipulasi data absensi serta menghilangkan resiko hilangnya data. Penggunaan teknologi RFID ini sendiri bisa digunakan dalam berbagai bidang kehidupan serta instansi yang membutuhkannya. Kolaborasi antara teknologi RFID dan IoT dapat menjadi solusi yang inovatif sekaligus cerdas dalam mengatasi berbagai tantangan yang ada dalam masyarakat modern.

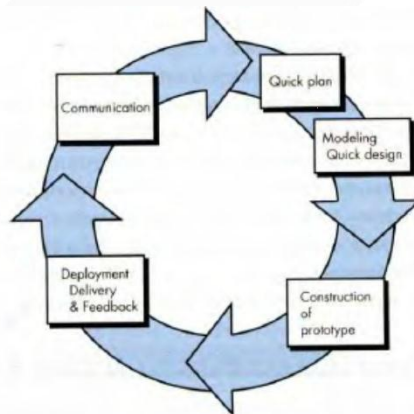


BAB III

METODE PELAKSANAAN

3.1 Metode Pengembangan

Metode pengembangan yang digunakan dalam pembuatan sistem manajemen presensi TK Darma Bakti adalah metode prototipe. Penggunaan metode ini memungkinkan sistem dikembangkan secara cepat dan bertahap, metode ini memungkinkan prototipe yang telah dihasilkan dapat segera ditinjau dan dievaluasi ulang oleh pengguna. Dengan menggunakan pendekatan ini, umpan balik yang diberikan oleh pengguna dapat langsung diterapkan untuk memperbaiki dan menyempurnakan sistem yang telah dibuat sebelum menuju tahap akhir. Tahapan dalam metode prototipe [29] dapat dilihat pada gambar 3.1 berikut ini



Gambar 3.1 Tahap Pelaksanaan Metode Prototype

3.1.1 Communication

Tahap ini dilaksanakan dengan melakukan komunikasi antara pengembang dan pengguna yang dalam hal ini adalah pihak TK Darma Bakti, tahap ini dilakukan dengan tujuan untuk mengumpulkan data awal, kemudian menganalisis kebutuhan sistem yang akan dirancang dan dibuat. Langkah ini dilakukan dengan tujuan untuk memberikan informasi kepada *developer* dan pengguna mengenai kebutuhan dari sistem yang akan dibangun. Metode pengumpulan data merupakan langkah yang digunakan untuk mengumpulkan data yang berkaitan dengan masalah yang sedang dihadapi. Pada tahap *communication* ini metode pengumpulan data yang penulis

lakukan selama proses pengembangan sistem manajemen presensi untuk guru TK Darma Bakti berbasis teknologi RFID adalah sebagai berikut :

1. Identifikasi Masalah dan Studi Literatur

Pada tahap ini dilakukan identifikasi terhadap permasalahan mengenai sistem absensi dan manajemen presensi yang terdapat di TK Darma Bakti. Adapun pengumpulan informasi dilakukan dengan mempelajari jurnal penelitian terkait untuk mengetahui prinsip kerja dan tujuan dari aplikasi manajemen presensi yang terintegrasi dengan teknologi RFID, serta untuk menentukan bagaimana cara untuk mengatasi permasalahan yang terjadi dan menjadikan permasalahan yang di hadapi sebagai landasan untuk melaksanakan penelitian.

Pencarian informasi mengenai setiap komponen yang akan digunakan juga dilakukan dengan tujuan ketika pembuatan proyek akhir ini dilaksanakan, maka dapat berjalan dengan lancar dan sistematis. Informasi tersebut mencakup penggunaan komunikasi serial dalam pemrograman *interface*, prinsip kerja mikrokontroller yang digunakan, software dan bahasa pemrograman yang akan digunakan untuk merancang sistem informasi berupa website serta aplikasi *smartphone* nantinya.

2. Observasi

Observasi dilakukan dengan melakukan kunjungan langsung menuju TK Darma Bakti. Pada tahap ini dilakukan pengamatan singkat mengenai mekanisme absensi yang biasanya dilakukan oleh guru TK Darma Bakti. Observasi ini dilakukan dengan tujuan untuk mendapatkan gambaran yang jelas bagaimana proses absensi dilaksanakan di TK Darma Bakti.

3.1.2 Quick Plan

Tahap *Quick Plan* ini akan berfokus pada identifikasi dan analisis terhadap kebutuhan sistem serta fitur-fitur yang diinginkan ada pada perangkat keras dan perangkat lunak yang akan dirancang. Tahapan ini melibatkan pengumpulan informasi spesifik mengenai spesifikasi sistem yang dibutuhkan serta fungsionalitas

sistem yang diinginkan. Langkah ini perlu dilaksanakan agar semua kebutuhan yang dibutuhkan dapat diakomodasikan dan dipahami dengan jelas sebelum pada akhirnya dilanjutkan ke tahap perancangan yang mendetail.

3.1.2.1 Kebutuhan Perangkat Keras dan Perangkat Lunak

Pada tahap ini dilakukan analisis terhadap kebutuhan yang diperlukan dalam perancangan sistem yang diantaranya meliputi spesifikasi fungsional setiap komponen yang akan digunakan, perangkat keras (*Hardware*) dan perangkat lunak (*Software*) yang akan digunakan. Analisa terhadap spesifikasi fungsional dari sistem ini yakni sistem mampu melakukan autentifikasi manajemen presensi guru berdasarkan data yang diperoleh dari RFID. Pada sistem yang dirancang ini digunakan beberapa perangkat keras diantaranya adalah Step Down LM2596, RFID Card dan Keychain Sensor Module, ESP 32, Baterai berukuran 3,7 Volt, Buzzer, Indikator Kapasitas Tegangan Baterai, serta ESP-32 CAM Module. Dalam proyek akhir ini perangkat lunak pada sistem ini yakni komunikasi dengan mikrokontroler, kemudian akses database menggunakan firebase serta pengolahan data gambar.

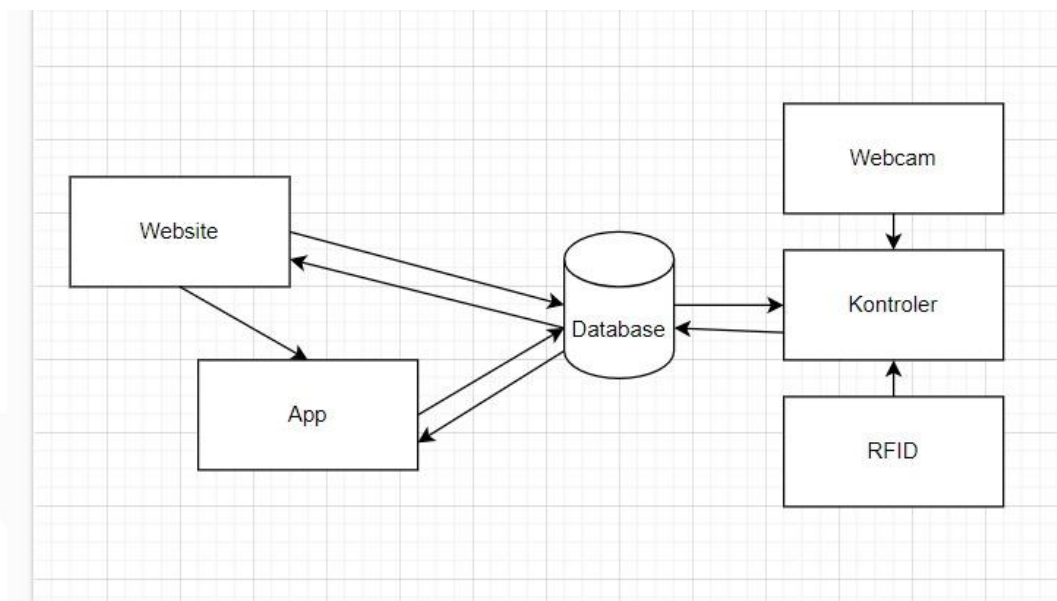
3.1.3 Modeling Quick Design

Modeling Quick Design merupakan tahap perancangan cepat yang menggambarkan secara detail mengenai alur kerja dari sistem yang akan dikembangkan. Berikut merupakan tahapan pelaksanaan yang akan dijelaskan dalam alur kerja sistem :

3.1.3.1 Pengembangan Sistem

Pengembangan sistem manajemen presensi untuk guru TK Darma bakti dilakukan berdasarkan analisa kebutuhan yang sudah dilakukan pada tahapan sebelumnya. Proses pengembangan sistem dilakukan dengan membuat perencanaan yang meliputi perancangan terhadap desain arsitektur sistem, mengidentifikasi komponen yang dibutuhkan, serta menentukan teknologi yang diperlukan. Pengembangan sistem ini terbagi menjadi beberapa tahap, yakni perancangan mekanisme penggunaan RFID, pengembangan sistem manajemen

presensi, pengolahan citra, pembuatan sistem informasi berbasis website dan pengembangan aplikasi android. Berikut merupakan desain rancangan sistem manajemen presensi guru TK Darma Bakti yang seluruh gambarannya dapat dilihat pada blok diagram 3.2 di bawah ini.



Gambar 3.2 Blok Diagram Sistem Manajemen Presensi

3.1.3.2 Perancangan *Hardware*

Pada tahap perancangan perangkat keras ini akan dilakukan perancangan perangkat keras yang meliputi pembuatan dan penentuan konstruksi RFID yang akan digunakan, dan pengembangan sistem kontrol. Langkah selanjutnya adalah memetakan bagaimana perangkat keras yang dirancang akan berkomunikasi satu sama lain, hal ini dilakukan dengan melibatkan pengembangan konstruksi RFID yang melibatkan pengembangan struktur fisik dari perangkat yang dikembangkan untuk selanjutnya dipastikan bahwa komponen yang digunakan dapat dipasang dan diletakkan pada konstruksi yang dirancang. Selain itu komunikasi antar perangkat juga dilaksanakan dengan melakukan pengembangan terhadap sistem kontrol yang meliputi tahap pengembangan hingga pengujian.

3.1.3.2.1 Pembuatan dan Penentuan Konstruksi RFID

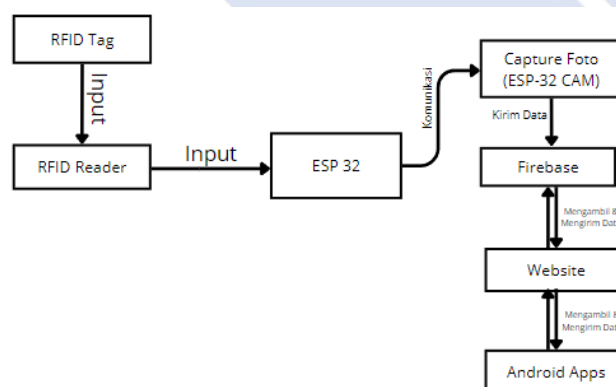
Dalam penelitian ini, komponen RFID yang digunakan disusun ke dalam sebuah box elektronik yang berukuran 14,5cm x 9,5cm x 5cm sebagai bagian dari konstruksi mekanik yang dirancang untuk sistem manajemen presensi. Seperti pada pembuatan blok diagram sebelumnya, penggunaan box elektronik ini akan mengintegrasikan teknologi RFID dalam implementasinya.

3.1.3.2.2 Peletakan Komponen Konstruksi RFID

Peletakan komponen elektrik untuk sistem manajemen presensi guru TK Darma Bakti yang digunakan akan diletakkan pada box dengan masing-masing komponen didalamnya adalah Step Down LM2596, RFID Card dan Keychain Sensor Module, ESP 32, Baterai berukuran 3,7 Volt, Buzzer, Indikator Kapasitas Tegangan Baterai, serta ESP-32 CAM Module Camera yang disusun dan diletakkan pada box elektrik yang diletakkan pada ruang guru TK Darma Bakti.

3.1.3.2.3 Pengembangan Sistem Kontrol

Perancangan sistem kontrol presensi guru TK Darma Bakti meliputi pada masukan data yang diperoleh dari RFID Tag serta masukan foto yang diperoleh melalui ESP-32 CAM yang dimana keluaran pada sistem berupa pengelolaan perizinan meninggalkan TK Darma Bakti oleh kepala sekolah. Skema rangkaian dalam perancangan sistem manajemen presensi dapat dilihat melalui blok diagram 3.3 berikut.



Gambar 3.3 Blok Diagram Perancangan Hardware

3.1.3.2.4 Perakitan dan Pengujian Sistem Kontrol

Setelah tahap perancangan skematik sistem kontrol selesai dilakukan, maka pada tahap ini akan dilakukan perakitan komponen berdasarkan pada skema rancangan yang telah dibuat sebelumnya. Adapun skematik rangkaian sistem yang terhubung dengan ESP32 terdiri dari RFID sedangkan ESP-32 CAM akan melakukan komunikasi dengan ESP 32 untuk menangkap data berupa gambar, kemudian Buzzer dan baterai sebagai daya juga akan terhubung dengan ESP32. Secara keseluruhan komponen diatas akan terhubung melalui komunikasi serial agar dapat saling berkomunikasi.

Pengujian pada sistem kontrol dilakukan pada kontrol RFID, Tag RFID dan RFID Reader, ESP32 dan ESP-32 CAM yang digunakan dengan tujuan untuk memastikan bahwa sistem RFID yang dibuat sudah dapat berfungsi dengan baik dan sesuai dengan kriteria yang telah dibuat sebelumnya. Pengujian ini dapat dilakukan dengan mencoba Tag RFID apakah berkomunikasi dengan RFID reader untuk membaca data untuk selanjutnya dikomunikasikan dengan ESP 32 dan ESP-32 CAM akan melakukan komunikasi juga dengan ESP 32 untuk menangkap data berupa gambar, jika berhasil data akan masuk ke dalam database dan dapat dikelola baik itu melalui website ataupun aplikasi android. Pengujian RFID juga dapat dilakukan dengan mengakses tag RFID pada jarak tertentu apakah RFID Reader mampu melakukan pembacaan apabila jaraknya divariasikan. Hal ini dilakukan dengan tujuan untuk memastikan apakah sistem dapat mendeteksi dan mengeluarkan keluaran yang sesuai dengan Tag RFID yang digunakan.

3.1.3.3 Perancangan Software

Tahap perancangan perangkat lunak meliputi melakukan pemodelan terhadap sistem yang akan dibangun dengan menggunakan diagram UML seperti *Use Case Diagram*, hingga pemodelan perangkat lunak menggunakan *mock-up* sebagai pengantar. Perancangan *mock-up* merupakan *interface* awal yang akan digunakan dalam perancangan aplikasi. Rancangan ini tidak hanya digunakan sebagai pedoman pembuatan tetapi juga merupakan dasar untuk membangun konstruksi pembuatan prototipe sistem manajemen presensi, dengan memastikan

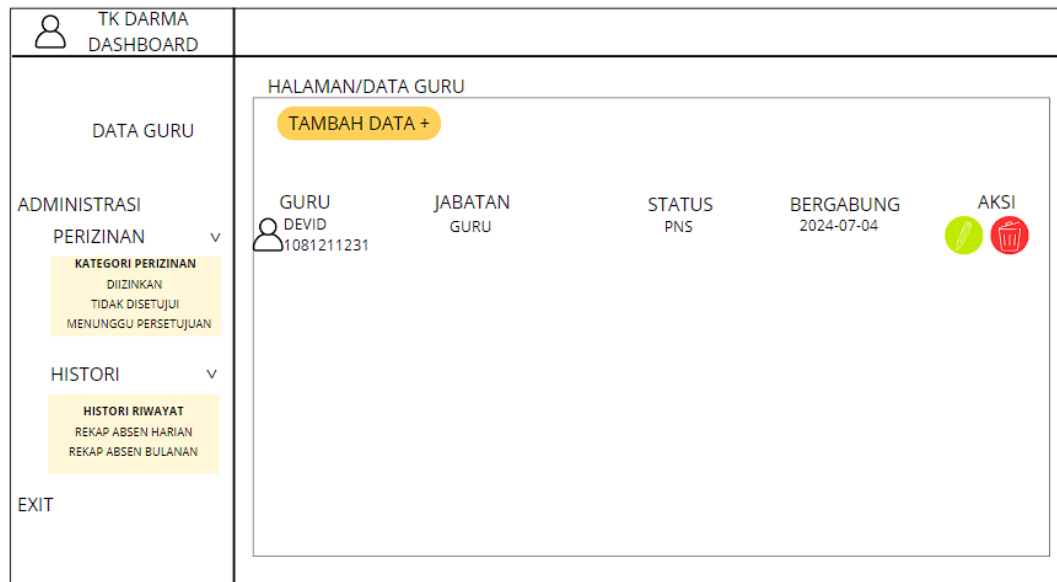
bahwa semua kebutuhan sistem yang telah dipetakan sebelumnya telah terlaksana dengan baik sebelum pada akhirnya dilakukan tahapan implementasi perancangan perangkat lunak. Dalam pengembangan sistem manajemen presensi untuk guru TK Darma Bakti mencakup tahapan sebagai berikut :

3.1.3.3.1 Perancangan Sistem Informasi

Proses pengembangan sistem kontrol presensi ini dimulai dengan pembuatan desain website sistem. Desain ini dirancang untuk mengakomodasi kebutuhan operasional yang spesifik, termasuk beberapa menu utama yang tersedia di halaman beranda. Menu-menu ini dirancang untuk memenuhi kebutuhan sistem secara efektif, yang mencakup :

- Data Guru : Memungkinkan admin dan pihak berkepentingan seperti kepala sekolah untuk menambahkan data guru dan mengelola perizinan mereka, baik yang disetujui, tidak disetujui, maupun yang sedang menunggu persetujuan.
- Info Presensi : Menampilkan informasi tentang presensi dan perizinan guru berdasarkan data yang diperoleh dari RFID Tag.
- Histori Presensi : Berisi rekap absen harian dan bulanan guru TK Darma Bakti untuk memudahkan monitoring dan evaluasi.

Sistem ini juga akan memperoleh dan menyimpan informasi melalui ESP-32 CAM ke dalam database, yang dimana data ini dapat diakses dan dikelola melalui antarmuka aplikasi android serta halaman website yang telah dirancang. Rancangan desain sistem dapat dilihat pada gambar mockup 3.4, 3.5 dan 3.6 di bawah ini.



Gambar 3.4 Mockup Dashboard Admin



Gambar 3.5 Mockup Halaman Perizinan

TK DARMA

DASHBOARD

DATA GURU

ADMINISTRASI

PERIZINAN

KATEGORI PERIZINAN

DIIZINKAN

TIDAK DISETUJUI

MENUNGGU PERSETUJUAN

HISTORI

HISTORI RIWAYAT

REKAP ABSEN HARIAN

REKAP ABSEN BULANAN

EXIT

HALAMAN/REKAP ABSEN HARIAN/BULANAN

GURU	JAM MASUK	JAM PULANG	TANGGAL	FOTO MASUK	FOTO PULANG	KET	AKSI
DEVID 1081211231	GURU	GURU	2024-07-04			HADIR	

Gambar 3.6 Mockup Halaman Histori Absensi

Berdasarkan pada *mock-up* 3.4, 3.5 dan 3.6, pengembangan sistem informasi untuk manajemen presensi guru TK Darma Bakti terdiri dari pembuatan website dan aplikasi android yang terhubung dengan database. Pada perancangan sistem ini digunakan software Visual Studio Code sebagai teks editor untuk membuat tampilan website sementara itu untuk aplikasi android menggunakan software android studio untuk mengembangkannya. Untuk database digunakan database php my admin yang selanjutnya dihosting secara online.

3.1.4 Construction of Prototype

Pada tahap ini setelah konstruksi dari perangkat keras dan perangkat lunak telah dilaksanakan, maka langkah selanjutnya adalah mengkonstruksi prototipe dari kedua perangkat dengan melakukan pengembangan terhadap pengolahan *image* yang diperoleh berdasarkan mekanisme keterhubungan antara perangkat keras dan perangkat lunak yang telah dikembangkan.

3.1.4.1 Pengolahan dan Pengujian Image

Pada proses pengolahan image bahasa pemrograman yang digunakan adalah PHP yang dimana pemrosesan data image ini didapatkan dari ESP-32 CAM untuk

menangkap image yang selanjutnya data diteruskan menuju database untuk selanjutnya dikelola pada sistem informasi baik itu website ataupun aplikasi android.

Pengujian pengolahan image untuk sistem manajemen presensi guru TK Darma Bakti dilakukan untuk memastikan bahwa sistem yang dibuat dapat menangkap image yang dalam konteks ini nantinya adalah wajah dari guru-guru TK Darma Bakti. Pengujian ini dilakukan dengan tujuan untuk memastikan bahwa sistem dapat bekerja sesuai kebutuhan, sehingga dapat membantu proses pengolahan gambar untuk manajemen presensi guru TK Darma Bakti.

3.1.5 Deployment Delivery and Feedback

Setelah tahap pengembangan, pembangunan dan pengujian selesai dilakukan, maka tahap selanjutnya adalah melimpahkan sistem yang telah dibuat kepada pihak TK Darma Bakti untuk digunakan dan diuji coba untuk selanjutnya mendapatkan umpan balik dari TK Darma Bakti selaku pengguna. Proses pengujian dan umpan balik ini selanjutnya akan digunakan untuk mengevaluasi dan menyempurnakan sistem berdasarkan metode prototipe yang telah dijadikan landasan sebelumnya. Adapun pengujian yang digunakan dalam proses proyek akhir ini adalah sebagai berikut :

3.1.5.1 Black Box Testing (Software)

Metode *black box testing* merupakan metode pengujian yang berfokus pada fungsional dari sistem yang dikembangkan. Metode ini melihat setiap eksekusi yang terjadi pada perangkat lunak apakah berhasil atau tidak.. Jika hasil *output* sudah sesuai dengan kebutuhan fungsional yang diharapkan, hal ini mengindikasikan bahwa sistem yang dirancang sudah sesuai dan valid untuk digunakan. Pengujian ini dilakukan dengan memberikan angket kepada responden.

3.1.5.2 User Acceptance Testing (UAT)

Metode UAT adalah proses pengujian untuk memverifikasi bahwa sistem siap digunakan, yang juga dikenal sebagai uji coba pengguna. Setelah pengujian

kepuasan pengguna selesai, data yang diperoleh akan dikelola untuk mendapatkan hasil dari penilaian. Pengujian UAT dilakukan dengan membagikan kuesioner dalam bentuk Google Form kepada responden. Dalam pengujian metode UAT ini, digunakan Skala Likert, yang mengukur respons individu terhadap suatu pernyataan dengan pilihan tanggapan positif dan negatif. Dalam Skala Likert ini, kuesioner memberikan penilaian pada setiap jawaban dengan rentang skor dari 1 sampai 5. Skor 1 menunjukkan pendapat atau jawaban sangat tidak setuju, sementara skor 5 menunjukkan pendapat atau jawaban sangat setuju [28]. Berikut ini adalah skala yang digunakan :

Tabel 3.1 Skala Likert

No	Pernyataan		Nilai
1	SS	Sangat Setuju	5
2	S	Setuju	4
3	N	Netral	3
4	TS	Tidak Setuju	2
5	STS	Sangat Tidak Setuju	1

Berdasarkan pada tabel 3.1 nilai skala dapat dihitung dengan sebagai berikut :

$T \times P_n$

T = Jumlah responden yang memilih

P_n = Pilihan skor *Likert*

Tahap selanjutnya adalah menginterpretasikan nilai yang dapat diukur dengan rumus *indeks* 100% sebagai berikut :

$$\text{Rumus Indeks \%} = \frac{\text{Total Skor}}{\text{Nilai tertinggi Likert}} \times 100\%$$

Setelah menetapkan hasil Skala *Likert*, perlu dipahami bahwa interpretasi skor yang didapatkan perlu diketahui interval skor persentasenya. Berikut adalah rumus yang digunakan untuk menghitung *interval*

$$\text{Interval Skor\% (I)} = \frac{100}{\text{Jumlah Skor Likert}(5)}$$

Nilai interval yang didapatkan memiliki interval persentase sebesar 20% yang artinya dapat dilihat pada tabel 3.2 di bawah ini

Tabel 3.2 Pernyataan Persentase Skala Likert [28]

No	Persentase	Interval
1	0% - 19,99%	Sangat Buruk
2	20% - 39,99%	Buruk
3	40% - 59,99%	Cukup
4	60% - 79,99%	Baik
5	80% - 100%	Sangat Baik

Tabel 3.3 Pertanyaan Skala Penelitian

No	Pertanyaan	Skala Penilaian				
		1	2	3	4	5
1	Seberapa puas Anda dengan aplikasi manajemen presensi ini?					
2	Seberapa relevan dan berguna aplikasi ini bagi Anda?					
3	Seberapa pentingkah aplikasi ini bagi kebutuhan presensi Anda?					
4	Seberapa lengkap fitur yang terdapat dalam aplikasi ini?					
5	Apakah aplikasi ini memudahkan Anda dalam mengajukan izin?					
6	Bagaimana kualitas tampilan aplikasi Android ini?					
7	Apakah aplikasi ini mudah digunakan oleh Anda sebagai guru?					

8	Apakah aplikasi ini membantu Anda dalam memantau jam masuk dan jam keluar?
9	Apakah aplikasi ini membantu Anda dalam melihat riwayat absensi dan perizinan?
10	Secara keseluruhan, apakah aplikasi manajemen presensi ini memuaskan bagi Anda?

3.2 Evaluasi dan Perbaikan

Tahapan evaluasi dilakukan dengan menganalisa data yang diperoleh berdasarkan hasil dari pengujian *hardware* dan *software*, serta keseluruhan sistem untuk selanjutnya ditentukan apakah sistem yang telah selesai dikembangkan sudah sesuai dengan yang diharapkan. Jika terdapat kekurangan maka selanjutnya akan dilakukan proses perbaikan pada sistem untuk menyempurnakan sistem manajemen presensi guru TK Darma Bakti dengan tingkat akurasi yang lebih baik dan akurat.

BAB IV

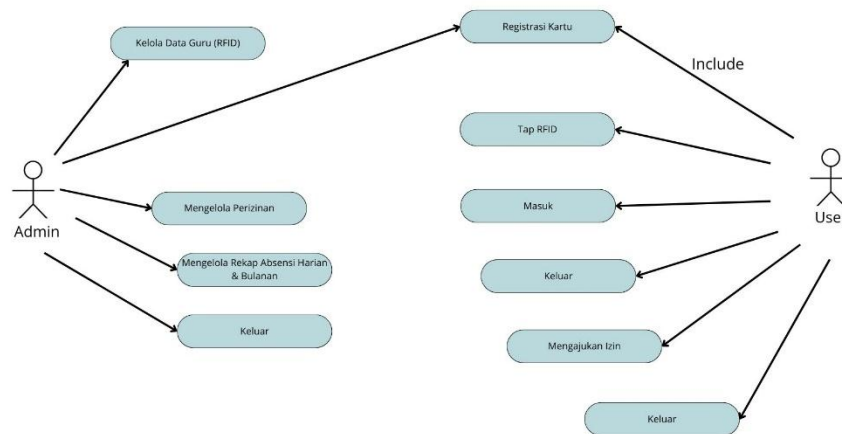
PEMBAHASAN

Pada bab ini akan dibahas mengenai proses pembuatan aplikasi manajemen presensi Guru TK Darma Bakti yang terdiri atas analisis kebutuhan sistem, implementasi sistem manajemen presensi hingga pengujian terhadap keseluruhan sistem untuk dilakukan analisa terhadap sistem berdasarkan data yang diperoleh. Berikut penjelasannya :

4.1 Analisa Kebutuhan Sistem

Sistem manajemen presensi berbasis RFID untuk guru TK Darma Bakti dibangun dengan mengintegrasikan dua entitas utama, yakni admin dan user. Kedua aktor ini memiliki hak akses yang berbeda dalam penggunaan sistem sesuai dengan fungsi dan tugas masing-masing. Administrator atau pihak berwenang, seperti Kepala Sekolah, dapat menggunakan website untuk mengelola berbagai aspek sistem. Fitur-fitur yang tersedia untuk administrator meliputi penambahan data guru, pemberian izin keluar bagi guru dengan kategori perizinan (disetujui, tidak disetujui dan menunggu persetujuan), serta pengelolaan histori atau riwayat rekap absensi harian dan bulanan.

Sementara itu, user atau guru TK Darma Bakti dapat menggunakan aplikasi android yang telah disediakan. Melalui aplikasi ini, guru dapat melihat histori atau rekap absensi dan riwayat perizinan mereka, waktu masuk serta waktu pulang dan mengajukan izin keluar. Apabila izin yang diajukan disetujui, maka izin tersebut selanjutnya tercatat dalam fitur perizinan pada halaman website menu perizinan dan akan dikelola oleh administrator. Analisis kebutuhan sistem yang diterapkan dapat dilihat di *use case diagram* berikut ini :



Gambar 4.1 Use Case Diagram

4.2 Implementasi Sistem Manajemen Presensi

Implementasi teknologi manajemen presensi dengan menggunakan RFID pada proyek akhir ini terdiri dari beberapa bagian. Pada pelaksanaannya terdapat beberapa tahapan yang akan diuraikan dalam pembahasan ini yakni :

1. Konstruksi RFID
2. Sistem Kontrol
3. Pengolahan Capture Image
4. Sistem Informasi Manajemen Presensi

4.2.1 Pengembangan Konstruksi RFID

Pengembangan konstruksi RFID merupakan tahapan untuk membuat tempat *prototype* alat. Perancangan konstruksi RFID dapat dilihat pada gambar di bawah ini.



Gambar 4.2 Perancangan Sistem RFID

Proyek akhir ini proses pembuatan konstruksi ditujukan untuk peletakan prototype RFID seperti pada gambar 4.2 sehingga dalam pelaksanaanya perlu diperhatikan ukuran dan bahan yang akan digunakan. Adapun pada proyek akhir ini digunakan box elektronik yang berukuran 14,5cm x 9,5cm x 5cm.

4.2.1 Pengembangan Sistem Kontrol

Pada bagian ini pengembangan sistem kontrol dilaksanakan dalam dua proses yakni

1. Sistem kontrol RFID, dan
2. Monitoring sistem

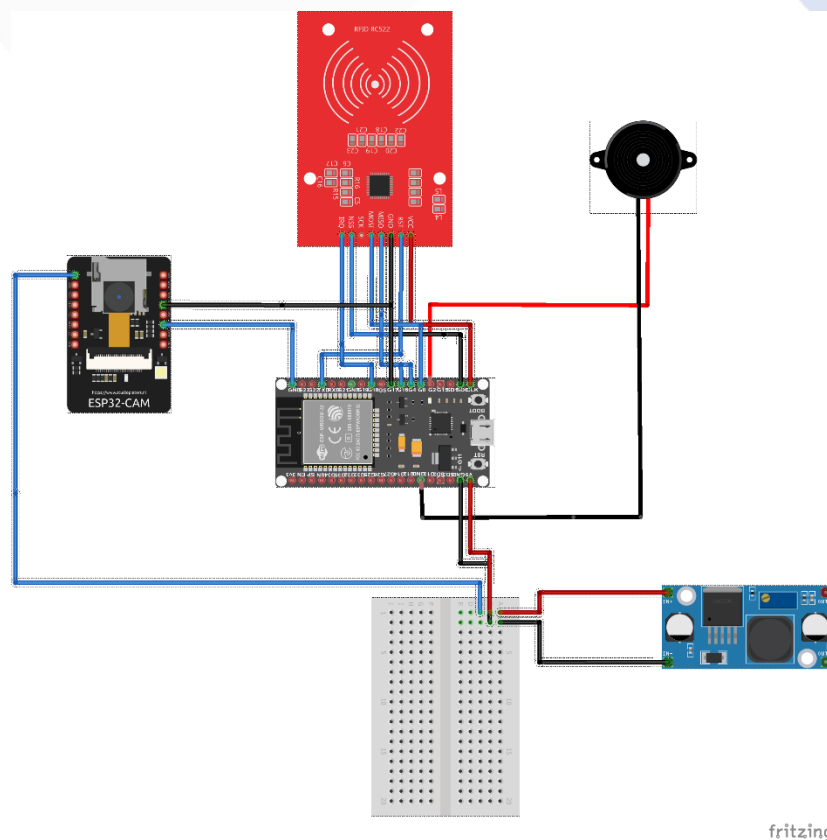
Pada proyek akhir ini perancangan skema elektrik dilakukan dengan menggunakan aplikasi fritzing yang bertujuan untuk memvisualisasikan dan mengatur tata letak

rangkaian elektrik di alat dan juga memahami hubungan setiap komponen yang saling berhubungan.

