**RANCANG BANGUN APLIKASI PRESENSI GURU BERBASIS WEB MENGGUNAKAN FRAMEWORK LARAVEL**

**(Studi Kasus: MA Nurul Huda Pacet)**

**SKRIPSI**

Usulan Penelitian sebagai syarat mengikuti Skripsi

Disusun oleh:

**MUHAMAD IQBAL RAFLY**

**NPM.** **301200003**



PROGAM STRATA 1

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS TEKNOLOGI INFORMASI

UNIVERSITAS BALE BANDUNG

BANDUNG

2024

# 

# LEMBAR PERSETUJUAN PEMBIMBING

**RANCANG BANGUN APLIKASI PRESENSI GURU BERBASIS WEB MENGGUNAKAN FRAMEWORK LARAVEL**

**(Studi Kasus:** **MA Nurul Huda Pacet)**

Disusun Oleh :

MUHAMAD IQBAL RAFLY

NPM. 301200003

Telah diterima dan disetujui untuk memenuhi persyaratan mencapai gelar

**SARJANA KOMPUTER**

Pada

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS TEKNOLOGI INFORMASI**

**UNIVERSITAS BALE BANDUNG**

Baleendah, Juli 2024

Disetujui oleh:

|  |  |
| --- | --- |
| Pembimbing Utama | Pembimbing Pendamping |
|  |  |
| Rustiyana, S.T., M.T. | Sukiman, S.Tr., Kom., M.Kom., MCE. |
| NIK. 04104808015 | NIK. 04104821001 |

# LEMBAR PERSETUJUAN PENGUJI

**RANCANG BANGUN APLIKASI PRESENSI GURU BERBASIS WEB MENGGUNAKAN FRAMEWORK LARAVEL**

**(Studi Kasus: MA Nurul Huda Pacet)**

Disusun Oleh :

MUHAMAD IQBAL RAFLY

NPM. 301200003

Telah diterima dan disetujui untuk memenuhi persyaratan mencapai gelar

**SARJANA KOMPUTER**

Pada

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS TEKNOLOGI INFORMASI**

**UNIVERSITAS BALE BANDUNG**

Baleendah, Juli 2024

Disetujui oleh:

|  |  |
| --- | --- |
| Penguji 1 | Penguji 2 |
|  |  |
| Yudi Herdiana, S.T, M.T. | Sutiyono, S.T., M.Kom. |
| NIK. 04104808008 | NIK. 01043180002 |

**LEMBAR PENGESAHAN PROGRAM STUDI**

**RANCANG BANGUN APLIKASI PRESENSI GURU BERBASIS WEB MENGGUNAKAN FRAMEWORK LARAVEL**

**(STUDI KASUS : MA NURUL HUDA PACET)**

Oleh :

Muhamad Iqbal Rafly

NPM. 301200003

telah diterima dan disetujui untuk memenuhi persyaratan mencapai gelar

**SARJANA KOMPUTER**

Baleendah, Juli 2024

|  |  |
| --- | --- |
| Dekan Fakultas teknologi Informasi | Ketua Program Studi Teknik Informatika |
|  |  |
| Yudi Herdiana, S.T., M.T., | Yusuf Muharam, S.Kom.,M. Kom. |
| NIK. 04104808008 | NIK. 04104820003 |

**HALAMAN PERNYATAAN**

Yang bertanda tangan di bawah ini, saya:

|  |  |  |
| --- | --- | --- |
| Nama | : | Muhamad Iqbal Rafly |
| NIM | : | 301200003 |
| Jurusan | : | Teknik Informatika |
| Fakultas | : | Fakultas Teknologi Informasi |
| Judul | : | Rancang Bangun Aplikasi Presensi Guru Berbasis Web Menggunakan *Framework Laravel* (Studi Kasus: MA Nurul Huda Pacet) |

Dengan ini penulis menyatakan sebenar-benarnya bahwa penulisan skripsi ini berdasarkan hasil penelitian, pemikiran, dan pemaparan asli dari penulis sendiri, baik untuk naskah laporan maupun kegiatan programming yang tercantum sebagai bagian dari skripsi ini. Jika terdapat karya orang lain, penulis mencantumkan sumber yang jelas mengikuti tata penulisan karya ilmiah yang seharusnya.

Bandung, 28 Mei 2024

Materai 10.000

Muhamad Iqbal Rafly

NPM. 301200003

# ABSTRAK

Pemanfaatan teknologi informasi dan komunikasi sudah menjadi kunci untuk meningkatkan efektivitas pengelolaan presensi. Pengelolaan presensi secara efisien sering kali menjadi tantangan bagi para pelaku di sebuah perusahaan ataupun instansi. Banyak perusaan ataupun instansi yang belum mengadopsi aplikasi presensi yang terdigitalisasi secara modern yang mengandalkan teknologi salah satunya adalah MA Nurul Huda Pacet. Hal tersebut menimbulkan beberapa masalah diantaranya ketidakakuratan data, kesulitan dalam melacak waktu kehadiran, serta keterlambatan dalam menyajikan laporan, dengan adanya penelitian ini diharapkan dapat mengatasi masalah tersebut.

Penelitian ini bertujuan untuk merancang dan membangun aplikasi presensi guru berbasis web yang dapat membantu dalam manajemen presensi secara terkomputerisasi. Penelitian ini menggunakan metode pengumpulan data metode kualitatif yaitu observasi, interview dan studi pustaka, dengan waterfall sebagai pengembangan nya dengantahapan analisis kebutuhan, perancangan sistem, implementasi, dan pengujian. Alat bantu yang digunakan dalam perancangan sistem adalah *Unified Modelling Language* (UML) yang dimana meliputi use case diagram, activity diagram, dan class diagram. Basis data yang digunakan adalah MySQL dengan struktur tabel yang dirancang sesuai dengan kebutuhan sistem. Sedangkan bahasa pemrograman yang digunakan adalah PHP dan framework Laravel. Pada tahapan perancangan antarmuka, desain aplikasi dibuat menggunakan Figma, dengan desain antarmuka yang dapat mudah di pahami oleh pengguna dan tentunya dengan memperhatikan aspek *user experience* Fitur-fitur utama aplikasi meliputi pengelolaan data presensi dengan, pengelolaan data pengguna, pengelola perizinan, laporan. pembuatan laporan Pengujian aplikasi dilakukan dengan metode *black box* testing untuk memastikan fungsionalitas dan non-fungsionalitas aplikasi berjalan dengan baik serta menggunakan metode validitas untuk mendapatkan nilai kelayakan dari aplikasi yang telah dibuat.

|  |  |
| --- | --- |
| Kata Kunci **:** | aplikasi web, *Laravel*, MySQL, *Presensi* , PHP, UML ,  *Waterfall* |

# ABSTRACT

*The use of information and communication technology has become the key to improving the effectiveness of presence management. Efficient presence management is often a challenge for perpetrators in a company or agency. Many firms or agencies have not adopted a modernly digitized presence application that relies on technology, one of which is MA Nurul Huda Pacet. This raises a number of problems, including data inaccuracy, difficulties in tracking attendance times, and delays in reporting, with the research expected to address the problem.*

*The research aims to design and build a web-based teacher presence application that can help in computerized presence management. The research uses qualitative methods of data collection such as observation, interview and library study, with waterfall as its development phase-by-phase needs analysis, system design, implementation, and testing. The utility used in system design is the Unified Modelling Language (UML) which includes the use case diagram, activity diagram and class diagram. The database used is MySQL with a table structure designed according to system requirements. The programming languages used are PHP and the Laravel framework. At the stage of interface design, application design is made using Figma, with an interface design that is easy to understand by the user and, of course, taking into account the user experience aspect. The main features of the application include presence data management with, user data management, authorization manager, reporting. reporting Application testing is done using the black box testing method to ensure the functionality and non-functionality of the application is running properly as well as using the validity method to obtain validity values from the app that has been created.*

|  |  |
| --- | --- |
| *Keywords* **:** | *Web applicatio, Laravel, MySQL, Presence,PHP,UML , Waterfall* |

**KATA PENGANTAR**

Puji syukur kehadirat Allah SWT yang senantiasa melimpahkan rahmat, dan karunia-nya, sehingga penulis dapat menyelesaikan skripsi yang berjudul **“RANCANG BANGUN APLIKASI PRESENSI GURU BERBASIS WEB MENGGUNAKAN FRAMEWORK LARAVEL (STUDI KASUS: MA NURUL HUDA PACET)”** ini dengan baik. Dalam penyusunan-nya penulis memperoleh banyak bantuan dari berbagai pihak, maka dari itu penulis mengucapkan terimakasih yang sebanyak-banyak nya kepada.

1. Allah SWT yang telah memberikan kemudahan serta kelancaran sehingga penulis dapat menjalani proses demi proses pelaksanakaan Tugas Akhir ini.
2. Kedua orang tua yaitu Bapak Herry Suhaeri dan Ibu Masitoh serta Keluarga yang selalu meberikan dukungan dan do’anya
3. Bapak Yudi Herdiana, S.T., M.T., selaku Dekan Fakultas Teknologi Informasi Universitas Bale Bandung.
4. Bapak Yusuf Muharam, S.Kom., M.Kom., selaku Ketua Program Studi Teknik Informatika Fakultas Teknologi Informasi Universitas Bale Bandung.
5. Semua pihak yang telah membantu dan memberikan dukungan kepada penulis untuk menyelesaikan proposal ini.

Penyusun menyadari bahwa masih banyak kekurangan dalam skripsi ini dan segala kritik dan saran yang membangun akan penyusun terima dengan baik. Akhir kata semoga skripsi ini bisa diterima dan bermanfaat bagi kita semua.

Bandung, April 2024

Penulis

# DAFTAR ISI

[ABSTRAK v](#_Toc174496446)

[ABSTRACT vi](#_Toc174496447)

[DAFTAR ISI viii](#_Toc174496448)

[DAFTAR TABEL xi](#_Toc174496449)

[DAFTAR GAMBAR xii](#_Toc174496450)

[DAFTAR LAMPIRAN xii](#_Toc174496451)

[BAB I PENDAHULUAN 1](#_Toc174496452)

[1.1 Latar Belakang 1](#_Toc174496453)

[1.2 Rumusan Masalah 2](#_Toc174496454)

[1.3 Batasan Masalah 2](#_Toc174496455)

[1.4 Tujuan Penelitian 3](#_Toc174496456)

[1.5 Metode Penelitian 3](#_Toc174496457)

[1.5.1 Metode Pengumpulan Data 3](#_Toc174496458)

[1.5.2 Metode Pengembangan Sistem 3](#_Toc174496459)

[1.6 Sistematika Penulisan 3](#_Toc174496460)

[BAB I PENDAHULUAN 4](#_Toc174496461)

[BAB II TINJAUAN PUSTAKA 4](#_Toc174496462)

[BAB III METODOLOGI PENELITIAN 4](#_Toc174496463)

[BAB IV ANALISIS, PERANCANGAN DAN HASIL 4](#_Toc174496464)

[BAB V IMPLEMENTASI DAN PENGUJIAN 4](#_Toc174496465)

[BAB VI PENUTUP 4](#_Toc174496466)

[BAB II TINJAUAN PUSTAKA 5](#_Toc174496467)

[2.1 Landasan Teori 5](#_Toc174496468)

[2.2.1 Acuan Penelitian Pertama 8](#_Toc174496469)

[2.2.2 Acuan Penelitian Kedua 8](#_Toc174496470)

[2.2.3 Acuan Penelitian Ketiga 9](#_Toc174496471)

[2.2 Dasar Teori 9](#_Toc174496472)

[2.2.1 Presensi 9](#_Toc174496473)

[2.2.2 MySQL 10](#_Toc174496474)

[2.2.3 Website 11](#_Toc174496475)

[2.2.4 Perancangan Sistem 12](#_Toc174496476)

[2.2.5 HeidiSQL 18](#_Toc174496477)

[2.2.6 PHP 18](#_Toc174496478)

[2.2.7 Motode SDLC *Waterfall* 18](#_Toc174496479)

[2.2.8 XAMPP 20](#_Toc174496480)

[2.2.9 Framework 21](#_Toc174496481)

[2.2.10 Laravel 21](#_Toc174496482)

[2.2.11 Visual Studio Code 23](#_Toc174496483)

[2.2.12 Figma 23](#_Toc174496484)

[2.2.13 Pengujian Blackbox 23](#_Toc174496485)

[BAB III ANALISIS DAN PERANCANGAN 24](#_Toc174496486)

[3.1 Kerangka Pikir 24](#_Toc174496487)

[3.2 Deskripsi 24](#_Toc174496488)

[3.1.1 Identifikasi Masalah 25](#_Toc174496489)

[3.1.2 Pengumpulan Data 25](#_Toc174496490)

[3.1.3 Analisis Kebutuhan 26](#_Toc174496491)

[3.1.3 Perancangan 26](#_Toc174496492)

[3.1.4 Implementasi 29](#_Toc174496493)

[3.1.5 Pengujian 29](#_Toc174496494)

[3.1.6 Pelaporan 30](#_Toc174496495)

[BAB IV ANALISIS DAN PERANCANGAN 31](#_Toc174496496)

[4.1 Analisis 31](#_Toc174496497)

[4.1.1 Analisis Masalah 31](#_Toc174496498)

[4.1.2 Analisis Software 31](#_Toc174496499)

[4.1.3 Analisis Pengguna 32](#_Toc174496500)

[4.1.4 User Interface 32](#_Toc174496501)

[4.1.5 Fitur-fitur 33](#_Toc174496502)

[4.1.6 Analisis Data 34](#_Toc174496503)

[4.1.7 Analisis Biaya 34](#_Toc174496504)

[4.2 Perancangan 35](#_Toc174496505)

[4.2.1 Unified Modeling Language (UML) 35](#_Toc174496506)

[4.2.1 Struktur Tabel 43](#_Toc174496507)

[4.2.2 Desain Sistem 46](#_Toc174496508)

[BAB V IMPLEMENTASI DAN PENGUJIAN 53](#_Toc174496509)

[5.1 Implementasi 53](#_Toc174496510)

[5.1.2 Implementasi Sistem 53](#_Toc174496511)

[5.1.3 Spesifikasi Sistem 53](#_Toc174496512)

[5.1.4 Installasi Sistem 54](#_Toc174496513)

[5.1.5 Menjalankan Sistem 55](#_Toc174496514)

[5.2 Pengujian 62](#_Toc174496515)

[BAB VI KESIMPULAN 65](#_Toc174496516)

[6.1 Kesimpulan 65](#_Toc174496517)

[6.2 Saran 65](#_Toc174496518)

[DAFTAR PUSTAKA 67](#_Toc174496519)

[LAMPIRAN 73](#_Toc174496520)

# DAFTAR TABEL

[Tabel 4.1 Analisis Biaya 35](#_Toc174499700)

[Tabel 4.2 Deskripsi Aktor 36](#_Toc174499701)

[Tabel 4.3 Tabel Pengguna 43](#_Toc174499702)

[Tabel 4.4 Tabel user 44](#_Toc174499703)

[Tabel 4.5 Tabel admin 45](#_Toc174499704)

[Tabel 4.6 Tabel perizinan 46](#_Toc174499705)

[Tabel 2.1 Acuan Penelitian 6](#_Toc174499651)

[Tabel 2.2 Simbol Use Case Diagram 12](#_Toc174499652)

[Tabel 2.3 Simbol Class Diagram 14](#_Toc174499653)

[Tabel 2.4 Simbol Activity Diagram 15](#_Toc174499654)

[Tabel 2.5 Simbol Sequence Diagram 17](#_Toc174499655)

[Tabel 5. 1 Spesifikasi Perangkat Keras 53](#_Toc174499887)

[Tabel 5.2 Spesifikasi Perangkat Lunak 54](#_Toc174499888)

[Tabel 5. 3 Hasil Pengujian Aplikasi Dengan Metode Blackbox 63](#_Toc174499889)

# DAFTAR GAMBAR

[Gambar 2.1 Arsitektur Website 11](#_Toc174499083)

[Gambar 2.2 Model SDLC Waterfall 19](#_Toc174499084)

[Gambar 2.3 Arsitektur Laravel 22](#_Toc174499085)

[Gambar 3.1 Kerangka Fikir 24](#_Toc174499076)

[Gambar 4.1 Use case Diagram 36](#_Toc174499048)

[Gambar 4.2 Activity Diagram Register 37](#_Toc174499049)

[Gambar 4.3 Activity Diagram Login 38](#_Toc174499050)

[Gambar 4.4 Activity Diagram Data Absensi 39](#_Toc174499051)

[Gambar 4.5 Activity Diagram Data Perizinan 40](#_Toc174499052)

[Gambar 4.6 Activity Diagram Data Admin 41](#_Toc174499053)

[Gambar 4.7 Activity Diagram Data Guru 42](#_Toc174499054)

[Gambar 4.8 *Class Diagram* 42](#_Toc174499055)

[Gambar 4.9 User Interface Halaman Register 47](#_Toc174499056)

[Gambar 4.10 *User Interface* Halaman Login 48](#_Toc174499057)

[Gambar 4.11 User Interface Halaman Dashboard 48](#_Toc174499058)

[Gambar 4. 12 User Interface Halaman Absensi 49](#_Toc174499059)

[Gambar 4. 13 User Interface Halaman admin 49](#_Toc174499060)

[Gambar 4.14 User interface halaman login user 50](#_Toc174499061)

[Gambar 4. 15 User Interface Halaman Login user 50](#_Toc174499062)

[Gambar 4.16 User interface halaman dashboard 51](#_Toc174499063)

[Gambar 4. 17 User Interface Halaman absensi user 52](#_Toc174499064)

[Gambar 4. 18 User Interface Halaman perizinan user 53](#_Toc174499065)

[Gambar 4. 19 Halaman Register 56](#_Toc174499066)

[Gambar 4.20 Halaman Login 56](#_Toc174499067)

[Gambar 4.21 Halaman Dashboard 57](#_Toc174499068)

[Gambar 4. 22 Halaman Absensi 57](#_Toc174499069)

[Gambar 4. 23 Halaman Admin 58](#_Toc174499070)

[Gambar 4. 24 Halaman Guru 58](#_Toc174499071)

[Gambar 4. 25 Halaman Login User 59](#_Toc174499072)

[Gambar 4. 26 Halaman Dashboard User 60](#_Toc174499073)

[Gambar 4. 27 Halaman Absensi User 61](#_Toc174499074)

[Gambar 4. 28 Halaman Perizinan User 62](#_Toc174499075)

# DAFTAR LAMPIRAN

[Lampiran: 1: Wawancara 73](#_Toc174498841)

[Lampiran: 2 : ToR (Term Of Reference) 75](#_Toc174498842)

[Lampiran:3 : Dokumentasi Wawancara 76](#_Toc174498843)

[Lampiran: 4 : Listing Program 77](#_Toc174498844)

[Lampiran: 5 : Riwayat Hidup 103](#_Toc174498845)

# BAB I PENDAHULUAN

## 1.1 Latar Belakang

Kemajuan teknologi informasi dan komunikasi mempengaruhi aktivitas manusia yang mengalami banyak perubahan dan perkembangan. Perkembangan ilmu pengetahuan di bidang teknologi saat ini telah telah mencuptakan aplikasi dan sistem oprasi baru dengan produktivitas dan biaya teknologi yang lebih efisien (atmaja & aziz 2019).

Presensi merupakan sebuah aktivitas yang sering diterapkan baik untuk mahasiswa, karyawan, dan staf pengajar, metode presensi memiliki berbagai model yang cukup beragam mulai penggunaan presensi manual, sidik jari, dan lain sebagainya. Beberapa metode tersebut masih memiliki kekurangan salah satunya pencatatan yang di lakukan secara manual yaitu penulisan tangan. Kekurangan dari metode penggunaan absensi manual adalah waktu pengolahan data lebih lama dan hal tersebut beresiko ketidakakuratan dalam pengolahan data presensi pegawai. (Emanual 2023)

Nurul Huda Pacet merupakan Yayasan Pendidikan Islam yang terletak di kecamatan Pacet atau yang memiliki alamat lengkap di Rt.02 RW.15, Desa Pangauban, Kecamatan Pacet, Kabupaten Bandung selatan. Bedasarkan observasi yang dilakukan di lapangan, di ketahui proses presensi guru yang berjalan pada MA Nurul Huda Pacet masih menggunakan cara yang konfensional yaitu dengan cara metode manual atau tanda tangan sehingga sistem penilaian sangat sensitif terhadap kesalahan manusia. Pencatatan manual yang dilakukan oleh guru atau pengelola dapat menyebabkan kesalahan pencatatan waktu kehadiran, dan pengisian data yang tidak lengkap. Hal ini yang menjadi alasan penulis melakukan sebuah penelitian.

Dengan adanya permasalahan terebut peneliti peneliti menganggap perlu adanya suatu inovasi dan solusi dalam proses presensi. Sebagai upaya penyelesaian masalah yang terjadi dengan membuat sistem aplikasi presensi dengan menggunakan perangkat *smartphone* dan *shoftware* laravel. Dengan adanya aplikasi presensi yang terkomputerisasi diharapkan dapat bekerja lebih efiseien dan efektif dalam memonitoring dan mengelola sistem presensi guru di MA Nurul Huda Pacet.

Berdasarkan latar belakang tersebut maka penulis mempertimbangkan untuk membuat presensi guru berbasis android dengan judul **“RANCANG BANGUN APLIKASI PRESENSI GURU BERBASIS WEB MENGGUNAKAN FRAMEWORK LARAVEL (Studi Kasus Di Ma Nurul Huda Pacet)"** dimana dengan adanya sistem ini nantinya akan memudahkan pihak Ma Nurul Huda ataupun staf oprator dalam proses monitoring presensi dan pengelolaan absensi guru.

## 1.2 Rumusan Masalah

Adapun berdasarkan latar belakang di atas yang telah diuraikan, permasalahan yang terjadi adalah sebagai berikut :

* + 1. Bagaimana membangun sebuah aplikasi yang dapat melakukan presensi guru menggunakan website dan framework laravel?
    2. Bagaimana membangun sebuah aplikasi yang dapat menampilkan laporan presensi guru?

## 1.3 Batasan Masalah

* + 1. Bagaimana membangun aplikasi yang dapat mengelola waktu kehadiran guru?
    2. Bagaimana membangun aplikasi yang dapat mengelola data jadwal jam kerja guru?
    3. Bagaimana membangun sebuah aplikasi yang dapat melakukan absensi menggunakan *smartphone*?
    4. Bagaimana membangun sebuah aplikasi yang dapat menampilkan laporan presensi guru?

## 1.4 Tujuan Penelitian

Berdasarkan rumusan masalah sebelumnya, maka tujuan penelitian yang ingin dicapai adalah:

* + 1. Merancang dan membangun aplikasi Presensi Guru berbasis *website* menggunakan *framework* laravel.
    2. Mengimplementasikan aplikasi Presensi Guru yang akan memberikan kemudahan dan pegelolaan data yang lebih baik dan detail kepada guru dan pelaksana pendidikan di sekolah MA Nurul Huda Pacet.

## 1.5 Metode Penelitian

### 1.5.1 Metode Pengumpulan Data

Pengumpulan data dilakukan melalui observasi langsung, wawancara dengan pihak terkait, dan studi literatur untuk mendukung pengembangan Rancang Bangun Aplikasi Presensi Guru Berbasis Website Menggunakan *Framework* laravel(Studi Kasus: MA Nurul Huda Pacet).

### 1.5.2 Metode Pengembangan Sistem

Adapun metode pengembangan yang digunakan adalah metode pengembangan perangkat lunak yang dikenal sebagai waterfall. Waterfall merupakan pendekatan di mana setiap tahapan harus diselesaikan sepenuhnya sebelum melanjutkan ke tahap berikutnya. Proses ini meliputi analisis sistem, desain, implementasi, pengujian, dan pemeliharaan. Hasil dari setiap tahap menjadi landasan untuk tahap berikutnya, dan umumnya, perubahan besar tidak dapat dilakukan sampai tahap sebelumnya selesai.

## 1.6 Sistematika Penulisan

Sistematika penulisan bertujuan untuk menyajikan informasi secara terstruktur dan mudah dipahami oleh pembaca. Adapun sistematika penulisan yang digunakan adalah sebagai berikut:

### BAB I PENDAHULUAN

Bab ini merupakan bagian utama yang mengemukakan latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, metodologi penelitian, dan sistematika penulisan laporan hasil penelitian.

### BAB II TINJAUAN PUSTAKA

Bab ini menjelaskan tentang landasan teori dan dasar teori pendukung dalam penelitian. Tinjauan Pustaka ini bersumber dari beberapa sumber seperti buku, jurnal, dan website.

### BAB III METODOLOGI PENELITIAN

Bab ini menjelaskan metodologi penelitian yang dipakai dan kerangka penelitian atau tahap-tahap penulis dalam melakukan penelitian di SMK Negeri 7 Baleendah.

### BAB IV ANALISIS, PERANCANGAN DAN HASIL

Bab ini berisi tentang analisis dan perancangan perangkat lunak, perancangan database, perancangan antarmuka serta penjelasan tentang perancangan perangkat lunak yang akan dibangun.

### BAB V IMPLEMENTASI DAN PENGUJIAN

Bab ini berisi penyajian tahap pengembangan aplikasi yang akan dijelaskan tiap langkahnya sertca contoh tampilan dari setiap aplikasi.

### BAB VI PENUTUP

Bab ini berisi kesimpulan dan penyajian tahap pengembangan yang dilakukan serta saran untuk pengembangan aplikasi selanjutnya.

# BAB II TINJAUAN PUSTAKA

## 2.1 Landasan Teori

Landasan teori ini berisi ringkasan berdasarkan jurnal yang berkaitan dengan penelitian yang dilakukan dan pengetahuan yang diterima selama perkuliahan untuk pengembangan perangkat lunak aplikasi. Pengetahuan digunakan meliputi:

1. Teori Algoritma 1 dan Algoritma 2.

Dalam penelitian ini, teori algoritma dan pemrograman digunakan dalam pembuatan kode program, karena algoritma berfungsi sebagai kerangka dasar dalam pengembangan kode program.

1. Basis Data.

Teori sistem basis data membantu dalam pembuatan dan pengelolaan basis data, yang berguna untuk mengatur data yang masuk maupun keluar.

1. Teori Interaksi Manusia dan Komputer.

Teori Interaksi Manusia dan Komputer digunakan untuk menciptakan desain antarmuka yang intuitif dan fungsional.

1. Teori Pemrograman Internet.

Dalam penelitian ini, penulis mengembangkan aplikasi berbasis website, di mana teori pemrograman internet memainkan peran penting dalam proses pembuatannya.

1. Rekayasa Perangkat Lunak

Dalam penelitian ini, penulis mengembangkan aplikasi berbasis website, dimana teori rekayasa perangkat lunak berperan untuk perancangan, pengembangan, pengujian, dan pemeliharaan pada perangkat lunak.

Pada penelitian ini, penyusun mempelajari penelitian terdahulu sebagai acuan dalam penelitian Rancang Bangun Aplikasi Presensi Guru Berbasis Website Menggunakan Framework Laravel (Studi Kasus : MA Nurul Huda Pacet) sebagai berikut:

Tabel 2.1 Acuan Penelitian

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Jurnal Penelitian | Masalah | Metode | Kesimpulan |
| 1 | **Judul:** SISTEM INFORMASI PRESENSI ONLINE MENGGUNAKAN TEKNOLOGI FACE RECOGNITION DAN GPS.  **Penulis:** Yusuf Wahyu Setiya Putra, Muhammad Fadlil Adhim.  **Tahun:** 2022 | Sistem presensi di SMK Muhammadiyah 1 Weleri masih menggukan cara konvensional sehingga membutuhkan waktu yang lama dan rentan adanya manipulasi data maupun data ganda. | Metode Waterfall | Sistem pengambilan presensi berbasis web ini dapat dikembangkan untuk mempermudah guru dan karyawan dalam mengontrol presensi agar bisa menjadi efektif dan efisien. |
| 2 | **Judul:** PERANCANGAN SISTEM INFORMASI ABSENSI DEWAN GURU TENAGA HARIAN LEPAS BERBASIS WEB PADA SEKOLAH DASAR NEGERI KUNCIRAN 6 KOTA TANGERANG.  **Penulis :** Aji Afriansyah dan Ari Syaripudin  **Tahun:** 2022 | Di sekolah ini masih menggunakan sistem absensi secara manual sehingga sering terhambatnya proses rekaptulasi absensi. Tujua dibuatnya Sistem Informasi Absensi Dewan Guru Tenaga Harian Lepas Berbasis Web Sekolah Dasar Negeri Kunciran 6 Kota Tangerang yakni untuk mempermudah Dewan Guru melakukan absensi | Waterfall | kini sistem yang ada di Sekolah Dasar Negeri Kunciran 6 kota Tanggerang sudah terkomputerisasi sehingga data absensi Dewan Guru Tenaga Harian Lepas dapat tersimpan menggunakan database dan kegiatan tersebut menjadi lebih efisien, cepat, dan akurat |
| 3 | **Judul:** Evaluasi Penerapan Inovasi Presensi Guru berbasis Online di Sekolah Dasar  **Penulis:** Resti Nadia Putri, Anggita Rahmawati, dan Diva Iftidiani  **Tahun:** 2023 |  | Metode Penelitian Kualitatif | Inovasi Aplikasi SiBeMo antara lain peningkatan akurasi kehadiran melalui fitur titik lokasi dan selfie, Rekapitulasi Absensi, dan Penyesuaian Kehadiran. Penerapan ini mencerminkan komitmen terhadap efisiensi operasional dan penggunaan teknologi informasi dalam pendidikan, dengan hasil wawancara menunjukkan penerimaan dan manfaat positif bagi guru. |

### 2.2.1 Acuan Penelitian Pertama

Pada acuan penelitian kedua terdapat beberapa persamaan, selain penelitian bertujuan meningkatkan pelayanan abseni, yaitu dengan menggunakan teknologi *smartphone* sehingga memberikan kemudahan bagi guru dan karyawan yang ingin melakukan absensi. . Pada penelitian ini metode pengembangan yang digunakan adalah Prototype dan menggunakan bahasa pemrograman PHP dengan framework Laravel serta menggunakan database MySQL

Sementara itu terdapat beberapa perbedaan dalam pendekatan pengembangan perangkat lunak. Pada penelitian ini metode pengembangan yang digunakan adalah face recognition.

### 2.2.2 Acuan Penelitian Kedua

Pada acuan penelitian kedua terdapat beberapa persamaan, selain penelitian bertujuan meningkatkan pelayanan abseni,yaitu dengan menggunakan teknologi face recognition serta memberikan kemudahan bagi guru dan karyawan yang ingin melakukan absensi.

Sementara itu terdapat beberapa perbedaan dalam pendekatan pengembangan perangkat lunak. Pada penelitian ini menggunakan bahasa pemograman *HyperText markup language* (HTML).

### 2.2.3 Acuan Penelitian Ketiga

Pada acuan penelitian ketiga terdapat beberapa persamaan, yaitu penelitian bertujuan meningkatkan pelayanan presensi guru, serta penggunaan fitur titik lokasi dan selfie, namun penelitian ini memiliki perbedaan penggunaan metode yaitu metode yang digunakan wawancara dan penelitian kualitatif yang bersifat evaluasi.

## 2.2 Dasar Teori

Pada pembuatan sistem informasi ini, penyusun mempunyai bahan dan alat untuk pembuatan sistem informasi, maka penyusun mempelajari teori-teori yang ada sebagai acuan dalam implementasi sistem informasi yang akan penulis buat yaitu sebagai berikut:

### 2.2.1 Presensi

Menurut kamus Besar Bahasa Indonesia (KBBI) presensi diartikan sebagai kehadiran. Sedangkan kehadiran sendiri mengacu pada kata hadir (seseorang, sekelompok orang) di suatu tempat. Absensi adalah suatu metode pencatatan kehadiran seseorang pada suatu kegiatan atau acara. Dapat digunakan dengan berbagai cara, baik secara manual maupun otomatis. Istilah ini biasanya digunakan dalam situasi seperti pekerjaan, sekolah, dan acara lainnya dengan audiens yang besar. (Meidiana 2023)

Presensi adalah suatu kegiatan mendokumentasikan kehadiran karyawan di perusahaan, setiap hari kerja pegawai diharuskan melakukan kehadiran karyawan perusahaan, setiap hari pegawai diharuskan melakukan presensi pada waktu datang dan pulangm dalam satu periode waktu. Jadi, presensi karyawan adalah suatu proses pencatatan data kehadiran karyawan melalui dari waktu masuk dan pulang karyawan pada periode waktu tertentu. (Bastian., 2007)

Bedasarkan defiisi di atas, maka dapat disimpulkan presensi adalah mengelola data kehadiran. Hal ini dapat membantu manajer atau supervisor untuk mengetahui kapan karyawan masuk atau keluar, serta memonitor tingkat kehadiran karyawan secara keseluruhan.