4.2.1.1 Sistem Kontrol RFID

1) Perakitan Sistem Kontrol RFID

Sistem manajemen presensi guru TK Darma Bakti dibangun dengan menggunakan komponen utama yakni tag RFID yang berfungsi sebagai sistem kontrol untuk memperoleh data untuk selanjutnya data tersebut disimpan dalam database. Data yang diperoleh dari tag RFID akan di baca oleh RFID Reader RC522 yang selanjutnya data akan diteruskan oleh modul wifi yang ada pada ESP32 menuju database. Data yang akan diteruskan menuju database juga mencakup data gambar yang diperoleh dari ESP32 CAM 8825N. Data yang ada pada database selanjutnya akan digunakan untuk melakukan pengontrolan proses perizinan guru TK Darma Bakti yang terdiri dari tiga kategori perizinan yakni disetujui, tidak disetujui dan menunggu persetujuan.

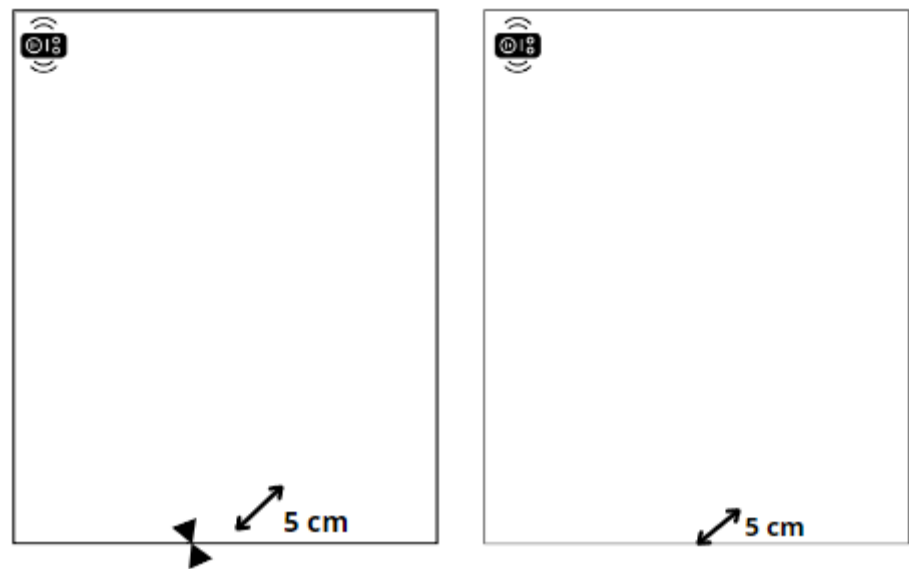


Gambar 4.3 Skematik Kontrol RFID

Gambar 4.3 merupakan desain skematik dari rangkaian kontrol RFID yang digunakan. Adapun pada rangkaian ini reader RFID akan terhubung dengan ESP 32 CAM dan ESP 32 sehingga nantinya mampu mengolah dan menampilkan data pada firebase nantinya.

2) Pengujian RFID

Pengujian pada RFID dilakukan dengan tujuan untuk mengetahui dan meninjau kemampuan dan mengetahui batas jarak minimal reader bisa membaca tag RFID yang dilaksanakan dengan menggunakan tag RFID berbentuk *keychain* dan *card*. Pengujian ini dilakukan dengan cara mengintegrasikan koneksi yang ada pada *reader* dan tag RFID dengan skema jarak tertentu.



Gambar 4.4 Pengujian Jarak RFID

Pada gambar 4.4 diatas dapat dilihat bahwa pengujian terhadap pendeteksian *reader* terhadap tag RFID diukur dengan variasi jarak tertentu megggunakan penggaris untuk mengetahui jangkauan yang dapat ditangkap oleh tag RFID baik itu yang berbentuk *card* maupun *keychain*.

Tabel 4.1 Pengujian *Reader* dan Tag RFID

No	Jarak	Koneksi	
		RFID Card	RFID Key
1	0 – 1 cm	Terhubung	Terhubung
2	1 – 2 cm	Terhubung	Terhubung
3	2 – 3 cm	Terhubung	Terhubung
4	3 – 4 cm	Terhubung	Terhubung
5	4 – 5 cm	Terhubung	Terhubung
6	6 – 7 cm	Tidak Terhubung	Tidak Terhubung
7	7 – 8 cm	Tidak Terhubung	Tidak Terhubung

Dari hasil pengujian pada table 4.1 Diketahui bahwa jarak maksimal reader RFID dapat membaca data dari RFID Tag baik itu yang berbentuk card dan key masing-masing adalah 1 cm dan 5cm. Selanjutnya pengukuran yang melebihi 5 cm tidak mampu terdeteksi baik itu oleh RFID card dan juga key. Hal ini disebabkan oleh kualitas antena pada RFID reader dan RFID Tag, frekuensi kerja serta bentuk dari Tag RFID yang digunakan memiliki batasan jangkauan untuk bisa digunakan. Pada sistem ini RFID berbentuk Tag memiliki prioritas utama dalam penggunaannya untuk sistem manajemen presensi.

4.2.2.2 Monitoring Sistem

Sistem monitoring manajemen presensi guru TK Darma Bakti pada proyek akhir ini berupa website yang dikelola oleh admin atau pihak berwenang yang dalam hal ini adalah kepala sekolah selaku pemangku kepentingan tertinggi. Pada website ini nantinya akan menginformasikan setiap data guru yang ada di TK Darma Bakti berdasarkan RFID Tag mereka, kemudian mampu memantau pengelolaan perizinan untuk guru TK Darma Bakti serta mengelola rekap absensi guru dalam bentuk harian dan bulanan.



Gambar 4.5 Rancangan Monitoring Sistem Manajemen Presensi

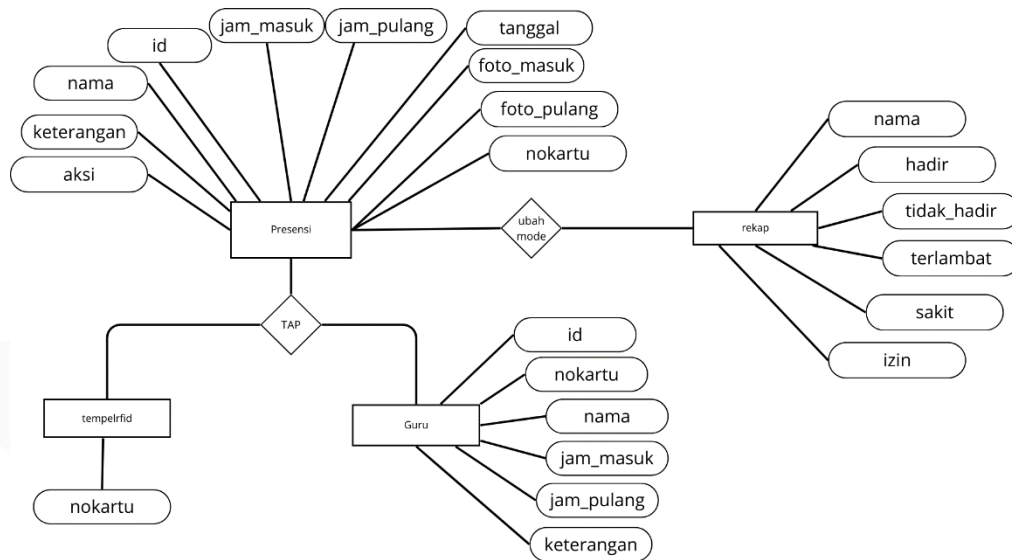
Sistem manajemen presensi Guru TK Darma Bakti ditinjau berdasarkan data RFID Tag Guru yang masuk ke dalam sistem. Jika guru melakukan Tapping pada RFID Reader maka guru akan dianggap masuk dan sistem akan mendata jam masuk serta jam pulang. Tetapi, jika tidak melakukan tapping dan tidak mengajukan perizinan pada sistem maka akan secara otomatis sistem menganggap guru tidak mengajukan perizinan sehingga dianggap tidak masuk pada hari bersangkutan. Selanjutnya sistem juga dapat memantau rekap absensi harian dan bulanan dari guru TK Darma Bakti dan mengelolanya secara keseluruhan seperti yang ada pada *mock-up* 4.5.

4.2.2 Sistem Informasi Manajemen Presensi

Pada bagian sistem informasi manajemen presensi akan dibahas mengenai perancangan serta implementasi *interface* pada halaman website untuk admin untuk memantau aktivitas guru TK Darma Bakti (perizinan, rekap absensi, dan data guru) serta pada aplikasi android untuk user (mengajukan izin, histori tap masuk dan tap keluar, serta riwayat absensi dan perizinan).

4.2.3.1 Entity Relationship Diagram

Entity Relationship Diagram (ERD) digunakan untuk memetakan relasi antara entitas yang ada dalam database. Pada sistem manajemen presensi ini, ERD yang dibuat mencakup entitas seperti user (guru), admin (kepala sekolah), perizinan dan histori yang bisa dilihat pada gambar 4.6 di bawah ini.



Gambar 4.6 Entity Relationship Diagram

4.2.3.2 Struktur Database

Pada tahap pembuatan database digunakan dengan menggunakan phpMyAdmin yang selanjutnya digunakan untuk mengelola dan mengakses ke dalam database. Berikut ini adalah tabel yang digunnakan dalam pembuatan sistem manajemen presensi, yakni sebagai berikut :

1) Table Database tb_absen

Tabel data absen pada gambar 4.7 digunakan untuk menyimpan data yang berkaitan dengan absensi guru TK Darma Bakti. Informasi yang ada didalamnya mencakup id_absen, nama, waktu, jam_masuk, uid, jam_pulang, foto_masuk, foto_pulang serta keterangan.

T				id_absen	waktu	nama	jam_masuk	jam_pulang	foto_masuk	foto_pulang	keterangan	
☐	Ubah	☐	Salin	Hapus	1	2024-07-05 12345	admin	07:00:00	NULL	foto-masuk.jpg	NULL	hadir
☐	Ubah	☐	Salin	Hapus	2	2024-07-05 123456789	ki arjuna	07:00:00	12:00:00	foto-bukti.jpg	foto-bukti.jpg	sakit
☐	Ubah	☐	Salin	Hapus	4	2024-07-07 435435	acil kecil	07:00:00	12:00:00	NULL	NULL	tidak hadir
☐	Ubah	☐	Salin	Hapus	5	2024-06-11 1102334455	josi	07:00:00	12:00:00	foto.jpg	foto.jpg	sakit
☐	Ubah	☐	Salin	Hapus	7	2024-07-11 453423	jensil	NULL	NULL	NULL	NULL	tidak hadir
☐	Ubah	☐	Salin	Hapus	9	2024-07-12 123453	acil kecil	07:00:00	12:00:00	IMG_20240707_170122.jpg	IMG_20240707_170122.jpg	sakit
☐	Ubah	☐	Salin	Hapus	10	2024-07-12 123453	acil kecil	07:00:00	12:00:00	Screenshot_2024-07-04-16-22-11-017_com.instagram.a...	Screenshot_2024-07-04-16-22-11-017_com.instagram.a...	izin
☐	Ubah	☐	Salin	Hapus	11	2024-07-12 123453	acil kecil	07:00:00	12:00:00	IMG_20240702_124054.jpg	IMG_20240702_124054.jpg	hadir
☐	Ubah	☐	Salin	Hapus	12	2024-07-12 123453	acil kecil	07:00:00	12:00:00	329924229_923387172148721_4084182758716825092_n_92...	329924229_923387172148721_4084182758716825092_n_92...	-C-
☐	Ubah	☐	Salin	Hapus	13	2024-07-12 123453	acil kecil	07:00:00	12:00:00	329924229_923387172148721_4084182758716825092_n_92...	329924229_923387172148721_4084182758716825092_n_92...	tidak hadir
☐	Ubah	☐	Salin	Hapus	17	2024-07-19 1234567899	acil kecil	08:30:00	12:00:00	NULL	NULL	terlambat

Gambar 4.7 Tabel Data Absen

2) Tabel Database tb_izin

Tabel izin pada gambar 4.8 digunakan pada database untuk menyimpan informasi yang berkaitan dengan pengajuan izin guru TK Darma Bakti. Informasi yang ada didalamnya mencakup id_izin, uid, nama, nip, alasan, tanggal, status, serta foto_bukti.

		Ters		id_izin	uid	nama	nip	alasan	tanggal	status	foto_bukti			
☐		Ubah		Salin		Hapus	18	123453	acil kecil	359888564	=	2024-07-12	disetujui	329500370_915374259774396_3211230071242185626_n_91...
☐		Ubah		Salin		Hapus	19	123453	acil kecil	359888564	=	2024-07-12	tidak disetujui	329924229_923387172148721_4084182758716825092_n_92...
☐		Ubah		Salin		Hapus	20	123453	acil kecil	359888564	cuti	2024-07-13	disetujui	329924229_923387172148721_4084182758716825092_n_92...
☐		Ubah		Salin		Hapus	21	123453	acil kecil	359888564	Cuti	2024-07-14	tidak disetujui	Flower & Bow Decor French Clip.jpeg
☐		Ubah		Salin		Hapus	22	123453	acil kecil	359888564	Sakit	2024-07-15	disetujui	68_0US \$ _Free shipping floor length evening dress...
☐		Ubah		Salin		Hapus	23	123452	josi	123459876	Cuti	2024-07-15	menunggu persetujuan	replicate-prediction-m3akqrbtqjmsjoatasz26d44.pn...

Gambar 4.8 Tabel Data Izin

3) Tabel Database tb_pegawai

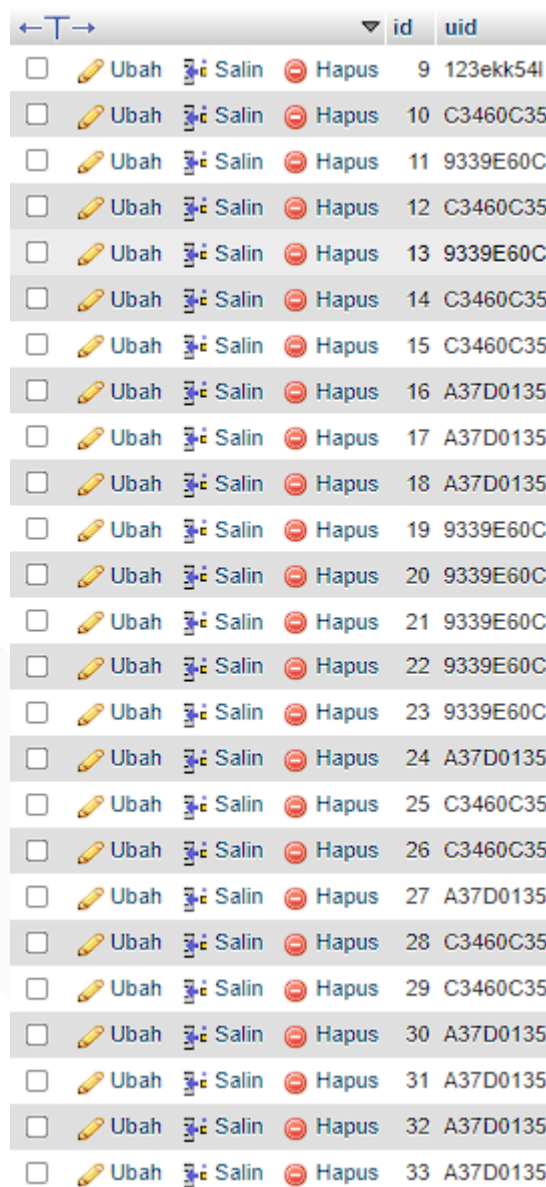
Tabel database pegawai seperti pada gambar 4.9 adalah tabel yang berisi informasi mengenai guru TK Darma Bakti. Informasi yang ada di dalamnya meliputi uid, username, password, role, nama, nip, jabatan, status, bergabung serta foto.

← T →				uid	username	password	role	nama	nip	jabatan	status	bergabung	foto		
☐	Ubah	☐	Salin	☐	Hapus	123452	josi	ee11cbb19052e40b07aac0ca060c23ee	user	josi	123459876	pengajar	pns	2015-07-16	download-modified.jpg
☐	Ubah	☐	Salin	☐	Hapus	123453	acil	ee11cbb19052e40b07aac0ca060c23ee	user	acil kecil	359888564	staf tu	honorer	2024-07-04	kocak.jpg
☐	Ubah	☐	Salin	☐	Hapus	123454	vigoun	ee11cbb19052e40b07aac0ca060c23ee	user	virgoun sebiak	1236754	pengajar	pns	2020-07-16	kocak.jpg
☐	Ubah	☐	Salin	☐	Hapus	123456	arjuna	ee11cbb19052e40b07aac0ca060c23ee	user	ki arjuna	837598547694	staf tu	honorer	2024-07-25	kocak.jpg
☐	Ubah	☐	Salin	☐	Hapus	123458	bambang	ee11cbb19052e40b07aac0ca060c23ee	user	xiao	84375894379	pengajar	honorer	2024-07-20	download.jpg
☐	Ubah	☐	Salin	☐	Hapus	12345admin	admin	21232f297a57a5a743694a0e4a801fc3	admin	admin	12345admin	admin	pns	2024-07-12	download.jpg
☐	Ubah	☐	Salin	☐	Hapus	9339E60C	asep	dc855efb0dc747676afaa1fb281665f1	user	asep	909090909	staf tu	honorer	2024-07-17	Halaman Account.png

Gambar 4.9 Tabel Data Pegawai

4) Tabel Database get_uid

Tabel database get_uid adalah tabel yang menyimpan informasi mengenai UID yang berperan sebagai primary key agar bisa berelasi dengan tabel-tabel lainnya. Pada gambar 4.10 berisi informasi yang ada didalamnya meliputi id dan uid.



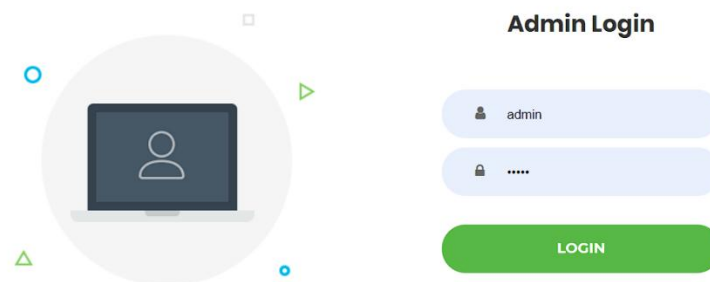
						id	uid
☐		Ubah		Salin		Hapus	9 123ekk54l
☐		Ubah		Salin		Hapus	10 C3460C35
☐		Ubah		Salin		Hapus	11 9339E60C
☐		Ubah		Salin		Hapus	12 C3460C35
☐		Ubah		Salin		Hapus	13 9339E60C
☐		Ubah		Salin		Hapus	14 C3460C35
☐		Ubah		Salin		Hapus	15 C3460C35
☐		Ubah		Salin		Hapus	16 A37D0135
☐		Ubah		Salin		Hapus	17 A37D0135
☐		Ubah		Salin		Hapus	18 A37D0135
☐		Ubah		Salin		Hapus	19 9339E60C
☐		Ubah		Salin		Hapus	20 9339E60C
☐		Ubah		Salin		Hapus	21 9339E60C
☐		Ubah		Salin		Hapus	22 9339E60C
☐		Ubah		Salin		Hapus	23 9339E60C
☐		Ubah		Salin		Hapus	24 A37D0135
☐		Ubah		Salin		Hapus	25 C3460C35
☐		Ubah		Salin		Hapus	26 C3460C35
☐		Ubah		Salin		Hapus	27 A37D0135
☐		Ubah		Salin		Hapus	28 C3460C35
☐		Ubah		Salin		Hapus	29 C3460C35
☐		Ubah		Salin		Hapus	30 A37D0135
☐		Ubah		Salin		Hapus	31 A37D0135
☐		Ubah		Salin		Hapus	32 A37D0135
☐		Ubah		Salin		Hapus	33 A37D0135

Gambar 4.10 Tabel Database get_uid

4.2.3.3 Implementasi Interface Website

Implementasi desain sistem manajemen presensi untuk guru TK Darma Bakti dibagi menjadi dua tahapan. Pada tahap pertama adalah pembuatan website yang merupakan halaman admin yang dirancang dengan menggunakan bahasa pemrograman PHP dan memanfaatkan *software* Visual Studio Code. Tampilan utama dari interface sistem presensi untuk halaman admin ini adalah halaman login

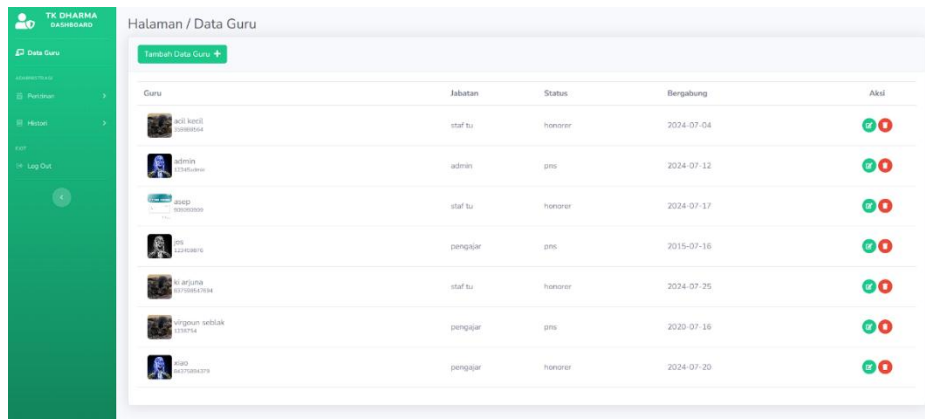
yang mewajibkan admin untuk melakukan login terlebih dahulu agar bisa menggunakan menu dan mengelola data guru, mengelola riwayat presensi, serta memberikan persetujuan terkait dengan pengajuan perizinan guru.



Gambar 4.11 Login Panel Admin

1) Menu Data Guru

Menu data guru merupakan salah satu menu yang terdapat pada *interface* sistem manajemen presensi untuk guru TK Darma Bakti, menu ini digunakan untuk mengelola data guru baik itu yang telah terdaftar ataupun yang belum terdaftar yang dalam konteks ini masih baru. Pada menu ini administrator (kepala sekolah) dapat memantau dan mengelola data dengan cara menambahkan data guru baru, mengubah data guru yang sudah terdaftar serta menghapus data guru yang sekiranya sudah tidak dibutuhkan lagi. Halaman menu data guru dapat dilihat pada gambar 4.12 berikut ini



Guru	Jabatan	Status	Bergabung	Aksi
 Aji Kandi	staf tu	honorer	2024-07-04	 
 Admin	admin	gms	2024-07-12	 
 Dadi	staf tu	honorer	2024-07-17	 
 Iis	pengajar	gms	2015-07-16	 
 Ariyana	staf tu	honorer	2024-07-25	 
 Wigena Seblak	pengajar	gms	2020-07-16	 
 Iis	pengajar	honorer	2024-07-20	 

Gambar 4.12 Halaman Data Guru

Pada menu ini terdapat informasi berupa nama guru dan uid, jabatan, status, serta tanggal bergabung. Admin dapat menambahkan data dengan melakukan aksi tambah data guru lalu selanjutnya mengisi informasi yang diperlukan. Setiap guru yang sudah didaftarkan nantinya akan menerima RFID Tag sebagai identitas unic yang dapat digunakan untuk mengakses sistem manajemen presensi TK Darma Bakti. Halaman penambahan data baru ditunjukkan oleh gambar 4.13 berikut ini

form tambah data guru

×

UID

9339E60C

Username

admin

Password

•••••

👁

Nama

Masukan nama

NIP

Masukan NIP

Jabatan

Pilih Jabatan

▼

Status

Pilih Status

▼

Bergabung

dd/mm/yyyy

📅

Foto

Choose File

No file chosen

Batal

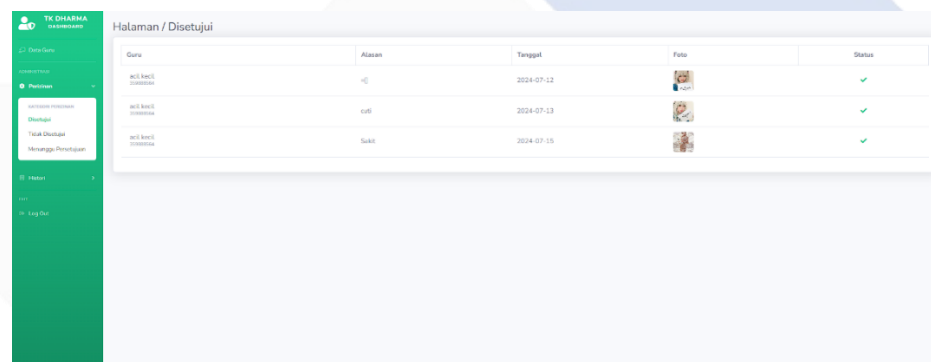
Simpan data

Gambar 4.13 Form Pengisian Data Guru Baru

Pada gambar 4.13 diketahui Form diatas berisi informasi yang harus diisi terlebih dahulu diantaranya adalah UID, nama, NIP, jabatan, status, bergabung serta foto.

2) Menu Perizinan

Menu perizinan digunakan untuk memantau dan mengelola perizinan yang diajukan oleh guru TK Darma Bakti, pada menu ini memungkinkan kepala sekolah selaku administrator untuk mengelola aktivitas perizinan ini. Pada menu ini terdapat tiga buah status peninjauan diantaranya adalah diizinkan, tidak diizinkan serta menunggu persetujuan. Tampilan dari menu perizinan adalah sebagai berikut.

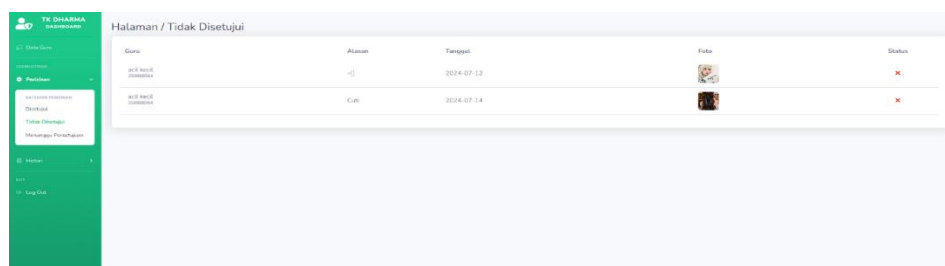


Guru	Alasan	Tanggal	Foto	Status
ari kesi gustiana	lib	2024-07-12		✓
ari kesi gustiana	cuti	2024-07-13		✓
ari kesi gustiana	Sakit	2024-07-15		✓

Gambar 4.14 Menu Perizinan Disetujui

Pada bagian perizinan untuk halaman disetujui menandakan bahwa pengajuan izin yang dilakukan oleh guru TK Darma Bakti telah diterima oleh Kepala Sekolah, halaman ini berisi informasi yang terdiri dari nama guru dan uid, alasan cuti, tanggal, foto serta status pengajuan yang sudah diterima.

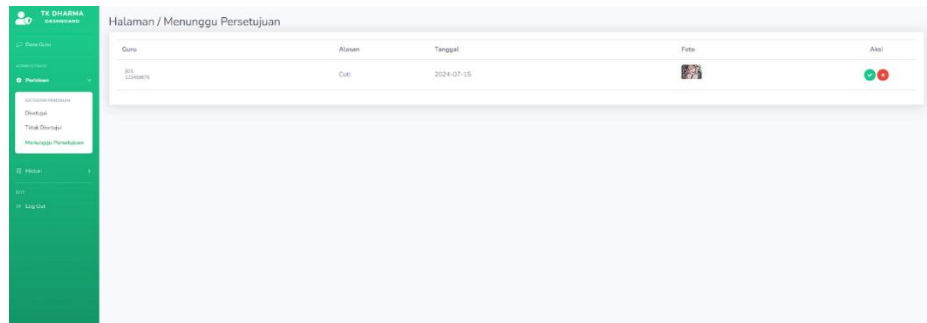
Selanjutnya adalah menu perizinan untuk halaman tidak disetujui dapat dilihat pada gambar 4.15 berikut ini.



Guru	Alasan	Tanggal	Foto	Status
ari kesi gustiana	lib	2024-07-12		✗
ari kesi gustiana	Cuti	2024-07-14		✗

Gambar 4.15 Menu Perizinan Tidak Disetujui

Apabila pengajuan perizinan yang dilakukan oleh guru TK Darma Bakti tidak disetujui oleh kepala sekolah maka akan muncul pada halaman ini dan menampilkan informasi berupa nama guru dan uid, alasan, tanggal, foto serta status yang telah ditolak. Selanjutnya apabila menunggu persetujuan dapat dilihat pada gambar 4.16 berikut ini.



Gambar 4.16 Menu Perizinan Menunggu Persetujuan

Apabila pengajuan izin yang dilakukan oleh guru TK Darma Bakti belum mendapatkan respon sama sekali berarti sedang menunggu persetujuan. Seperti pada gambar 4.16 terdapat informasi berupa nama guru dan uid, alasan, tanggal, foto serta aksi untuk disetujui atau ditolak.

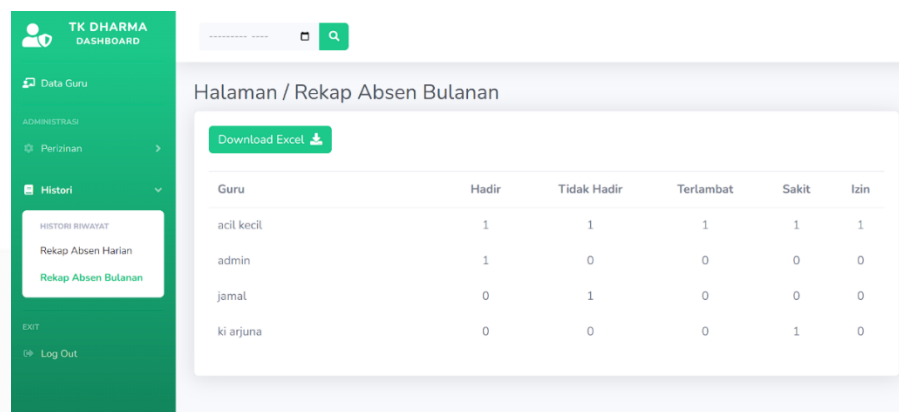
3) Menu Histori Absensi

Menu history rekap absensi ini merupakan riwayat absensi dari guru TK Darma Bakti. Pada menu ini administrator dapat melihat informasi rekap absensi baik itu secara harian ataupun secara bulanan. Berikut adalah menu history presensi harian pada gambar 4.17

Guru	Jam Masuk	Jam Pulang	Tanggal	Foto Masuk	Foto Pulang	Keterangan	Aksi
adrian	07:00:00		2024-07-09			hadi	
ki adiana	07:00:00	12:00:00	2024-07-09			hadi	
ark heidi	07:00:00	12:00:00	2024-07-07			tidak hadir	
jira	07:00:00	12:00:00	2024-06-11			hadi	
jessal			2024-07-11			tidak hadir	
ark heidi	07:00:00	12:00:00	2024-07-12			hadi	
ark heidi	07:00:00	12:00:00	2024-07-12			hadi	
ark heidi	07:00:00	12:00:00	2024-07-12			hadi	
ark heidi	07:00:00	12:00:00	2024-07-12			hadi	
ark heidi	07:00:00	12:00:00	2024-07-12			hadi	
ark heidi	08:30:00	12:00:00	2024-07-19			terlambat	

Gambar 4.17 Menu Histori Presensi Harian

Pada menu ini administrator yang dalam hal ini adalah kepala sekolah dapat melihat dan mengelola histori presensi harian. Adapun informasi yang ada pada menu ini diantaranya adalah nama guru, jam masuk, jam pulang, tanggal, foto masuk, foto pulang, keterangan dan juga aksi. Selanjutnya, jika rekap secara harian adalah melihat informasi presensi berdasarkan satu hari kerja, maka pada menu rekap bulanan seperti yang terlihat pada gambar 4.18 memiliki sedikit perbedaan.