### 2.2.2 MySQL

MySQL adalah sebuah DBMS (Database Management System) menggunakan perintah SQL (Structured Query Language) yang banyak digunakan saat ini dalam pembuatan aplikasi berbasis website. MySQL dibagi menjadi dua lisensi, pertama adalah Free Software dimana perangkat lunak dapat diakses oleh siapa saja. Dan kedua adalah Shareware dimana perangkat lunak berpemilik memiliki batasan dalam penggunaannya

“MySQL adalah salah satu aplikasi DBMS yang sudah sangat banyak digunakan para pemrogram aplikasi web. Kelebihan dari MySQL adalah gratis, handal, selalu di-update dan banyak forum yang memfasilitasi para pengguna jika memiliki kendala. MySQL juga menjadi DBMS yang sering di bundling dengan web server sehingga proses instalasinya jadi lebih mudah” (Daniel Dido Jantce TJ Sitinjak, Maman, Jaka Suwita, 2020)

Beberapa fungsi utama MySql sebagai berikut:

Fungsi utama dari materi MySQL yang disampaikan adalah sebagai berikut:

**1.** Database Management System (DBMS): MySQL adalah sistem manajemen basis data yang memungkinkan pengguna untuk menyimpan, mengatur, dan mengakses data dengan efisien. Ini berarti MySQL menyediakan alat untuk membuat, mengedit, dan mengelola database serta tabel dan relasinya.

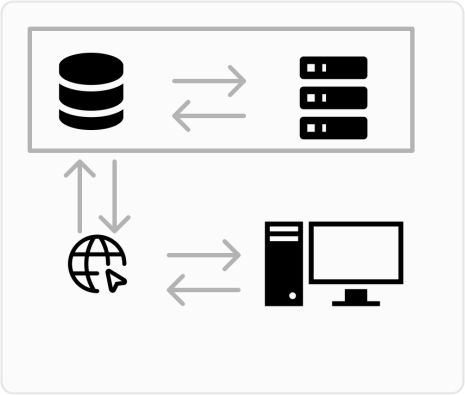
**2.** Penggunaan SQL: MySQL menggunakan bahasa kueri SQL (Structured Query Language) untuk berinteraksi dengan database. SQL memungkinkan pengguna untuk melakukan berbagai operasi seperti pengambilan data, penambahan data, pembaruan, penghapusan, dan banyak lagi.

**3.** Popularitas dalam Pembuatan Aplikasi Berbasis Web: MySQL banyak digunakan dalam pembuatan aplikasi berbasis web karena kemudahan penggunaannya dan kompatibilitasnya dengan banyak platform web. Ini membuatnya menjadi pilihan yang populer di kalangan pengembang web.

4. Dua Lisensi: MySQL memiliki dua jenis lisensi yang berbeda, yaitu Free Software dan Shareware. Free Software memungkinkan pengguna untuk mengakses perangkat lunak secara gratis tanpa batasan penggunaan, sementara Shareware memiliki batasan penggunaan karena dimiliki oleh pihak tertentu. (Suhartanto, no date)

### 2.2.3 Website

Perkembangan media internet sekarang ini semakin pesat dengan ditandai dengan sangat mudahnya informasi diperoleh (Prasetyo & Trisyanti, 2019). Saat ini penggunaan internet tidak hanya sebagai media informasi saja, akan tetapi sudah merambah kepada dunia bisnis dan pendidikan (Pradnyana et al., 2017).



Gambar 2.1 Arsitektur Website

(sumber : Wardani, Sirojul Hadi, Jian Budiarto,2021)

menunjukkan perancangan artisektur aplikasi web yaitu dengan menghubungkan antara server dengan komputer pengguna melalui jaringan internet. Data sensor suhu dan kelembaban akan di kirim ke database serta di operasikan pada VPS. Tidak jauh beda dengan aplikasi web, hanya saja aplikasi mobile view dirancang agar dapat di operasikan pada aplikasi android, aplikasi mobile view juga melakukan request ke server untuk mendapatkan data suhu terakhir.

### 2.2.4 Perancangan Sistem

1. *Unified Modeling Language* (UML)

UML digunakan untuk menggambarkan perancangan awal dari sistem yang akan dibangun. UML memiliki banyak jenis permodelan, tetapi hanya beberapa yang digunakan saja yang akan dibahas. UML (Unified Modeling Language) adalah salah satu alat bantu yang sangat handal di dunia pengembangan sistem yang berorientasi obyek. Hal ini disebabkan UML menyediakan bahasa pemodelan visual yang memungkinkan bagi pengembang sistem untuk membuat cetak biru atas visi mereka dalam bentuk yang baku, mudah dimengerti serta dilengkapi dengan mekanisme yang efektif untuk berbagi dan mengkonsumsikan rancangan mereka dengan yang lain (Fauzi, Indriyani and Hasta Yanto, 2020).

1. *Use Case* Diagram

Use Case Diagram adalah sesuatu atau proses merepresentasikan hal-hal yang dapat dilakukan oleh aktor dalam menyelesaikan sebuah pekerjaan. Menurut Shalahuddin dalam jurnal (Stinjak and Masya, 2021) mengungkapkan : “Diagram use case merupakan pemodelan untuk kelakuan (behavior) sistem informasi yang akan dibuat. Use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, use case digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut.” Berikut adalah simbol-simbol yang ada pada diagram use case:

Tabel 2.2 Simbol Use Case Diagram

|  |  |  |
| --- | --- | --- |
| No | Simbol | Keterangan |
| 1 |  | Aktor yaitu mewakilkan tugas manusia, sistem maupun perangkat lain dalam mengkomunikasikan *use case*. |
| 2 |  | *Use case* yaituAbstraksi dan hubungan antara sistem dan aktor. |
| 3 |  | *Association* yaitu Abstraksi dari penghubung antara aktor dengan *use case*. |
| 4 | - - - - - - - -> | *Generalisasi* yaitu Menunjukkan spesialisasi aktor sehingga mereka dapat berpartisipasi dengan *use case*. |
| 5 | <<**include**>> | Menunjukkan bahwa *use case* sepenuhnya merupakan fungsi dari *use case* lain. |
| 6 | <<extend>> | Menunjukkan bahwa *use case* adalah pelengkap fungsional untuk *use case*  lain jika suatu kondisi terpenuhi. |

1. Class Diagram

Class Diagram merupakan kumpulan objek-objek yang serupa. Setiap objek memiliki keadaan (state) dan perilaku (behavior). State dari objek adalah kondisinya yang dijelaskan melalui atributnya. Perilaku objek menentukan bagaimana objek tersebut bertindak dan merespons situasi tertentu. Secara umum, class memiliki tiga area utama: nama (stereotype), atribut, dan metode (operasi).

Menurut (Oktaviani, Widiarta and Nurlaily, 2019) mengungkapkan: “Class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi. Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas, sedangkan operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas”.

Tabel 2.3 Simbol Class Diagram

|  |  |  |  |
| --- | --- | --- | --- |
| No | Gambar | Nama | Keterangan |
| 1 |  | *Generalization* | Yaitu hubungan objek anak (descendent) berbagi perilaku dan struktur data objek diatas objek induk (ancestor). |
| 2 |  | *Nary Association* | Yaitu upaya untuk menghindari asosiasi dengan lebih dari 2 objek. |
| 3 |  | *Class* | Yaitu kumpulan dari objek yang memiliki atribut dan fungsi yang sama. |
| 4 |  | *Collaboration* | Yaitu deskripsi urutan tindakan yang dilakukan oleh sistem yang menghasilkan hasil yang terukur bagi suatu aktor. |
| 5 | <- - - - - - | *Realization* | Yaitu tindakan yang sebenarnya dilakukan objek. |
| 6 | - - - - - -> | *Dependency* | Yaitu hubungan dimana perubahan elemen mandiri (*independent*) mempengaruhi elemen yang bergantung pada elemen yang tidak mandiri. |
| 7 |  | *Association* | Apa yang menghubungkan antara objek satu dengan objek lainnya. |

1. Activity Diagram

Menurut Rosa dalam jurnal (Handayani *et al.*, 2023) mengungkapkan: “Activity Diagram menggambarkan work flow (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas dapat dilakukan oleh sistem”.

Activity Diagram memiliki peran yang sangat penting dalam mengilustrasikan alur aktivitas dari system yang sedang berjalan. Dengan demikian, activity diagram menjadi alat yang efektif dalam memahami dan mengkomunikasikan cara interaksi antara pengguna dan system aplikasi berlangsung dengan efisien.

Tabel 2.4 Simbol Activity Diagram

|  |  |  |  |
| --- | --- | --- | --- |
| No | Simbol | Nama | Keterangan |
| 1 |  | Status Awal | Yaitu sebuah diagram aktivitas memiliki keadaan awal. |
| 2 |  | Aktivitas | Yaitu aktivitas yang dilakukan oleh ocus, aktivitas biasanya dimulai dengan kata kerja. |
| 3 |  | Percabangan / Decision | Yaitu cabang ocus ada pilihan aktivitas yang lebih dari satu ocusar. |
| 4 |  | Percabangan / Join | Yaitu penggabungan ocus yang mana lebih dari satu kegiatan digabungkan menjadi satu. |
| 5 |  | Status Akhir | Yaitu keadaan akhir dari ocus, diagram aktivitas memiliki status akhir. |
| 6 |  | Swimlane | Yaitu memisahkan organisasi komersial yang bertanggung jawab atas aktivitas yang berlangsung. |

1. Sequence Diagram

Sequence Diagram menjelaskan interaksi objek yang disusun dalam suatu urutan waktu. Diagram ini secara khusus berasosiasi dengan usecase. Sequence diagram memperlihatkan tahap demi tahap apa yang seharusnya terjadi untuk menghasilkan sesuatu di dalam usecase. Tipe diagram ini sebaiknya digunakan diawal tahap desain atau analisis karena kesederhanaannya dan mudah untuk dimengerti (Widiyanto, 2022).

Tabel 2.5 Simbol Sequence Diagram

|  |  |  |  |
| --- | --- | --- | --- |
| No | Simbol | Nama | Keterangan |
| 1 |  | Aktor | Orang atau divisi yang terlibat dalam suatu sistem |
| 2 |  | Lifeline | Menyatakan kehidupan suatu objek dalam basis waktu. |
| 3 |  | Activation | Menyatakan objek dalam keadaan aktif dan berinteraksi. |
| 4 |  | message | Menyatakan arah tujuan antara object lifeline. |
| 5 |  | message (return) | Menyatakan arah kembali antara object lifeline. |

### 2.2.5 HeidiSQL

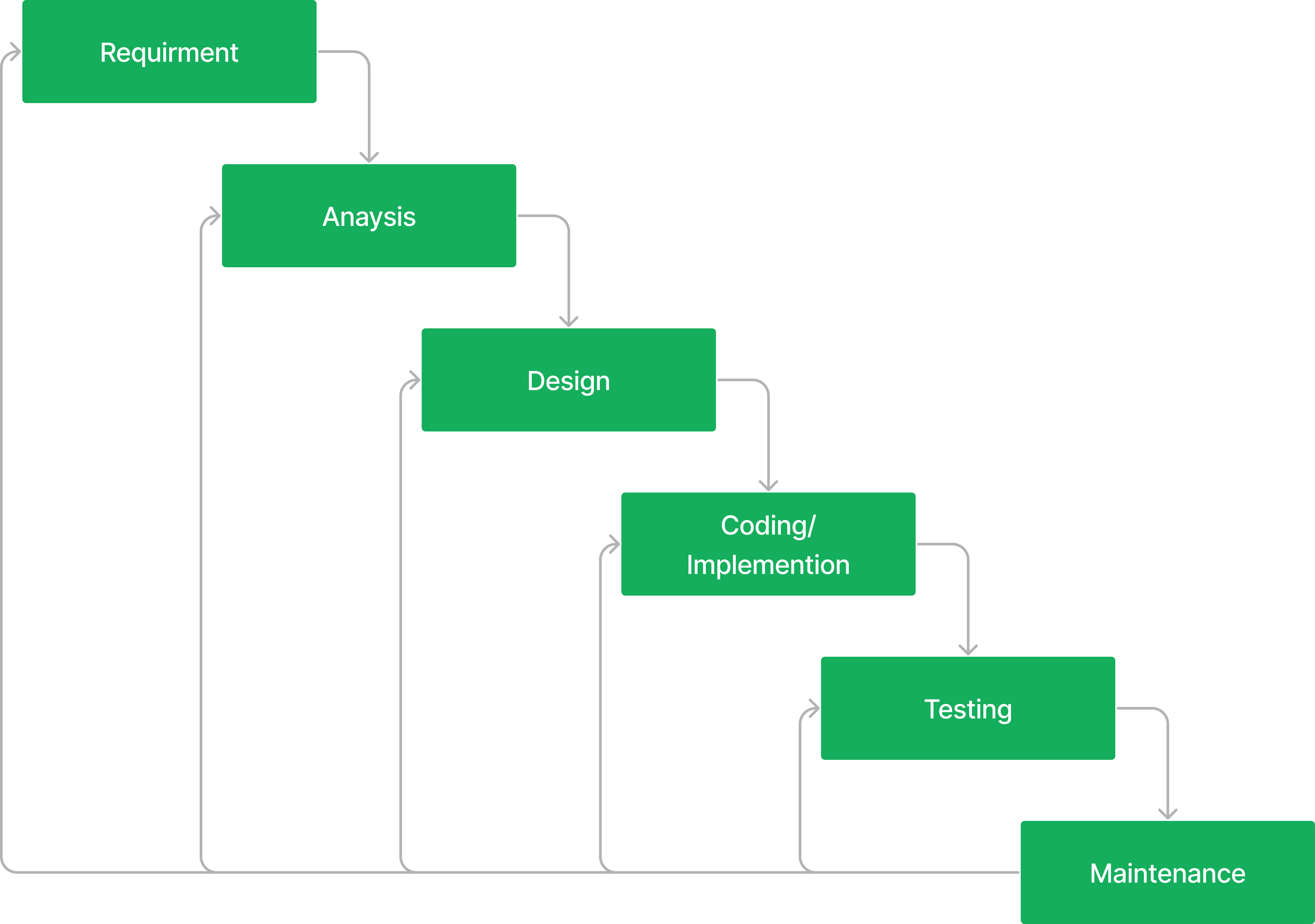
HeidiSQL adalah sebuah DBMS (*Database Management System*) menggunakan perintah SQL (*Structured Query Language*) yang banyak digunakan saat ini dalam pembuatan aplikasi berbasis android. HeidiSQL dibagi menjadi dua lisensi, pertama adalah *FreeSoftware*dimana perangkat lunak dapat diakses oleh siapa saja. Dan kedua adalah *Shareware* dimana perangkat lunak berpemilik memiliki batasan dalam penggunaannya “HeidiSQL adalah salah satu aplikasi DBMS yang sudah sangat banyak digunakan para pemrogram aplikasi. Kelebihan dari HidiSQL adalah gratis, handal, selalu di-update dan banyak forum yang memfasilitasi para pengguna jika memiliki kendala. HeidiSQL juga menjadi DBMS yang sering di bundling dengan web server sehingga proses instalasinya jadi lebih mudah” (Daniel Dido Jantce TJ Sitinjak, Maman, Jaka Suwita, 2020)

### 2.2.6 PHP

PHP (*Hypertext Prepocessor*) merupakan bahasa pemrograman berbasis *server–side* yang dapat melakukan parsing *script* php menjadi *script* web sehingga dari sisi client menghasilkan suatu tampilan yang menarik. PHP adalah PHP (*Hypertext Preprocessor*) adalah bahasa pemrograman web berbasis *server* (*server slide*) yang mampu memparsing kode PHP dari kode dengan ekstensi PHP sehingga menghasilkan tampilan website yang dinamis disisi *client* (Rusi, Iqbal and Febrianto, 2019)*.*

### 2.2.7 Motode SDLC *Waterfall*

SDLC *(Software Development Life Cycle)* merupakan sebuah proses pembuatan dan pengubahan pada system. Yang biasanya Sistem terssbut adalah Sistem Komputer atau Sistem Informasi. Sistem tersebut memeliki tahapan-tahapan yang terstruktur dari perenacanaan, Analisa, Design, Implementasi, Testing & Maintanace.



Gambar 2.2 Model SDLC Waterfall

Model *waterfall* ini memiliki beberapa langkah-langkah terstruktur dalam mengembangan sistem sehingga menjadi produk yang siap pakai oleh pengguna. Pada model ini tahapan yang harus dilalui oleh pengembang yaitu:

Requirement (Perencanaan)

Tahap ini mencakup pengumpulan dan dokumentasi kebutuhan pengguna serta persyaratan sistem. Tujuannya adalah untuk memahami apa yang diinginkan dan dibutuhkan oleh pengguna dari sistem yang akan dibangun.

Analysis (Analisis)

Pada tahap ini, kebutuhan yang telah dikumpulkan dianalisis secara mendalam untuk memastikan kelayakan dan kelengkapannya. Analisis ini juga mencakup identifikasi masalah, peluang, dan batasan yang mungkin muncul selama pengembangan sistem.