Guru	Hadir	Tidak Hadir	Terlambat	Sakit	Izin
acil kecil	1	1	1	1	1
admin	1	0	0	0	0
jamal	0	1	0	0	0
ki arjuna	0	0	0	1	0

Gambar 4.18 Menu Histori Presensi Bulanan

Pada menu histori presensi secara bulanan, administrator bisa melihat total rekap absen selama satu bulan yang berisi informasi hadir, tidak hadir, terlambat, sakit serta izin dan dapat mendownloadnya kedalam bentuk data *excel*.

4.2.3.4 Implementasi Interface Aplikasi Android

Implementasi *interface* dari aplikasi Android terdiri dari beberapa halaman dan menu yakni sebagai berikut :

4.2.3.4.1 Halaman User

Selanjutnya adalah interface sistem manajemen presensi yang dibuat untuk user dalam hal ini guru dalam bentuk aplikasi android. Interface untuk aplikasi android ini dirancang dengan menggunakan aplikasi android studio dan ditujukan untuk diakses oleh guru. Tampilan utama dari interface pada aplikasi ini adalah landing page yang terdapat informasi dan menu yang dapat digunakan untuk mengajukan izin dan melihat riwayat absensi.

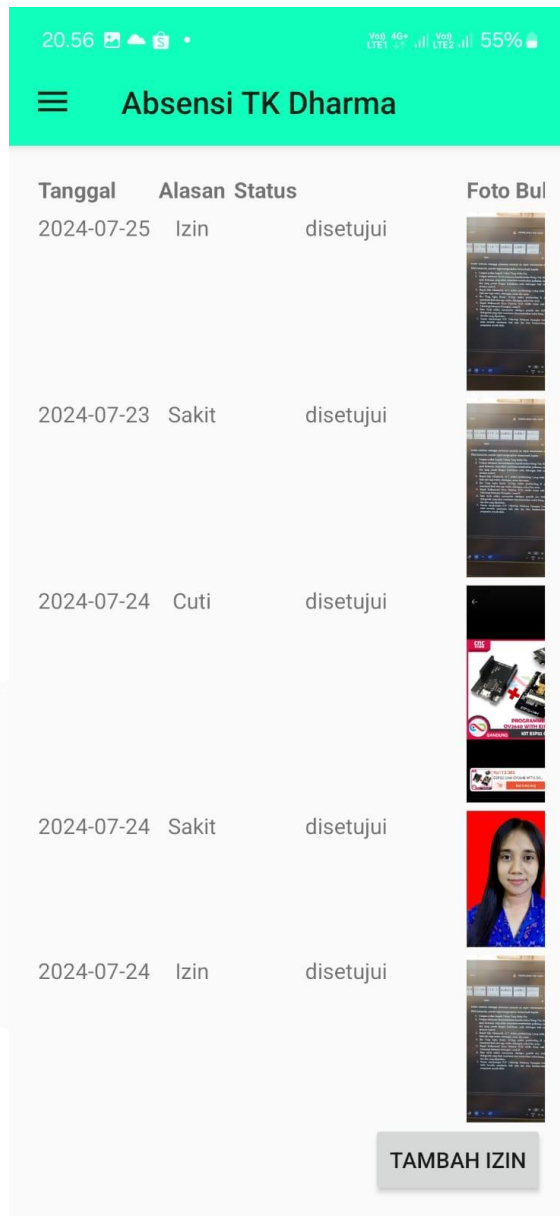


Gambar 4.19 Landing Page

berdasarkan gambar 4.19 guru juga ketika melakukan tapping dapat mengetahui waktu masuk dan waktu mereka pulang. User juga dapat melihat riwayat absensi dan riwayat perizinan yang mereka lakukan.

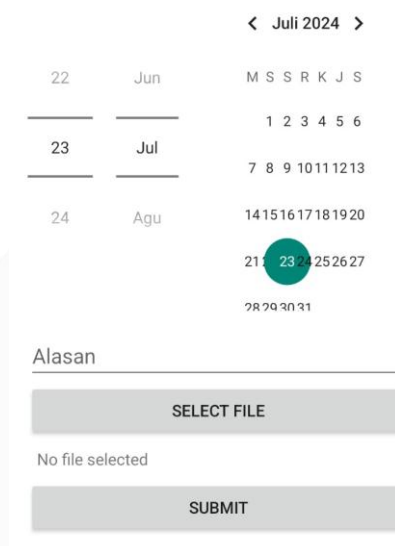
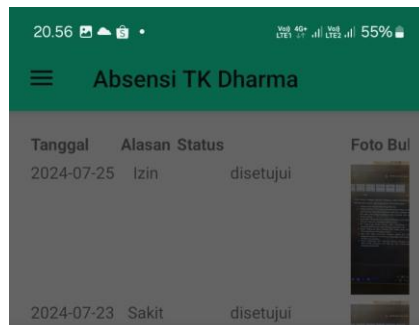
1) Menu Pengajuan Perizinan

Pada menu ini user yakni guru TK Darma Bakti dapat mengajukan perizinan apabila terdapat keperluan, nantinya pengajuan perizinan yang telah dilakukan akan masuk ke dalam halaman admin untuk selanjutnya diproses perizinannya oleh Kepala Sekolah, pada gambar 4.20 berikut merupakan interface dari menu pengajuan izin:



Gambar 4.20 Halaman Pengajuan Perizinan

Apabila guru ingin mengajukan izin maka bisa langsung dengan mengklik tombol tambah izin dan akan muncul tampilannya seperti gambar 4.21 berikut ini :



Gambar 4.21 Halaman Detail Perizinan

2) Menu Riwayat Absensi

Selain halnya dapat mengajukan perizinan, user yang dalam hal ini guru juga bisa melihat riwayat rekap absensi mereka seperti pada gambar 4.22 di bawah ini.

Waktu	Jam Masuk	Jam Pulang	Foto Masuk	Foto Pulang
2024-07-21	06:15:40	12:26:28		
2024-07-22	06:04:40	12:01:48		
2024-07-23	06:00:40	12:01:48		

Gambar 4.22 Riwayat Absensi

4.2.3.4.2 Halaman Admin

Selain user yang bisa menggunakan aplikasi android, admin juga bisa menggunakan aplikasi android untuk mengakses sistem presensi, pada gambar 4.23 adalah halaman landing page dari halaman admin.



Gambar 4.23 Landing Page Admin

Selain guru, kepala sekolah juga memiliki kesamaan dengan guru yakni tetap melakukan absen masuk dan absen keluar. namun yang menjadi pembeda disini adalah fitur yang ada pada halaman admin, berikut adalah menu yang terdapat pada dashboard admin.

1) Menu Pengelolaan Perizinan

Pada halaman ini admin dapat melakukan pengelolaan terhadap pengajuan izin dari guru TK Darma Bakti



Gambar 4.24 Menu Perizinan

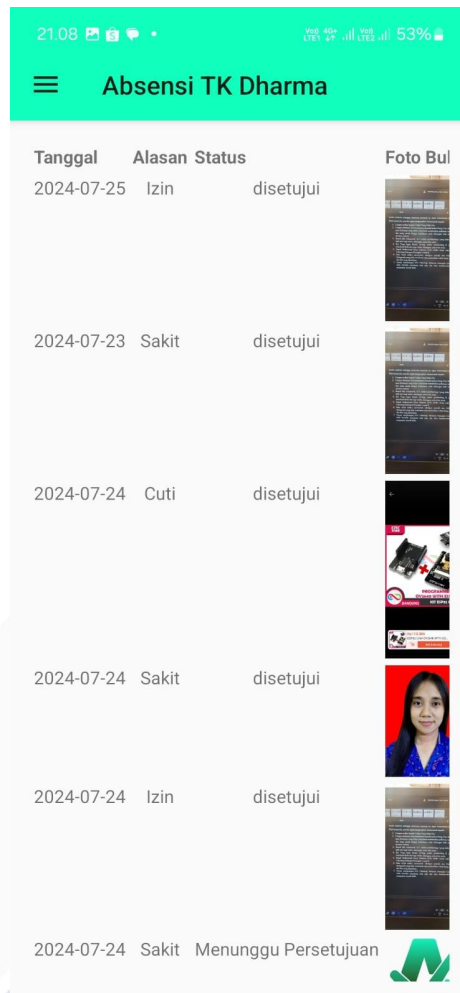
Berdasarkan gambar 4.24 diatas dapat diketahui bahwa admin dapat mengecek status perizinan yang masih menunggu persetujuan dan riwayat persetujuan.

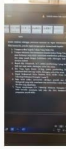
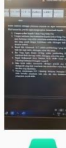


ID	Nama	Tanggal	Alasan	Status
30	Nanda	2024-07-24	Sakit	Menunggu Persetujuan
31	Nanda	2024-07-11	Sakit	Menunggu Persetujuan

Gambar 4.25 Menu Menunggu Perizinan

Berdasarkan gambar 4.25 jika pengajuan perizinan telah disetujui maka kita dapat melihat bagian riwayat perizinan yang berisi informasi siapa saja yang melakukan pengajuan perizinan. Selanjutnya bila pengajuan disetujui maka akan masuk ke riwayat absensi seperti pada gambar 4.26 berikut ini

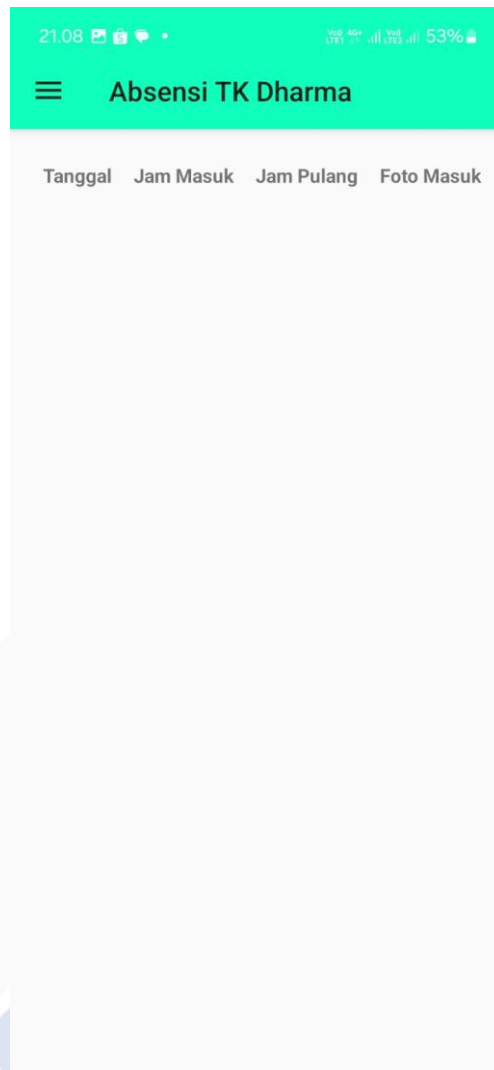


Tanggal	Alasan	Status	Foto Bul
2024-07-25	Izin	disetujui	
2024-07-23	Sakit	disetujui	
2024-07-24	Cuti	disetujui	
2024-07-24	Sakit	disetujui	
2024-07-24	Izin	disetujui	
2024-07-24	Sakit	Menunggu Persetujuan	

Gambar 4.26 Riwayat Absensi

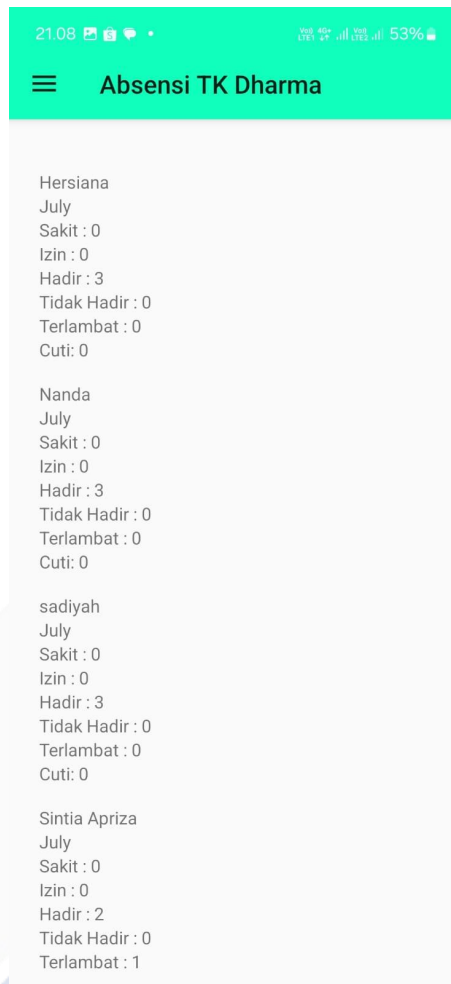
2) Menu Riwayat Perizinan

Pada menu ini admin dapat memantau riwayat perizinan dalam dua bentuk yakni secara harian dan secara bulanan.



Gambar 4.27 Riwayat Absensi Harian

Riwayat absensi harian dapat dilihat pada gambar 4.27 yang dimana rekapannya dilakukan selama harian, sementara itu untuk riwayat absensi bulanan pada dilihat pada gambar 4.28 berikut ini.



Gambar 4.28 Riwayat Absensi Bulanan

4.2.3.5 Pengujian Sistem Informasi dengan Metode Black Box

Metode blackbox merupakan salah satu metode pengujian yang bertujuan untuk memastikan kualitas dari perangkat lunak yang dikembangkan. Pengujian blackbox ini digunakan untuk menguji proses input dan output dari website dan juga aplikasi android yang dikembangkan. Hasil dari pengujian blackbox adalah sebagai berikut

Tabel 4.2 Pengujian Black Box Admin

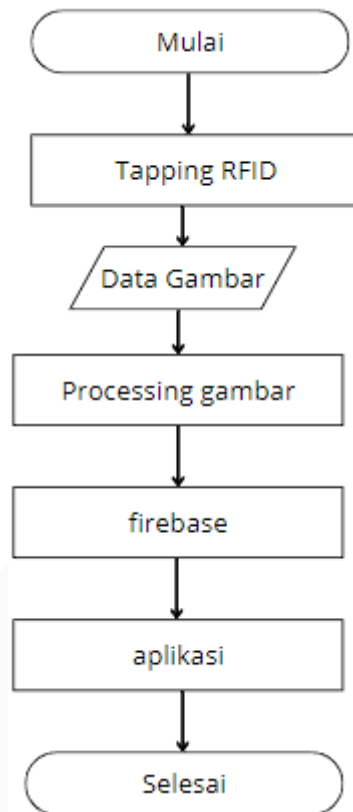
No	Fitur	Deskripsi	Hasil Pengujian
1	Login	Admin dapat melakukan login untuk masuk ke dashboard admin	Berhasil
2	Data Guru	Sistem mampu menampilkan data guru yang sudah terdaftar	Berhasil
3	Tambah Data Guru	Sistem akan menampilkan form penambahan data guru baru	Berhasil
4	Menyetujui pengajuan perizinan berdasarkan kategori	Jika izin disetujui akan masuk pada kategori perizinan disetujui	Berhasil
5	Menyetujui pengajuan perizinan berdasarkan kategori	Jika izin tidak disetujui maka akan masuk pada kategori perizinan tidak disetujui	Berhasil
6	Menyetujui pengajuan perizinan berdasarkan kategori	Jika izin belum disetujui akan masuk pada kategori menunggu perizinan	Berhasil
7	Mengelola Histori	Sistem mampu melakukan pengelolaan rekap absensi secara harian	Berhasil
8	Mengelola Histori	Sistem mampu melakukan pengelolaan terhadap rekap absensi selama per bulan	Berhasil
9	Logout	Admin melakukan logout dan kembali ke halaman login admin	Berhasil

Tabel 4.3 Pengujian Blackbox User

No	Fitur	Deskripsi	Hasil Pengujian
1	Login	User dapat melakukan login untuk masuk ke dashboard User	Berhasil
2	Data Guru	Sistem mampu menampilkan data guru sesuai informasi pengguna yang login	Berhasil
3	Mengajukan pengajuan perizinan	Jika izin disetujui akan masuk pada kategori perizinan disetujui, jika tidak maka akan masuk kategori tidak disetujui, dan jika tidak keduanya maka menunggu persetujuan admin	Berhasil
4	Riwayat Absensi harian	Melihat riwayat absensi harian yang dilakukan	Berhasil
5	Riwayat Absensi bulanan	Melihat riwayat absensi bulanan yang dilakukan selama periode satu bulan	Berhasil
6	Logout	User melakukan logout dan kembali ke halaman login user	Berhasil

4.2.3.6 Pengolahan Image atau Gambar

Alur pengolahan Image dapat dilihat pada skema 4.29 di bawah ini.



Gambar 4.29 Tahapan Pengolahan Gambar

Pengolahan image dilakukan secara *real-time* menggunakan citra foto wajah masukan dari perangkat ESP-32 CAM yang sudah terhubung dengan database, lalu selanjutnya akan diterukan ke halaman admin atau website. Pengolahan gambar ini dimaksudkan agar pada pengelolaan histori riwayat untuk rekap absensi harian dapat terdeteksi oleh admin.

4.2.3.7 Pengujian Pengolahan Gambar

Pengujian pengolahan gambar pada sistem presensi ini ditujukan untuk mengetahui pada absensi harian terdeteksi gambar secara *real-time* ketika melakukan tapping masuk dan pulang. Untuk mengetahui pengujian ini berhasil atau tidak maka dilakukan uji coba terhadap semua guru yang terdaftar di TK Darma Bakti yakni berjumlah 5 data.

Tabel 4.3 Hasil Pengujian Pengambilan Gambar ESP32-CAM

Guru	Jam Masuk	Jam Pulang	Tanggal	Foto Masuk	Foto Pulang	Keterangan	Hasil Pengujian
Admin	07:00:00	12:00:00	2024/07/23	Berhasil	Berhasil	Hadir	Berhasil
Hersiana	07:00:00	12:00:00	2024/07/23	Berhasil	Berhasil	Hadir	Berhasil
Nanda	07:00:00	12:00:00	2024/07/23	Berhasil	Berhasil	Hadir	Berhasil
Sadiyah	07:00:00	09:00:00	2024/07/23	Berhasil	Berhasil	Hadir	Berhasil
Sintia Apriza	07:00:00	09:00:00	2024/07/23	Berhasil	Berhasil	Hadir	Berhasil

Berdasarkan hasil pengujian pada tabel 4.3 diatas diketahui bahwa sistem mampu mendeteksi RFID Tag setiap guru dan melakukan pencatatan terhadap absensi mereka. Untuk guru yang hadir sesuai jam masuk dan jam pulang maka akan ketika melakukan tapping maka foto masuk dan foto pulang akan masuk ke dalam sistem, sedangkan guru yang tidak hadir atau tidak melakukan tapping maka tidak akan menyimpan data gambar di database sehingga tidak akan tampil pada halaman admin.

4.2.3.8 Sistem Keseluruhan

Secara keseluruhan proses pembuatan sistem manajemen presensi untuk guru TK Darma Bakti ini dilakukan secara sistematis dan terstruktur melalui beberapa tahapan, baik itu mulai dari tahap perancangan, implelementasi serta pengujian untuk menghasilkan skema kerja yang efektif dan sistematis serta dapat berfungsi dengan baik dan mampu memenuhi kebutuhan presensi dari TK Darma Bakti sebagaimana yang telah diharapkan sebelumnya.

4.2.3.9 Pengujian Sistem Keseluruhan

Pengujian sistem keseluruhan perlu dilakukan dengan tujuan untuk mengetahui apakah sistem dapat bekerja sesuai dengan alur rancangan yang telah direncanakan sebelumnya. Pengujian ini dilakukan dengan dua tahapan yakni sebagai berikut.

4.2.3.9.1 Pengujian RFID Sistem Manajemen Presensi

Pengujian ini dilakukan untuk mengetahui apakah sistem yang dikembangkan sudah mampu bekerja sesuai dengan alur rancangan yang dibuat. Pengujian pertama dilakukan dengan melihat bagian data guru TK Darma Bakti. Berikut ini merupakan data guru yang sudah masuk ke dalam database sistem :

Halaman / Data Guru

[Tambah Data Guru +](#)

Guru	Jabatan	Status	Bergabung	Aksi
 admin 12345admin	admin	pns	2024-07-12	 
 Hersiana 90903030	guru	karyawan	2016-07-17	 
 Nanda 12345678	guru	honorer	2015-07-15	 
 Foto Sadiyah Profil 12387678	guru	guru	2016-07-17	 
 Gintia Apriza 90980504	staf tu	Karyawan	2011-07-04	 

Gambar 4.30 Data Guru TK Darma Bakti

Berdasarkan data diatas perlu dilakukan pengujian apakah sistem mampu untuk menambahkan data baru dengan ID RFID yang baru dan belum terdaftar pada sistem

Tabel 4.4 Pengujian ID RFID

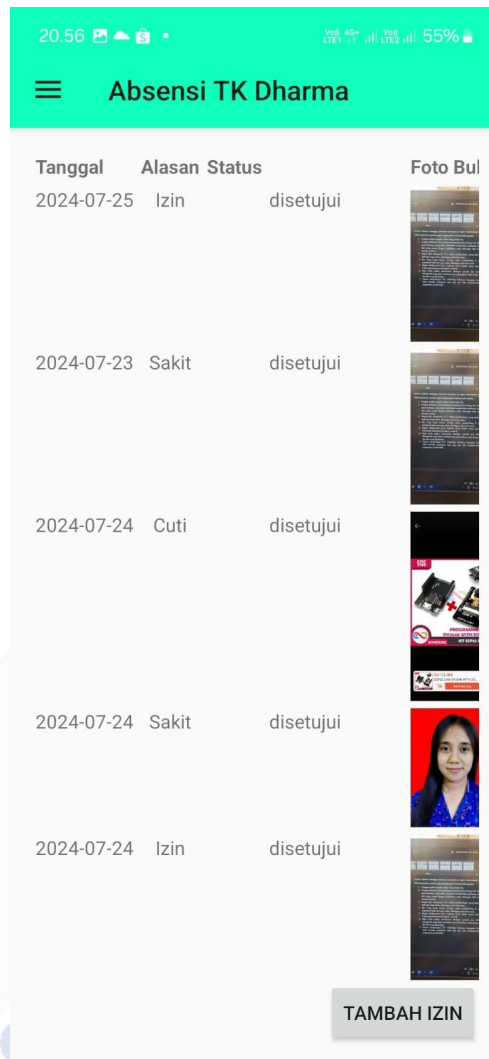
No	Guru	ID RFID	Reader RFID	Hasil Pengujian
1	Murihawa	12345admin	Tersimpan	Berhasil
2	Hersiana	9339E60C	Tersimpan	Berhasil
3	Nanda	A37D0135	Tersimpan	Berhasil
4	Sadiyah	530CED0C	Tersimpan	Berhasil
5	Sintia Apriza	C3460C35	Tersimpan	Berhasil
6	Devid Wijaya	530CED0C	Tidak Tersimpan	Gagal
7	Juanda Renaldi	210803	Tersimpan	Berhasil

8	Novramdan	C3460C35	Tidak	Gagal
	Ilham Alan		Tersimpan	

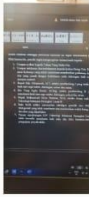
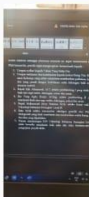
Berdasarkan hasil pengujian data pada tabel 4.4 diatas bisa kita ketahui bahwa ID RFID sudah didaftarkan tidak bisa untuk digunakan kembali untuk menyimpan data. Sebaliknya ID yang belum terdaftar sama sekali dan belum disimpan pada sistem bisa digunakan untuk menyimpan data guru baru di TK Darma Bakti. Untuk dapat melakukannya pada database yang dibuat cukup menambahkan data ID RFID baru yang belum pernah digunakan. Berdasarkan hasil pengujian diatas dapat kita tarik kesimpulan bahwa sistem ini sudah mampu dalam mengatur inputan data RFID untuk Guru TK Darma Bakti.

4.2.3.9.1 Pengujian Pengajuan Perizinan

Pengujian selanjutnya adalah yakni percobaan untuk mengajukan perizinan bagi guru TK Darma Bakti. Pengujian ini dilakukan apabila guru ingin izin keluar nantinya sistem akan mendata pada jam berapa yang bersangkutan keluar serta apakah izin untuk keluar ini diterima atau tidak diterima atau menunggu persetujuan dari kepala sekolah selaku administrator. Berikut ini adalah hasil pengujian pengajuan perizinan untuk guru TK Darma Bakti dapat dilihat pada gambar 4.31 untuk pengajuan izin, dan gambar 4.32 yang merupakan halaman persetujuan pengajuan izin



Gambar 4.31 Halaman Pengajuan Izin

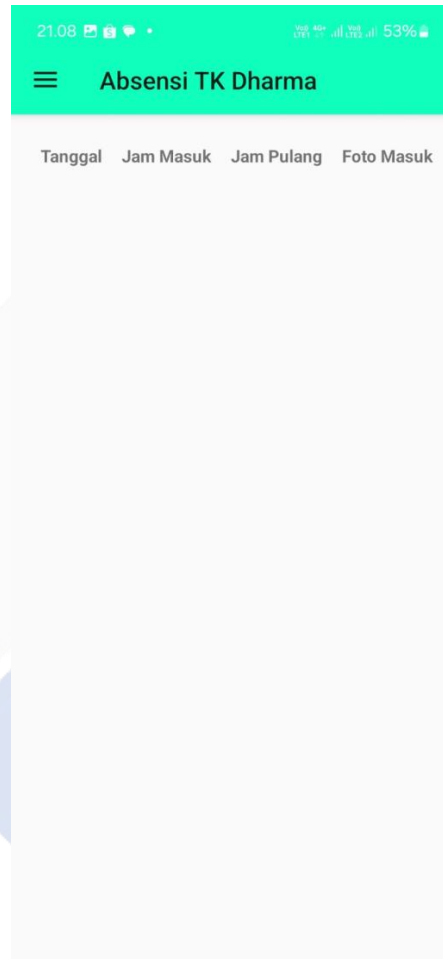
Absensi TK Dharma			
Tanggal	Alasan	Status	Foto Bul
2024-07-25	Izin	disetujui	
2024-07-23	Sakit	disetujui	
2024-07-24	Cuti	disetujui	
2024-07-24	Sakit	disetujui	
2024-07-24	Izin	disetujui	
2024-07-24	Sakit	Menunggu Persetujuan	

Gambar 4.32 Halaman Persetujuan Pengajuan Izin

4.2.3.9.2 Pengujian History Manajemen Presensi

Pada tahap ini akan dilakukan dua pengujian terhadap manajemen presensi yakni terhadap presensi harian dan terhadap presensi bulanan.

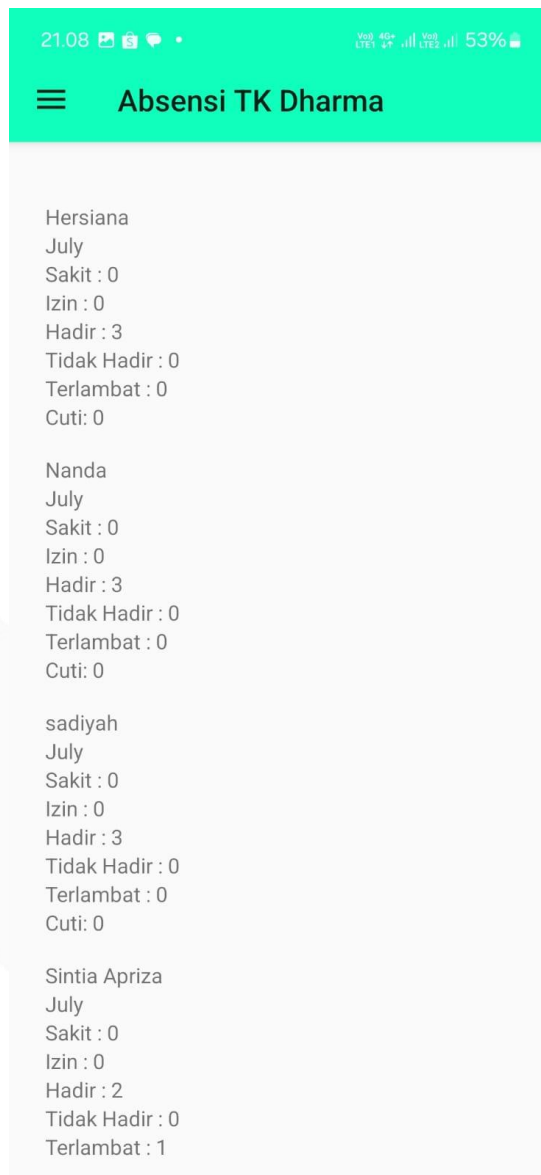
1) Presensi Harian



Gambar 4.33 Riwayat Absensi Harian

Berdasarkan pada gambar 4.33 diatas berkaitan dengan presensi harian akan muncul beberapa informasi seperti tanggal, jam masuk, jam pulang hingga foto masuk

2) Presensi Bulanan



Absensi TK Dharma	
Hersiana	
July	
Sakit : 0	
Izin : 0	
Hadir : 3	
Tidak Hadir : 0	
Terlambat : 0	
Cuti: 0	
Nanda	
July	
Sakit : 0	
Izin : 0	
Hadir : 3	
Tidak Hadir : 0	
Terlambat : 0	
Cuti: 0	
sadiyah	
July	
Sakit : 0	
Izin : 0	
Hadir : 3	
Tidak Hadir : 0	
Terlambat : 0	
Cuti: 0	
Sintia Apriza	
July	
Sakit : 0	
Izin : 0	
Hadir : 2	
Tidak Hadir : 0	
Terlambat : 1	

Gambar 4.34 Riwayat Absensi Bulanan

Berdasarkan pada gambar 4.34 diatas berkaitan dengan rekap absensi bulanan akan dihitung berdasarkan jumlah sakit, izin, kehadiran, tidak hadir, serta terlambat.

4.2.3.10 Pengujian Sistem

Berikut adalah hasil kuesioner dari pengujian sistem manajemen presensi guru TK Darma Bakti, setelah angket diberikan kepada pengguna, maka angket tersebut akan diolah datanya untuk mendapatkan hasil akhir penelitian dari uji coba pengguna. Data pengujian sistem kepada 33 orang pengguna dapat dilihat pada tabel 4.5 berikut ini.

Tabel 4.5 Jumlah Jawaban Kuesioner

Pertanyaan	Jumlah Jawaban					Bobot
	1	2	3	4	5	
1	-	-	6	20	7	133
2	-	-	12	12	9	129
3	-	-	7	4	14	107
4	-	-	4	16	13	141
5	-	-	8	10	15	139
6	-	-	7	14	12	137
7	1	-	5	9	18	142
8	-	-	4	11	18	146
9	-	-	6	12	15	141
10	-	-	4	13	16	144
						1489

$$\text{Hasil Akhir} = \frac{1489}{1650} \times 100\% = 0,9024 \times 100\% = 90,24\%$$

Dari hasil pengujian tiga puluh tiga responden pada tabel 4.5 diatas dengan masing-masing diberikan 10 pertanyaan yang sama, maka akan mendapatkan total score sebesar 1489 dan didapatlah total perhitungan dengan persentase total skor sebesar 90,24% Berdasarkan perhitungan tersebut dapat kita tarik kesimpulan bahwa sistem manajemen presensi yang dibuat untuk guru TK Darma Bakti ini dapat digunakan

sesuai dengan kebutuhan dan berada pada persentase 80% - 100% yang berarti sangat baik berdasarkan *skala likert* yang telah ditetapkan sebelumnya.



BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil pembuatan “Aplikasi dan Website untuk Manajemen Presensi Guru TK Darma Bakti”, maka dapat kita tarik kesimpulan bahwa sistem manajemen presensi dapat menggunakan teknologi RFID agar lebih mutakhir dan modern mengikuti perkembangan zaman. Penggunaan teknologi ini bukan tanpa alasan melainkan dapat meningkatkan pengelolaan presensi yang ada di TK Darma Bakti.

Adapun hasil penelitian berdasarkan pengujian sistem dapat dirangkum sebagai berikut :

1. Dalam proses perancangan, sistem presensi menggunakan perangkat RFID Reader dan Tag untuk otentikasi kehadiran guru. Selain itu, ESP32 cam digunakan sebagai alat untuk menangkap foto wajah saat melakukan presensi. Sistem ini dirancang untuk memastikan bahwa setiap guru yang hadir diidentifikasi dengan tepat melalui kombinasi teknologi RFID dan pengenalan wajah. Proses perancangan meliputi pemrograman perangkat keras dan lunak untuk memastikan integrasi yang lancar antara RFID Reader, ESP32 cam, dan database presensi. Pengujian terhadap sistem dilakukan melalui User Acceptance Testing (UAT) yang melibatkan tiga puluh lima responden, dengan hasil pengujian menunjukkan total skor 1489 dan persentase 90,24%. Hasil ini mengindikasikan bahwa sistem siap untuk digunakan dan berada pada skala sangat baik.
2. Setelah proses perancangan selesai, sistem absensi diimplementasikan di TK Darma Bakti. Implementasi ini mencakup pemasangan perangkat keras seperti RFID Reader dan ESP32 cam di lokasi sekolah, serta instalasi perangkat lunak pada server dan aplikasi. Guru melakukan absensi dengan mendekatkan RFID Tag mereka ke RFID Reader, diikuti dengan pengambilan foto wajah untuk validasi identitas. Hasil absensi kemudian dapat diakses melalui website dengan alamat tkdarma.site, yang menyediakan interface bagi admin untuk mengelola dan memantau presensi. Selain itu, aplikasi Android dikembangkan untuk

memudahkan admin dalam mengelola presensi dan bagi guru untuk melihat riwayat absensi mereka masing-masing serta mengajukan izin. Sistem ini memastikan bahwa data presensi tersimpan secara akurat dan dapat diakses secara real-time oleh pihak yang berwenang.

3. Fitur pemantauan riwayat presensi dirancang untuk menyediakan data presensi harian dan bulanan, yang mencakup perhitungan izin, keterlambatan, sakit, dan ketidakhadiran lainnya. Sistem pengelolaan izin memungkinkan admin untuk memantau dan menyetujui atau menolak permohonan izin yang diajukan oleh guru. Jika permohonan diterima, maka statusnya akan tercatat sebagai diterima; jika ditolak, akan tercatat sebagai tidak diterima; dan jika masih menunggu, akan berada dalam status menunggu persetujuan.

5.2 Saran

Dalam proyek akhir ini, disarankan untuk mengembangkan lebih lanjut aplikasi dan website yang telah dibuat agar pengelolaan presensi menjadi semakin efektif dan efisien. Berikut adalah beberapa saran yang dapat diberikan :

1. Mengoptimalkan penggunaan kartu RFID untuk beberapa keperluan. Penulis menyarankan untuk terus mengembangkan dan memperbarui sistem ini agar tetap sesuai dengan perkembangan teknologi dan kebutuhan pengguna.
2. Meskipun sistem ini cukup intuitif, pelatihan rutin bagi pengguna (guru dan admin) diperlukan untuk memastikan bahwa semua fitur yang digunakan telah maksimal dan masalah teknis dapat diatasi dengan cepat

Daftar Pustaka

1. Mustofa, M., *Upaya pengembangan profesionalisme guru di indonesia*. Jurnal Ekonomi dan Pendidikan, 2007. **4**(1): p. 17245.
2. Izzati, U.A., Y. Lolita, and O. Prabandini, *Komitmen Profesional pada Guru Taman Kanak-Kanak*. Jurnal Obsesi: Jurnal Pendidikan Anak Usia Dini, 2022. **6**(6): p. 6746-6755.
3. Harahap, F., et al., *Pembuatan Aplikasi Absensi Berbasis Flutter untuk Meningkatkan Efisiensi Monitoring Kehadiran*. Jurnal Penelitian Teknologi Informasi dan Sains, 2023. **1**(3): p. 85-93.
4. Sitorus, E.N., Jamaluddin, J., & Harianja, E.J.G. (2023). Sistem Informasi Kehadiran Siswa Menggunakan QR Kode Berbasis Android: Studi Kasus SD Negeri 105270. TAMIKA: Jurnal Tugas Akhir Manajemen Informatika & Komputerisasi Akuntansi, 3(1), 24-39.
5. Zeng, W., Q. Meng, and R. Li. *Design of intelligent classroom attendance system based on face recognition*. in *2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*. 2019. IEEE.
6. Cahyadi, D., *Desain Sistem Absensi PNS Berbasis Teknologi RFID*. Jurnal Informatika Mulawarman, 2009. **4**(3): p. 29-36.
7. Arianingrum, F. and F. Farindika Metandi, *Aplikasi Absensi Penyiar Radio dengan menggunakan Teknologi RFID (Radio Frequency Identification) berbasis Web Studi Kasus Radio Polnes FM*. JUST TI, 2018. **10**(2): p. 1-4.
8. Hidayat, R., *Rancang Bangun Alat Absensi Karyawan menggunakan RFID dan ESP32Cam Berbasis Internet of Things*. 2022, ITN MALANG.
9. Santika, R.R., Wiharto, Y., & Irawan, A. (2021). Desain Sistem Absensi Karyawan dengan Radio Frequency Identification Berbasis Web. Research Lembaran Publikasi Ilmiah, 4(1), 18-22.
10. Hunaepi, A., A. Roihan, and A. Nurtursina, *Perancangan Sistem Kehadiran Pendidik Dan Tenaga Kependidikan Berbasis Mikrokontroler Esp32Cam*. Jurnal Sistem Informasi dan Teknologi (SINTEK), 2023. **3**(2): p. 61-67.

- 11 Harumy, T.H.F., Julham Sitorus, M. L. (2018). Sistem Informasi Absensi Pada Pt . Cospar Sentosa Jaya Menggunakan Bahasa Pemrograman Java. *Jurnal Teknik Informartika*, 5(1), 63–70.
- 12 Fitriyadi, F., & Hariono, H. (2021). Perancangan Sistem Absensi Perkuliahan Dengan Menggunakan Radio Frequency Identification. *Progresif: Jurnal Ilmiah Komputer*, 17(1), 55. <https://doi.org/10.35889/progresif.v17i1.573>
- 13 Husain, A., Prastian, A. H. A., & Ramadhan, A. (2017). Perancangan Sistem Informasi Reservasi Sewa Kamar Hotel Berbasis Web Pada Lumbung Padi Resort. *Technomedia Journal*, 2(1), 105–116.
- 14 Dinasari, W., Budiman, A., & Megawaty, D. A. (2020). Sistem Informasi Manajemen Absensi Guru Berbasis Mobile (Studi Kasus : Sd Negeri 3 Tangkit Serdang). *Jurnal Teknologi Dan Sistem Informasi*, 1(2), 50–57. <https://doi.org/10.33365/jtsi.v1i2.558>
- 15 Sikumbang, M. A. R., Habibi, R., & Pane, S. F. (2020). Sistem Informasi Absensi Pegawai Menggunakan Metode RAD dan Metode LBS Pada Koordinat Absensi. *Jurnal Media Informatika Budidarma*, 4(1), 59. <https://doi.org/10.30865/mib.v4i1.1445>
- 16 Niswah, M., & Devitra, J. (2023). Sistem Informasi Manajemen Presensi Kehadiran Karyawan Berbasis Web Dan SMS Gateway Pada SMK Dr. Indra Adnan Indragiri Collage Tembilahan. *Manajemen Sistem Informasi*, 8(2), 232–241.
- 17 Susanto, R., Ananta, A., Santoso, A., & Trianto, M. (2009). Sistem Absensi Berbasis RFID. *Teknik Komputer*, 17(1), 67–74.
- 18 Damara, T.M.D. (2022). Sistem Presensi dan Peminjaman Perpustakaan Politeknik Manufaktur Negeri Bangka Belitung Berbasis Radio Frequency Identification (RFID). Bangka Belitung.
- 19 Saputra, F. H. (2008). Sistem Absensi Menggunakan Teknologi RFID. *Sistem Absensi Menggunakan Teknologi RFID*, 1–82.
- 20 Rahman La Mai, & Muchlis. (2021). Sistem Absensi Menggunakan Teknologi Radio Frequency Identification (RFID) pada CV. Kereta Laju

- Kota Tangerang. *Jurnal Teknik Informatika*, 7(1), 23–29.
<https://doi.org/10.51998/jti.v7i1.348>
- 21 Klaus Finkenzeller. “RFID Handbook :Fundamentals and Application in Contacless Smart Cards and Identification, Second Edition”. (West Sussex: John Wiley & Sons Ltd, 2003).
 - 22 Ramdani, D., Mukti Wibowo, F., & Adi Setyoko, Y. (2020). Journal of Informatics, Information System, Software Engineering and Applications Rancang Bangun Sistem Otomatisasi Suhu Dan Monitoring pH Air Aquascape dengan IoT (Internet Of Thing) Menggunakan Nodemcu Esp8266 Pada Aplikasi Telegram. *Journal of Informatics, Information System, Software Engineering and Applications*, 3(1), 59–068.
<https://doi.org/10.20895/INISTA.V2I2>
 - 23 Ibadurrahman, S. (2020). Rancangan Bangun Smart Home dengan Konsep Internet of Things (IoT) Berbasis Android. University of Technology Yogyakarta.
 - 24 Jurnal, H., Huda, Q., Fahmi, H., & Pardiansyah, A. S. (2022). Jurnal Publikasi Ilmu Komputer Dan Multimedia Prototype Sistem Absensi Menggunakan Rfid Berbasis Iot. *Jupikom*, 1(1).
 - 25 Setyawan, E., Dajamaludin, D., & ... (2021). Sistem Alat Absensi Menggunakan RFID dan Kamera Berbasis Internet of Things. *Jurnal Ilmiah Fakultas ...*, 2, 123–129.
<http://www.ejournal.unis.ac.id/index.php/jimtek/article/view/1671%0Ahttp://www.ejournal.unis.ac.id/index.php/jimtek/article/download/1671/1560>
 - 26 Hariono, F. (2021). Perancangan Sistem Absensi Perkuliahan Dengan Menggunakan Radio Frequency Identification (Studi Kasus pada: STMIK Banjarbaru). *Progresif: Jurnal Ilmiah Komputer*, 55–66. <http://ojs.stmik-banjarbaru.ac.id/index.php/progresif/article/view/573/0>
 - 27 Fitriyadi, F., & Maulidi, M. (2020). Model Perekaman Pengunjung Perpustakaan STMIK Banjarbaru Menggunakan RFID Berbasis Mikrokontroler. *Progresif: Jurnal Ilmiah Komputer*, 16(2), 103-112.

- 28 Harahap, K. S., dkk. (n.d.). Kajian Pengendalian Mutu Produk Tuna Loin Precooked Frozen Menggunakan Metode Skala Likert di Perusahaan Pembekuan Tuna X [Study of Quality Control of Tuna Loin Precooked Frozen Products Using the Likert Scale Method in Tuna Freezing Company X]
- 29 Aditya, R., Pranatawijaya, V. H., & Putra, P. B. A. A. (2021). Rancang Bangun Aplikasi Monitoring Kegiatan Menggunakan Metode Prototype. *Journal of Information Technology and Computer Science*, 1(1), 47-57.



LAMPIRAN 1

DAFTAR RIWAYAT HIDUP

Daftar Riwayat Hidup

1. Data Pribadi

Nama Lengkap : Devid Wijaya
Tempat Tanggal Lahir : Sungailiat, 3 Maret 2003
Jl. Belinyu, Simpan
Alamat Rumah : Bedukang
No. HP : 0821 7590 5109
Email : devidwijaya9999@gmail.com
Jenis Kelamin : Laki-laki
Agama : Buddha



2. Riwayat Pendidikan

SD Negeri 6 Deniang	Lulus 2015
SMP Negeri 3 Sungailiat	Lulus 2018
SMK Negeri 1 Sungailiat	Lulus 2021
Politeknik Manufaktur Negeri Bangka Belitung	2021 Sekarang

3. Pendidikan non Formal

-

Sungailiat, 24 Juli 2024

LAMPIRAN 2

PENGUJIAN BLACKBOX TESTING ADMIN & USER

BLACKBOX TESTING USER

Nama Penguji : Dwi Atep
 Jabatan : IT Mega Mart
 Tanggal : 28 Juli 2023

No	Modul	Test Case	Test Script	Diharapkan	Didapat	Hasil Uji
1	Login	Login dengan data yang benar	Memasukkan username & password yang sesuai, melakukan login	Berhasil login dan masuk ke dashboard user	Sesuai	✓
		Login dengan data yang tidak sesuai	Memasukkan username & password yang tidak sesuai, melakukan login	Gagal melakukan login dan muncul alert gagal login	Sesuai	✓
2	Home User	Menguji fitur melihat data diri	Melihat data guru yang terdiri dari nama, nip, jabatan, status, waktu masuk, waktu pulang, dan foto diri	Data diri guru berhasil muncul pada halaman user	Sesuai	✓
		Menguji fitur riwayat absensi	Melihat riwayat absensi yang sudah dilakukan	Data riwayat absensi muncul pada halaman user	Sesuai	✓
		Menguji fitur riwayat perizinan	Melihat riwayat perizinan yang sudah dilakukan	Data riwayat absensi muncul pada halaman user	Sesuai	✓
		Menguji fitur mengajukan perizinan	Melakukan tambah izin kemudian memilih alasan perizinan	Data perizinan masuk ke halaman admin untuk selanjutnya ditindaklanjuti	Sesuai	✓
		Menguji fitur histori absensi harian	Melihat absensi harian yang telah dilakukan	Rekap absensi harian ditampilkan pada halaman user dan muncul pada halaman admin	Sesuai	✓
		Menguji fitur histori absensi bulanan	Melihat rekap absensi selama periode bulan	Rekap absensi bulanan ditampilkan pada halaman user dengan informasi nama guru, bulan, dan keterangan perizinan selama berapa hari	Sesuai	✓

		Menguji fitur administrasi perizinan kategori menunggu perizinan	Memberikan persetujuan terhadap guru yang mengajukan izin	Berhasil memberikan persetujuan dan data guru yang tidak disetujui perizinannya akan muncul di halaman tidak disetujui	Sesuai	✓
		Menguji Fitur Histori riwayat rekap absensi harian	Menampilkan data rekap absensi harian guru yang terdiri dari nama, jam masuk, jam pulang, tanggal, foto masuk, foto keluar, keterangan dan mampu menghapus data yang tidak dibutuhkan	Merupakan data absensi harian guru dan menghapus data absensi harian pada button hapus	Sesuai	✓
		Menguji fitur histori riwayat rekap absensi bulanan	Menampilkan data rekap absensi bulanan	Mampu menampilkan jumlah total absensi yang terdiri dari kehadiran, tidak hadir, terlambat, sakit, izin, dan cuti kemudian mendownloadnya ke dalam format excel	Sesuai	✓
3	Logout	Menguji fitur logout	Admin keluar dari sistem	Admin diarahkan kembali ke halaman login, sesi admin berakhir	Sesuai	✓

Sungailat, Juli 2023


 Mega Mart
 Jl. Merdeka No. 100, Sungailat, Kalimantan Tengah

BLACKBOX TESTING ADMIN

Nama Penguji : Dwi Ardi
Jabatan : IT Klaten Map
Tanggal : 28 2024

No	Modal	Test Case	Test Script	Diharapkan	Didapat	Hasil Uji
1	Login	Login dengan data yang benar	Memasukkan username & password yang sesuai, melakukan login	Berhasil login dan masuk ke dashboard admin	Sesuai	✓
		Login dengan data yang tidak sesuai	Memasukkan username & password yang tidak sesuai, melakukan login	Gagal melakukan login dan muncul alert gagal login	Sesuai	✓
2	Home Admin	Menguji fitur input data guru untuk ditambahkan ke dalam database	Menginput form tambah data guru yang terdiri dari nama, nip, jabatan, status, waktu bergabung, dan foto diri	Data guru baru muncul di halaman admin	Sesuai	✓
		Menguji fitur edit data guru diperbarui dalam database	Mengedit informasi data guru yang akan diperbarui	Perubahan terlihat daftar guru di halaman admin	Sesuai	✓
		Menguji fitur hapus data guru	Menghapus data guru yang dipilih dalam database	Daftar guru yang dihapus tidak lagi muncul di halaman admin	Sesuai	✓
		Menguji fitur administrasi perizinan kategori disetujui	Melengkapi daftar guru yang mengajukan perizinan yang sudah mendapatkan persetujuan	Data guru yang mengajukan perizinan akan muncul di halaman admin kategori perizinan disetujui	Sesuai	✓
		Menguji fitur administrasi perizinan kategori tidak disetujui	Melengkapi daftar guru yang mengajukan perizinan yang tidak mendapatkan persetujuan	Data guru yang mengajukan perizinan namun ditolak akan muncul di halaman admin kategori perizinan tidak disetujui	Sesuai	✓
		Menguji fitur administrasi perizinan kategori menunggu perizinan	Memberikan persetujuan terhadap guru yang mengajukan izin	Berhasil memberikan persetujuan dan data guru yang disetujui muncul di halaman disetujui	Sesuai	✓

3	Logout	Menguji fitur logout	User keluar dari sistem	User diarahkan kembali ke halaman login, saat User diakhiri	Sesuai	✓
---	--------	----------------------	-------------------------	---	--------	---



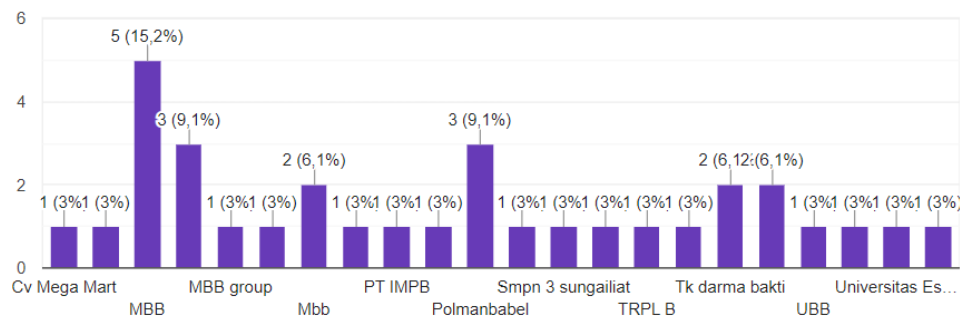
LAMPIRAN 3

PENGUJIAN UAT

Instansi

[Salin](#)

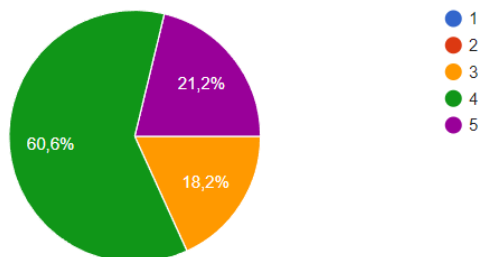
33 jawaban



Seberapa puas anda dengan aplikasi manajemen presensi ini

[Salin](#)

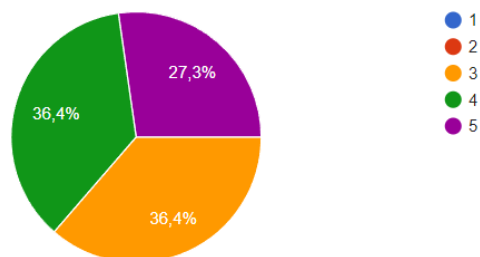
33 jawaban



Seberapa relevan dan berguna aplikasi ini bagi anda

[Salin](#)

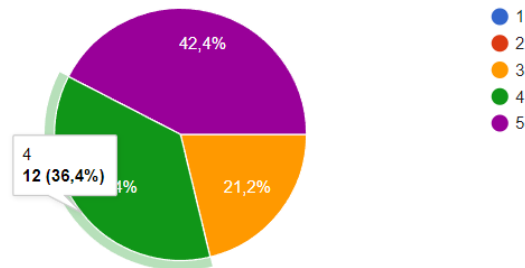
33 jawaban



Seberapa pentingkah aplikasi bagi kebutuhan presensi anda

 Salin

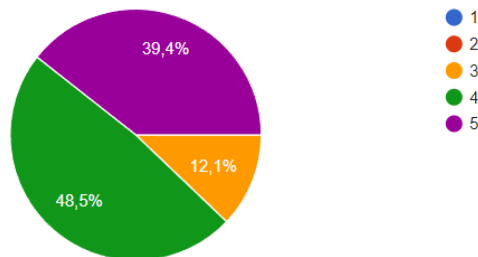
33 jawaban



Seberapa lengkap fitur yang terdapat dalam aplikasi ini

 Salin

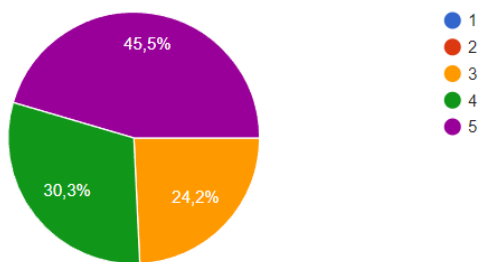
33 jawaban



Apakah aplikasi ini memudahkan anda dalam mengajukan izin

 Salin

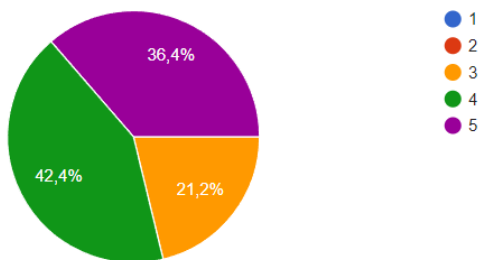
33 jawaban



Bagaimana kualitas tampilan aplikasi android ini

 Salin

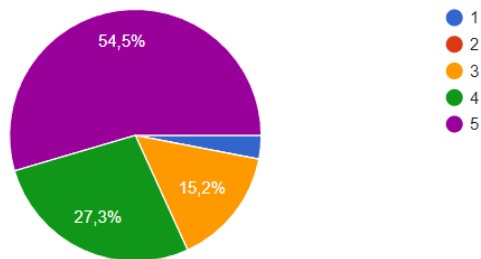
33 jawaban



Apakah aplikasi ini mudah digunakan oleh anda sebagai guru

 Salin

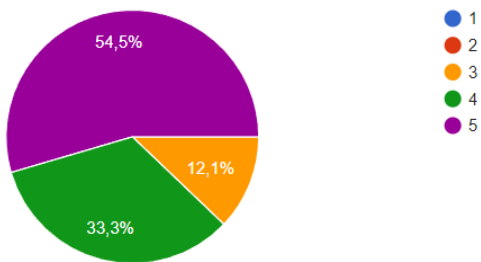
33 jawaban



Apakah aplikasi ini membantu anda dalam memantau jam masuk dan jam keluar

 Salin

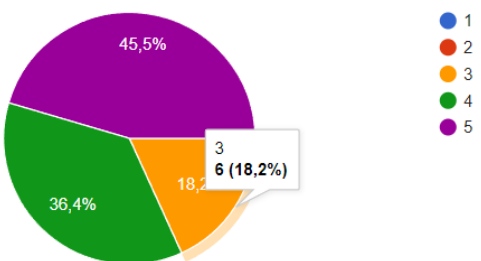
33 jawaban



Apakah aplikasi ini membantu anda dalam melihat riwayat absensi dan perizinan

 Salin

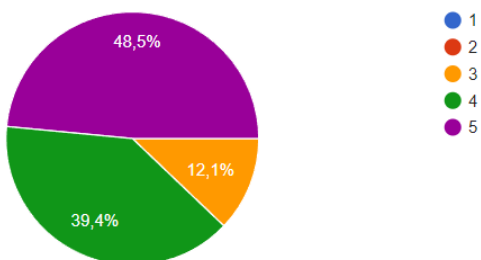
33 jawaban



Secara keseluruhan, apakah aplikasi manajemen presensi ini memuaskan bagi anda

 Salin

33 jawaban



LAMPIRAN 4

KODE PROGRAM

Esp 32

```
#include <SPI.h>
#include <MFRC522.h>
#include <WiFi.h>
#include <NTPClient.h>
#include <WiFiUdp.h>
#include <HTTPClient.h>

// Definisi pin yang digunakan
#define RST_PIN 22
#define SS_PIN 21
#define buzzer 15

// Kredensial jaringan WiFi
const char* ssid = "TUF SERIES";
const char* password = "12345672";

// URL server untuk mengambil gambar dan mengirim data
const char* cam_server = "http://192.168.137.10/capture";
const char* last_image_server = "http://192.168.137.10/last_image";
const char* SNamePulang = "https://tkdharma.site/SNamePulang.php";
const char* SNameMasuk = "https://tkdharma.site/SNameMasuk.php";
const char* AbsensiTidakHadir = "https://tkdharma.site/AbsensiTidakHadir.php";

// Fungsi untuk mengaktifkan buzzer
// frequency: frekuensi suara buzzer
```

```

// duration: durasi suara buzzer
// volume: volume suara buzzer
// void buzz(int frequency, int duration, int volume) {
//   int period = 1000000 / frequency;
//   int onTime = (period * volume) / 255;
//   int offTime = period - onTime;

//   unsigned long start = millis();
//   while (millis() - start < duration) {
//     digitalWrite(BUZZER_PIN, HIGH);
//     delayMicroseconds(onTime);
//     digitalWrite(BUZZER_PIN, LOW);
//     delayMicroseconds(offTime);
//   }
// }

// Offset waktu untuk zona waktu lokal
const long utcOffsetInSeconds = 7 * 3600;

WiFiUDP ntpUDP;
NTPClient  timeClient(ntpUDP, "pool.ntp.org", utcOffsetInSeconds,
60000);
MFRC522 rfid(SS_PIN, RST_PIN);

String last_image_filename = "";
bool hasCheckedAbsence = false;

// Fungsi setup yang dijalankan saat memulai perangkat
void setup() {
  pinMode(buzzer, OUTPUT);

```

```
Serial.begin(115200);
SPI.begin();
rfid.PCD_Init();
WiFi.begin(ssid, password);
while (WiFi.status() != WL_CONNECTED) {
    delay(1000);
    Serial.println("Connecting to WiFi...");
}
Serial.println("Connected to WiFi");
digitalWrite(buzzer, HIGH);
delay(2000);
digitalWrite(buzzer, LOW);
delay(2000);
Serial.print("ESP32 IP Address: ");
Serial.println(WiFi.localIP());

timeClient.begin();

// Mengambil nama file gambar terakhir
retrieveLastImageFilename();
}

// Fungsi loop yang dijalankan berulang kali
void loop() {
    timeClient.update();

    // Mendapatkan waktu dan tanggal saat ini
    time_t rawTime = timeClient.getEpochTime();
    struct tm *timeInfo = localtime(&rawTime);

    // Cek apakah hari ini adalah Sabtu (6) atau Minggu (0)
```

```
if (timeInfo->tm_wday == 6 || timeInfo->tm_wday == 0) {  
    Serial.println("Absensi tidak tersedia pada hari Sabtu atau Minggu.");  
    delay(60000); // Tunggu 1 menit sebelum cek lagi  
    return;  
}
```

```
int currentHour = timeClient.getHours();  
Serial.println("Tempelkan Kartu");
```

```
if (currentHour >= 12) {  
    if ( currentHour >= 20) {  
        if (!hasCheckedAbsence) {  
            checkAbsence();  
            hasCheckedAbsence = true;  
        }  
    }  
    else {  
        pulang();  
    }  
} else {  
    hasCheckedAbsence = false;  
    masuk();  
}
```

```
delay(1000);  
}
```

// Fungsi untuk mengambil nama file gambar terakhir dari server

```
void retrieveLastImageFilename() {  
    if (WiFi.status() == WL_CONNECTED) {  
        HTTPClient http;
```

```

http.begin(last_image_server);
int httpStatusCode = http.GET();
if (httpStatusCode > 0) {
    last_image_filename = http.getString();
    Serial.println("Last image filename: " + last_image_filename);
} else {
    Serial.print("Error retrieving last image filename: ");
    Serial.println(httpStatusCode);
}
http.end();
} else {
    Serial.println("WiFi disconnected");
}
}

// Fungsi untuk mengecek absensi tidak hadir
void checkAbsence() {
    if (WiFi.status() == WL_CONNECTED) {
        HTTPClient http;
        http.begin(AbsensiTidakHadir);
        int httpStatusCode = http.GET();
        if (httpStatusCode > 0) {
            String response = http.getString();
            Serial.println(httpStatusCode);
            Serial.println(response);
            Serial.println("Absence data sent successfully");
        } else {
            Serial.print("Error sending absence data: ");
            Serial.println(httpStatusCode);
        }
        http.end();
    }
}

```

```

    } else {
        Serial.println("WiFi disconnected");
    }
}

// Fungsi untuk proses absensi pulang
void pulang() {
    if (!rfid.PICC_IsNewCardPresent() || !rfid.PICC_ReadCardSerial()) {
        delay(50);
        return;
    }

    // Membaca UID kartu RFID
    String cardUID = "";
    for (byte i = 0; i < rfid.uid.size; i++) {
        cardUID += String(rfid.uid.uidByte[i] < 0x10 ? "0" : "");
        cardUID += String(rfid.uid.uidByte[i], HEX);
    }
    cardUID.toUpperCase();
    Serial.println("Card UID: " + cardUID);

    // Mendapatkan waktu dan tanggal saat ini
    String formattedTime = timeClient.getFormattedTime();
    time_t rawTime = timeClient.getEpochTime();
    struct tm * timeInfo = localtime(&rawTime);
    String formattedDate = String(timeInfo->tm_year + 1900) + "-" +
String(timeInfo->tm_mon + 1) + "-" + String(timeInfo->tm_mday);

    Serial.println("Current time: " + formattedTime);
    Serial.println("Current date: " + formattedDate);

```

```

// Mengirim data ke server jika terhubung ke WiFi
if (WiFi.status() == WL_CONNECTED) {
    HTTPClient http;
    http.begin(cam_server);
    int httpStatusCode = http.GET();
    if (httpStatusCode == 200) {
        Serial.println("Picture taken");
        retrieveLastImageFilename();
    } else {
        Serial.print("Error taking picture: ");
        Serial.println(httpStatusCode);
    }
    http.end();

    // Mengirim data absensi pulang ke server
    http.begin(SNamePulang);
    http.addHeader("Content-Type", "application/x-www-form-
urlencoded");

    String httpRequestData = "rfid=" + cardUID + "&date=" + formattedDate
+ "&time=" + formattedTime + "&image=" + last_image_filename;
    httpStatusCode = http.POST(httpRequestData);

    if (httpStatusCode > 0) {
        String response = http.getString();
        Serial.println(httpStatusCode);
        Serial.println(response);
        Serial.println("RFID data sent successfully");

        digitalWrite(buzzer, HIGH);
        delay(1000);
        digitalWrite(buzzer, LOW);
    }
}

```

```

        delay(1000);

        digitalWrite(buzzer, HIGH);
        delay(1000);
        digitalWrite(buzzer, LOW);
        delay(1000);
    } else {
        Serial.print("Error sending RFID data: ");
        Serial.println(httpResponseCode);
        digitalWrite(buzzer, HIGH);
        delay(3000);
        digitalWrite(buzzer, LOW);
        delay(3000);
    }
    http.end();
} else {
    Serial.println("WiFi disconnected");
}

delay(3000);
}

// Fungsi untuk proses absensi masuk
void masuk() {
    if (!rfid.PICC_IsNewCardPresent() || !rfid.PICC_ReadCardSerial()) {
        delay(50);
        return;
    }

    // Membaca UID kartu RFID
    String cardUID = "";