Design (Perancangan)

Tahap ini melibatkan perancangan arsitektur sistem dan spesifikasi teknis secara detail. Desain ini mencakup struktur data, antarmuka pengguna, dan alur kerja sistem yang akan diimplementasikan.

Implementation (Implementasi)

Pada tahap ini, desain yang telah dibuat diterjemahkan ke dalam kode program. Pengembang menulis dan mengintegrasikan berbagai komponen perangkat lunak sesuai dengan spesifikasi yang telah ditentukan pada tahap perancangan.

Testing (Pengujian)

Tahap ini bertujuan untuk memastikan bahwa perangkat lunak yang telah diimplementasikan bekerja sesuai dengan kebutuhan dan spesifikasi yang telah ditentukan. Pengujian meliputi verifikasi dan validasi untuk menemukan dan memperbaiki kesalahan atau bug

Maintanance (Pemeliharaan)

Setelah perangkat lunak selesai dan diimplementasikan, tahap ini memastikan bahwa sistem tetap berfungsi dengan baik. Pemeliharaan mencakup perbaikan bug, penyesuaian terhadap perubahan lingkungan, dan peningkatan fitur berdasarkan umpan balik pengguna.(Ridwan, Fitri and Benrahman, 2021)

### 2.2.8 XAMPP

XAMPP adalah perangkat lunak (*free software*) bebas, yang mendukung untuk banyak sistem operasi, yang merupakan kompilasi dari beberapa program. Fungsi XAMPP sendiri adalah sebagai server yang berdiri sendiri (*localhost*), yang terdiri beberapa program antara lain: *Apache HTTP Server*, HeidiSQL database, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl. Nama XAMPP sendiri merupakan singkatan dari X (empat sistem operasi apapun), Apache, HeidiSQL, PHP dan Perl. Program ini tersedia dalam GNU *General Public License* dan bebas, merupakan web server yang mudah untuk digunakan yang dapat menampilkan halaman web yang dinamis. Untuk mendapatkanya XAMPP anda dapat mendownload langsung dari web resminya. Dan berikut beberapa definisi program lainnya yang terdapat dalam XAMPP.

XAMPP ini merupakan project non-profit yang dikembangkan oleh Apache Friends yang didirikan Kai 'Oswalad' Seidler dan Kay Vogelgesang pada tahun 2002, project mereka ini bertujuan mempromosikan pengunaan Apache web server XAMPP adalah paket software yang didalamnya sudah terkandung Web Server Apache, database HeidiSQL, dan PHP interpreter” (Al Zikri, 2021).

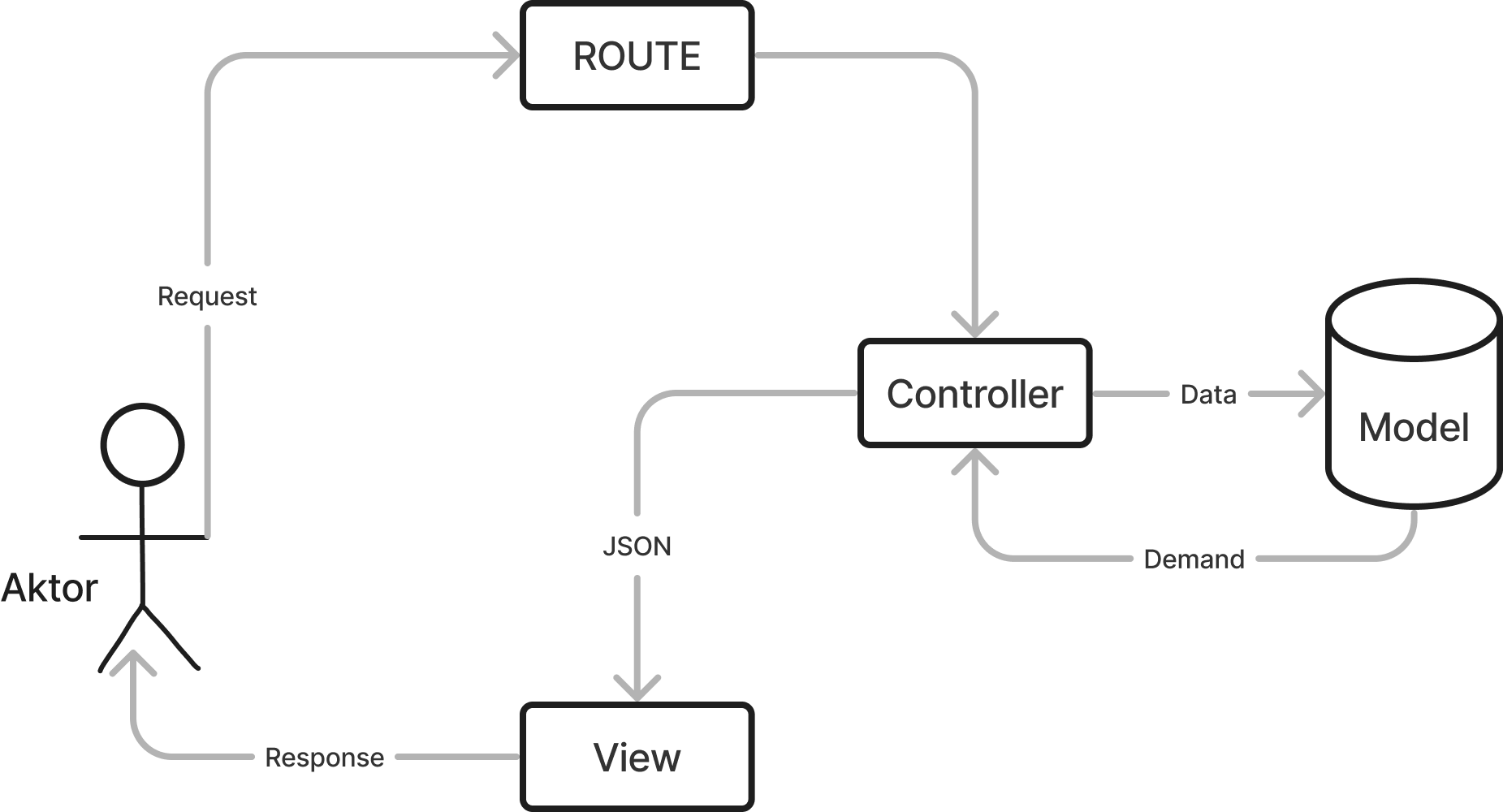
### 2.2.9 Framework

*Web Application Framework* (*WAF*), atau sering disingkat *web* *framework* adalah suatu kumpulan kode berupa pustaka *(library)* dan alat *(tool)* yang dipadukan sedemikian rupa kerangka kerja *(framework)* guna memudahkan dan mempercepat proses pengembangan aplikasi *web*. Proses pengembangan *web* itu sendiri dapat dilakukan dengan beragam bahasa pemrograman, yaitu dengan *PHP, Python, Ruby, Perl, C++, Java,* dan sebagainya. Saat ini, banyak bermunculan *framework* *web* yang dirancang untuk bahasa-bahasa pemrograman tersebut. Diantara bahasa pemrograman yang lain, *framework* untuk *PHP* memiliki varian paling banyak (Arhandi, Arief and Firdausi, 2022).

### 2.2.10 Laravel

Laravel merupakan framework berbasis PHP yang dirilis oleh MIT yang menekankan kesederhanaan dan fleksibilitas pada desainnya. Laravel dibangun dengan konsep MVC (Model-ControllerView), kemudian dilengkapi juga dengan command line tool yang bisa digunakan untuk packaging bundle dan instalasi bundle melalui command prompt.

Laravel adalah kerangka kerja (framework) perangkat lunak sumber terbuka (open-source) yang ditulis dalam bahasa pemrograman PHP. Kerangka kerja ini dikembangkan oleh Taylor Otwell dan dirilis pertama kali pada tahun 2011. Laravel dirancang dengan tujuan menyederhanakan proses pengembangan aplikasi web dengan menyediakan seperangkat alat (tools) dan fitur-fitur yang kuat namun mudah digunakan. Laravel menonjol dengan keunggulan dalam mempermudah pengembangan aplikasi web melalui pendekatan yang menekankan kesederhanaan dan fleksibilitas. Dengan desain yang mengikuti konsep Model-View-Controller (MVC) (Widiarta, Mulyanto and Sutrianto, 2023).



Gambar 2.3 Arsitektur Laravel

Dapat dijelaskan bahwa struktur pola MVC framework Laravel terdapat perbedaan dengan pola MVC pada umumnya. Perbedaan terletak pada route yang menghubungkan antara request dari user dan controller. Sehingga request dari user tidak diterima langsung oleh controller [14]. Setelah membuat model dan controller langkah selanjutnya untuk pengkodean RESTful web service agar dapat di gunakan adalah pembuatan route.(Arianto and Susetyo, 2022)

### 2.2.11 Visual Studio Code

Visual Studio Code merupakan software kode editor lintas platform yang ringan namun kuat yang dapat digunakan oleh siapa saja untuk membangun aplikasi Web. Muncul dengan builtin dukungan untuk JavaScript, naskah dan Node.js dan memiliki array beragam ekstensi yang tersedia untuk bahasa lain, termasuk C ++, C# , Python, dan PHP. Visual Studio Code menggunakan open source NET untuk memberikan dukungan ASP. Visual Studio Code dapat digunakan untuk berbagai bahasa pemrograman seperti JavaScript, HTML, CSS, PHP, Python, C++, dan masih banyak lagi. Visual Studio Code bekerja pada berbagai sistem operasi seperti Windows, macOS, dan Linux (Hariyanto *et al.*, 2021).

### 2.2.12 Figma

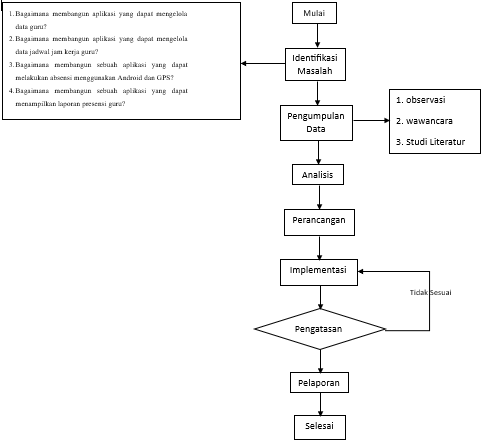
Figma merupakan sebuah platform inovatif yang telah dirancang khusus untuk memfasilitasi proses desain grafis yang komprehensif. Dengan fokus utama pada pembuatan wireframes, prototipe, serta antarmuka pengguna, aplikasi ini telah menjadi pilihan utama bagi para profesional desain untuk menghasilkan produk digital yang memukau, baik itu aplikasi smartphone maupun situs web. Keunggulan utama dari Figma terletak pada kemampuannya untuk diakses melalui browser, memungkinkan kolaborasi yang mudah antara tim desain yang tersebar geografis. Dengan demikian, Figma tidak hanya menjadi alat desain yang andal, tetapi juga menjadi platform yang mempercepat proses pengembangan produk digital dengan kualitas yang optimal (Wijaya, 2021).

### 2.2.13 Pengujian Blackbox

Black box adalah suatu konsep yang mengacu pada suatu sistem atau perangkat yang dianggap sebagai entitas tunggal yang menerima input dan menghasilkan output tanpa memberikan penjelasan detail tentang bagaimana proses internalnya bekerja. Dalam konteks ini, meskipun input dan output diamati, pengetahuan tentang mekanisme kerja internal dari black box tidak tersedia atau tidak diungkapkan kepada pengguna eksternal. Dengan kata lain, black box berfungsi sebagai suatu "kotak hitam" di mana pengguna hanya dapat mengamati apa yang dimasukkan ke dalamnya dan apa yang keluar darinya, tanpa mengetahui secara spesifik bagaimana proses di dalamnya terjadi atau bekerja (Titi Komalasari, 2021).

# BAB III ANALISIS DAN PERANCANGAN

## 3.1 Kerangka Pikir



Gambar 3.1 Kerangka Fikir

## 3.2 Deskripsi

Deskripsi adalah prinsip yang terlibat dalam proses mengubah data menjadi format yang dapat dijelaskan dan disajikan dengan jelas dan akurat untuk mencapai tujuan tertentu. Hal ini memungkinkan informasi tersebut dapat dipahami dan dimengerti oleh pembaca dengan baik.

### 3.1.1 Identifikasi Masalah

Tahap pertama dalam merancang aplikasi presensi Guru dalam kerangka pikir penulis adalah mengidentifikasi masalah atau menetapkan perumusan masalah. Dalam hal ini penulis meminta izin kepada pihak staf tata usaha untuk melakukan penelitian, serta melakukan observasi dan wawancara guna memahami masalah yang dihadapi oleh bagian sarana prasarana. Permasalahan yang terdeteksi di MA Nurul Pacet khususnya di bidang dokumentasi kehadiran adalah penggunaan metode masih dilakukan secara manual menggunakan tulis tangan, menyebabkan kesulitan dalam pencarian arsip data absensi. Terjadi ketidak sesuaian antara kebutuhan dan implementasi presensi karena terkendala oleh waktu.

### 3.1.2 Pengumpulan Data

1. Observasi

Metode pengambilan data melibatkan pengamatan langsung terhadap sistem pengolahan data yang telah ada di MA Nurul Huda Pacet khususnya di bidang sarana dokumentasi kehadiran, dengan tujuan mendapatkan informasi yang tepat dan akurat yang diperlukan untuk pengembangan aplikasi pengolahan data baru. Melalui metode observasi ini, peluang diberikan untuk mengdapatkan pemahaman mendalam tentang praktik yang dilakukan secara nyata, memungkinkan analisis rinci tentang aspek-aspek yang relevan.

1. Wawancara (*Interview*)

Metode Pengumpulan Data melibatkan interaksi langsung melalui wawancara dengan petugas sarana prasarana yang berkaitan dengan topik penelitian. Penulis melakukan serangkaian pertanyaan kepada petugas absensi untuk mengetahui kesulitan dan kebutuhan apa saja yang diperlukan.

1. Studi Pustaka

Pengumpulan data pada penelitian ini berkaitan dengan analisis dari berbagai tulisan dalam jurnal ilmiah dan berkaitan dengan topik penelitian yang penulis ambil. Teknik pada pengumpulan data dilakukan dengan mencari referensi dan jurnal-jurnal. Dalam penelitian ini penulis menggunakan 3 jurnal yang terdiri dari Yusuf Wahyu Satiya Putra (2022) dengan judul “Sistem Presensi Online Menggunakan Teknologi Face Recognition Dan GPS” menggunakan metode Waterfall. Berikutnya penelitian dari Aji Afriansyah (2022) dengan judul Perancangan Sistem Informasi Absensi Dewan Guru Tenaga Harian Lepas Berbasis WEB Pada Sekolah Dasar Negeri Kunciran 6 Kota Tangerang. Dan Terakhir penelitian dari Resti nadia putri, Anggita Rahmawati, dan Diva Iftidiani (2023) dengan judul “ Evaluasi Penerapan Inovasi Presensi Guru berbasis Online di Sekolah Dasar”.

## 3.1.3 Analisis Kebutuhan

Tahap selanjutnya adalah merancang dan merencanakan aplikasi presensi guru yang dimulai dengan analisis kebutuhan perangkat lunak dan perangkat keras yang diperlukan untuk pengembangan aplikasi presensi guru. Pada tahap ini, dilakukan pengumpulan kebutuhan elemen-elemen pada tingkat perangkat lunak.

### 3.1.3 Perancangan

Pada tahap ini penulis diharapkan untuk merumuskan rencana detail terkait pengembangan aplikasi presensi guru. Rencana ini mencakup berbagai tahapan dalam proses pengembangan serta aspek perancangan yang melibatkan:

1. Perancangan Diagram

Pada tahap ini dilakukan dengan membuat desain menggunakan bahasa pemodelan *Unified Modelling Language* (UML). UML yang digunakan dalam penelitian terdiri dari:

1. *Use case diagram*  digunakan untuk menggambarkan secara umum hubungan antara aktor dan kasus pengguna dalam suatu sistem aplikasi. Diagram ini membantuk dalam pemahaman tentang interaksi antara aktor (entitas yang berinteraksi dengan sistem) dan berbagai fungsi (*use case*) yang ada dalam sistem tersebut. Dalam konteks ini, terdapat satu aktor yaitu “Admin” dan kasus pengguna (*use case*) yang mencakup:

* Aktor User

1. Halaman *Login*: Halaman ini digunakan untuk masuk ke dalam sistem dengan menggunakan kredensial yang valid, seperti nama pengguna dan kata sandi, sehingga pengguna dapat mengakses fitur-fitur yang terkait.
2. Menu Beranda: Menu ini membawa pengguna kembali ke halaman utama atau beranda sistem, di mana mereka dapat melihat fitur-fitur utama lainnya.
3. Menu Absensi: Menu ini digunakan untuk melakukan proses pencatatan kehadiran seperti absen masuk (*check-in*) dan absen keluar (*check-out*).
4. Menu Perizinan: Menu ini terletak dibagan menu presensi digunakan aktor user untuk melakukan perizinan, seperti cuti, izin dan sakit.

* Aktor Admin

1. Menu Data Absensi: membawa aktor admin melihat aktivitas kegiatan presensi ya sedang berjalan.
2. Menu Daftar Data Admin: Menu ini memungkinkan admin untuk menambah jumlah admin pada aplikasi presensi guru, serta mengatur hak akses aplikasi mobile pada ponsel pengguna agar menjadi status kredensial yang valid.
3. Menu Data Guru: Menu ini digunakan untuk menyimpan data absensi guru, admin dapat mengedit data guru agar menjadi status kredensial yang valid.
4. *Activity diagram* digunakan untuk memodelkan rangkain aktivitas dalam suatu sistem, mencakup representasi visual dari urutan aktivitas yang dilakukan oleh pengguna atau entitas lain dalam aplikasi. Diagram ini memberikan pandangan keseluruhan tentang bagaimana aktivitas berlangsung, berinteraksi, dan mengalir dalam sistem. Penulis akan membuat beberapa diagram aktivitas yang berkaitan dengan sistem aplikasi diantaranya:

1. User

1. Proses akses *register*
2. Proses akses *login*
3. Proses akses menu beranda
4. Proses akses absensi
5. Proses akses menu izin

2. Admin

1. Proses akses *login*
2. Proses akses data presensi
3. Proses akses data admin
4. Proses akses data admin
   * 1. *User*
        1. *Activity Diagram Register*
        2. *Activity Diagram login*
        3. *Activity Diagram* menu beranda
        4. *Activity Diagram* menu absensi
        5. *Activity Diagram* menu Izin
     2. *Admin*
        1. *Activity Diagram login*
        2. *Activity Diagram* data presensi
        3. *Activity Diagram* kelola data admin
        4. *Activity Diagram* kelola data guru
5. *Class diagram*, untuk menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. *Class diagram*  yang akan dibuat adalah sebagai berikut:
6. User
7. *Class Diagram Register*
8. *Class Diagram* *Login*
9. *Class Diagram* Beranda
10. *Class Diagram* Absensi
11. *Class Diagram* Perizinan
12. admin
13. *Class diagram* data presensi
14. *Class diagram* data admin
15. *Class diagram* data guru
16. Perancangan *Interface*

Pada tahapan perancangan antarmuka, desain aplikasi dibuat sederhana menggunakan Figma, namun tetap mampu mengilustrasikan aplikasi yang dibangun dan memperhatikan interaksi antara pengguna dan sistem. Dalam proses ini, desain aplikasi dibuat dengan menggunakan Figma yang lebih canggih, tetapi masih mampu menggambarkan fungsionalitas aplikasi serta menunjukkan bagaimana interaksi antara pengguna dan sistem akan terjadi di dalamnya.

### 3.1.4 Implementasi

Setelah melakukan perancangan diagram dan antarmuka, tahap selanjutnya adalah melakukan implementasi perancangan yang telah dibuat selanjutnya dengan cara menerjemahkan kedalam bentuk pengkodean secara nyata memanfaatkan bahasa pemrograman PHP, *Visual Studio Code* (VS Code) dan database menggunakan MySQL *management* database serta *framework Laravel*.

### 3.1.5 Pengujian

Pada tahap ini, pengujian dilakukan untuk memastikan bahwa hasilnya sesuai dengan desain yang diinginkan. Jika terdapat masalah, koreksi dilakukan kembali pada tahap implementasi hingga semua berjalan dengan baik sesuai yang diharapkan. Pengujian sistem menggunakan metode Black Box Testing, sementara pengujian perangkat menggunakan metode fungsionalitas, kinerja, dan keamanan. Apabila terdapat kekurangan, kembali ke tahap implementasi untuk diperbaiki sehingga sesuai dengan desain yang diinginkan.

### 3.1.6 Pelaporan

Langkah terakhir adalah menyusun laporan skripsi yang merupakan satu-satunya komponen yang dinilai sesuai dengan kriteria yang telah ditetapkan dalam formulir pengajuan proposal, dan juga merupakan syarat untuk mencapai kelulusan di Fakultas Teknologi dan Informasi Universitas Bale Bandung. Laporan skripsi ini dipersiapkan sesuai dengan panduan penulisan skripsi yang diberikan oleh fakultas. Laporan ini adalah hasil akhir dari proses penelitian yang menunjukkan kualitas ilmiah dan ketaatan terhadap standar yang telah ditetapkan dalam panduan tersebut.

# BAB IV ANALISIS DAN PERANCANGAN

## 4.1 Analisis

Pada tahap ini, analisis yang dilakukan dalam proses penelitian ini meliputi beberapa tahapan yaitu Analisis masalah, analisis software, analisis pengguna, user interface, fitur-fitur, analisis data serta analisis biaya.

### 4.1.1 Analisis Masalah

Berdasarkan observasi dan wawancara, Permasalahan yang terdeteksi di MA Nurul Pacet khususnya di bidang dokumentasi kehadiran adalah penggunaan metode masih dilakukan secara manual menggunakan tulis tangan, menyebabkan kesulitan dalam pencarian data absensi. Terjadi ketidak sesuaian antara kebutuhan dan implementasi presensi karena terkendala oleh waktu.

### 4.1.2 Analisis Software

Berdasarkan analisis kebutuhan maka untuk memenuhi kebutuhan dalam pengembangan sistem informasi presensi guru memerlukan software sebagai alat penunjang pengembangan sistem informasi absensi guru. Adapun software yang dibutuhkan sebagai alat penunjang pada penelitian ini adalah sebagai berikut:

1. Sistem operasi windows 10
2. MySQL dengan penggunaan *tools* HeidiSQL untuk management *Database* dengan pengaplikasiaan yg lebih mudah dan cepat.
3. Visual Studio Code sebagai text editor digunakan untuk melakukan pengkodean dalam pembuatan aplikasi presensi guru
4. XAMPP digunakan untuk web server
5. Web browser digunakan untuk melihat tampilan sistem yang sudah dibuat
6. Draw.Io digunakan untuk membuat/merancang diagram UML
7. Figma digunakan untuk membuat rancangan antarmuka sistem informasi Presensi guru

### 4.1.3 Analisis Pengguna

Analisis pengguna adalah yang berkaitan dengan yang akan memakai sistem informasi presensi. Terdapat dua kelompok pengguna utama yaitu bagian staf tata usaha sebagai admin dan pegawai sekolah, khususnya guru dan jajarannya, sebagai user. Bagian staf tata usaha absensi bertanggung jawab atas pengelolaan presensi di sekolah dengan akses penuh terhadap sistem. Bagian staff tata usaha bagian absensi, mengedit, dan menghapus data presensi, memantau status presensi secara real-time, serta membuat laporan untuk keperluan audit dan administrasi. Selain itu, bagian staf tatausaha bagian absensi juga mengatur data pegawai, laporan presensi pegawai, dan mengelola hak akses pengguna lain, sehingga memastikan data presensi selalu terbaru dan akurat, serta meminimalkan kesalahan dan duplikasi data.

Masyarakat sekolah, khususnya guru dan staf, memiliki kebutuhan untuk mengakses informasi mengenai data diri dan *history* presensi. Mereka dapat melihat data diri, mengajukan permohonan izin seperti izin, cuti, dan sakit. Dengan adanya sistem ini, diharapkan proses presensi guru menjadi lebih cepat dan mudah, serta memberikan umpan balik yang berguna bagi admin dalam menjaga dan merawat sistem presesni. Secara keseluruhan, sistem ini bertujuan untuk meningkatkan efisiensi, transparansi, dan akurasi dalam pengelolaan presensi, sehingga mendukung kelancaran kegiatan belajar mengajar di sekolah.

### 4.1.4 User Interface

User Interface dari aplikasi presensi guru sangat berpengaruh untuk kenyamanan user saat menggunakannya. Hal ini mencakup perangkat yang dibuat ditunjukan untuk menghasilkan informasi yang cepat, maka user interface juga harus menyesuaikan dengan kebutuhan user berupa tampilan yang mudah dipahami. Berikut adalah user Interface aplikasi presensi guru berbasis android, meliputi:

1. User Interface User
2. User interface Register
3. User interface Halaman Login
4. User interface Halaman Beranda
5. User interface Halaman Data Absensi User
6. User interface Halaman Perizinan
7. User Interface Admin
8. User Interface Halaman Register
9. User Interface Halaman Login
10. User Interface Halaman Beranda
11. User Interface Halaman Data Presensi
12. User Interface Halaman Data Admin
13. User Interface Halaman Data Guru

### 4.1.5 Fitur-fitur

Fitur-fitur yang digunakan dalam aplikasi presensi guru ini dibuat agar user dapat dengan mudah untuk mendapatkan informasi dan mengetahui bagaimana melakukan proses absensi. Berikut adalah fitur-fitur yang disediakan aplikasi presensi guru:

1. User
2. Halaman Register berfungsi untuk menambah user pada apliksai presensi guru untuk bisa mengakses login dan masuk ke halaman branda aplikasi.
3. Halaman Login berfungsi untuk validasi akses yang akan menampilkan email dan password sebagai prosedur validasi akses masuk menuju ke halaman beranda.
4. Halaman Beranda berfungsi untuk menampilkan ringkasan informasi penting seperti tanggal dan waktu, *clock-in dan clock-out* foto, informasi data diri pengguna dan menu izin. Menyediakan akses cepat ke menu-menu lain.
5. Halaman Absensi berfungsi untuk melakukan proses absensi pada user dengan melakukan absen *clock-in dan clock-out* foto.
6. Halaman perizinan berfungsi untuk user melakukan perizinan terkait kegiatan di sekolah, pada menu perizinan akan menampilkan beberpa pilihan jenis izin di antaranya izin, cuti, dan sakit dengan tambahan data keterangan.
7. Admin
8. Halaman Informasi Data Presensi berfungsi untuk memonitoring aktivitas presensi user (guru dan staf). Mengatur hak akses dan peran masing-masing pengguna, serta menyimpan informasi status kehadiran pegawai.
9. Halaman Informasi Data Admin untuk menambahkan admin dan mengedit data pengguna seperti user admin. Mengatur hak akses dan peran masing-masing pengguna, serta menyimpan informasi kontak dan identitas admin.
10. Halaman Informasi Data Guru berfungsi untuk menyimpan informasi data guru yang terverifikasi dalam aplikasi presensi.

### 4.1.6 Analisis Data

Untuk mendukung perancangan aplikasi presensi guru, penulis memerlukan beberapa data sebagai bahan penelitian dan perancangan aplikasi. Adapun data-data yang digunakan dan diolah pada aplikasi presensi guru ini adalah sebagai berikut:

1. Data Presensi

Analisis data Presensi akan memberikan detail mengenai informasi data diri pegawai seperti nama, tipe, status, dan tanggal, saat selesai melakukan absensi.

1. Data Pengguna (*All User)*

Analisis data *All User* akan menghasilkan informasi mengenai data pegawai, seperti nama, nomer identitas pegawai, dan alamat email. Informasi ini ditujukan untuk admin atau bagian staf presensi sekolah, yang bertugas memantau dan mengelola data pegawai. Dengan data ini, admin dapat mengetahui status pegawai sesuai data resmi di sekolah.

1. Data Perizinan

Analisis data Perizinan mencakup informasi detail nama user, type izin, status, tanggal izin, dan catatan keterangan izin.

### 4.1.7 Analisis Biaya

Pada penelitian kali ini ada beberapa rincian biaya yang dibutuhkan dalam proses pengerjaan penelitian diantaranya adalah:

Tabel 4.1 Analisis Biaya

|  |  |  |
| --- | --- | --- |
| No | Jenis Kebutuhan | Biaya |
| 1 | Biaya ATK | Rp. 550.000 |
| 2 | Internet | Rp. 500.000 |
| 3 | *Programming* | Rp. 1.200.000 |
| 4 | Biaya Bahan Bakar | Rp. 1.200.000 |
| Jumlah | | Rp. 3.450.000 |

## 4.2 Perancangan

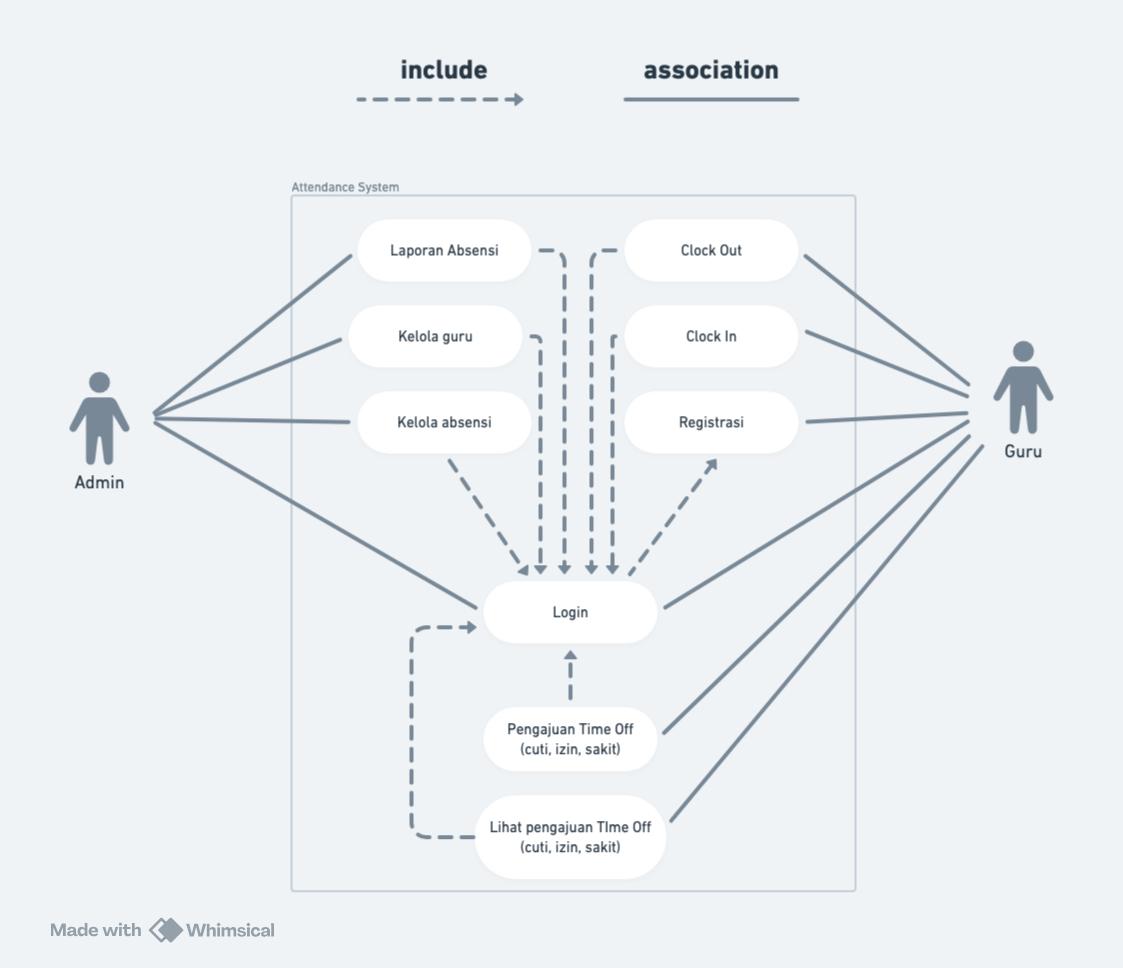
Sebelum masuk dalam pembuatan aplikasi dibutuhkan perancangan yang bertujuan untuk memudahkan pengguna agar mendapatkan informasi mengenai aplikasi presensi guru berbasis android yang diantaranya yaitu merancang software dalam bentuk Unified Modelling Language (UML) yang terdiri dari use case, activity diagram, dan class diagram.