```

```

for (byte i = 0; i < rfid.uid.size; i++) {
    cardUID += String(rfid.uid.uidByte[i] < 0x10 ? "0" : "");
    cardUID += String(rfid.uid.uidByte[i], HEX);
}
cardUID.toUpperCase();
Serial.println("Card UID: " + cardUID);

// Mendapatkan waktu dan tanggal saat ini
String formattedTime = timeClient.getFormattedTime();
time_t rawTime = timeClient.getEpochTime();
struct tm * timeInfo = localtime(&rawTime);
String formattedDate = String(timeInfo->tm_year + 1900) + "-" +
String(timeInfo->tm_mon + 1) + "-" + String(timeInfo->tm_mday);

Serial.println("Current time: " + formattedTime);
Serial.println("Current date: " + formattedDate);

// Mengirim data ke server jika terhubung ke WiFi
if (WiFi.status() == WL_CONNECTED) {
    HTTPClient http;
    http.begin(cam_server);
    int httpStatusCode = http.GET();
    if (httpStatusCode == 200) {
        Serial.println("Picture taken");
        retrieveLastImageFilename();
    } else {
        Serial.print("Error taking picture: ");
        Serial.println(httpStatusCode);
    }
    http.end();
}

```

```

// Mengirim data absensi masuk ke server
http.begin(SNameMasuk);
http.addHeader("Content-Type", "application/x-www-form-
urlencoded");

String httpRequestData = "rfid=" + cardUID + "&date=" + formattedDate
+ "&time=" + formattedTime + "&image=" + last_image_filename;
httpResponseCode = http.POST(httpRequestData);

if (httpResponseCode > 0) {
    String response = http.getString();
    Serial.println(httpResponseCode);
    Serial.println(httpRequestData);
    Serial.println("RFID data sent successfully");
    digitalWrite(buzzer, HIGH);
    delay(1000);
    digitalWrite(buzzer, LOW);
    delay(1000);

    digitalWrite(buzzer, HIGH);
    delay(1000);
    digitalWrite(buzzer, LOW);
    delay(1000);
} else {
    Serial.print("Error sending RFID data: ");
    Serial.println(httpResponseCode);
    digitalWrite(buzzer, HIGH);
    delay(3000);
    digitalWrite(buzzer, LOW);
    delay(3000);
}
http.end();

```

```

    } else {
        Serial.println("WiFi disconnected");
    }

    delay(3000);
}

```

esp 32 cam

```

#include <WiFi.h>
#include <WebServer.h>
#include <esp_camera.h>
#include <HTTPClient.h>

// Gantilah dengan kredensial jaringan Anda
const char* ssid = "Red";
const char* password = "12093487";
int i = 1;
String last_image_filename = "";

// Gantilah dengan URL endpoint file manager online Anda (menggunakan
api.php)
const char* upload_url = "https://tkdharma.site/take.php";

WebServer server(80); // Membuat server di port 80

// Konfigurasi pin kamera
#define PWDN_GPIO_NUM 32
#define RESET_GPIO_NUM -1
#define LK_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

// Fungsi setup yang dijalankan saat memulai perangkat
void setup() {
  Serial.begin(115200);

  // Menghubungkan ke Wi-Fi
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(1000);
    Serial.println("Connecting to WiFi...");
  }
  Serial.println("Connected to WiFi");
  Serial.print("ESP32-CAM IP Address: ");
  Serial.println(WiFi.localIP());

  // Konfigurasi kamera
  camera_config_t config;

```

```

config.ledc_channel = LEDC_CHANNEL_0;
config.ledc_timer = LEDC_TIMER_0;
config.pin_d0 = Y2_GPIO_NUM;
config.pin_d1 = Y3_GPIO_NUM;
config.pin_d2 = Y4_GPIO_NUM;
config.pin_d3 = Y5_GPIO_NUM;
config.pin_d4 = Y6_GPIO_NUM;
config.pin_d5 = Y7_GPIO_NUM;
config.pin_d6 = Y8_GPIO_NUM;
config.pin_d7 = Y9_GPIO_NUM;
config.pin_lk = LK_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.lk_freq_hz = 20000000;
config.pixel_format = PIXFORMAT_JPEG;

config.frame_size = FRAMESIZE_SVGA; // Resolusi lebih rendah
config.jpeg_quality = 20; // Angka lebih tinggi berarti kualitas lebih rendah
config.fb_count = 1;

// Inisialisasi kamera
esp_err_t err = esp_camera_init(&config);
if (err != ESP_OK) {
    Serial.printf("Camera init failed with error 0x%x\n", err);
    return;
} else {

```

```

        Serial.println("Camera initialized successfully");
    }

    // Memulai server
    server.on("/capture", HTTP_GET, captureImage);
    server.on("/last_image", HTTP_GET, getLastImageFilename);
    server.begin();
}

// Fungsi loop yang dijalankan berulang kali
void loop() {
    server.handleClient();
}

// Fungsi untuk menangkap gambar dari kamera dan mengirimnya ke server
void captureImage() {
    camera_fb_t * fb = NULL;
    fb = esp_camera_fb_get(); // Mengambil gambar dari kamera
    if (!fb) {
        Serial.println("Camera capture failed");
        server.send(500, "text/plain", "Camera capture failed");
        return;
    }

    Serial.printf("Captured image size: %d bytes\n", fb->len);

    HTTPClient http;
    http.begin(upload_url);

    String boundary = "-----boundary123";
    String filename = "capture" + String(i) + ".jpg";

```

```

String head = "--" + boundary + "\r\n" +
    "Content-Disposition: form-data; name=\"image\"; filename=\""
+ filename + "\"\r\n" +
    "Content-Type: image/jpeg\r\n\r\n";
String tail = "\r\n--" + boundary + "--\r\n";

i++;
int contentLength = head.length() + fb->len + tail.length();

http.addHeader("Content-Type", "multipart/form-data; boundary=" +
boundary);
http.addHeader("Content-Length", String(contentLength));

// Menggabungkan payload
String payload = head;
payload += String((char*)fb->buf, fb->len);
payload += tail;

int  httpResponseCode  =  http.POST((uint8_t  *)payload.c_str(),
payload.length());

if (httpResponseCode > 0) {
    String response = http.getString();
    Serial.println("Image sent successfully");
    Serial.printf("Response code: %d\n", httpResponseCode);
    Serial.printf("Response: %s\n", response.c_str());
    server.send(200, "text/plain", "Image captured and sent successfully");
    last_image_filename = filename;
} else {
    Serial.printf("Failed to send image, response code: %d\n",
httpResponseCode);

```

```

        Serial.printf("HTTP          error:          %s\n",
http.errorToString(httpResponseCode).c_str());
        server.send(500, "text/plain", "Image capture failed");
    }

    esp_camera_fb_return(fb); // Mengembalikan frame buffer ke driver
kamera
    http.end();
}

// Fungsi untuk mendapatkan nama file gambar terakhir yang dikirim
void getLastImageFilename() {
    server.send(200, "text/plain", last_image_filename);
}

```

Absesitidakhadir.php

```

<?php
    // Koneksi ke database
    $servername = "localhost";
    $username = "u639002938_devidwij";
    $password = "Juangae3.";
    $dbname = "u639002938_absensi";

    $conn = new mysqli($servername, $username, $password, $dbname);

    // Cek koneksi
    if ($conn->connect_error) {
        die("Connection failed: " . $conn->connect_error);
    }

```

```

// Mendapatkan tanggal hari ini
$date = date("Y-m-d");

// Query untuk mendapatkan daftar semua pengguna
$sql_users = "SELECT uid, nama FROM tb_pegawai";
$result_users = $conn->query($sql_users);

if ($result_users->num_rows > 0) {
    while($row_user = $result_users->fetch_assoc()) {
        $suid = $row_user["uid"];
        $nama = $row_user["nama"];

        // Cek apakah pengguna ini telah melakukan absensi masuk atau tidak
        $sql_check = "SELECT * FROM tb_absen WHERE uid='$suid' AND
waktu='$date' AND jam_masuk IS NOT NULL";
        $result_check = $conn->query($sql_check);

        if ($result_check->num_rows == 0) {
            // Jika pengguna belum melakukan absensi masuk, maka tambahkan
data absensi tidak hadir
            $sql_absent = "INSERT INTO tb_absen (uid, waktu, nama,
keterangan) VALUES ('$suid', '$date', '$nama', 'tidak hadir')";
            if ($conn->query($sql_absent) === TRUE) {
                echo "Data tidak hadir berhasil ditambahkan untuk UID: $suid\n";
            } else {
                echo "Error: " . $sql_absent . "<br>" . $conn->error . "\n";
            }
        }
    }
} else {

```

```

        echo "Tidak ada data pengguna ditemukan\n";
    }

    $conn->close();
?>

```

add_tb_izin

```

<?php
    // Database connection details
    $servername = "localhost";
    $username = "u639002938_devidwij";
    $password = "Juangae3.";
    $dbname = "u639002938_absensi";

    // Create a new database connection
    $conn = new mysqli($servername, $username, $password, $dbname);

    if ($conn->connect_error) {
        die("Connection failed: " . $conn->connect_error);
    }

    // Validate and sanitize input data
    $uid = isset($_POST['uid']) ? $conn->real_escape_string($_POST['uid']) :
";
    $nama = isset($_POST['nama']) ? $conn->real_escape_string($_POST['nama']) :
";
    $nip = isset($_POST['nip']) ? $conn->real_escape_string($_POST['nip']) :
";
    $alasan = isset($_POST['alasan']) ? $conn->real_escape_string($_POST['alasan']) :
";

```

```

        $tanggal          =          isset($_POST['tanggal'])          ?          $conn-
>real_escape_string($_POST['tanggal']) : ";
        $status           =          isset($_POST['status'])           ?          $conn-
>real_escape_string($_POST['status']) : ";
        $foto_bukti = "";

        if(isset($_FILES['foto_bukti']['name'])                                &&
!empty($_FILES['foto_bukti']['name'])) {
            $target_dir = "img-uploads/img-perizinan/";
            $image_name = basename($_FILES['foto_bukti']['name']);
            $target_file = $target_dir . $image_name;
            if
                (move_uploaded_file($_FILES['foto_bukti']['tmp_name'],
$target_file)) {
                $foto_bukti = $image_name;
            } else {
                echo "Error uploading file.";
                $conn->close();
                exit();
            }
        }

        // Prepare and bind
        $stmt = $conn->prepare("INSERT INTO tb_izin (uid, nama, nip, alasan,
tanggal, status, foto_bukti) VALUES (?, ?, ?, ?, ?, ?, ?)");
        $stmt->bind_param("ssssss", $uid, $nama, $nip, $alasan, $tanggal,
$status, $foto_bukti);

        if ($stmt->execute()) {
            echo "New record created successfully";
        } else {
            echo "Error: " . $stmt->error;

```

```
}

// Close connections
$stmt->close();
$conn->close();
?>
```

add_uid.php

```
<?php
// Database connection details
$servername = "localhost";
$username = "u639002938_devidwij";
$password = "Juangae3.";
$dbname = "u639002938_absensi";

// Membuat koneksi
$conn = new mysqli($servername, $username, $password, $dbname);

// Cek koneksi
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $uid = $_POST['uid'];

    // Masukkan UID ke dalam tabel
    $sql = "INSERT INTO get_uid (uid) VALUES ('$uid')";

    if ($conn->query($sql) === TRUE) {
```

```

        echo "New record created successfully";
    } else {
        echo "Error: " . $sql . "<br>" . $conn->error;
    }
    $conn->close();
}
?>

```

Api.php

```

<?php
header("Content-Type: application/json; charset=UTF-8");

$servername = "localhost";
$username = "u639002938_devidwij";
$password = "Juangae3.";
$dbname = "u639002938_absensi";

$conn = new mysqli($servername, $username, $password, $dbname);

if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

$sql = "SELECT nama,
        MONTHNAME(waktu) as bulan,
        SUM(keterangan = 'sakit') as sakit,
        SUM(keterangan = 'izin') as izin,
        SUM(keterangan = 'hadir') as hadir,
        SUM(keterangan = 'tidak hadir') as tidak_hadir,
        SUM(keterangan = 'cuti') as cuti,

```

```
SUM(keterangan = 'terlambat') as terlambat
FROM tb_absen
GROUP BY nama, MONTH(waktu)";
```

```
$result = $conn->query($sql);
```

```
$data = array();
```

```
if ($result->num_rows > 0) {
```

```
    while($row = $result->fetch_assoc()) {
```

```
        $data[] = $row;
```

```
    }
```

```
}
```

```
echo json_encode($data);
```

```
$conn->close();
```

```
?>
```

get_tb_absen.php

```
<?php
```

```
$servername = "localhost";
```

```
$username = "u639002938_devidwij";
```

```
$password = "Juangae3.";
```

```
$dbname = "u639002938_absensi";
```

```
// Create a new database connection
```

```
$conn = new mysqli($servername, $username, $password, $dbname);
```

```
// Check the connection
```

```
if ($conn->connect_error) {
```

```
    http_response_code(500);
```

```

        echo json_encode(["error" => "Connection failed: " . $conn-
>connect_error]);
        exit();
    }

    // Get the username parameter from the URL
    $nama = isset($_GET['nama']) ? $_GET['nama'] : "";

    if (empty($nama)) {
        http_response_code(400);
        echo json_encode(["error" => "nama parameter is required"]);
        exit();
    }

    // Prepare and execute the SQL query
    $stmt = $conn->prepare("SELECT * FROM tb_absen WHERE nama = ?");
    $stmt->bind_param("s", $nama);
    $stmt->execute();
    $result = $stmt->get_result();

    $data = [];
    if ($result->num_rows > 0) {
        while ($row = $result->fetch_assoc()) {
            $data[] = $row;
        }
    } else {
        http_response_code(404);
        echo json_encode(["error" => "No data found for user: $nama"]);
        exit();
    }
}

```

```

$stmt->close();
$conn->close();

// Set the content type to application/json and output the data
header('Content-Type: application/json');
echo json_encode($data);
?>

```

Get_tb_izin.php

```

<?php
$servername = "localhost";
$username = "u639002938_devidwij";
$password = "Juangae3.";
$dbname = "u639002938_absensi";

// Create a new database connection
$conn = new mysqli($servername, $username, $password, $dbname);

// Check the connection
if ($conn->connect_error) {
    http_response_code(500);
    echo json_encode(["error" => "Connection failed: " . $conn-
>connect_error]);
    exit();
}

// Get the username parameter from the URL
$nama = isset($_GET['nama']) ? $_GET['nama'] : "";

if (empty($nama)) {
    http_response_code(400);

```

```

        echo json_encode(["error" => "nama parameter is required"]);
        exit();
    }

    // Prepare and execute the SQL query
    $stmt = $conn->prepare("SELECT * FROM tb_izin WHERE nama = ?");
    $stmt->bind_param("s", $nama);
    $stmt->execute();
    $result = $stmt->get_result();

    $data = [];
    if ($result->num_rows > 0) {
        while ($row = $result->fetch_assoc()) {
            $data[] = $row;
        }
    } else {
        http_response_code(404);
        echo json_encode(["error" => "No data found for user: $nama"]);
        exit();
    }

    $stmt->close();
    $conn->close();

    // Set the content type to application/json and output the data
    header('Content-Type: application/json');
    echo json_encode($data);
    ?>

```

Get_tb_izin_kepsek.php

```
<?php
$servername = "localhost";
$username = "u639002938_devidwij";
$password = "Juangae3.";
$dbname = "u639002938_absensi";

// Create a new database connection
$conn = new mysqli($servername, $username, $password, $dbname);

if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

$sql = "SELECT * FROM tb_izin";
$result = $conn->query($sql);

$data = array();

if ($result->num_rows > 0) {
    while($row = $result->fetch_assoc()) {
        $data[] = $row;
    }
}

$conn->close();

header('Content-Type: application/json');
echo json_encode($data);
?>
```

Get_tb_izin_persetujuan.php

```
<?php
$servername = "localhost";
$username = "u639002938_devidwij";
$password = "Juangae3.";
$dbname = "u639002938_absensi";

// Create a new database connection
$conn = new mysqli($servername, $username, $password, $dbname);

// Check the connection
if ($conn->connect_error) {
    http_response_code(500);
    echo json_encode(["error" => "Connection failed: " . $conn-
>connect_error]);
    exit();
}

// Get the username parameter from the URL
$status = isset($_GET['status']) ? $_GET['status'] : "";

if (empty($status)) {
    http_response_code(400);
    echo json_encode(["error" => "nama parameter is required"]);
    exit();
}

// Prepare and execute the SQL query
$stmt = $conn->prepare("SELECT * FROM tb_izin WHERE status = ?");
```

```
$stmt->bind_param("s", $status);
$stmt->execute();
$result = $stmt->get_result();

$data = [];
if ($result->num_rows > 0) {
    while ($row = $result->fetch_assoc()) {
        $data[] = $row;
    }
} else {
    http_response_code(404);
    echo json_encode(["error" => "No data found for user: $status"]);
    exit();
}

$stmt->close();
$conn->close();

// Set the content type to application/json and output the data
header('Content-Type: application/json');
echo json_encode($data);
?>
```

Index.php

```
<!DOCTYPE html>
<html lang="en">

<head>
    <title>TK Dharma Login Page</title>
    <meta charset="UTF-8" />
```

```

<meta name="viewport" content="width=device-width, initial-scale=1" />
<!--
=====
=====-->
<link rel="icon" type="image/x-icon" href="content/img/favicon.ico">
<!--
=====
=====-->
<link      rel="stylesheet"      type="text/css"      href="login-
page/vendor/bootstrap/css/bootstrap.min.css" />
<!--
=====
=====-->
<link  rel="stylesheet"  type="text/css"  href="login-page/fonts/font-
awesome-4.7.0/css/font-awesome.min.css" />
<!--
=====
=====-->
<link      rel="stylesheet"      type="text/css"      href="login-
page/vendor/animate/animate.css" />
<!--
=====
=====-->
<link  rel="stylesheet"  type="text/css"  href="login-page/vendor/css-
hamburgers/hamburgers.min.css" />
<!--
=====
=====-->
<link      rel="stylesheet"      type="text/css"      href="login-
page/vendor/select2/select2.min.css" />

```

```

<!--
=====
=====-->

<link rel="stylesheet" type="text/css" href="login-page/css/util.css" />
<link rel="stylesheet" type="text/css" href="login-page/css/main.css" />
<!--
=====
=====-->

<?php
session_start();
if (isset($_SESSION['username'])) {
    header("Location: content/pages/page-data-guru/view-data-guru.php");
}
?>
</head>

<body>
<div class="limiter">
<div class="container-login100">
<div class="wrap-login100">
<div class="login100-pic js-tilt" data-tilt>

</div>
<form class="login100-form validate-form" action="config/config-
login/login.php" method="POST">
<span class="login100-form-title"> Admin Login </span>

<div class="wrap-input100 validate-input" data-validate="masukan
username yang benar">
<input class="input100" type="text" name="username"
placeholder="Username" id="username" required />

```

```

        <span class="focus-input100"></span>
        <span class="symbol-input100">
            <i class="fa fa-user" aria-hidden="true"></i>
        </span>
    </div>

    <div class="wrap-input100 validate-input" data-validate="Password
wajib diisi">
        <input class="input100" type="password" name="pass"
placeholder="Password" id="pass" required />
        <span class="focus-input100"></span>
        <span class="symbol-input100">
            <i class="fa fa-lock" aria-hidden="true"></i>
        </span>
    </div>

    <div class="container-login100-form-btn">
        <button class="login100-form-btn" type="submit">Login</button>
    </div>

    <div class="text-center p-t-12"><br /><br /><br /><br /></div>
</form>
</div>
</div>
</div>

<!--
=====
=====-->

<script src="login-page/vendor/jquery/jquery-3.2.1.min.js"></script>

```

```

<!--
=====
=====-->
<script src="login-page/vendor/bootstrap/js/popper.js"></script>
<script src="login-page/vendor/bootstrap/js/bootstrap.min.js"></script>
<!--
=====
=====-->
<script src="login-page/vendor/select2/select2.min.js"></script>
<!--
=====
=====-->
<script src="login-page/vendor/tilt/tilt.jquery.min.js"></script>
<script>
    $(".js-tilt").tilt({
        scale: 1.1,
    });
</script>
<!--
=====
=====-->
<script src="login-page/js/main.js"></script>
</body>

</html>
?>

```

Login.php

```

<?php
    header('Content-Type: application/json');

```

```

$response = array();

if ($_SERVER['REQUEST_METHOD'] == 'POST') {
    $username = isset($_POST['username']) ? $_POST['username'] : "";
    $password = isset($_POST['password']) ? $_POST['password'] : "";

    if (!empty($username) && !empty($password)) {
        // Database connection
        $host = 'localhost';
        $db = 'u639002938_absensi';
        $user = 'u639002938_devidwij';
        $pass = 'Juangae3.';
        $charset = 'utf8mb4';

        $dsn = "mysql:host=$host;dbname=$db;charset=$charset";
        $options = [
            PDO::ATTR_ERRMODE            => PDO::ERRMODE_EXCEPTION,
            PDO::ATTR_DEFAULT_FETCH_MODE  =>
PDO::FETCH_ASSOC,
            PDO::ATTR_EMULATE_PREPARES   => false,
        ];

        try {
            $pdo = new PDO($dsn, $user, $pass, $options);

            // Check if user exists
            $stmt = $pdo->prepare('SELECT uid, username, password, role,
nama, nip, jabatan, foto FROM tb_pegawai WHERE username = ?');
            $stmt->execute([$username]);
            $user = $stmt->fetch();

```

```
if ($user) {
    // Check if the stored password is MD5 or bcrypt
    if (strlen($user['password']) == 32) {
        // Password is MD5 hashed
        if (md5($password) === $user['password']) {

        } else {
            $response['response'] = false;
            $response['message'] = 'Invalid username or password';
            echo json_encode($response);
            exit();
        }
    } elseif (password_verify($password, $user['password'])) {
        // Password is bcrypt hashed and verified
    } else {
        $response['response'] = false;
        $response['message'] = 'Invalid username or password';
        echo json_encode($response);
        exit();
    }

    // Successful login
    $response['response'] = true;
    $response['payload'] = array(
        'id_user' => $user['uid'],
        'level_user' => $user['role'],
        'nama' => $user['nama'],
        'nip' => $user['nip'],
        'jabatan' => $user['jabatan'],
        'foto' => $user['foto']
    );
}
```

```

    );
    $response['message'] = 'Login successful';
} else {
    $response['response'] = false;
    $response['message'] = 'Invalid username or password';
}
} catch (PDOException $e) {
    $response['response'] = false;
    $response['message'] = 'Database error: ' . $e->getMessage();
}
} else {
    $response['response'] = false;
    $response['message'] = 'Username and password are required';
}
} else {
    $response['response'] = false;
    $response['message'] = 'Invalid request method';
}

echo json_encode($response);
?>

```

sendPriod.php

```

<?php
// Database credentials
$servername = "localhost";
$username = "u639002938_devidwij";
$password = "Juangae3.";
$dbname = "u639002938_absensi";

```

```

// Create connection
$conn = new mysqli($servername, $username, $password, $dbname);

// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

// Check if the request method is POST
if ($_SERVER['REQUEST_METHOD'] == 'POST') {
    // Get the raw POST data
    $input = file_get_contents('php://input');
    // Decode JSON data
    $data = json_decode($input, true);

    // Extract date and time from the JSON data
    if (isset($data['date']) && isset($data['time'])) {
        $date = $data['date'];
        $time = $data['time'];

        // Prepare and bind
        $stmt = $conn->prepare("INSERT INTO tb_absen (date, time) VALUES
(?, ?)");
        $stmt->bind_param("ss", $date, $time);

        // Execute the statement
        if ($stmt->execute()) {
            echo json_encode(array("message" => "Data inserted successfully."));
        } else {
            echo json_encode(array("message" => "Failed to insert data."));
        }
    }
}

```

```
// Close statement
$stmt->close();
} else {
    echo json_encode(array("message" => "Invalid data.));
}
} else {
    echo json_encode(array("message" => "Invalid request method.));
}
```

```
// Close connection
$conn->close();
?>
```

Snamemasuk.php

<?php

```
ini_set('display_errors', 1);
ini_set('display_startup_errors', 1);
error_reporting(E_ALL);
```

```
// Database connection details
$servername = "localhost";
$username = "u639002938_devidwij";
$password = "Juangae3.";
$dbname = "u639002938_absensi";
```

```
// Create a new database connection
$conn = new mysqli($servername, $username, $password, $dbname);
```

```
// Check the connection
```

```

if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

// Check if the request method is POST
if ($_SERVER['REQUEST_METHOD'] == 'POST') {
    // Check if the required POST parameters are set
    if (isset($_POST['rfid']) && isset($_POST['date']) &&
isset($_POST['time']) && isset($_POST['image'])) {
        $rfid = $_POST['rfid'];
        $date = $_POST['date'];
        $time = $_POST['time'];
        $image = $_POST['image'];

        // Retrieve the name from the tb_pegawai table
        $stmt = $conn->prepare("SELECT nama FROM tb_pegawai WHERE
uid = ?");
        $stmt->bind_param("s", $rfid);
        $stmt->execute();
        $stmt->bind_result($nama);
        $stmt->fetch();
        $stmt->close();

        if ($nama) {
            // Check if there is already a jam_masuk for the given date
            $stmt = $conn->prepare("SELECT jam_masuk FROM tb_absen
WHERE uid = ? AND waktu = ?");
            $stmt->bind_param("ss", $rfid, $date);
            $stmt->execute();
            $stmt->bind_result($masuk);
            $stmt->fetch();

```

```

$stmt->close();

if ($masuk) {
    // User has already clocked in, prompt to wait for clock out
    $stmt = $conn->prepare("INSERT INTO get_uid (uid) VALUES
(?)");

    $stmt->bind_param("s", $rfid);
    if ($stmt->execute()) {
        echo "Data updated successfully for " . $rfid . ". Silahkan
tunggu waktu pulang.";
    } else {
        echo "Error: " . $stmt->error;
    }
    $stmt->close();
} else {
    // Insert clock in data
    $stmt = $conn->prepare("INSERT INTO tb_absen (uid, waktu,
nama, jam_masuk, foto_masuk) VALUES (?, ?, ?, ?, ?)");
    $stmt->bind_param("sssss", $rfid, $date, $nama, $time, $image);

    if ($stmt->execute()) {
        echo "Data inserted successfully for " . $nama . ".";
    } else {
        echo "Error: " . $stmt->error;
    }
    $stmt->close();
}
} else {
    // No matching employee found
    $stmt = $conn->prepare("INSERT INTO get_uid (uid) VALUES
(?)");

```

```

$stmt->bind_param("s", $rfid);
if ($stmt->execute()) {
    echo "Data updated successfully for " . $rfid . ".";
} else {
    echo "Error: " . $stmt->error;
}
echo "No matching employee found.";
$stmt->close();
}
} else {
    // Missing required parameters
    echo "Missing required parameters.";
}
}

// Close the database connection
$conn->close();
?>

```

SnamePulang.php

<?php

```

ini_set('display_errors', 1);
ini_set('display_startup_errors', 1);
error_reporting(E_ALL);

$servername = "localhost";
$username = "u639002938_devidwij";
$password = "Juangae3.";
$dbname = "u639002938_absensi";

```

```

$conn = new mysqli($servername, $username, $password, $dbname);

if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

if ($_SERVER['REQUEST_METHOD'] == 'POST') {
    if (isset($_POST['rfid']) && isset($_POST['date']) &&
isset($_POST['time']) && isset($_POST['image'])) {
        $rfid = $_POST['rfid'];
        $date = $_POST['date'];
        $time = $_POST['time'];
        $image = $_POST['image'];

        $stmt = $conn->prepare("SELECT nama FROM tb_pegawai WHERE
uid = ?");
        $stmt->bind_param("s", $rfid);
        $stmt->execute();
        $stmt->bind_result($name);
        $stmt->fetch();
        $stmt->close();

        $stmt = $conn->prepare("SELECT jam_masuk FROM tb_absen
WHERE uid = ? AND waktu = ?");
        $stmt->bind_param("ss", $rfid, $date);
        $stmt->execute();
        $stmt->bind_result($masuk);
        $stmt->fetch();
        $stmt->close();

        if ($name && $masuk) {

```

```

// Check if the employee is late
$stmt = null;
if (strtotime($masuk) > strtotime("07:00:00")) {
    $stmt = $conn->prepare("UPDATE tb_absen SET jam_pulang =
?, foto_pulang = ?, keterangan = 'terlambat' WHERE waktu = ? AND nama = ?");
} else {
    $stmt = $conn->prepare("UPDATE tb_absen SET jam_pulang =
?, foto_pulang = ?, keterangan = 'hadir' WHERE waktu = ? AND nama = ?");
}
$stmt->bind_param("ssss", $time, $image, $date, $name);

if ($stmt->execute()) {
    echo "Data updated successfully for " . $name . ".";
} else {
    echo "Error: " . $stmt->error;
}
$stmt->close();
} else {
    $stmt = null;
    $stmt = $conn->prepare("INSERT INTO get_uid (uid) VALUES
(?)");

    $stmt->bind_param("s", $rfid);
    if ($stmt->execute()) {
        echo "Data updated successfully for " . $rfid . ".";
    } else {
        echo "Error: " . $stmt->error;
    }
    echo "No matching employee found.";
    $stmt->close();
}

```

```

    } else {
        $stmt = null;
        $stmt = $conn->prepare("INSERT INTO get_uid (uid) VALUES
(?)");

        $stmt->bind_param("s", $rfid);
        if ($stmt->execute()) {
            echo "Data updated successfully for " . $rfid . ".";
        } else {
            echo "Error: " . $stmt->error;
        }
        echo "Missing required parameters.";
        $stmt->close();
    }
}

$conn->close();
?>

```

Take.php

```

<?php
header('Content-Type: application/json');

$target_dir = "img-uploads/img-absen/";
if (!is_dir($target_dir)) {
    mkdir($target_dir, 0777, true);
}

$response = array();

```

```

        if      ($_SERVER['REQUEST_METHOD']      ==      'POST'      &&
isset($_FILES['image'])) {
            $target_file = $target_dir . basename($_FILES["image"]["name"]);
            $uploadOk = 1;
            $imageFileType      =      strtolower(pathinfo($target_file,
PATHINFO_EXTENSION));

            // Check if image file is a actual image or fake image
            $check = getimagesize($_FILES["image"]["tmp_name"]);
            if ($check !== false) {
                $uploadOk = 1;
            } else {
                $response['error'] = "File is not an image.";
                $uploadOk = 0;
            }

            // Check if file already exists
            if (file_exists($target_file)) {
                $response['error'] = "Sorry, file already exists.";
                $uploadOk = 0;
            }

            // Check file size
            if ($_FILES["image"]["size"] > 5000000) { // 5MB limit
                $response['error'] = "Sorry, your file is too large.";
                $uploadOk = 0;
            }

            // Allow certain file formats
            if ($imageFileType != "jpg" && $imageFileType != "jpeg" &&
$imageFileType != "png") {

```

```

        $response['error'] = "Sorry, only JPG, JPEG & PNG files are
allowed.";
        $uploadOk = 0;
    }

    // Check if $uploadOk is set to 0 by an error
    if ($uploadOk == 0) {
        $response['error'] = "Sorry, your file was not uploaded.";
        // if everything is ok, try to upload file
    } else {
        if (move_uploaded_file($_FILES["image"]["tmp_name"],
$target_file)) {
            $response['success'] = "The file " .
htmlspecialchars(basename($_FILES["image"]["name"])) . " has been uploaded.";
        } else {
            $response['error'] = "Sorry, there was an error uploading your file.";
        }
    }
} else {
    $response['error'] = "Invalid request.";
}

echo json_encode($response);
?>

```

Update_tb_izin_status.php

```

<?php
$servername = "localhost";
$username = "u639002938_devidwij";
$password = "Juangae3.";
$dbname = "u639002938_absensi";