### 4.2.1 Unified Modeling Language (UML)

1. User Case Diagram

Pada rancangan Use case diagram bertujuan untuk memberikan gambaran interaksi aktor dengan aplikasi presensi guru berbasis android. Berikut adalah penjelasan Use Case diagram aplikasi presensi guru berbasis android:

1. Use case Diagram



Gambar 4.1 Use case Diagram

Penjelasan dari gambar Use case diagram di atas dapat dijelaskan dalam tabel deskripsi berikut:

1. Deskripsi Aktor

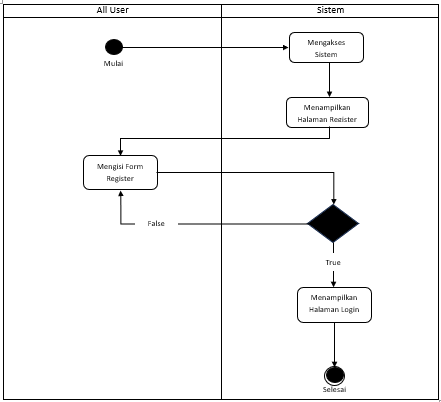
Tabel 4.2 Deskripsi Aktor

|  |  |  |
| --- | --- | --- |
| No | Aktor | Deskripsi |
| 1. | Admin | Admin merupakan pengguna yang dapat melakukan berbagai operasi pada aplikasi Presensi Guru berupa melihat aktivitas presensi yang sedang berjalan, menambah data pegawai, mengubah data pegawai, menghapus data pegawai serta mencetak laporan presensi data pegawai. |
| 2. | Pengguna/pegawai | Pengguna dapat melihat seluruh informasi yang ditampilkan pada halaman aplikasi dengan mengakses absensi yang ada pada halaman aplikasi. |

2. Activity Diagram

Activity Diagram menggambarkan urutan aktivitas proses pada sebuah sistem. Berikut adalah activity diagram pada aplikas presensi guru.

1. Activity Diagram Register (*all user*)

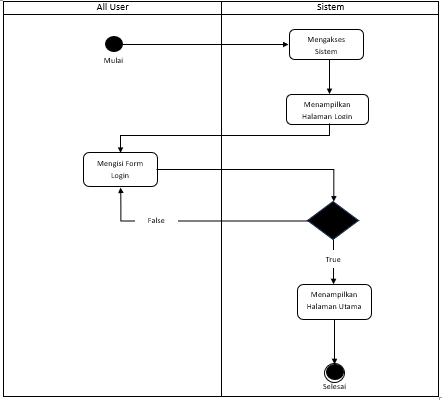


Gambar 4.2 Activity Diagram Register

Pada gambar 4.2 menunjukan bahwa semua user akan melakukan register pada aplikasi dengan memasukan username, alamat email user dan kata sandi (password minimal berisi delapan karakter), kemudian sistem akan melakukan validasi terhadap input yang sudah dimasukan.

1. Activity Diagram Login (*all user*)

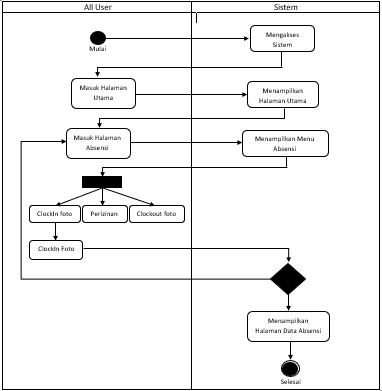
Pada gambar 4.3 menjelaskan bahwa user yang sudah melakukan login setelah sistem melakukan validasi terhadap input yang dimasukan, apabila sesuai maka akan langsung menampilkan halaman utama.



Gambar 4.3 Activity Diagram Login

1. Activity Diagram Data Absensi

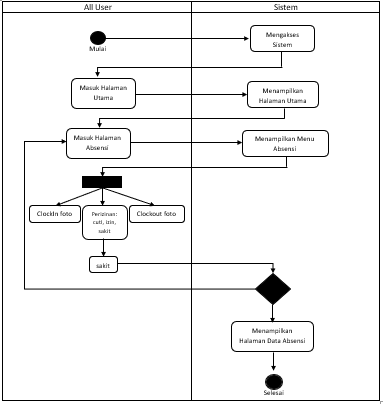
Pada gambar 4.4 menjelaskan bahwa admin dapat mengelola data absensi. Dalam modul ini, admin dapat menambah, memperbarui, atau menghapus informasi tentang user yang terkait dengan aplikasi presensis.



Gambar 4.4 Activity Diagram Data Absensi

1. Activity Diagram data Perizinan

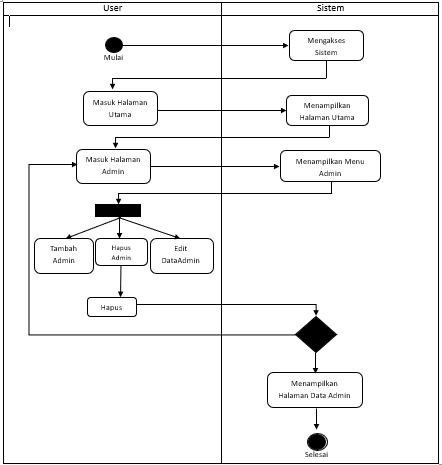
Pada gambar 4.5 menjelaskan bahwa user memiliki akses untuk perizinan. User dapat memasukkan data perizinan, yang terdiri dari tiga golongan yaitu cuti, izin, sakit. maka sistem akan menampilkan keterangan izin yang dipilih user dan menampilkannya pada halaman data absensi.



Gambar 4.5 Activity Diagram Data Perizinan

1. Activity Diagram data admin

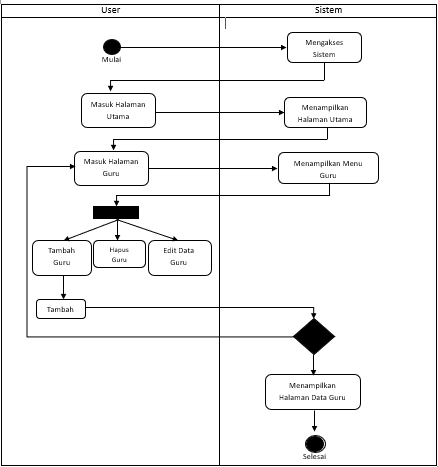
Pada gambar 4.6 menjelaskan bahwa admin dapat mengelola data admin itu sendiri. Ini mencakup penambahan admin baru, pembaruan informasi data admin yang ada, atau penghapusan admin yang tidak lagi memiliki wewenang terhadap akses admin aplikasi presensi guru.



Gambar 4.6 Activity Diagram Data Admin

1. Activity Diagram data guru

Pada gambar 4.7 menjelaskan bahwa admin dapat melihat dan mengelola data admin yang sudah di input sebelumnya. admin dapat melakukan pembaruan atau menghapus data guru sesuai kebutuhan.



Gambar 4.7 Activity Diagram Data Guru

1. *Class* Diagram



Gambar 4.8 *Class Diagram*

Pada gambar 4.8 Class diagram aplikasi presensi guru yang dibangun.

### 4.2.1 Struktur Tabel

Tabel-tabel yang terdapat dalam basis data yang digunakan dalam aplikasi presensi guru adalah sebagai berikut:

1. Perancangan tabel pengguna

Tabel 4.3 Tabel Pengguna

| **Field** | **Type** | **Size** | **Indeks** | **Deskripsi** |
| --- | --- | --- | --- | --- |
| id | integer | 20 | *Primary key* | Id unik pengguna |
| role | varchar | 255 | - | Peran pengguna |
| name | varchar | 255 | - | Nama pengguna |
| email | email | 255 | - | Email pengguna |
| email\_verified\_at | date\_time | timestamp | - | Waktu verifikasi email |
| password | text | 255 | - | Kata sandi pengguna |
| deleted\_at | date\_time | timestamp | - | Waktu penghapusan data |
| created\_at | date\_time | timestamp | - | Waktu pembuatan data |
| updated\_at | date\_time | timestamp | - | Waktu pembaruan terakhir |

1. Perancangan tabel absensi

*Tabel 4.4 Tabel user*

| **Field** | **Type** | **Size** | **Indeks** | **Deskripsi** |
| --- | --- | --- | --- | --- |
| id | integer | 20 | *Primary key* | Id unik user |
| location\_id | integer | 10 | - | Id lokasi item |
| name\_id | integer | 10 | *Foreign key* | Id name item |
| user\_id | integer | 10 | *Foreign key* | Id user item |
| user\_id | varchar | 255 |  | Nomor user |
| attendence\_index | integer | 20 | - | Indeks urutan absensi |
| name | varchar | 255 | - | Nama item |
| reg\_number | integer | 20 | - | Nomor registrasi item |
| Attendnce\_at | date\_time | timestamp | - | Tanggal Absensi item |
| source | varchar | 255 | - | Sumber item |
| status | varchar | 255 | - | Status item |
| deleted\_at | date\_time | timestamp | - | Waktu penghapusan data |
| created\_at | date\_time | timestamp | - | Waktu pembuatan data |
| updated\_at | date\_time | timestamp | - | Waktu pembaruan terakhir |

1. Perancangan tabel admin

Tabel 4.5 Tabel admin

| **Field** | **Type** | **Size** | **Indeks** | **Deskripsi** |
| --- | --- | --- | --- | --- |
| id | integer | 20 | *Primary key* | Id unik lokasi |
| code | varchar | 255 | - | Kode lokasi |
| parent\_id | integer | 10 | *Foreign key* | Id lokasi induk |
| name | varchar | 255 | - | Nama lokasi |
| order\_index | integer | 20 | - | Indeks urutan lokasi |
| deleted\_at | date\_time | timestamp | - | Waktu penghapusan data |
| created\_at | date\_time | timestamp | - | Waktu pembuatan data |
| updated\_at | date\_time | timestamp | - | Waktu pembaruan terakhir |

1. Perancangan tabel perizinan

Tabel 4.6 Tabel perizinan

| **Field** | **Type** | **Size** | **Indeks** | **Deskripsi** |
| --- | --- | --- | --- | --- |
| id | integer | 20 | *Primary key* | Id unik izin |
| user\_id | integer | 10 | *Foreign key* | Id user |
| parent\_id | integer | 10 | *Foreign key* | Id produk induk |
| code | varchar | 250 | - | Kode user |
| name | varchar | 250 | - | Nama user |
| status | varchar | 250 | - | Status user |
| deleted\_at | date\_time | timestamp | - | Waktu penghapusan data |
| created\_at | date\_time | timestamp | - | Waktu pembuatan data |
| updated\_at | date\_time | timestamp | - | Waktu pembaruan terakhir |

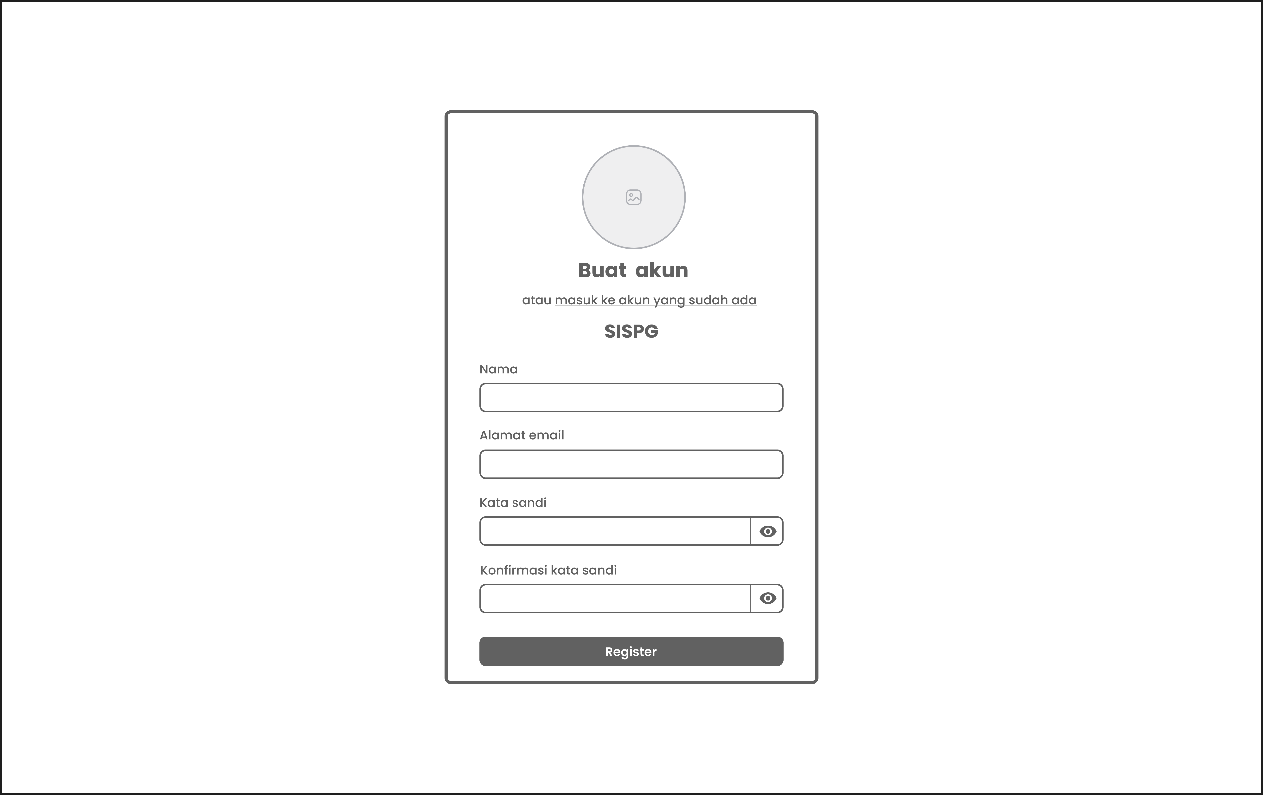
### 4.2.2 Desain Sistem

Sebelum memasuki menu aplikasi presensi guru, admin/user diharuskan untuk melakukan login dengan memasukan Username dan Password aplikasi presensi guru.

Setelah admin/user mengisi form login dengan benar maka tampilan yang pertama kali muncul yaitu halaman dashboard sebagai halaman utama, pada halaman dashboard aplikasi presensi guru.

1. User interface halaman register

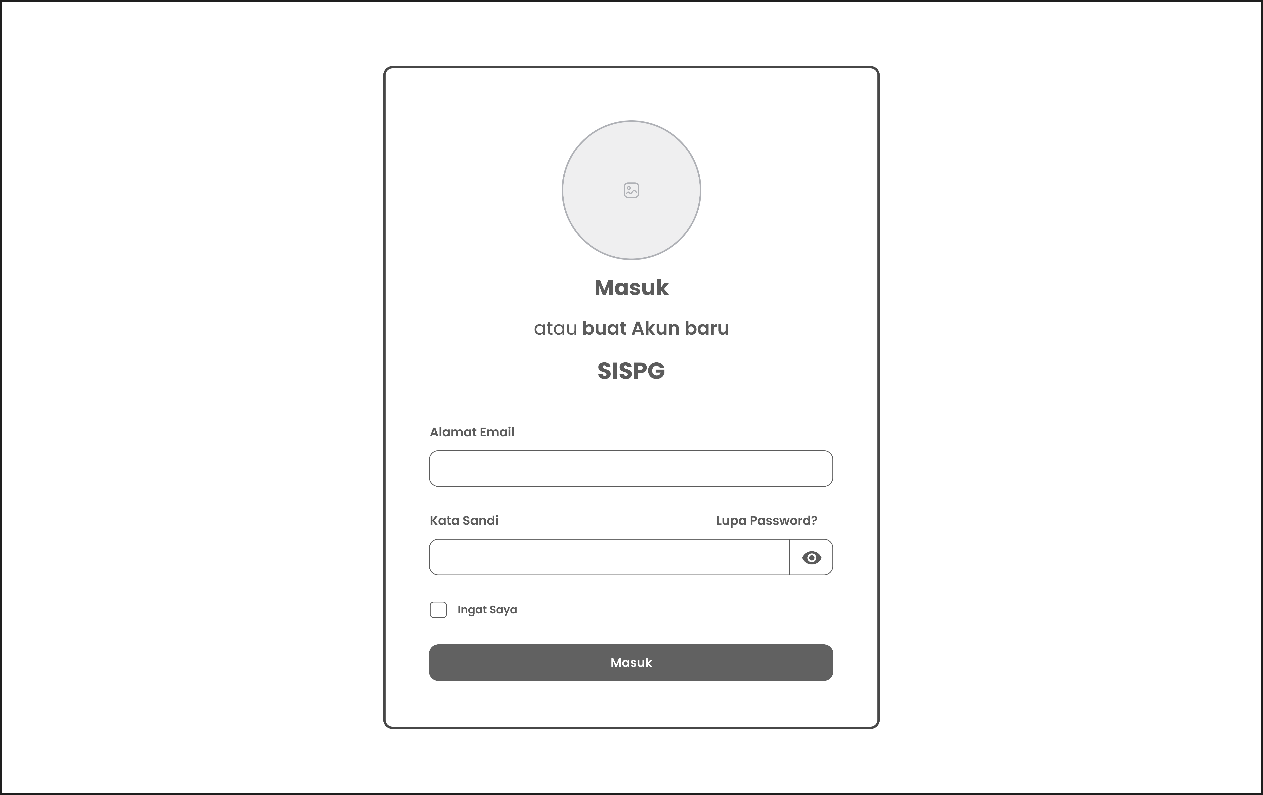
Sebelum memasuki menu aplikasi presensi guru, admin atau user diharuskan untuk melakukan register dengan memasukan Username dan Password aplikasi presensi guru.



Gambar 4.9 User Interface Halaman Register

1. User interface halaman login

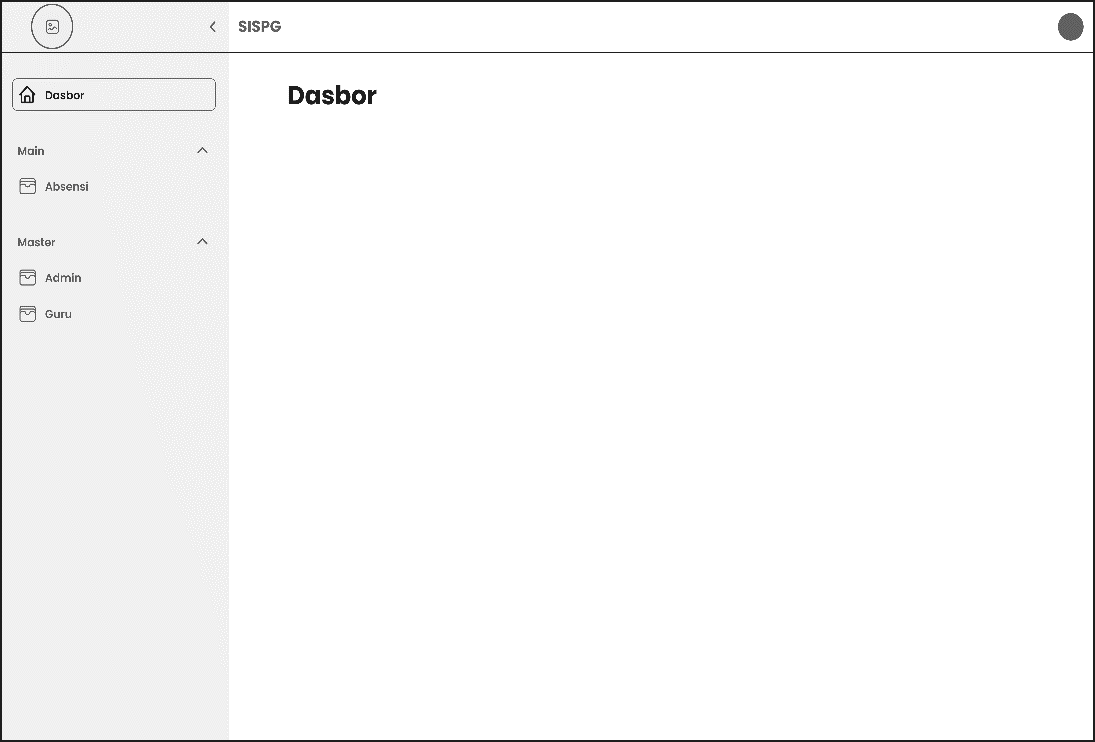
Setelah admin mengisi form register dengan benar maka tampilan yang pertama kali muncul yaitu halaman login sebagai halaman akses utama menuju dashboard aplikasi presensi guru.



Gambar 4.10 *User Interface* Halaman Login

1. User interface halaman *dashboard*

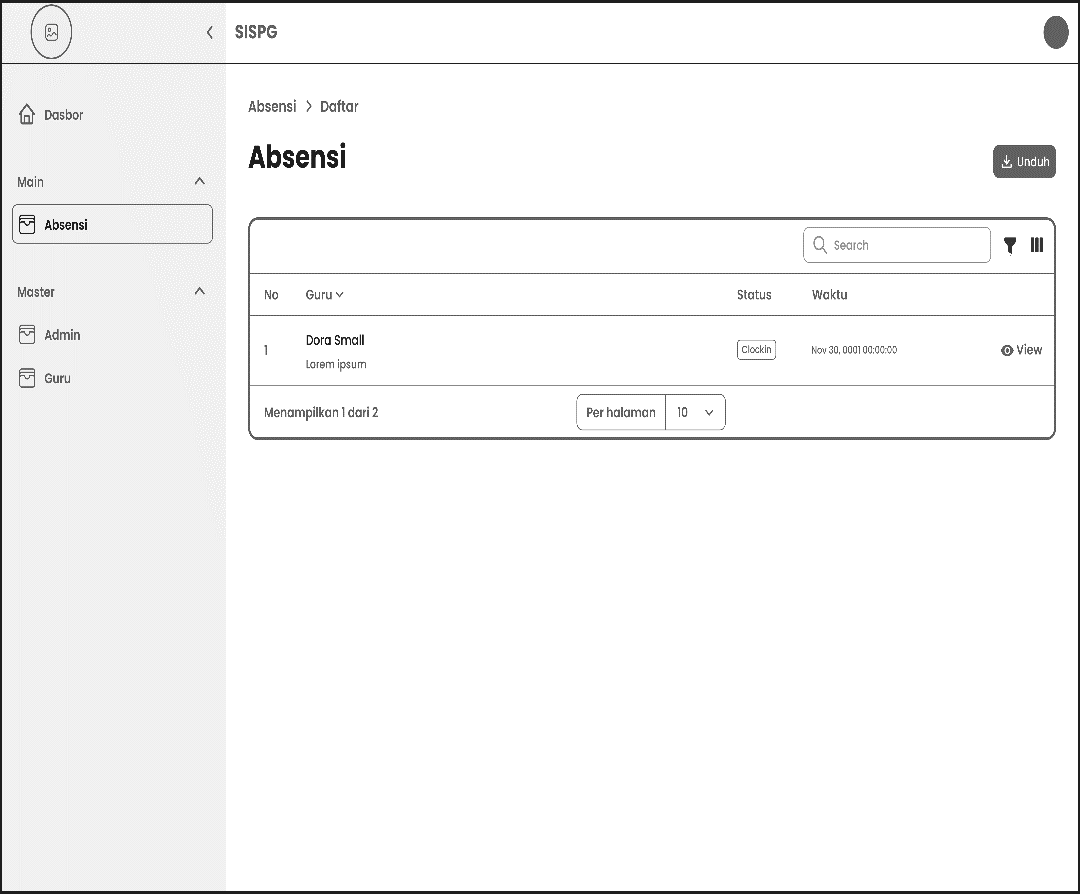
Pada desain halaman *dashborard* admin dapat menampilkan akses tampilan menu absensi, admin dan menu guru.



Gambar 4.11 User Interface Halaman Dashboard

1. User interface halaman absensi

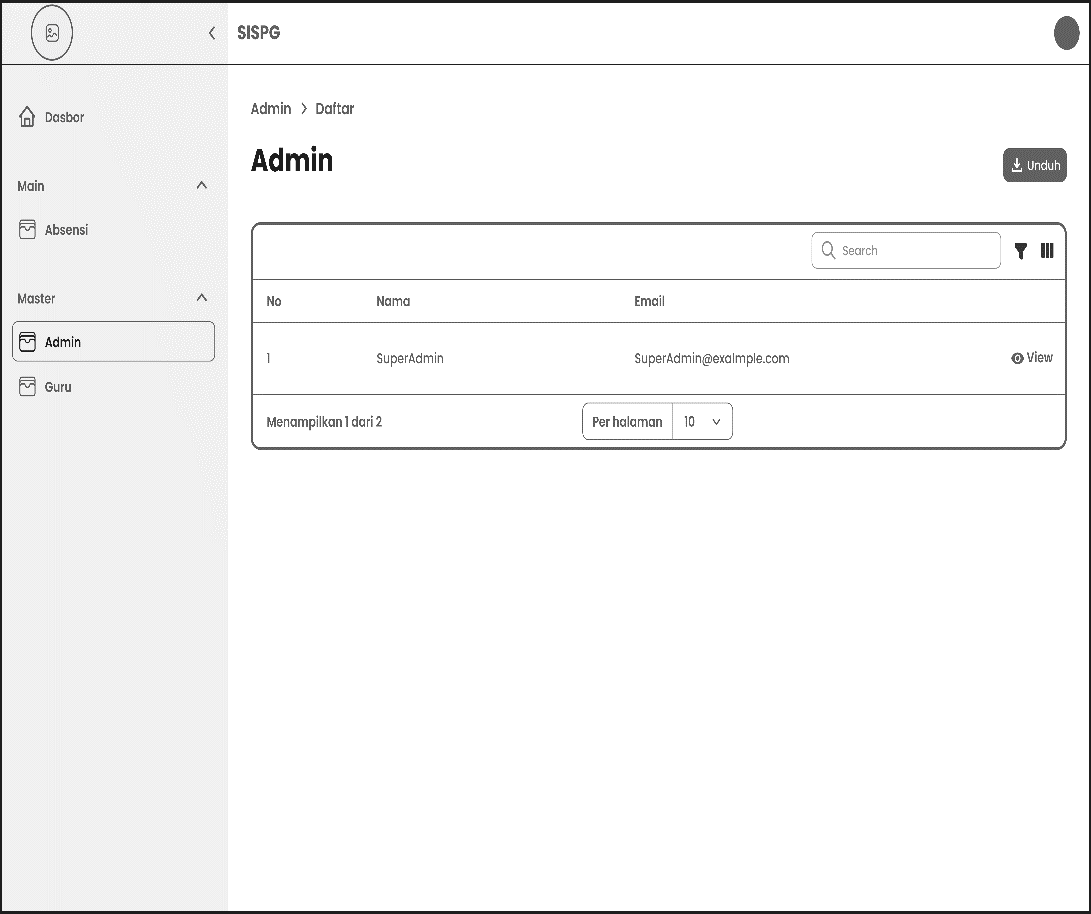
Pada desain halaman absensi admin dapat melakukan tinjauan aktivitas pencatatan absensi yang sedang berjalan seperti di bawah ini.



Gambar 4. 12 User Interface Halaman Absensi

1. User interface halaman admin

Pada desain halaman guru, admin dapat menambah, mengedit, menambah keanggotaan baru dan menghapus data sesuai kebutuhan sekolah.

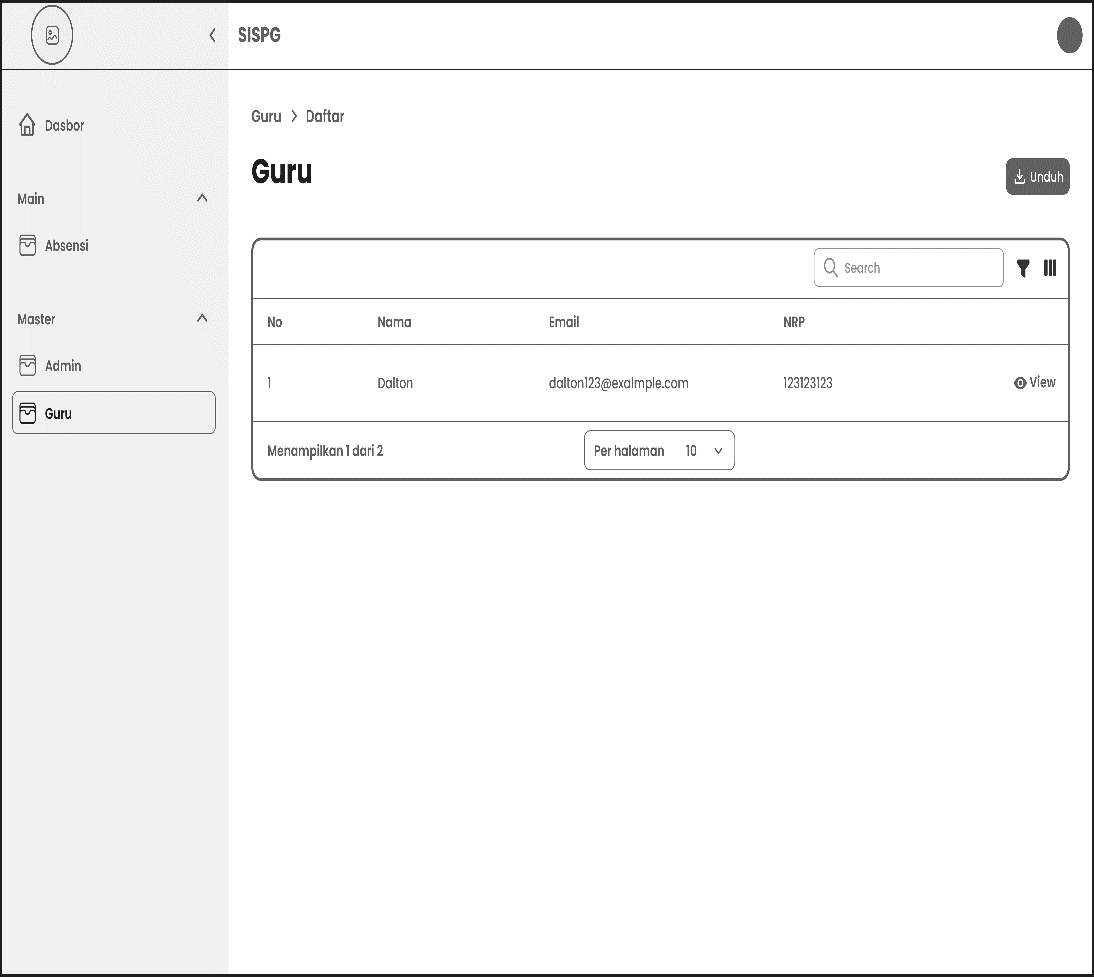


Gambar 4. 13 User Interface Halaman admin

1. User interface halaman guru

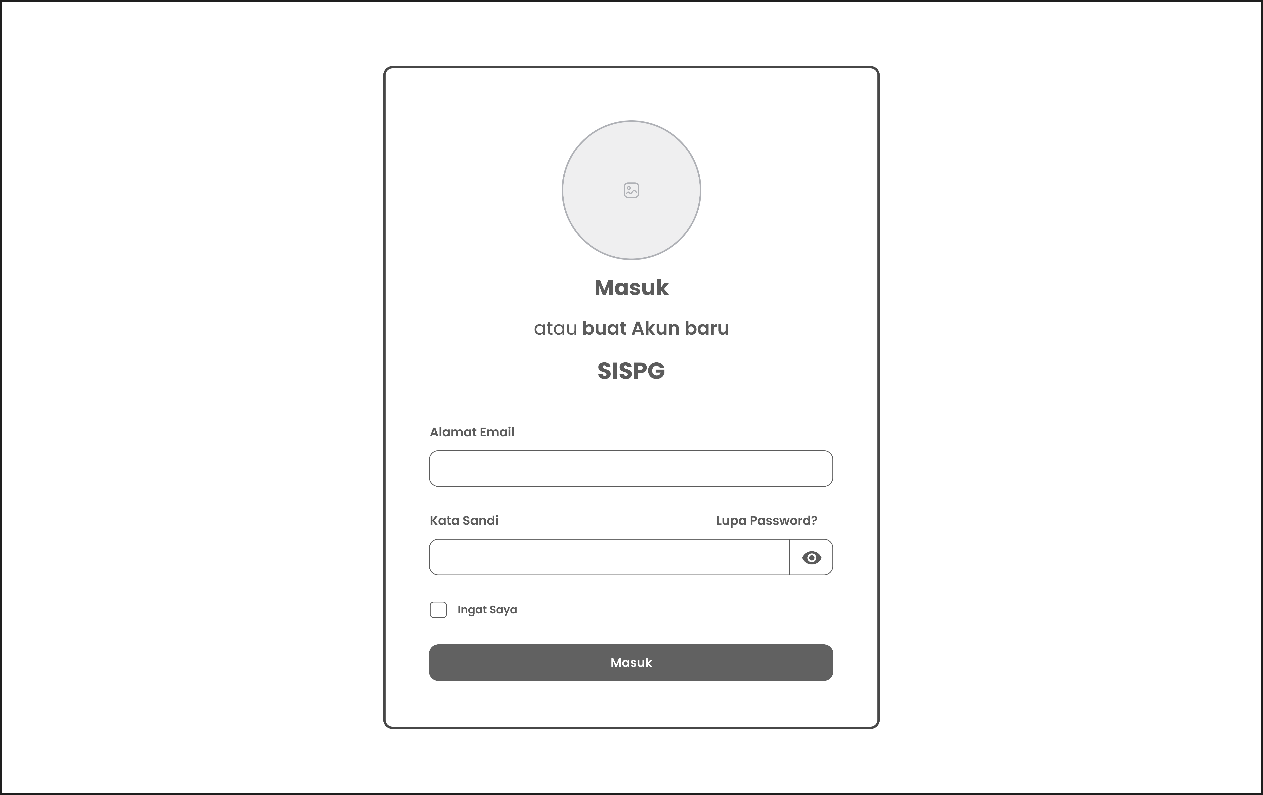
Pada desain halaman guru, admin dapat menambah, mengedit, menambah keanggotaan baru dan menghapus data sesuai kebutuhan sekolah.