```

```

// Create a new database connection
$conn = new mysqli($servername, $username, $password, $dbname);

// Check the connection
if ($conn->connect_error) {
    http_response_code(500);
    echo json_encode(["error" => "Connection failed: " . $conn-
>connect_error]);
    exit();
}

// Get the id and status parameters from the request
$id_izin = isset($_POST['id_izin']) ? $_POST['id_izin'] : "";
$status = isset($_POST['status']) ? $_POST['status'] : "";

if (empty($id_izin) || empty($status)) {
    http_response_code(400);
    echo json_encode(["error" => "id_izin and status parameters are
required"]);
    exit();
}

// Prepare and execute the SQL query to update the status
$stmt = $conn->prepare("UPDATE tb_izin SET status = ? WHERE id_izin
= ?");
$stmt->bind_param("si", $status, $id_izin);
$stmt->execute();

if ($stmt->affected_rows > 0) {
    // Fetch the related data from tb_izin

```

```

$stmt = $conn->prepare("SELECT tanggal, uid, nama, alasan FROM
tb_izin WHERE id_izin = ?");
$stmt->bind_param("i", $id_izin);
$stmt->execute();
$result = $stmt->get_result();
if ($result->num_rows > 0) {
    $row = $result->fetch_assoc();
    $tanggal = $row['tanggal'];
    $uid = $row['uid'];
    $nama = $row['nama'];
    $keterangan = $row['alasan'];

    // Insert the data into tb_absen
    $stmt = $conn->prepare("INSERT INTO tb_absen (waktu, uid, nama,
keterangan) VALUES (?, ?, ?, ?)");
    $stmt->bind_param("ssss", $tanggal, $uid, $nama, $keterangan);
    $stmt->execute();

    if ($stmt->affected_rows > 0) {
        echo json_encode(["success" => "Status updated and data inserted
into tb_absen successfully"]);
    } else {
        http_response_code(500);
        echo json_encode(["error" => "Failed to insert data into tb_absen"]);
    }
} else {
    http_response_code(404);
    echo json_encode(["error" => "No data found for id: $id_izin"]);
}
} else {
    http_response_code(404);

```

```

        echo json_encode(["error" => "No data found for id: $id_izin"]);
    }

```

```

$stmt->close();
$conn->close();
?>

```

Config/config-tambah-data/tambah-data.php

```

<?php
include('../config-connection/config.php');

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $uid = $_POST['uid'];
    $username = $_POST['username'];
    $password = md5($_POST['password']); // Menggunakan MD5 untuk
enkripsi password
    $nama = $_POST['nama'];
    $nip = $_POST['nip'];
    $jabatan = $_POST['jabatan'];
    $status = $_POST['status'];
    $bergabung = $_POST['bergabung'];

    // Proses upload file
    $target_dir = "../img-uploads/img-profile/";
    $target_file = $target_dir . basename($_FILES["foto"]["name"]);
    $imageFileType = strtolower(pathinfo($target_file,
PATHINFO_EXTENSION));

    // Validasi file gambar
    $check = getimagesize($_FILES["foto"]["tmp_name"]);
    if ($check === false) {

```

```
die(json_encode(["error" => "File bukan gambar."]));  
}  
  
if ($_FILES["foto"]["size"] > 500000) {  
    die(json_encode(["error" => "Maaf, file terlalu besar."]));  
}  
  
if ($imageFileType != "jpg" && $imageFileType != "png" &&  
$imageFileType != "jpeg" && $imageFileType != "gif") {  
    die(json_encode(["error" => "Maaf, hanya file JPG, JPEG, PNG &  
GIF yang diperbolehkan."]));  
}  
  
// Check if file already exists  
// if (file_exists($target_file)) {  
//     die(json_encode(["error" => "Maaf, file sudah ada."]));  
// }  
  
// Move uploaded file to target directory  
if (!move_uploaded_file($_FILES["foto"]["tmp_name"], $target_file)) {  
    die(json_encode(["error" => "Maaf, terjadi kesalahan saat  
mengunggah file."]));  
}  
  
// Menyimpan data ke database hanya jika proses upload file berhasil  
$sql = "INSERT INTO tb_pegawai (uid, username, password, role, nama,  
nip, jabatan, status, bergabung, foto)  
VALUES ('$uid', '$username', '$password', 'user', '$nama', '$nip',  
'$jabatan', '$status', '$bergabung', '' . basename($_FILES["foto"]["name"]) . ".");"  
  
if (mysqli_query($conn, $sql)) {
```

```

        echo json_encode(["success" => true]);
    } else {
        // Jika terjadi kesalahan saat menyimpan ke database, tambahkan
        penanganan sesuai kebutuhan
        // Misalnya, bisa menghapus file yang sudah terupload
        unlink($target_file); // Hapus file yang sudah terupload
        die(json_encode(["error" => "Error: " . $sql . "<br>" .
        mysqli_error($conn)]));
    }

```

```

    mysqli_close($conn);
}

```

Config/config-logout/logout.php

```

<?php
session_start();
session_unset();
session_destroy();

header("Location: ../../index.php");
exit();

```

config-config-login/login.php

```

<?php
include '../config-connection/config.php';

// Pastikan method yang digunakan adalah POST
if ($_SERVER['REQUEST_METHOD'] == 'POST') {
    $username = $_POST['username'];
    // $password = $_POST['pass'];

```

```

$password = md5($_POST['pass']);

// Query SQL untuk mencari pengguna dengan username dan password
tertentu
$query = "SELECT * FROM tb_pegawai WHERE
username='$username' AND password='$password' AND role = 'admin'";

// Menjalankan query
$login = mysqli_query($conn, $query);
$cek = mysqli_num_rows($login);

session_start();
// Memeriksa apakah pengguna ditemukan
if ($cek > 0) {

    $_SESSION['username'] = $username;
    $_SESSION['role'] = 'admin';
    header("Location: ../../content/pages/page-data-guru/view-data-
guru.php");
    exit;
} else {
    // Jika tidak ditemukan, tampilkan pesan error menggunakan
SweetAlert2
    echo <<<HTML
    <!DOCTYPE html>
    <html lang="en">

    <head>
    <title>Login Page</title>
    <meta charset="UTF-8" />

```

```

<meta name="viewport" content="width=device-width, initial-
scale=1" />
<!--
=====
=====-->
<link rel="icon" type="image/x-icon"
href="../../content/img/favicon.ico">
<!--
=====
=====-->
<link rel="stylesheet" type="text/css" href="../../login-
page/vendor/bootstrap/css/bootstrap.min.css" />
<!--
=====
=====-->
<link rel="stylesheet" type="text/css" href="../../login-
page/fonts/font-awesome-4.7.0/css/font-awesome.min.css" />
<!--
=====
=====-->
<link rel="stylesheet" type="text/css" href="../../login-
page/vendor/animate/animate.css" />
<!--
=====
=====-->
<link rel="stylesheet" type="text/css" href="../../login-
page/vendor/css-hamburgers/hamburgers.min.css" />
<!--
=====
=====-->

```

```

        <link      rel="stylesheet"      type="text/css"      href="../../login-
page/vendor/select2/select2.min.css" />
        <!--
=====
=====-->
        <link      rel="stylesheet"      type="text/css"      href="../../login-
page/css/util.css" />
        <link      rel="stylesheet"      type="text/css"      href="../../login-
page/css/main.css" />
        <link
            rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/sweetalert2@11/dist/sweetalert2.min.css"
        />
        <!--
=====
=====-->
        <?php
include 'config/config-connection/config.php';
?>

</head>

<body>
<div class="limiter">
    <div class="container-login100">
        <div class="wrap-login100">
            <div class="login100-pic js-tilt" data-tilt>
                
            </div>

```

```

        <form          class="login100-form          validate-form"
action="config/config-login/login.php" method="POST">
        <span class="login100-form-title"> Admin Login </span>

        <div          class="wrap-input100          validate-input"          data-
validate="masukan username yang benar">
                <input          class="input100"          type="text"          name="username"
placeholder="Username" id="username" required />
                <span class="focus-input100"></span>
                <span class="symbol-input100">
                        <i class="fa fa-user" aria-hidden="true"></i>
                </span>
        </div>

        <div          class="wrap-input100          validate-input"          data-
validate="Password wajib diisi">
                <input          class="input100"          type="password"          name="pass"
placeholder="Password" id="pass" required />
                <span class="focus-input100"></span>
                <span class="symbol-input100">
                        <i class="fa fa-lock" aria-hidden="true"></i>
                </span>
        </div>

        <div class="container-login100-form-btn">
                <button
                                class="login100-form-btn"
type="submit">Login</button>
        </div>

        <div class="text-center p-t-12"><br /><br /><br /><br /></div>
</form>

```

```

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

<!--
=====
=====-->

    <script src="../../login-page/vendor/jquery/jquery-
3.2.1.min.js"></script>
    <!--
=====
=====-->

    <script src="../../login-page/vendor/bootstrap/js/popper.js"></script>
    <script src="../../login-
page/vendor/bootstrap/js/bootstrap.min.js"></script>
    <!--
=====
=====-->

    <script src="../../login-page/vendor/select2/select2.min.js"></script>
    <!--
=====
=====-->

    <script src="../../login-page/vendor/tilt/tilt.jquery.min.js"></script>
    <script>
        $($.js-tilt).tilt({
            scale: 1.1,
        });
    </script>
    <!--
=====
=====-->

```

```

<script src="../../../login-page/js/main.js"></script>
<script src="https://cdn.jsdelivr.net/npm/sweetalert2@11"></script>
<script>
    Swal.fire({
        title: 'OPPSS!',
        text: 'Sepertinya username dan password salah.',
        icon: 'error',
        confirmButtonText: 'OK'
    }).then(() => {
        window.location.href = '../../../index.php'
    });
</script>
</body>

</html>
?>
HTML;
}
} else {
    // handel jika field username dan pass kosong
    header("Location: ../../../index.php");
    exit;
}

// Tutup koneksi ke database
mysqli_close($conn);

config/config-hapus-data/hapus-data.php
<?php
include('../../../config-connection/config.php');

```

```

if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    $id = $_POST['id'];

    // Periksa apakah ID valid
    if (!empty($id)) {

        // Periksa koneksi
        if ($conn->connect_error) {
            die("Koneksi gagal: " . $conn->connect_error);
        }

        // Query untuk menghapus data berdasarkan ID
        $sql = "DELETE FROM tb_pegawai WHERE uid = ?";
        $stmt = $conn->prepare($sql);
        $stmt->bind_param('i', $id);

        if ($stmt->execute()) {
            // Redirect kembali ke halaman sebelumnya setelah penghapusan
            header("Location: ../../content/pages/page-data-guru/view-data-
guru.php");
            exit();
        } else {
            echo "Gagal menghapus data.";
        }

        $stmt->close();
        $conn->close();
    } else {
        echo "ID tidak valid.";
    }
}

```

Config/config-get-uid/fetch-uid.php

```
<?php
header('Content-Type: application/json');

// Include file konfigurasi database
include('../config-connection/config.php');

// Pastikan tidak ada output tambahan sebelum JSON
ob_start();

$sql = "SELECT uid FROM get_uid ORDER BY id DESC LIMIT 1";
$result = $conn->query($sql);

if ($result->num_rows > 0) {
    $row = $result->fetch_assoc();
    echo json_encode(array('uid' => $row['uid']));
} else {
    echo json_encode(array('uid' => ""));
}

$conn->close();
?>
```

Config/config-edit-data/edit-data.php

```
<?php
// Include file konfigurasi database
include('../config-connection/config.php');

// Pastikan file ini dipanggil dengan metode POST
if ($_SERVER["REQUEST_METHOD"] == "POST") {
```

```

// Tangkap data yang dikirim dari form
$uid = $_POST['uid'];
$nama = $_POST['nama'];
$nip = $_POST['nip'];
$jabatan = $_POST['jabatan'];
$status = $_POST['status'];
$bergabung = $_POST['bergabung'];

// Proses untuk update data ke dalam database
$sql = "UPDATE tb_pegawai SET nama='$nama', nip='$nip',
jabatan='$jabatan', status='$status', bergabung='$bergabung' WHERE uid='$uid'";

// Handle upload foto jika ada file yang diunggah
if ($_FILES['foto']['name']) {
    $target_dir = "../img-uploads/img-profile/";
    $target_file = $target_dir . basename($_FILES["foto"]["name"]);
    $uploadOk = 1;
    $imageFileType = strtolower(pathinfo($target_file,
PATHINFO_EXTENSION));

    // Check if image file is an actual image or fake image
    $check = getimagesize($_FILES["foto"]["tmp_name"]);
    if ($check !== false) {
        $uploadOk = 1;
    } else {
        echo "File is not an image.";
        $uploadOk = 0;
    }

    // Check file size
    if ($_FILES["foto"]["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.";
    } else {
        if (move_uploaded_file($_FILES["foto"]["tmp_name"],
        $target_file)) {
            echo "The file " .
            htmlspecialchars(basename($_FILES["foto"]["name"])) . " has been uploaded.";
            // Update nama file foto di database
            $trimmed_filename =
            trim(basename($_FILES["foto"]["name"])); // Trim the filename
            $sql_update_foto = "UPDATE tb_pegawai SET
            foto='$trimmed_filename' WHERE uid='$uid'";
            mysqli_query($conn, $sql_update_foto);
        } else {
            echo "Sorry, there was an error uploading your file.";
        }
    }

```

```

    }
}

if (mysqli_query($conn, $sql)) {
    mysqli_close($conn);
    // Redirect atau kembali ke halaman sebelumnya
    header('Location: ../../content/pages/page-data-guru/view-data-
guru.php');
    exit;
} else {
    // Jika query update gagal
    echo "Terjadi kesalahan: " . mysqli_error($conn);
}
} else {
    // Jika file ini diakses langsung tanpa melalui metode POST
    echo "Metode akses tidak diizinkan.";
}
?>

```

Config/config-download-data/download-data.php

```

<?php
// Koneksi ke database dan query untuk mengambil data
include('../../config-connection/config.php');

// Query untuk mengambil data
$result = mysqli_query($conn, "SELECT
    nama,
    SUM(CASE WHEN keterangan = 'hadir' THEN 1 ELSE 0
END) AS total_hadir,
    SUM(CASE WHEN keterangan = 'tidak hadir' THEN 1
ELSE 0 END) AS total_tidak_hadir,

```

```

SUM(CASE WHEN keterangan = 'sakit' THEN 1 ELSE 0
END) AS total_sakit,
SUM(CASE WHEN keterangan = 'izin' OR keterangan =
'cuti' THEN 1 ELSE 0 END) AS total_izin
FROM
tb_absen
WHERE
MONTH(waktu) = 7 AND YEAR(waktu) = 2024
GROUP BY
nama");

```

```

// Buat file CSV sementara di server
$filename = 'absensi_guru.csv';
$filepath = '/path/to/your/files/' . $filename; // Sesuaikan dengan path yang
sesuai di server Anda

```

```

$output = fopen($filepath, 'w');
// Tulis header kolom
fputcsv($output, ['Guru', 'Total Hadir', 'Total Tidak Hadir', 'Total Sakit',
'Total Izin']);

```

```

// Tulis data dari hasil query
while ($row = mysqli_fetch_assoc($result)) {
    fputcsv($output, [$row['nama'], $row['total_hadir'],
$row['total_tidak_hadir'], $row['total_sakit'], $row['total_izin']]);
}

```

```

fclose($output);

```

```

// Set header untuk unduhan
header('Content-Type: text/csv');

```

```
header('Content-Disposition: attachment; filename="' . $filename . '"');
header('Cache-Control: no-cache, no-store, must-revalidate'); // No cache
header('Pragma: no-cache');
header('Expires: 0');
header('Content-Length: ' . filesize($filepath));
```

```
// Baca dan kirimkan file CSV ke output
readfile($filepath);
```

```
// Hapus file setelah selesai diunduh
unlink($filepath);
```

config/config-connection/config.php

```
<?php
$servername = "localhost";
$username = "u639002938_devidwij";
$password = "Juangae3.";
$dbname = "u639002938_absensi";
```

```
// Membuat koneksi
```

```
$conn = new mysqli($servername, $username, $password, $dbname);
```

```
// Mengecek koneksi
```

```
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
```

Config/config-approve-reject/reject.php

```
<?php
include('../config-connection/config.php');
```

```

$id = $_GET['id'];

$sql = "UPDATE tb_izin SET status='tidak disetujui' WHERE
id_izin='$id'";
if (mysqli_query($conn, $sql)) {
    echo "Status berhasil diubah menjadi tidak disetujui";
} else {
    echo "Error: " . mysqli_error($conn);
}

$sql2 = "SELECT * FROM tb_izin where id_izin = '$id'";
$result = mysqli_query($conn, $sql2);
if (mysqli_num_rows($result) > 0) {
    while ($row = mysqli_fetch_assoc($result)) {
        $uid = $row['uid'];
        $foto = $row['foto_bukti'];
        $nama = $row['nama'];
        $nip = $row['nip'];
        $jenis_cuti = $row['alasan'];
        $tanggal_cuti = $row['tanggal'];
    }
    $sql_insert = "INSERT INTO tb_absen (waktu, uid, nama, jam_masuk,
jam_pulang, foto_masuk, foto_pulang, keterangan)
VALUES ('$tanggal_cuti', '$uid', '$nama', '07:00', '12:00', '$foto', '$foto',
'tidak hadir') ";
    mysqli_query($conn, $sql_insert);
}
mysqli_close($conn);

// Redirect kembali ke halaman sebelumnya

```

```
header('Location:    ../../content/pages/page-menunggu-persetujuan/view-  
menunggu-persetujuan.php');
```

```
config/config-approve-reject/approve.php
```

```
<?php
```

```
include('../../config-connection/config.php');
```

```
$id = $_GET['id'];
```

```
$sql = "UPDATE tb_izin SET status='disetujui' WHERE id_izin='$id'";
```

```
if (mysqli_query($conn, $sql)) {
```

```
    echo "Status berhasil diubah menjadi disetujui";
```

```
} else {
```

```
    echo "Error: " . mysqli_error($conn);
```

```
}
```

```
$sql2 = "SELECT * FROM tb_izin where id_izin = '$id'";
```

```
$result = mysqli_query($conn, $sql2);
```

```
if (mysqli_num_rows($result) > 0) {
```

```
    while ($row = mysqli_fetch_assoc($result)) {
```

```
        $uid = $row['uid'];
```

```
        $foto = $row['foto_bukti'];
```

```
        $nama = $row['nama'];
```

```
        $nip = $row['nip'];
```

```
        $jenis_cuti = $row['alasan'];
```

```
        $tanggal_cuti = $row['tanggal'];
```

```
    }
```

```
    $sql_insert = "INSERT INTO tb_absen (waktu, uid, nama, jam_masuk,  
jam_pulang, foto_masuk, foto_pulang, keterangan)
```

```
VALUES ('$tanggal_cuti', '$uid', '$nama', '07:00', '12:00', '$foto', '$foto',
'$jenis_cuti') ";
mysqli_query($conn, $sql_insert);
}
```

```
mysqli_close($conn);
```

```
// Redirect kembali ke halaman sebelumnya
header('Location: ../../content/pages/page-menunggu-persetujuan/view-
menunggu-persetujuan.php');
```

```
login-page/index.php
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Login Page</title>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1"
/>

    <!--
=====
=====-->

    <link rel="icon" type="image/png" href="images/icons/favicon.ico" />
    <!--

=====
=====-->

    <link
      rel="stylesheet"
      type="text/css"
      href="vendor/bootstrap/css/bootstrap.min.css"
/>
```

```
<!--  
=====
```

```
=====
```

```
----->
```

```
<link  
  rel="stylesheet"  
  type="text/css"  
  href="fonts/font-awesome-4.7.0/css/font-awesome.min.css"  
>
```

```
<!--  
=====
```

```
----->
```

```
<link          rel="stylesheet"          type="text/css"  
href="vendor/animate/animate.css" />
```

```
<!--  
=====
```

```
----->
```

```
<link  
  rel="stylesheet"  
  type="text/css"  
  href="vendor/css-hamburgers/hamburgers.min.css"  
>
```

```
<!--  
=====
```

```
----->
```

```
<link  
  rel="stylesheet"  
  type="text/css"  
  href="vendor/select2/select2.min.css"  
>
```

```

<!--
=====
=====-->

<link rel="stylesheet" type="text/css" href="css/util.css" />
<link rel="stylesheet" type="text/css" href="css/main.css" />
<!--
=====
=====-->

<script>
function validateAndRedirect(event) {
    event.preventDefault(); // Mencegah pengiriman form default
    const username = document.getElementById("username").value;
    const password = document.getElementById("pass").value;

    if (username && password) {
        window.location.href = "../admin-dashboard/pages/dashboard.html";
// Ganti dengan halaman tujuan
    }
}
</script>
</head>
<body>
<div class="limiter">
<div class="container-login100">
<div class="wrap-login100">
<div class="login100-pic js-tilt" data-tilt>

</div>

<form
    class="login100-form validate-form"

```

```
onsubmit="validateAndRedirect(event)"
>
<span class="login100-form-title"> Admin Login </span>

<div
  class="wrap-input100 validate-input"
  data-validate="masukan username yang benar"
>
  <input
    class="input100"
    type="text"
    name="username"
    placeholder="Username"
    id="username"
  />
  <span class="focus-input100"></span>
  <span class="symbol-input100">
    <i class="fa fa-user" aria-hidden="true"></i>
  </span>
</div>

<div
  class="wrap-input100 validate-input"
  data-validate="Password wajib diisi"
>
  <input
    class="input100"
    type="password"
    name="pass"
    placeholder="Password"
    id="pass"
```

```

/>
<span class="focus-input100"></span>
<span class="symbol-input100">
  <i class="fa fa-lock" aria-hidden="true"></i>
</span>
</div>

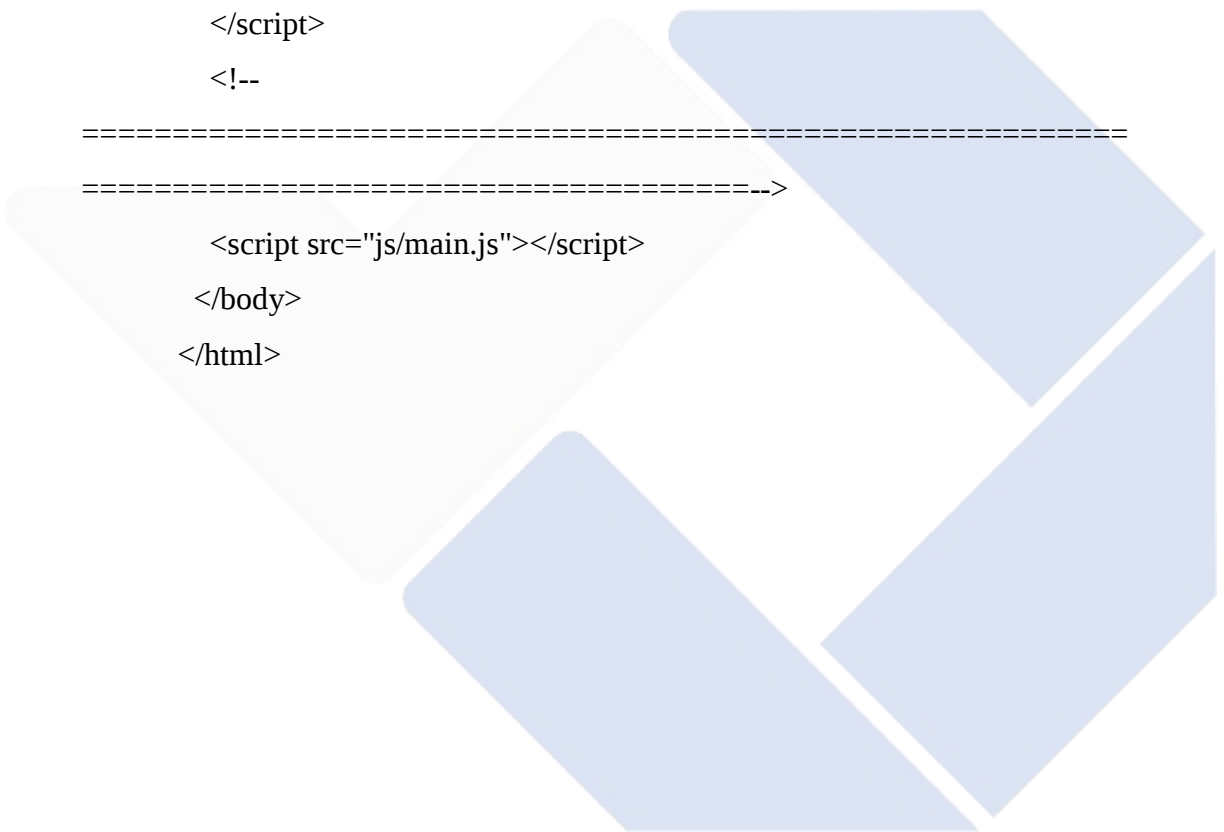
<div class="container-login100-form-btn">
  <button class="login100-form-btn">Login</button>
</div>

<div class="text-center p-t-12"><br /><br /><br /><br /></div>
</form>
</div>
</div>

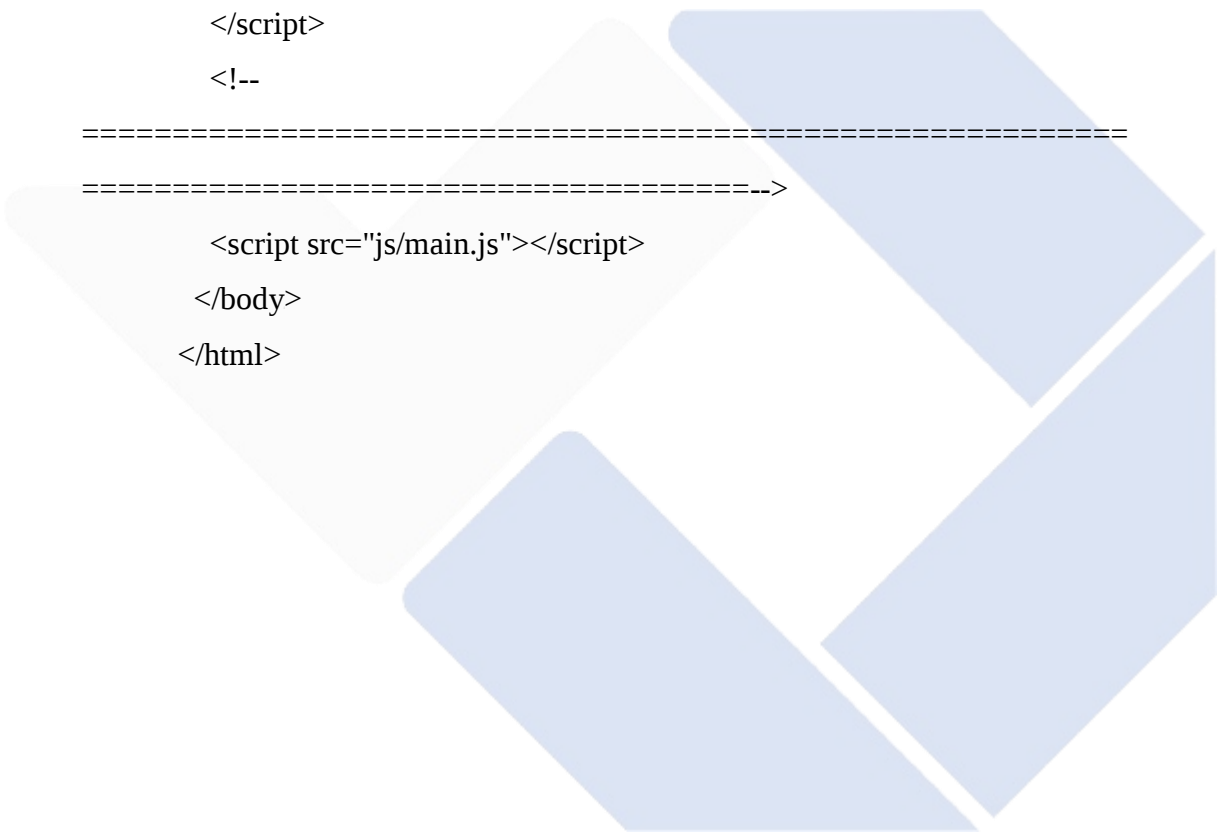
<!--
=====
=====-->
<script src="vendor/jquery/jquery-3.2.1.min.js"></script>
<!--
=====
=====-->
<script src="vendor/bootstrap/js/popper.js"></script>
<script src="vendor/bootstrap/js/bootstrap.min.js"></script>
<!--
=====
=====-->
<script src="vendor/select2/select2.min.js"></script>

```

```
<!--  
=====
```



```
=====-->  
  
    <script src="vendor/tilt/tilt.jquery.min.js"></script>  
    <script>  
        $(".js-tilt").tilt({  
            scale: 1.1,  
        });  
    </script>  
    <!--  
=====
```



```
=====-->  
  
    <script src="js/main.js"></script>  
    </body>  
</html>
```

HomeActivity.kt

```
package com.example.absensiapk.activity
```

```
import android.content.Intent
import android.os.Bundle
import android.view.MenuItem
import androidx.appcompat.app.ActionBarDrawerToggle
import androidx.appcompat.app.AppCompatActivity
import androidx.appcompat.widget.Toolbar
import androidx.core.view.GravityCompat
import androidx.drawerlayout.widget.DrawerLayout
import com.example.absensiapk.fragment.user.HomeFragment
import com.example.absensiapk.fragment.user.PerizinanFragment
import com.example.absensiapk.R
import com.example.absensiapk.fragment.RiwayatAbsensiFragment
import com.example.absensiapk.helper.PrefHelper
import com.google.android.material.navigation.NavigationView
```

```
/**
```

```
 * HomeActivity adalah aktivitas utama yang menampilkan navigation
drawer dan menangani transaksi fragment
```

```
 * berdasarkan pilihan navigasi pengguna.
```

```
*/
```

```
class HomeActivity : AppCompatActivity(),
NavigationView.OnNavigationItemSelectedListener {
```

```
    // Properti yang diinisialisasi kemudian untuk DrawerLayout dan
    PrefHelper
```

```
    private lateinit var drawerLayout: DrawerLayout
```

```
    private lateinit var prefHelper: PrefHelper
```

```

/**
 * Dipanggil saat aktivitas pertama kali dibuat.
 * @param savedInstanceState Jika aktivitas sedang diinisialisasi ulang
setelah sebelumnya ditutup, maka Bundle ini berisi data yang baru saja disediakan
di onSaveInstanceState(Bundle).
 */
override fun onCreate(savedInstanceState: Bundle?) {
    super.onCreate(savedInstanceState)
    setContentView(R.layout.activity_home)

    // Inisialisasi PrefHelper
    prefHelper = PrefHelper(this)

    // Inisialisasi DrawerLayout
    drawerLayout = findViewById(R.id.drawer_layout)

    // Menyiapkan toolbar sebagai app bar untuk aktivitas
    val toolbar = findViewById<Toolbar>(R.id.toolbar)
    setSupportActionBar(toolbar)

    // Menyiapkan NavigationView dan listener item terpilihnya
    val navigationView =
        findViewById<NavigationView>(R.id.nav_view)
    navigationView.setNavigationItemSelectedListener(this)

    // Menyiapkan ActionBarDrawerToggle untuk mengelola navigation
    drawer
    val toggle = ActionBarDrawerToggle(
        this,
        drawerLayout,

```

```

        toolbar,
        R.string.open_nav,
        R.string.close_nav
    )
    drawerLayout.addDrawerListener(toggle)
    toggle.syncState()

    // Memuat fragment default (HomeFragment) saat aktivitas pertama
    kali dibuat
    if (savedInstanceState == null) {
        supportFragmentManager.beginTransaction()
            .replace(R.id.fragment_container, HomeFragment())
            .commitNow()
    }
    navigationView.setCheckedItem(R.id.nav_home)
}

/**
 * Menangani pemilihan item navigasi.
 * @param item Menu item yang dipilih.
 * @return true untuk menampilkan item yang dipilih sebagai item saat
    ini.
 */
override fun onNavigationItemSelected(item: MenuItem): Boolean {
    when (item.itemId) {
        R.id.nav_home -> {
            // Memuat HomeFragment
            supportFragmentManager.beginTransaction()
                .replace(R.id.fragment_container, HomeFragment()).commit()
        }
        R.id.nav_setting -> {

```

```

        // Memuat PerizinanFragment
        supportFragmentManager.beginTransaction()
            .replace(R.id.fragment_container,
PerizinanFragment()).commit()
    }
    R.id.nav_riwayat -> {
        // Memuat RiwayatAbsensiFragment
        supportFragmentManager.beginTransaction()
            .replace(R.id.fragment_container,
RiwayatAbsensiFragment()).commit()
    }
    R.id.nav_logout -> {
        // Menangani logout pengguna
        logoutUser()
        // Memulai LoginActivity dan membersihkan stack aktivitas
        val intent = Intent(this, LoginActivity::class.java)
        intent.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP or
Intent.FLAG_ACTIVITY_NEW_TASK)
        startActivity(intent)
        finish()
    }
}

// Menutup navigation drawer
drawerLayout.closeDrawer(GravityCompat.START)
return true
}

/**
 * Melakukan logout pengguna dengan membersihkan sesi di PrefHelper.
 */
private fun logoutUser() {

```

```
        prefHelper.clearSession()  
    }  
}
```



HomeKepsekActivity.kt

```
package com.example.absensiapk.activity
```

```
import android.content.Intent
import android.os.Bundle
import android.view.MenuItem
import androidx.appcompat.app.ActionBarDrawerToggle
import androidx.appcompat.app.AppCompatActivity
import androidx.appcompat.widget.Toolbar
import androidx.core.view.GravityCompat
import androidx.drawerlayout.widget.DrawerLayout
import com.example.absensiapk.fragment.user.HomeFragment
import com.example.absensiapk.fragment.user.PerizinanFragment
import com.example.absensiapk.R
import com.example.absensiapk.fragment.RiwayatAbsensiFragment
import com.example.absensiapk.helper.PrefHelper
import com.google.android.material.navigation.NavigationView
```

```
/**
```

```
 * HomeActivity adalah aktivitas utama yang menampilkan navigation
drawer dan menangani transaksi fragment
```

```
 * berdasarkan pilihan navigasi pengguna.
```

```
*/
```

```
class HomeActivity : AppCompatActivity(),
NavigationView.OnNavigationItemSelectedListener {
```

```
    // Properti yang diinisialisasi kemudian untuk DrawerLayout dan
    PrefHelper
```

```
    private lateinit var drawerLayout: DrawerLayout
```

```
    private lateinit var prefHelper: PrefHelper
```

```

/**
 * Dipanggil saat aktivitas pertama kali dibuat.
 * @param savedInstanceState Jika aktivitas sedang diinisialisasi ulang
setelah sebelumnya ditutup, maka Bundle ini berisi data yang baru saja disediakan
di onSaveInstanceState(Bundle).
 */
override fun onCreate(savedInstanceState: Bundle?) {
    super.onCreate(savedInstanceState)
    setContentView(R.layout.activity_home)

    // Inisialisasi PrefHelper
    prefHelper = PrefHelper(this)

    // Inisialisasi DrawerLayout
    drawerLayout = findViewById(R.id.drawer_layout)

    // Menyiapkan toolbar sebagai app bar untuk aktivitas
    val toolbar = findViewById<Toolbar>(R.id.toolbar)
    setSupportActionBar(toolbar)

    // Menyiapkan NavigationView dan listener item terpilihnya
    val navigationView =
        findViewById<NavigationView>(R.id.nav_view)
    navigationView.setNavigationItemSelectedListener(this)

    // Menyiapkan ActionBarDrawerToggle untuk mengelola navigation
    drawer
    val toggle = ActionBarDrawerToggle(
        this,
        drawerLayout,

```

```

        toolbar,
        R.string.open_nav,
        R.string.close_nav
    )
    drawerLayout.addDrawerListener(toggle)
    toggle.syncState()

    // Memuat fragment default (HomeFragment) saat aktivitas pertama
    kali dibuat
    if (savedInstanceState == null) {
        supportFragmentManager.beginTransaction()
            .replace(R.id.fragment_container, HomeFragment())
            .commitNow()
    }
    navigationView.setCheckedItem(R.id.nav_home)
}

/**
 * Menangani pemilihan item navigasi.
 * @param item Menu item yang dipilih.
 * @return true untuk menampilkan item yang dipilih sebagai item saat
    ini.
 */
override fun onNavigationItemSelected(item: MenuItem): Boolean {
    when (item.itemId) {
        R.id.nav_home -> {
            // Memuat HomeFragment
            supportFragmentManager.beginTransaction()
                .replace(R.id.fragment_container, HomeFragment()).commit()
        }
        R.id.nav_setting -> {

```

```

        // Memuat PerizinanFragment
        supportFragmentManager.beginTransaction()
            .replace(R.id.fragment_container,
PerizinanFragment()).commit()
    }
    R.id.nav_riwayat -> {
        // Memuat RiwayatAbsensiFragment
        supportFragmentManager.beginTransaction()
            .replace(R.id.fragment_container,
RiwayatAbsensiFragment()).commit()
    }
    R.id.nav_logout -> {
        // Menangani logout pengguna
        logoutUser()
        // Memulai LoginActivity dan membersihkan stack aktivitas
        val intent = Intent(this, LoginActivity::class.java)
        intent.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP or
Intent.FLAG_ACTIVITY_NEW_TASK)
        startActivity(intent)
        finish()
    }
}

// Menutup navigation drawer
drawerLayout.closeDrawer(GravityCompat.START)
return true
}

/**
 * Melakukan logout pengguna dengan membersihkan sesi di PrefHelper.
 */
private fun logoutUser() {

```

```
        prefHelper.clearSession()  
    }  
}
```



LoginActivity.kt

```
package com.example.absensiapk.activity

import android.app.Dialog
import android.content.Intent
import android.graphics.Color
import android.graphics.drawable.ColorDrawable
import android.os.Bundle
import android.text.InputType
import android.util.Log
import android.view.Gravity
import android.view.View
import android.view.ViewGroup
import android.view.Window
import android.widget.Button
import android.widget.EditText
import android.widget.ImageView
import android.widget.ProgressBar
import android.widget.Toast
import androidx.appcompat.app.AppCompatActivity
import com.example.absensiapk.R
import com.example.absensiapk.helper.PrefHelper
import com.example.absensiapk.network.ResponseLogin
import com.example.absensiapk.network.RetrofitClient
import retrofit2.Call
import retrofit2.Callback
import retrofit2.Response
```

```
/**
```

* LoginActivity adalah aktivitas untuk menangani proses login pengguna.

```

*/
class LoginActivity : AppCompatActivity() {

    private lateinit var prefHelper: PrefHelper
    private lateinit var loading: ProgressBar
    private var user: String = ""
    private var pass: String = ""

    // Waktu tidak aktif yang diizinkan dalam milidetik (5 menit)
    private val ALLOWED_INACTIVE_TIME = 5 * 60 * 1000L

    /**
     * Dipanggil saat aktivitas pertama kali dibuat.
     * @param savedInstanceState Jika aktivitas sedang diinisialisasi ulang
     setelah sebelumnya ditutup, maka Bundle ini berisi data yang baru saja disediakan
     di onSaveInstanceState(Bundle).
     */
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_login)

        // Inisialisasi PrefHelper dan loading ProgressBar
        prefHelper = PrefHelper(this)
        loading = findViewById(R.id.loading)

        // Cek apakah pengguna sudah login
        if (prefHelper.getIsLogin()) {
            if
(prefHelper.isInactiveTimeEeded(ALLOWED_INACTIVE_TIME)) {
                performLogout()
            } else {

```

```

        navigateToHome()
        return // Keluar dari onCreate untuk mencegah tampilan dialog
login
    }
}

// Set onClickListener untuk tombol login
findViewById<Button>(R.id.LoginSheet).setOnClickListener {
    showDialog()
}

/**
 * Dipanggil saat aktivitas dipause.
 */
override fun onPause() {
    super.onPause()
    prefHelper.saveLastActiveTime()
}

/**
 * Dipanggil saat aktivitas dilanjutkan.
 */
override fun onResume() {
    super.onResume()
    if (prefHelper.getIsLogin() == false &&
prefHelper.isInactiveTimeEeeded(ALLOWED_INACTIVE_TIME)) {
        performLogout()
    }
}
}

```

```

/**
 * Menampilkan dialog login.
 */
private fun showDialog() {
    val dialog = Dialog(this)
    dialog.requestWindowFeature(Window.FEATURE_NO_TITLE)
    dialog setContentView(R.layout.loginsheetlayout)

    val usernameEditText: EditText =
dialog.findViewById(R.id.username)
    val passwordEditText: EditText =
dialog.findViewById(R.id.password)
    val showPasswordImageView: ImageView =
dialog.findViewById(R.id.showPassword)
    val loginButton: Button = dialog.findViewById(R.id.loginButton)

    // Set onClickListener untuk menampilkan atau menyembunyikan
password
    showPasswordImageView.setOnClickListener {
        if (passwordEditText.inputType ==
(InputType.TYPE_CLASS_TEXT or
InputType.TYPE_TEXT_VARIATION_PASSWORD)) {
            // Tampilkan password
            passwordEditText.inputType =
InputType.TYPE_TEXT_VARIATION_VISIBLE_PASSWORD

            showPasswordImageView.setImageResource(R.drawable.ic_visibility) // Ubah
ikon menjadi terlihat
        } else {
            // Sembunyikan password

```

```

        passwordEditText.inputType
        InputType.TYPE_CLASS_TEXT
        InputType.TYPE_TEXT_VARIATION_PASSWORD

showPasswordImageView.setImageResource(R.drawable.ic_visibility_off)
Ubah ikon menjadi tersembunyi
    }
    // Pindahkan kursor ke akhir teks
    passwordEditText.setSelection(passwordEditText.text.length)
}

// Set onClickListener untuk tombol login
loginButton.setOnClickListener {
    user = usernameEditText.text.toString()
    pass = passwordEditText.text.toString()
    if (validateInput(user, pass)) {
        dialog.dismiss()
        loading.visibility = View.VISIBLE
        performLogin()
    }
}

dialog.show()
dialog.window?.apply {
    setLayout(ViewGroup.LayoutParams.MATCH_PARENT,
    ViewGroup.LayoutParams.WRAP_CONTENT)

setBackgroundDrawable(ColorDrawable(Color.TRANSPARENT))
    attributes?.windowAnimations = R.style.DialogAnimation
    setGravity(Gravity.BOTTOM)
}

```

```

    }

    /**
     * Memvalidasi input username dan password.
     * @param username Username yang diinputkan.
     * @param password Password yang diinputkan.
     * @return true jika input valid, false jika tidak.
     */
    private fun validateInput(username: String, password: String): Boolean {
        return when {
            username.isEmpty() -> {
                Toast.makeText(this, "Username tidak boleh kosong",
                    Toast.LENGTH_SHORT).show()
                false
            }
            password.isEmpty() -> {
                Toast.makeText(this, "Password tidak boleh kosong",
                    Toast.LENGTH_SHORT).show()
                false
            }
            else -> true
        }
    }

    /**
     * Melakukan proses login dengan memanggil API.
     */
    private fun performLogin() {
        val api = RetrofitClient.instance

        api.login(user, pass).enqueue(object : Callback<ResponseLogin> {

```

```

        override fun onResponse(call: Call<ResponseLogin>, response:
Response<ResponseLogin>) {
            loading.visibility = View.GONE
            if (response.isSuccessful && response.body()?.response == true)
            {
                val userLevel = response.body()?.payload?.level_user ?: "0"
                val userId = response.body()?.payload?.id_user ?: "0"
                val usr = response.body()?.payload?.nama ?: "0"
                val nip = response.body()?.payload?.nip ?: "0"
                val jabatan = response.body()?.payload?.jabatan ?: "0"
                val foto = response.body()?.payload?.foto ?: "0"
                saveSession(user, pass, usr, true, userLevel, userId, nip,
jabatan, foto)
                navigateToHome()
            } else {
                Toast.makeText(
                    this@LoginActivity,
                    response.body()?.message ?: "Login gagal, silakan periksa
username dan password Anda",
                    Toast.LENGTH_LONG
                ).show()
            }
        }

        override fun onFailure(call: Call<ResponseLogin>, t: Throwable) {
            loading.visibility = View.GONE
            Log.e("Error", "${t.message}")
            Toast.makeText(this@LoginActivity, "Error:  ${t.message}",
Toast.LENGTH_SHORT).show()
        }
    })

```

```

    }

    /**
     * Menyimpan sesi login pengguna.
     * @param username Username pengguna.
     * @param password Password pengguna.
     * @param user Nama pengguna.
     * @param isLogin Status login pengguna.
     * @param userLevel Level pengguna.
     * @param userId ID pengguna.
     * @param nip NIP pengguna.
     * @param jabatan Jabatan pengguna.
     * @param foto Foto pengguna.
     */
    private fun saveSession(username: String, password: String, user: String,
        isLogin: Boolean, userLevel: String, userId: String, nip: String, jabatan: String,
        foto: String) {
        prefHelper.saveSession(username, password, user, isLogin, userLevel,
            userId, nip, jabatan, foto)
    }

    /**
     * Melakukan logout pengguna dan membersihkan sesi.
     */
    private fun performLogout() {
        prefHelper.clearSession()
        val intent = Intent(this, LoginActivity::class.java)
        startActivity(intent)
        finish()
    }
}

```

```
/**
 * Menavigasi pengguna ke halaman utama sesuai dengan level
 pengguna.
 */
private fun navigateToHome() {
    val userLevel = prefHelper.getUserLevel()
    val intent = if (userLevel == "admin") {
        Intent(this, HomeKepsekActivity::class.java)
    } else {
        Intent(this, HomeActivity::class.java)
    }
    startActivity(intent)
    finish()
}
```

HomeKepsekFragment.kt

```
package com.example.absensiapk.fragment.kepsek

import android.os.Bundle
import android.os.Handler
import android.os.Looper
import android.view.Gravity
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import android.widget.TableRow
import android.widget.TextView
import androidx.fragment.app.Fragment
import androidx.fragment.app.commit
import com.android.volley.Request
import com.android.volley.RequestQueue
import com.android.volley.toolbox.JsonArrayRequest
import com.android.volley.toolbox.Volley
import com.bumptech.glide.Glide
import com.example.absensiapk.R
import com.google.gson.Gson
import com.google.gson.reflect.TypeToken
import com.example.absensiapk.databinding.FragmentHomeBinding
import com.example.absensiapk.fragment.RiwayatAbsensiFragment
import com.example.absensiapk.fragment.user.PerizinanFragment
import com.example.absensiapk.helper.PrefHelper
import com.example.absensiapk.network.Absen
import com.example.absensiapk.network.Izin
import java.time.LocalDateTime
import java.time.format.DateTimeFormatter
```

```

/**
 * HomeKepsekFragment adalah fragment yang menampilkan halaman
utama untuk kepala sekolah.
 * Fragment ini menampilkan informasi pengguna dan data absensi serta
izin terbaru.
 */
class HomeKepsekFragment : Fragment() {

    private var _binding: FragmentHomeBinding? = null
    private val binding get() = _binding!!
    private val handler = Handler(Looper.getMainLooper())
    private val dateTimeFormatter = DateTimeFormatter.ofPattern("EEEE,
dd-MM-yyyy HH:mm:ss")
    private lateinit var requestQueue: RequestQueue

    // Tugas untuk memperbarui waktu dan tanggal setiap detik
    private val updateDateTimeTask = object : Runnable {
        override fun run() {
            val currentDateTime =
LocalDateTime.now().format(dateTimeFormatter)
            binding.dateTimeTextView.text = "$currentDateTime"
            handler.postDelayed(this, 1000)
        }
    }

}

/**
 * Dipanggil untuk menginisialisasi tampilan fragment.
 * @param inflater Objek LayoutInflater yang dapat digunakan untuk
mengembangkan tampilan fragment.

```

* @param container Jika tidak null, ini adalah parent view yang akan digunakan untuk mengembangkan tampilan fragment.

* @param savedInstanceState Jika tidak null, fragment ini sedang dibuat ulang dari status sebelumnya yang disimpan di Bundle ini.

* @return Mengembalikan View untuk UI fragment, atau null.

*/

override fun onCreateView(

inflater: LayoutInflater, container: ViewGroup?,

savedInstanceState: Bundle?

): View? {

_binding = FragmentHomeBinding.inflate(inflater, container, false)

return binding.root

}

/**

* Dipanggil segera setelah onCreateView().

* @param view Tampilan yang dikembalikan oleh onCreateView().

* @param savedInstanceState Jika tidak null, fragment ini sedang dibuat ulang dari status sebelumnya yang disimpan di Bundle ini.

*/

override fun onViewCreated(view: View, savedInstanceState: Bundle?)

{

super.onViewCreated(view, savedInstanceState)

requestQueue = Volley.newRequestQueue(requireContext())

// Mulai memperbarui waktu dan tanggal

handler.post(updateDateTimeTask)

// Mengikat data pengguna

bindUserData()

```

        // Mengambil dan menampilkan data terbaru
        fetchIzinData()
        fetchAbsensiData()
    }

    /**
     * Mengikat data pengguna dari PrefHelper.
     */
    private fun bindUserData() {
        val prefHelper = PrefHelper(requireContext())
        val username = prefHelper.getUsername() ?: "N/A"
        val nip = prefHelper.getNip() ?: "N/A"
        val jabatan = prefHelper.getJabatan() ?: "N/A"
        val fotoFilename = prefHelper.getFoto() ?: ""

        binding.nama.text = username
        binding.nip.text = nip
        binding.jabatan.text = jabatan

        val baseUrl = "https://tkdharma.site/img-uploads/img-profile/"
        val fullImageUrl = "$baseUrl$fotoFilename"

        if (fotoFilename.isNotEmpty()) {
            Glide.with(this)
                .load(fullImageUrl)
                .placeholder(R.drawable.placeholder_image)
                .error(R.drawable.error_image)
                .into(binding.imageView)
        } else {

binding.imageView.setImageResource(R.drawable.placeholder_image)

```

```

    }
}

/**
 * Mengambil data absensi dari server dan menampilkannya.
 */
private fun fetchAbsensiData() {
    val prefHelper = PrefHelper(requireContext())
    val nama = prefHelper.getUser() ?: return

    val url = "https://tkdharma.site/get_tb_absen.php?nama=$nama"

    val jsonArrayRequest = JsonRequest(
        Request.Method.GET, url, null,
        { response ->
            val dataType = object : TypeToken<List<Absen>>() {}.type
            val absenList: List<Absen> =
                Gson().fromJson(response.toString(), dataType)
            if (absenList.isNotEmpty()) {
                displayAbsensiData(absenList.first())
            } else {
                // Tangani kasus tidak ada data
                binding.errorTextView.visibility = View.VISIBLE
                binding.errorTextView.text = "Tidak Ada Data Dari: $nama"
            }
        },
        { error ->
            error.printStackTrace()
            // Tangani error
            binding.errorTextView.visibility = View.VISIBLE
            binding.errorTextView.text = "Anda Belum Melakukan Absensi"
        }
    )
}

```

```

    }
)

requestQueue.add(jsonArrayRequest)
}

/**
 * Menampilkan data absensi di dalam tabel.
 * @param absen Objek Absen yang berisi data absensi terbaru.
 */
private fun displayAbsensiData(absen: Absen) {
    binding.latestAbsensiTable.removeAllViews()
    val headerRow = TableRow(context)
    val headers = listOf("Waktu", "Jam Masuk", "Jam Pulang",
        "Keterangan")

    headers.forEach { header ->
        val textView = TextView(context).apply {
            text = header
            setPadding(8, 8, 8, 8)
            set Typeface(null, android.graphics.Typeface.BOLD)
            gravity = Gravity.CENTER
        }
        headerRow.addView(textView)
    }
    binding.latestAbsensiTable.addView(headerRow)

    val tableRow = TableRow(context)

    val textViewWaktu = TextView(context).apply {
        text = absen.waktu

```

```
        setPadding(8, 8, 8, 8)
        gravity = Gravity.CENTER
    }
    val textViewJamMasuk = TextView(context).apply {
        text = absen.jam_masuk
        setPadding(8, 8, 8, 8)
        gravity = Gravity.CENTER
    }
    val textViewJamPulang = TextView(context).apply {
        text = absen.jam_pulang
        setPadding(8, 8, 8, 8)
        gravity = Gravity.CENTER
    }
    val textViewKeterangan = TextView(context).apply {
        text = absen.keterangan
        setPadding(8, 8, 8, 8)
        gravity = Gravity.CENTER
    }

    tableRow.addView(textViewWaktu)
    tableRow.addView(textViewJamMasuk)
    tableRow.addView(textViewJamPulang)
    tableRow.addView(textViewKeterangan)

    tableRow.setOnClickListener {
        parentFragmentManager.commit {
            replace(R.id.fragment_container, RiwayatAbsensiFragment())
            addToBackStack(null)
        }
    }
}
```

```

        binding.latestAbsensiTable.addView(tableRow)
    }

    /**
     * Mengambil data izin dari server dan menampilkannya.
     */
    private fun fetchIzinData() {
        val prefHelper = PrefHelper(requireContext())
        val nama = prefHelper.getUser() ?: return

        val url = "https://tkdharma.site/get_tb_izin.php?nama=$nama"

        val jsonArrayRequest = JsonRequest(
            Request.Method.GET, url, null,
            { response ->
                val dataType = object : TypeToken<List<Izin>>() {}.type
                val izinList: List<Izin> = Gson().fromJson(response.toString(),
dataType)

                if (izinList.isNotEmpty()) {
                    displayIzinData(izinList.first())
                } else {
                    // Tangani kasus tidak ada data
                    binding.errorTextView2.visibility = View.VISIBLE
                    binding.errorTextView2.text = "Tidak Ada Data Dari: $nama"
                }
            },
            { error ->
                error.printStackTrace()
                // Tangani error
                binding.errorTextView2.visibility = View.VISIBLE
                binding.errorTextView2.text = "Anda Belum Melakukan Izin"
            }
        )
    }

```

```

    }
)

requestQueue.add(jsonArrayRequest)
}

/**
 * Menampilkan data izin di dalam tabel.
 * @param izin Objek Izin yang berisi data izin terbaru.
 */
private fun displayIzinData(izin: Izin) {
    binding.latestIzinTable.removeAllViews()
    val headerRow = TableRow(context)
    val headers = listOf("Tanggal", "Nama", "Alasan", "Status")

    headers.forEach { header ->
        val textView = TextView(context).apply {
            text = header
            setPadding(8, 8, 8, 8)
            setTypeface(null, android.graphics.Typeface.BOLD)
            gravity = Gravity.CENTER
        }
        headerRow.addView(textView)
    }
    binding.latestIzinTable.addView(headerRow)

    val tableRow = TableRow(context)

    val textViewTanggal = TextView(context).apply {
        text = izin.tanggal
        setPadding(8, 8, 8, 8)
    }

```

```
        gravity = Gravity.CENTER
    }
    val textViewNama = TextView(context).apply {
        text = izin.nama
        setPadding(8, 8, 8, 8)
        gravity = Gravity.CENTER
    }
    val textViewAlasan = TextView(context).apply {
        text = izin.alasan
        setPadding(8, 8, 8, 8)
        gravity = Gravity.CENTER
    }
    val textViewStatus = TextView(context).apply {
        text = izin.status
        setPadding(8, 8, 8, 8)
        gravity = Gravity.CENTER
    }

    tableRow.addView(textViewTanggal)
    tableRow.addView(textViewNama)
    tableRow.addView(textViewAlasan)
    tableRow.addView(textViewStatus)

    tableRow.setOnClickListener {
        parentFragmentManager.commit {
            replace(R.id.fragment_container, PerizinanFragment())
            addToBackStack(null)
        }
    }
}

binding.latestIzinTable.addView(tableRow)
```

```
}

/**
 * Dipanggil saat tampilan fragment dihancurkan.
 */
override fun onDestroyView() {
    super.onDestroyView()
    handler.removeCallbacks(updateDateTimeTask)
    _binding = null
}
}
```

PersetujuanFragment.kt

```
package com.example.absensiapk.fragment.kepsek

import android.app.AlertDialog
import android.os.Bundle
import android.util.Log
import androidx.fragment.app.Fragment
import android.view.Gravity
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import android.widget.Button
import android.widget.ImageView
import android.widget.LinearLayout
import android.widget.TableRow
import android.widget.TextView
import androidx.swiperefreshlayout.widget.SwipeRefreshLayout
import com.android.volley.Request
import com.android.volley.RequestQueue
import com.android.volley.toolbox.JsonArrayRequest
import com.android.volley.toolbox.StringRequest
import com.android.volley.toolbox.Volley
import com.bumptech.glide.Glide
import com.example.absensiapk.R
import com.example.absensiapk.databinding.FragmentPersetujuanBinding
import com.example.absensiapk.fragment.FullscreenImageFragment
import com.example.absensiapk.network.Izin
import com.google.gson.Gson
import com.google.gson.reflect.TypeToken
```

```

/**
 * PersetujuanFragment adalah fragment yang menampilkan daftar izin
 yang menunggu persetujuan kepala sekolah.
 */
class PersetujuanFragment : Fragment() {

    private lateinit var requestQueue: RequestQueue
    private var _binding: FragmentPersetujuanBinding? = null
    private val binding get() = _binding!!
    private lateinit var swipeRefreshLayout: SwipeRefreshLayout

    /**
     * Dipanggil untuk menginisialisasi tampilan fragment.
     * @param inflater Objek LayoutInflater yang dapat digunakan untuk
 mengembangkan tampilan fragment.
     * @param container Jika tidak null, ini adalah parent view yang akan
 digunakan untuk mengembangkan tampilan fragment.
     * @param savedInstanceState Jika tidak null, fragment ini sedang dibuat
 ulang dari status sebelumnya yang disimpan di Bundle ini.
     * @return Mengembalikan View untuk UI fragment, atau null.
     */
    override fun onCreateView(
        inflater: LayoutInflater, container: ViewGroup?,
        savedInstanceState: Bundle?
    ): View? {
        _binding = FragmentPersetujuanBinding.inflate(inflater, container,
false)

        return binding.root
    }

    /**

```

- * Dipanggil segera setelah onCreateView().
- * @param view Tampilan yang dikembalikan oleh onCreateView().
- * @param savedInstanceState Jika tidak null, fragment ini sedang dibuat ulang dari status sebelumnya yang disimpan di Bundle ini.

```

*/
override fun onCreateView(view: View, savedInstanceState: Bundle?)
{
    super.onCreateView(view, savedInstanceState)

    requestQueue = Volley.newRequestQueue(requireContext())
    swipeRefreshLayout = binding.root.findViewById(R.id.swipeRefreshLayout)

    // Set onRefreshListener untuk SwipeRefreshLayout
    swipeRefreshLayout.setOnRefreshListener {
        fetchIzinData()
    }

    fetchIzinData()
}

/**
 * Mengambil data izin dari server.
 */
private fun fetchIzinData() {
    val status = "Menunggu Persetujuan"
    val url = "https://tkdharma.site/get_tb_izin_persetujuan.php?status=$status"

    val jsonArrayRequest = JsonRequest(
        Request.Method.GET, url, null,

```

```

        { response ->
            val dataType = object : TypeToken<List<Izin>>() {}.type
            val izinList: List<Izin> = Gson().fromJson(response.toString(),
dataType)

            if (izinList.isNotEmpty()) {
                displayIzinData(izinList)
            } else {
                binding.errorTextView.visibility = View.VISIBLE
                binding.errorTextView.text = "Tidak ada data ditemukan untuk
pengguna ini."
            }
            swipeRefreshLayout.isRefreshing = false
        },
        { error ->
            error.printStackTrace()
            binding.errorTextView.visibility = View.VISIBLE
            binding.errorTextView.text = "Error mengambil data:
${error.networkResponse?.statusCode ?: "null"} - ${error.message ?: "Tidak ada
pesan"}"
            swipeRefreshLayout.isRefreshing = false
        }
    )

    requestQueue.add(jsonArrayRequest)
}

/**
 * Menampilkan data izin di dalam tabel.
 * @param izinList Daftar objek Izin yang berisi data izin.
 */
private fun displayIzinData(izinList: List<Izin>) {

```

```
binding.tableLayout.removeAllViews()
val headerRow = TableRow(context).apply {
    gravity = Gravity.CENTER
}
val headers = listOf("ID", "Nama", "Tanggal", "Alasan", "Status",
"Foto Bukti", "Aksi")
```

```
headers.forEach { header ->
    val textView = TextView(context).apply {
        text = header
        setPadding(8, 8, 8, 8)
        setTypeface(null, android.graphics.Typeface.BOLD)
        gravity = Gravity.CENTER
    }
    headerRow.addView(textView)
}
binding.tableLayout.addView(headerRow)
```

```
for (izin in izinList) {
    val tableRow = TableRow(context).apply {
        gravity = Gravity.CENTER
    }

    val textViewID = TextView(context).apply {
        text = izin.id_izin.toString()
        setPadding(8, 8, 8, 8)
        gravity = Gravity.CENTER
    }

    val textViewNama = TextView(context).apply {
        text = izin.nama
        setPadding(8, 8, 8, 8)
```

```

        gravity = Gravity.CENTER
    }
    val textViewTanggal = TextView(context).apply {
        text = izin.tanggal
        setPadding(8, 8, 8, 8)
        gravity = Gravity.CENTER
    }
    val textViewAlasan = TextView(context).apply {
        text = izin.alasan
        setPadding(8, 8, 8, 8)
        gravity = Gravity.CENTER
    }
    val textViewStatus = TextView(context).apply {
        text = izin.status
        setPadding(8, 8, 8, 8)
        gravity = Gravity.CENTER
    }
    val imageViewFotoBukti = ImageView(context).apply {
        setPadding(8, 8, 8, 8)
        val imageUrl = "https://tkdharma.site/img-uploads/img-
perizinan/" + izin.foto_bukti
        Glide.with(this)
            .load(imageUrl)
            .into(this)
        setOnClickListener {
            Log.d("PersetujuanFragment", "Image clicked:
${izin.foto_bukti}")
            val fragmentManager = parentFragmentManager
            val fullScreenFragment =
FullscreenImageFragment.newInstance(imageUrl)

```

```

        fullScreenFragment.show(fragmentManager,
"fullScreenImage")
    }
}

val buttonLayout = LinearLayout(context).apply {
    orientation = LinearLayout.VERTICAL
    setPadding(8, 8, 8, 8)
}

val buttonAgree = Button(context).apply {
    text = "Setujui"
    setOnClickListener { updateStatus(izin.id_izin, "disetujui") }
}

val buttonNo = Button(context).apply {
    text = "Tolak"
    setOnClickListener { updateStatus(izin.id_izin, "tidak disetujui")
}

}

buttonLayout.addView(buttonAgree)
buttonLayout.addView(buttonNo)

tableRow.addView(textViewID)
tableRow.addView(textViewNama)
tableRow.addView(textViewTanggal)
tableRow.addView(textViewAlasan)
tableRow.addView(textViewStatus)
tableRow.addView(imageViewFotoBukti)
tableRow.addView(buttonLayout)

binding.tableLayout.addView(tableRow)
}
}

```

```

/**
 * Memperbarui status izin di server.
 * @param id_izin ID izin yang akan diperbarui.
 * @param status Status baru izin.
 */
private fun updateStatus(id_izin: Int, status: String) {
    val url = "https://tkdharma.site/update_tb_izin_status.php"
    val request = object : StringRequest(Method.POST, url,
        { response ->
            showAlertAndRefresh(status)
        },
        { error ->
            error.printStackTrace()
            binding.errorTextView.visibility = View.VISIBLE
            binding.errorTextView.text = "Error memperbarui status:
            ${error.networkResponse?.statusCode ?: "null"} - ${error.message ?: "Tidak ada
            pesan"}"
        }) {
        override fun getParams(): MutableMap<String, String> {
            return hashMapOf("id_izin" to id_izin.toString(), "status" to
            status)
        }
    }

    requestQueue.add(request)
}

/**
 * Menampilkan dialog alert dan menyegarkan data izin.
 * @param status Status baru izin.

```

```

*/
private fun showAlertAndRefresh(status: String) {
    val alertDialog = AlertDialog.Builder(requireContext()).apply {
        setTitle("Status Update")
        setMessage("Status berhasil diperbarui menjadi $status.")
        setPositiveButton("OK") { dialog, _ ->
            dialog.dismiss()
            swipeRefreshLayout.isRefreshing = true
            fetchIzinData()
        }
    }.create()

    alertDialog.show()
}

/**
 * Dipanggil saat tampilan fragment dihancurkan.
 */
override fun onDestroyView() {
    super.onDestroyView()
    _binding = null
}
}

```

RecapFragment.kt

```

package com.example.absensiapk.fragment.kepsek

import android.os.Bundle
import android.view.LayoutInflater
import android.view.View

```

```

import android.view.ViewGroup
import androidx.fragment.app.Fragment
import androidx.fragment.app.viewModels
import androidx.recyclerview.widget.LinearLayoutManager
import com.example.absensiapk.databinding.FragmentRecapBinding
import com.example.absensiapk.helper.RecapAdapter
import com.example.absensiapk.helper.RecapViewModel

class RecapFragment : Fragment() {

    private var _binding: FragmentRecapBinding? = null
    private val binding get() = _binding!!
    private val viewModel: RecapViewModel by viewModels()

    override fun onCreateView(
        inflater: LayoutInflater, container: ViewGroup?,
        savedInstanceState: Bundle?
    ): View {
        _binding = FragmentRecapBinding.inflate(inflater, container, false)
        return binding.root
    }

    override fun onViewCreated(view: View, savedInstanceState: Bundle?)
    {
        super.onViewCreated(view, savedInstanceState)

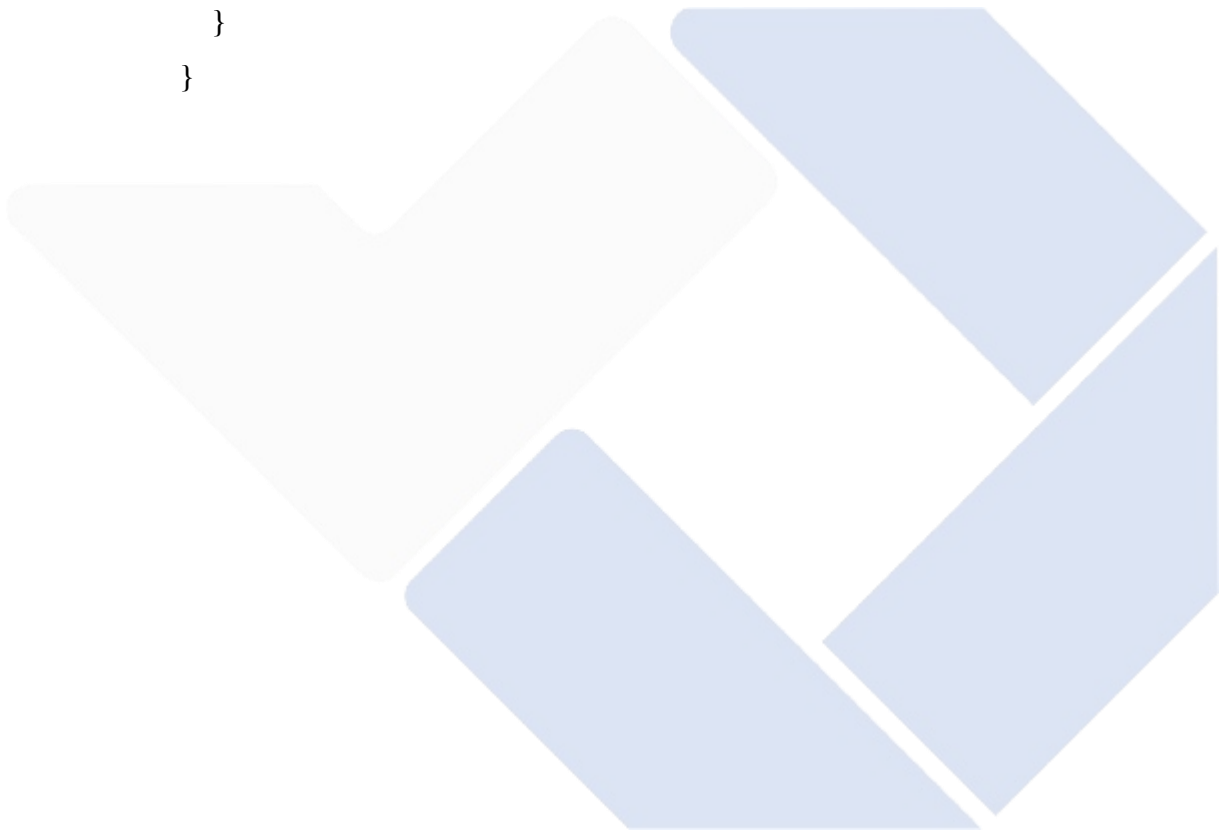
        // Set up the LayoutManager
        binding.recapList.layoutManager =
LinearLayoutManager(requireContext())

        // Observe ViewModel and bind data to RecyclerView

```

```
viewModel.monthlyRecap.observe(viewLifecycleOwner) { recaps ->
    binding.recapList.adapter = RecapAdapter(recaps)
}

override fun onDestroyView() {
    super.onDestroyView()
    _binding = null
}
}
```



RiwayatFragment.kt

```
package com.example.absensiapk.fragment.kepsek
```

```
import android.os.Bundle
import android.util.Log
import android.view.Gravity
import androidx.fragment.app.Fragment
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import android.widget.ImageView
import android.widget.TableRow
import android.widget.TextView
import androidx.swiperefreshlayout.widget.SwipeRefreshLayout
import com.android.volley.Request
import com.android.volley.RequestQueue
import com.android.volley.toolbox.JsonArrayRequest
import com.android.volley.toolbox.Volley
import com.bumptech.glide.Glide
import com.example.absensiapk.R
import com.example.absensiapk.databinding.FragmentRiwayatBinding
import com.example.absensiapk.fragment.FullscreenImageFragment
import com.example.absensiapk.network.Izin
import com.google.gson.Gson
import com.google.gson.reflect.TypeToken
```

```
/**
```

* RiwayatFragment adalah fragment yang menampilkan riwayat izin yang telah diajukan oleh pengguna.

```
*/
```

```

class RiwayatFragment : Fragment() {

    private lateinit var requestQueue: RequestQueue
    private var _binding: FragmentRiwayatBinding? = null
    private val binding get() = _binding!!
    private lateinit var swipeRefreshLayout: SwipeRefreshLayout

    /**
     * Dipanggil untuk menginisialisasi tampilan fragment.
     * @param inflater Objek LayoutInflater yang dapat digunakan untuk
    mengembangkan tampilan fragment.
     * @param container Jika tidak null, ini adalah parent view yang akan
    digunakan untuk mengembangkan tampilan fragment.
     * @param savedInstanceState Jika tidak null, fragment ini sedang dibuat
    ulang dari status sebelumnya yang disimpan di Bundle ini.
     * @return Mengembalikan View untuk UI fragment, atau null.
    */
    override fun onCreateView(
        inflater: LayoutInflater, container: ViewGroup?,
        savedInstanceState: Bundle?
    ): View? {
        _binding = FragmentRiwayatBinding.inflate(inflater, container, false)
        return binding.root
    }

    /**
     * Dipanggil segera setelah onCreateView().
     * @param view Tampilan yang dikembalikan oleh onCreateView().
     * @param savedInstanceState Jika tidak null, fragment ini sedang dibuat
    ulang dari status sebelumnya yang disimpan di Bundle ini.
    */

```

```

override fun onCreateView(view: View, savedInstanceState: Bundle?)
{
    super.onCreateView(view, savedInstanceState)

    requestQueue = Volley.newRequestQueue(requireContext())
    swipeRefreshLayout =
binding.root.findViewById(R.id.swipeRefreshLayout)

    // Set onRefreshListener untuk SwipeRefreshLayout
    swipeRefreshLayout.setOnRefreshListener {
        fetchIzinData()
    }

    // Mengambil data izin saat fragment pertama kali dibuka
    fetchIzinData()
}

/**
 * Mengambil data izin dari server.
 */
private fun fetchIzinData() {
    val url = "https://tkdharma.site/get_tb_izin_kepsek.php"

    val jsonArrayRequest = JsonRequest(
        Request.Method.GET, url, null,
        { response ->
            val dataType = object : TypeToken<List<Izin>>() {}.type
            val izinList: List<Izin> = Gson().fromJson(response.toString(),
dataType)

            if (izinList.isNotEmpty()) {
                displayIzinData(izinList)
            }
        })
}

```

```

    } else {
        // Tangani kasus tidak ada data
        binding.errorTextView.visibility = View.VISIBLE
        binding.errorTextView.text = "Tidak ada data ditemukan untuk
pengguna ini."
    }
    swipeRefreshLayout.isRefreshing = false
},
{ error ->
    // Log rincian error
    error.printStackTrace()
    // Tangani error
    binding.errorTextView.visibility = View.VISIBLE
    binding.errorTextView.text = "Error mengambil data:
${error.networkResponse?.statusCode ?: "null"} - ${error.message ?: "Tidak ada
pesan"}"
    swipeRefreshLayout.isRefreshing = false
}
)

requestQueue.add(jsonArrayRequest)
}

/**
 * Menampilkan data izin di dalam tabel.
 * @param izinList Daftar objek Izin yang berisi data izin.
 */
private fun displayIzinData(izinList: List<Izin>) {
    binding.tableLayout.removeAllViews() // Menghapus tampilan yang
ada
    val headerRow = TableRow(context)

```

```
val headers = listOf("Tanggal", "Alasan", "Status", "Foto Bukti")
```

```
headers.forEach { header ->
```

```
    val textView = TextView(context).apply {
```

```
        text = header
```

```
        setPadding(8, 8, 8, 8)
```

```
        setTypeface(null, android.graphics.Typeface.BOLD)
```

```
    }
```

```
    headerRow.addView(textView)
```

```
}
```

```
binding.tableLayout.addView(headerRow)
```

```
for (izin in izinList) {
```

```
    val tableRow = TableRow(context)
```

```
    val textViewTanggal = TextView(context).apply {
```

```
        text = izin.tanggal
```

```
        setPadding(8, 8, 8, 8)
```

```
        gravity = Gravity.CENTER
```

```
    }
```

```
    val textViewAlasan = TextView(context).apply {
```

```
        text = izin.alasan
```

```
        setPadding(8, 8, 8, 8)
```

```
        gravity = Gravity.CENTER
```

```
    }
```

```
    val textViewStatus = TextView(context).apply {
```

```
        text = izin.status
```

```
        setPadding(8, 8, 8, 8)
```

```
        gravity = Gravity.CENTER
```

```
    }
```

```
    val imageViewFotoBukti = ImageView(context).apply {
```

```

        setPadding(8, 8, 8, 8)
        val imageUrl = "https://tkdharma.site/img-uploads/img-
perizinan/" + izin.foto_bukti
        Glide.with(this)
            .load(imageUrl)
            .into(this)
        setOnClickListener {
            Log.d("PersetujuanFragment", "Image clicked:
${izin.foto_bukti}")
            val fragmentManager = parentFragmentManager
            val fullScreenFragment =
FullscreenImageFragment.newInstance(imageUrl)
            fullScreenFragment.show(fragmentManager,
"fullScreenImage")
        }
    }

    tableView.addView(textViewTanggal)
    tableView.addView(textViewAlasan)
    tableView.addView(textViewStatus)
    tableView.addView(imageViewFotoBukti)

    binding.tableLayout.addView(tableView)
}

/**
 * Dipanggil saat tampilan fragment dihancurkan.
 */
override fun onDestroyView() {

```

```
        super.onDestroyView()
        _binding = null
    }
}
```

HomeFragment.kt

```
package com.example.absensiapk.fragment.user
```

```
import android.os.Bundle
import android.os.Handler
import android.os.Looper
import android.view.Gravity
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import android.widget.TableRow
import android.widget.TextView
import androidx.fragment.app.Fragment
import androidx.fragment.app.commit
import com.android.volley.Request
import com.android.volley.RequestQueue
import com.android.volley.toolbox.JsonArrayRequest
import com.android.volley.toolbox.Volley
import com.bumptech.glide.Glide
import com.example.absensiapk.R
import com.google.gson.Gson
import com.google.gson.reflect.TypeToken
import com.example.absensiapk.databinding.FragmentHomeBinding
import com.example.absensiapk.fragment.RiwayatAbsensiFragment
import com.example.absensiapk.fragment.user.PerizinanFragment
```

```

import com.example.absensiapk.helper.PrefHelper
import com.example.absensiapk.network.Absen
import com.example.absensiapk.network.Izin
import java.time.LocalDateTime
import java.time.format.DateTimeFormatter

/**
 * HomeFragment adalah fragment yang menampilkan halaman utama
 untuk pengguna.
 * Fragment ini menampilkan informasi pengguna dan data absensi serta
 izin terbaru.
 */
class HomeFragment : Fragment() {

    private var _binding: FragmentHomeBinding? = null
    private val binding get() = _binding!!
    private val handler = Handler(Looper.getMainLooper())
    private val dateTimeFormatter = DateTimeFormatter.ofPattern("EEEE,
dd-MM-yyyy HH:mm:ss")
    private lateinit var requestQueue: RequestQueue

    // Tugas untuk memperbarui waktu dan tanggal setiap detik
    private val updateDateTimeTask = object : Runnable {
        override fun run() {
            val currentDateTime =
LocalDateTime.now().format(dateTimeFormatter)
            binding.dateTimeTextView.text = "$currentDateTime"
            handler.postDelayed(this, 1000)
        }
    }
}

```

```

/**
 * Dipanggil untuk menginisialisasi tampilan fragment.
 * @param inflater Objek LayoutInflater yang dapat digunakan untuk
mengembangkan tampilan fragment.
 * @param container Jika tidak null, ini adalah parent view yang akan
digunakan untuk mengembangkan tampilan fragment.
 * @param savedInstanceState Jika tidak null, fragment ini sedang dibuat
ulang dari status sebelumnya yang disimpan di Bundle ini.
 * @return Mengembalikan View untuk UI fragment, atau null.
 */
override fun onCreateView(
    inflater: LayoutInflater, container: ViewGroup?,
    savedInstanceState: Bundle?
): View? {
    _binding = FragmentHomeBinding.inflate(inflater, container, false)
    return binding.root
}

/**
 * Dipanggil segera setelah onCreateView().
 * @param view Tampilan yang dikembalikan oleh onCreateView().
 * @param savedInstanceState Jika tidak null, fragment ini sedang dibuat
ulang dari status sebelumnya yang disimpan di Bundle ini.
 */
override fun onViewCreated(view: View, savedInstanceState: Bundle?)
{
    super.onViewCreated(view, savedInstanceState)
    requestQueue = Volley.newRequestQueue(requireContext())

    // Mulai memperbarui waktu dan tanggal
    handler.post(updateDateTimeTask)

```

```

// Mengikat data pengguna
bindUserData()

// Mengambil dan menampilkan data terbaru
fetchAbsensiData()
fetchIzinData()
}

/**
 * Mengikat data pengguna dari PrefHelper.
 */
private fun bindUserData() {
    val prefHelper = PrefHelper(requireContext())
    val nama = prefHelper.getUser() ?: "N/A"
    val nip = prefHelper.getNip() ?: "N/A"
    val jabatan = prefHelper.getJabatan() ?: "N/A"
    val fotoFilename = prefHelper.getFoto() ?: ""

    binding.nama.text = nama
    binding.nip.text = nip
    binding.jabatan.text = jabatan

    val baseUrl = "https://tkdharma.site/img-uploads/img-profile/"
    val fullImageUrl = "$baseUrl$fotoFilename"

    if (fotoFilename.isNotEmpty()) {
        Glide.with(this)
            .load(fullImageUrl)
            .placeholder(R.drawable.placeholder_image)
            .error(R.drawable.error_image)
    }
}

```

```

        .into(binding.imageView)
    } else {

binding.imageView.setImageResource(R.drawable.placeholder_image)
    }
}

/**
 * Mengambil data absensi dari server dan menampilkannya.
 */
private fun fetchAbsensiData() {
    val prefHelper = PrefHelper(requireContext())
    val nama = prefHelper.getUser() ?: return

    val url = "https://tkdharma.site/get_tb_absen.php?nama=$nama"

    val jsonArrayRequest = JsonRequest(
        Request.Method.GET, url, null,
        { response ->
            val dataType = object : TypeToken<List<Absen>>() {}.type
            val absenList: List<Absen> =
Gson().fromJson(response.toString(), dataType)
            if (absenList.isNotEmpty()) {
                displayAbsensiData(absenList.first())
            } else {
                // Tangani kasus tidak ada data
                binding.errorTextView.visibility = View.VISIBLE
                binding.errorTextView.text = "Tidak ada data ditemukan untuk
pengguna: $nama"
            }
        },

```

```

        { error ->
            error.printStackTrace()
            // Tangani error
            binding.errorTextView.visibility = View.VISIBLE
            binding.errorTextView.text = "Error mengambil data:
            ${error.networkResponse?.statusCode ?: "null"} - ${error.message ?: "Tidak ada
            pesan"}"
        }
    )

    requestQueue.add(jsonArrayRequest)
}

/**
 * Menampilkan data absensi di dalam tabel.
 * @param absen Objek Absen yang berisi data absensi terbaru.
 */
private fun displayAbsensiData(absen: Absen) {
    binding.latestAbsensiTable.removeAllViews()
    val headerRow = TableRow(context)
    val headers = listOf("Waktu", "Jam Masuk", "Jam Pulang",
    "Keterangan")

    headers.forEach { header ->
        val textView = TextView(context).apply {
            text = header
            setPadding(8, 8, 8, 8)
            setTypeface(null, android.graphics.Typeface.BOLD)
            gravity = Gravity.CENTER
        }
        headerRow.addView(textView)
    }
}

```

```
}  
binding.latestAbsensiTable.addView(headerRow)  
  
val tableRow = TableRow(context)  
  
val textViewWaktu = TextView(context).apply {  
    text = absen.waktu  
    setPadding(8, 8, 8, 8)  
    gravity = Gravity.CENTER  
}  
val textViewJamMasuk = TextView(context).apply {  
    text = absen.jam_masuk  
    setPadding(8, 8, 8, 8)  
    gravity = Gravity.CENTER  
}  
val textViewJamPulang = TextView(context).apply {  
    text = absen.jam_pulang  
    setPadding(8, 8, 8, 8)  
    gravity = Gravity.CENTER  
}  
val textViewKeterangan = TextView(context).apply {  
    text = absen.keterangan  
    setPadding(8, 8, 8, 8)  
    gravity = Gravity.CENTER  
}  
  
tableRow.addView(textViewWaktu)  
tableRow.addView(textViewJamMasuk)  
tableRow.addView(textViewJamPulang)  
tableRow.addView(textViewKeterangan)
```

```

        tableRow.setOnClickListener {
            parentFragmentManager.commit {
                replace(R.id.fragment_container, RiwayatAbsensiFragment())
                addToBackStack(null)
            }
        }

        binding.latestAbsensiTable.addView(tableRow)
    }

    /**
     * Mengambil data izin dari server dan menampilkannya.
     */
    private fun fetchIzinData() {
        val prefHelper = PrefHelper(requireContext())
        val nama = prefHelper.getUser() ?: return

        val url = "https://tkdharma.site/get_tb_izin.php?nama=$nama"

        val jsonArrayRequest = JsonRequest(
            Request.Method.GET, url, null,
            { response ->
                val dataType = object : TypeToken<List<Izin>>() {}.type
                val izinList: List<Izin> = Gson().fromJson(response.toString(),
dataType)

                if (izinList.isNotEmpty()) {
                    displayIzinData(izinList.first())
                } else {
                    // Tangani kasus tidak ada data
                    binding.errorTextView2.visibility = View.VISIBLE
                }
            }
        )
    }

```

```

        binding.errorTextView2.text = "Tidak ada data ditemukan
untuk pengguna: $nama"
    }
},
{ error ->
    error.printStackTrace()
    // Tangani error
    binding.errorTextView2.visibility = View.VISIBLE
    binding.errorTextView2.text = "Error mengambil data:
${error.networkResponse?.statusCode ?: "null"} - ${error.message ?: "Tidak ada
pesan"}"
}
)

requestQueue.add(jsonArrayRequest)
}

/**
 * Menampilkan data izin di dalam tabel.
 * @param izin Objek Izin yang berisi data izin terbaru.
 */
private fun displayIzinData(izin: Izin) {
    binding.latestIzinTable.removeAllViews()
    val headerRow = TableRow(context)
    val headers = listOf("Tanggal", "Nama", "Alasan", "Status")

    headers.forEach { header ->
        val textView = TextView(context).apply {
            text = header
            setPadding(8, 8, 8, 8)
            setTypeface(null, android.graphics.Typeface.BOLD)

```

```
        gravity = Gravity.CENTER
    }
    headerRow.addView(textView)
}
binding.latestIzinTable.addView(headerRow)
```

```
val tableRow = TableRow(context)
```

```
val textViewTanggal = TextView(context).apply {
    text = izin.tanggal
    setPadding(8, 8, 8, 8)
    gravity = Gravity.CENTER
}
```

```
val textViewNama = TextView(context).apply {
    text = izin.nama
    setPadding(8, 8, 8, 8)
    gravity = Gravity.CENTER
}
```

```
val textViewAlasan = TextView(context).apply {
    text = izin.alasan
    setPadding(8, 8, 8, 8)
    gravity = Gravity.CENTER
}
```

```
val textViewStatus = TextView(context).apply {
    text = izin.status
    setPadding(8, 8, 8, 8)
    gravity = Gravity.CENTER
}
```

```
tableRow.addView(textViewTanggal)
tableRow.addView(textViewNama)
```

```

        tableView.addView(textViewAlasan)
        tableView.addView(textViewStatus)

        tableView.setOnClickListener {
            parentFragmentManager.commit {
                replace(R.id.fragment_container, PerizinanFragment())
                addToBackStack(null)
            }
        }

        binding.latestIzinTable.addView(tableView)
    }

    /**
     * Dipanggil saat tampilan fragment dihancurkan.
     */
    override fun onDestroyView() {
        super.onDestroyView()
        handler.removeCallbacks(updateDateTimeTask)
        _binding = null
    }
}

```

VolleyMultipartRequest.kt

```

package com.example.absensiapk.helper

import com.android.volley.AuthFailureError
import com.android.volley.NetworkResponse
import com.android.volley.Request
import com.android.volley.Response

```

```

import com.android.volley.VolleyError
import com.android.volley.toolbox.HttpHeaderParser
import java.io.*

/**
 * VolleyMultipartRequest adalah kelas yang memungkinkan pengiriman
 permintaan multipart/form-data menggunakan library Volley.
 * @param method Metode HTTP yang digunakan untuk permintaan.
 * @param url URL tujuan permintaan.
 * @param listener Listener untuk menangani respons dari permintaan.
 * @param errorListener Listener untuk menangani kesalahan selama
 permintaan.
 */
open class VolleyMultipartRequest(
    method: Int,
    url: String,
    private val listener: Response.Listener<NetworkResponse>,
    private val errorListener: Response.ErrorListener
) : Request<NetworkResponse>(method, url, errorListener) {

    private var headers: Map<String, String> = emptyMap()

    /**
     * Mengembalikan header HTTP yang digunakan dalam permintaan.
     * @return Peta yang berisi pasangan kunci-nilai header HTTP.
     */
    override fun getHeaders(): Map<String, String> {
        return headers
    }

    /**

```

```

* Mengatur header HTTP yang akan digunakan dalam permintaan.
* @param headers Peta yang berisi pasangan kunci-nilai header HTTP.
*/
fun setHeaders(headers: Map<String, String>) {
    this.headers = headers
}

/**
* Mengembalikan tipe konten dari body permintaan.
* @return String yang mewakili tipe konten dari body permintaan.
*/
@Throws(AuthFailureError::class)
override fun getBodyContentType(): String {
    return "multipart/form-data; boundary=$boundary"
}

/**
* Mengembalikan body dari permintaan dalam bentuk array byte.
* @return Array byte yang mewakili body dari permintaan.
*/
@Throws(AuthFailureError::class)
override fun getBody(): ByteArray {
    val bos = ByteArrayOutputStream()
    val dos = DataOutputStream(bos)

    try {
        // Mengisi payload teks
        getParams()?.forEach { (key, value) ->
            buildTextPart(dos, key, value)
        }
    }
}

```

```

        // Mengisi payload data
        getByteData()?.forEach { (key, data) ->
            buildDataPart(dos, data, key)
        }

        // Menutup multipart form data
        dos.writeBytes("--$boundary--\r\n")
        dos.flush()
    } catch (e: IOException) {
        e.printStackTrace()
    }

    return bos.toByteArray()
}

/**
 * Mem-parsing respons dari jaringan.
 * @param response Objek NetworkResponse yang berisi respons dari
jaringan.
 * @return Objek Response yang berisi respons yang telah di-parse.
 */
override fun parseNetworkResponse(response: NetworkResponse):
Response<NetworkResponse> {
    return try {
        Response.success(response,
HttpHeaderParser.parseCacheHeaders(response))
    } catch (e: Exception) {
        Response.error(e as VolleyError?)
    }
}

```

```

/**
 * Mengirimkan respons ke listener.
 * @param response Objek NetworkResponse yang berisi respons dari
jaringan.
 */
override fun deliverResponse(response: NetworkResponse) {
    listener.onResponse(response)
}

/**
 * Membangun bagian teks dari multipart/form-data.
 * @param dos Objek DataOutputStream yang digunakan untuk menulis
data.
 * @param parameterName Nama parameter teks.
 * @param parameterValue Nilai parameter teks.
 */
private fun buildTextPart(dos: DataOutputStream, parameterName:
String, parameterValue: String) {
    try {
        dos.writeBytes("--$boundary\r\n")
        dos.writeBytes("Content-Disposition: form-data;
name=\"$parameterName\"\r\n")
        dos.writeBytes("\r\n")
        dos.writeBytes(parameterValue + "\r\n")
    } catch (e: IOException) {
        e.printStackTrace()
    }
}

/**
 * Membangun bagian data dari multipart/form-data.

```

* @param dos Objek DataOutputStream yang digunakan untuk menulis data.

* @param data Objek DataPart yang berisi data file.

* @param inputName Nama input file.

*/

private fun buildDataPart(dos: DataOutputStream, data: DataPart, inputName: String) {

try {

dos.writeBytes("--\$boundary\r\n")

dos.writeBytes("Content-Disposition: form-data;

name=\"\$inputName\"; filename=\"\$data.fileName\"\r\n")

dos.writeBytes("Content-Type: \${data.type}\r\n")

dos.writeBytes("\r\n")

val fileInputStream = ByteArrayInputStream(data.content)

val buffer = ByteArray(1024)

var bytesRead: Int

while (fileInputStream.read(buffer).also { bytesRead = it } != -1) {

dos.write(buffer, 0, bytesRead)

}

dos.writeBytes("\r\n")

} catch (e: IOException) {

e.printStackTrace()

}

}

/**

* Mendapatkan data byte untuk multipart/form-data.

* @return Peta yang berisi pasangan kunci-nilai data byte.

*/

```
open fun getByteData(): Map<String, DataPart>? {  
    return emptyMap()  
}  
  
/**  
    * DataPart adalah kelas data untuk menyimpan informasi file yang akan  
    diunggah.  
    * @param fileName Nama file.  
    * @param content Konten file dalam bentuk byte array.  
    * @param type Tipe MIME dari file.  
    */  
data class DataPart(val fileName: String, val content: ByteArray, val  
type: String)  
  
companion object {  
    private val boundary = "apiclient-" + System.currentTimeMillis()  
}  
}
```

Prefhelper.kt

```
package com.example.absensiapk.helper

import android.content.Context
import android.content.SharedPreferences

/**
 * PrefHelper adalah kelas helper untuk mengelola sesi pengguna
 * menggunakan SharedPreferences.
 * @param context Konteks aplikasi.
 */
class PrefHelper(context: Context) {

    private val sharedPreferences: SharedPreferences =
        context.getSharedPreferences("user_session",
Context.MODE_PRIVATE)

    /**
     * Menyimpan sesi pengguna.
     * @param username Nama pengguna.
     * @param password Kata sandi pengguna.
     * @param user Nama lengkap pengguna.
     * @param isLogin Status login pengguna.
     * @param userLevel Tingkat pengguna.
     * @param userId ID pengguna.
     * @param nip Nomor Induk Pegawai pengguna.
     * @param jabatan Jabatan pengguna.
     * @param foto URL foto pengguna.
     */
    fun saveSession(
```

```

        username: String,
        password: String,
        user: String,
        isLogin: Boolean,
        userLevel: String,
        userId: String,
        nip: String,
        jabatan: String,
        foto: String
    ) {
        sharedPreferences.edit().apply {
            putString("user_uid", userId)
            putString("username", username)
            putString("password", password)
            putString("user", user)
            putBoolean("is_login", isLogin)
            putString("user_level", userLevel)
            putString("nip", nip)
            putString("jabatan", jabatan)
            putString("foto", foto)
            putLong("last_active_time", System.currentTimeMillis())
            apply()
        }
    }

/**
 * Mendapatkan ID pengguna yang disimpan.
 * @return ID pengguna atau null jika tidak ditemukan.
 */
fun getUserId(): String? {
    return sharedPreferences.getString("user_uid", null)
}

```

```

}

/**
 * Mendapatkan nama lengkap pengguna yang disimpan.
 * @return Nama lengkap pengguna atau null jika tidak ditemukan.
 */
fun getUser(): String? {
    return sharedPreferences.getString("user", null)
}

/**
 * Mendapatkan nama pengguna yang disimpan.
 * @return Nama pengguna atau null jika tidak ditemukan.
 */
fun getUsername(): String? {
    return sharedPreferences.getString("username", null)
}

/**
 * Mendapatkan NIP pengguna yang disimpan.
 * @return NIP pengguna atau null jika tidak ditemukan.
 */
fun getNip(): String? {
    return sharedPreferences.getString("nip", null)
}

/**
 * Mendapatkan jabatan pengguna yang disimpan.
 * @return Jabatan pengguna atau null jika tidak ditemukan.
 */
fun getJabatan(): String? {

```

```

        return sharedPreferences.getString("jabatan", null)
    }

    /**
     * Mendapatkan URL foto pengguna yang disimpan.
     * @return URL foto pengguna atau null jika tidak ditemukan.
     */
    fun getFoto(): String? {
        return sharedPreferences.getString("foto", null)
    }

    /**
     * Mendapatkan status login pengguna.
     * @return True jika pengguna sedang login, False jika tidak.
     */
    fun getIsLogin(): Boolean {
        return sharedPreferences.getBoolean("is_login", false)
    }

    /**
     * Mendapatkan tingkat pengguna yang disimpan.
     * @return Tingkat pengguna atau null jika tidak ditemukan.
     */
    fun getUserLevel(): String? {
        return sharedPreferences.getString("user_level", null)
    }

    /**
     * Menghapus sesi pengguna.
     */
    fun clearSession() {

```

```

        sharedPreferences.edit().apply {
            clear()
            apply()
        }
    }

    /**
     * Menyimpan waktu terakhir pengguna aktif.
     */
    fun saveLastActiveTime() {
        sharedPreferences.edit().apply {
            putLong("last_active_time", System.currentTimeMillis())
            apply()
        }
    }

    /**
     * Mengecek apakah waktu tidak aktif pengguna telah melebihi batas
     yang diizinkan.
     * @param allowedInactiveTime Batas waktu tidak aktif yang diizinkan
     dalam milidetik.
     * @return True jika waktu tidak aktif melebihi batas yang diizinkan,
     False jika tidak.
     */
    fun isInactiveTimeEeeded(allowedInactiveTime: Long): Boolean {
        val lastActiveTime = sharedPreferences.getLong("last_active_time",
0)

        val currentTime = System.currentTimeMillis()
        return (currentTime - lastActiveTime) > allowedInactiveTime
    }
}

```

LAMPIRAN 5

DOKUMENTASI KEGIATAN