.



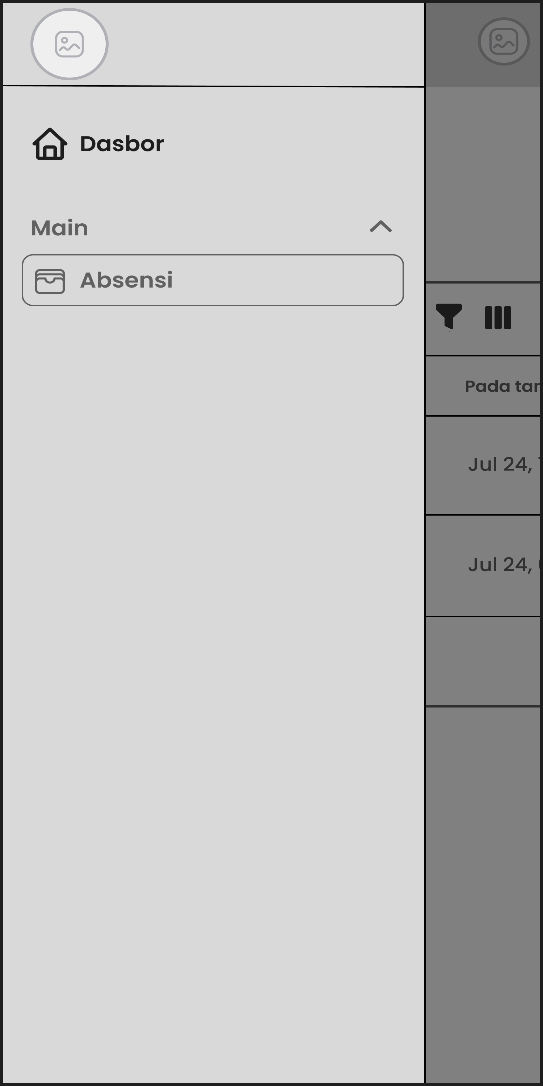
Gambar 4.14 User interface halaman login user

Pada desain halaman *login* *user* dapat melakukan akses masuk menuju aplikasi presensi bguru setalah melakukan register pada admin, lalu user dapat masuk ke halaman beranda aplikasi presensi guru.



Gambar 4. 15 User Interface Halaman Login user

1. User interface halaman *dashboard*

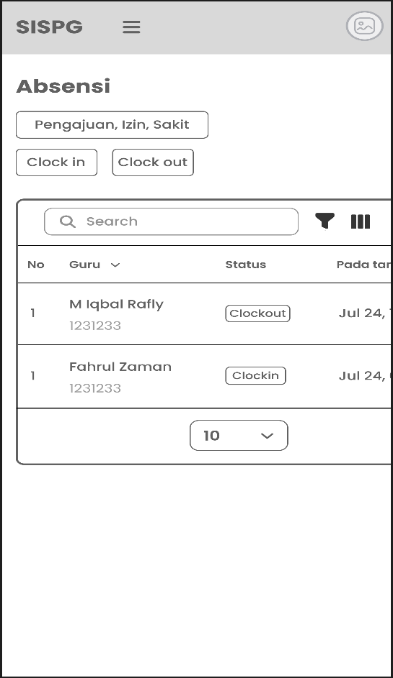


Gambar 4.16 User interface halaman dashboard

Setelah berhasil melakukan akses login, user dapat memasuki halaman dashboard *user* seperti yang sedang berjalan di bawah ini.

1. User interfacehalaman absensi *user*

Pada desain halaman absen user akan menampilkan dua kategori menu tambahan di dalamnya sebagai akses menuju proses abseni *user*, beberapa diantaranya terdapat menu perizinan dan *clock-In*/*clock-Out* sebagai proses absensi.



Gambar 4. 17 User Interface Halaman absensi user

1. User interface halaman perizinan

Pada desain halaman perizinan akan menampilkan halaman yang menampilkan tipe atau opsi jenis izin yang ungin user pergunakan, lalu tanggal perizinan dan catatan sebagai keterangan alsan melakukan perizinan.



Gambar 4. 18 User Interface Halaman perizinan user

# BAB V IMPLEMENTASI DAN PENGUJIAN

## 5.1 Implementasi

Setelah melaksanakan analisis dan perancangan maka selanjutnya adalah pengimplementasian untuk menjalankan analisis dan perancangan yang sudah dibuat ke dalam bentuk aplikasi.

### 5.1.2 Implementasi Sistem

Implementasi sistem adalah tahap penerapan sistem yang akan dilakukan jika sistem telah disetujui termasuk program yang telah dibuat pada tahap perancangan sistem agar siap untuk dioperasikan. Adapun waktu dan tempat penerapan sistem yang sudah dibuat sebagai berikut:

1. Waktu dan Tempat Implementasi

Tempat : MA Nurul Huda Pacet

Alamat : Jalan raya Pacet Rt.02 Rw.15 Desa Pangauban, Kecamatan Pacet, Kabupaten Bandung, Jawa Barat.

### 5.1.3 Spesifikasi Sistem

Spesifikasi sistem akan menjelaskan tentang spesifikasi perangkat keras dan perangkat lunak yang digunakan dalam pengimplementasian aplikasi Presensi Guru.

1. Spesifikasi Perangkat Keras

Di bawah ini merupakan spesifikasi perangkat keras yang digunakan sebagai berikut:

Tabel 5. 1 Spesifikasi Perangkat Keras

|  |  |
| --- | --- |
| Processor | AMD Ryzen 3 3200U with Radeon Vega Mobile Gfx (4 CPUs), 2.6GHz |
| RAM | 4 GB |
| SSD | 128 GB |

1. Spesifikasi Perangkat Lunak

Di bawah ini merupakan spesifikasi perangkat lunak yang digunakan sebagai berikut:

Tabel 5.2 Spesifikasi Perangkat Lunak

|  |  |
| --- | --- |
| Sistem Operasi | Windows 10 |
| Database | MySQL |
| Bahasa Pemrograman | PHP |

### 5.1.4 Installasi Sistem

Tabel-tabel yang terdapat dalam basis data yang digunakan dalam aplikasi presensi guru adalah sebagai berikut:

1. Installasi Aplikasi
2. XAMPP

XAMPP ini dibutuhkan untuk menjalankan aplikasi dengan memanggil localhost dan juga untuk melakukan running MySQL yang berada di dalam XAMPP.

Berikut di bawah ini merupakan tahapan-tahapan instalasi XAMPP:

1. Unduh installer aplikasi XAMPP melalui web apachefriends.org
2. Lakukan double klik pada file XAMPP yang sudah diunduh
3. Klik next pada jendela installer
4. Pilih komponen yang akan diinstal atau di biarkan default untuk menginstal keseluruhan. Disarankan untuk menginstal keseluruhan
5. Pilih folder instalasi, lalu klik next
6. Pilih Bahasa yang ingin digunakan, lalu klik next
7. Jalankan instalasi dengan mengklik next pada jendela berikutnya
8. Tunggu hingga proses instalasi selesai
9. Setelah instalasi selesai, klik finish dan XAMPP siap digunakan.

2). Browser (Google Chrome)

Browser ini berguna untuk mengaktifkan dan menjalankan aplikasi presensi yang sudah dibuat.

Berikut di bawah ini merupakan tahapan-tahapan instalasi Browser Google Chrome:

* 1. Unduh installer Google Chrome
  2. Jika diminta, klik jalankan atau simpan
  3. Jika memilih simpan maka lakukan klik dua kali pada tombol download untuk memulai proses instalasi
  4. Tunggu hingga proses instalasi selesai
  5. Setelah instalasi selesai, Google Chrome sudah bisa digunakan.

1. Installasi Database

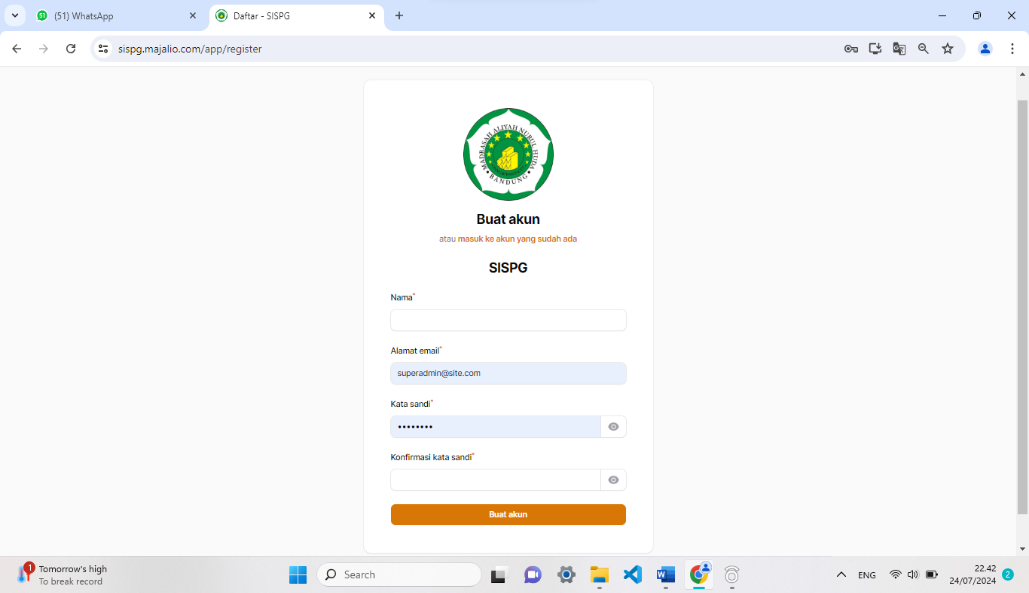
Berkaitan dengan database yang digunakan pada aplikasi Presensi Guru merupakan MySQL dan MySQL itu sendiri sudah otomatis terdapat di dalam aplikasi XAMPP maka yang harus dilakukan untuk menjalankan database MySQL hanyalah dengan mengaktifkan XAMPP dan memanggil PHP MyAdmin di *browser*.

### 5.1.5 Menjalankan Sistem

Pada bagian ini akan dijelaskan bagaimana cara-cara menjalankan aplikasi presensi guru.

1. Halaman *Regiter*

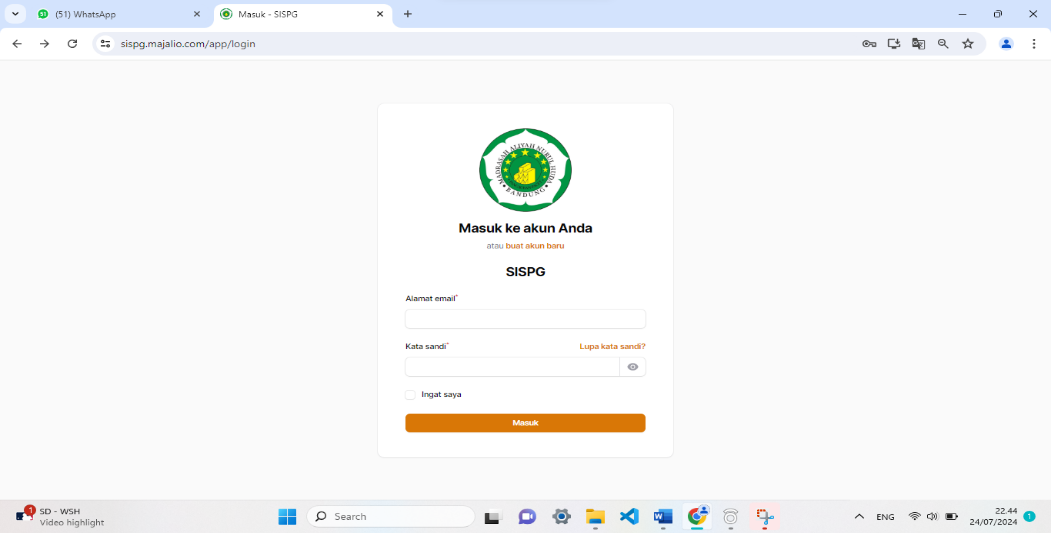
Untuk bisa memasuki aplikasi presensi guru *user* diwajibkan untuk melakukan proses autentikasi dengan cara memasukkan *username* serta *password* yang telah diatur sebelumnya.



Gambar 4. 19 Halaman Register

1. Halaman Login

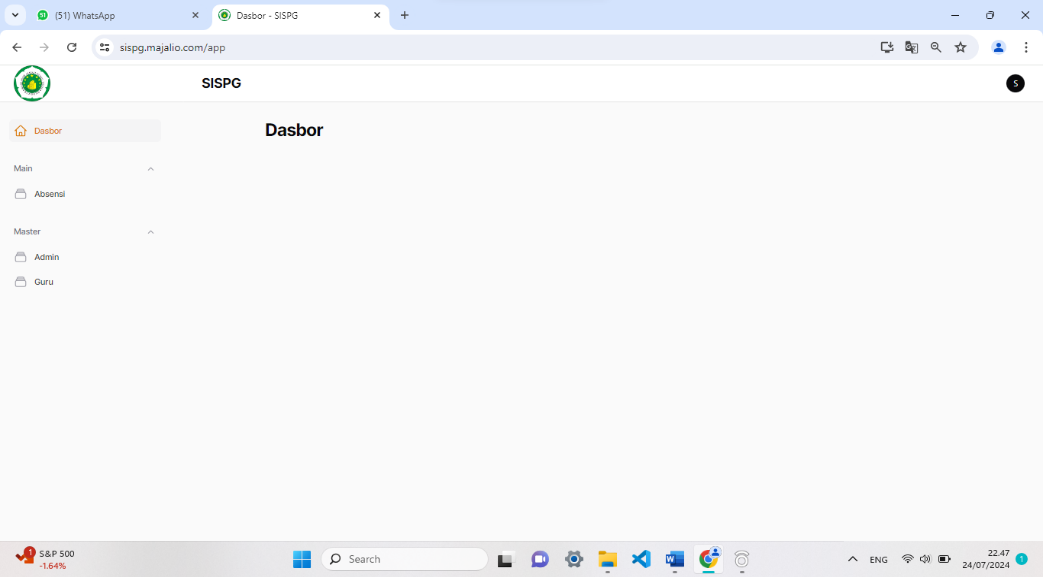
Setelah berhasil melakukan proses autentikasi *user* diarahkan ke halaman beranda *user*.



Gambar 4.20 Halaman Login

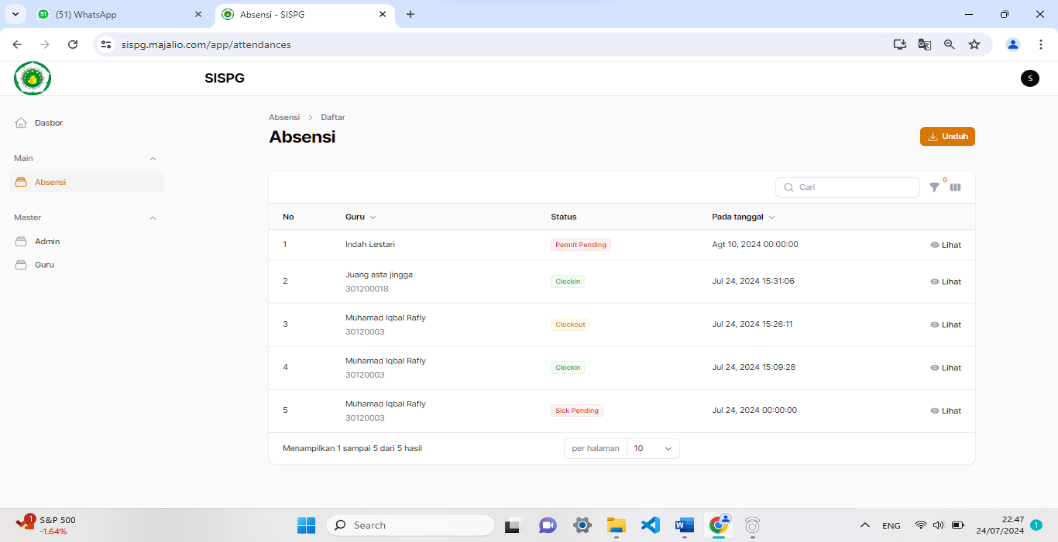
1. Halaman *Dashboard*

Halaman pengguna merupakan halaman yang digunakan untuk menambah, mengubah serta menghapus pengguna yang ada dalam aplikasi presensi guru.



Gambar 4.21 Halaman Dashboard

1. Halaman Absensi

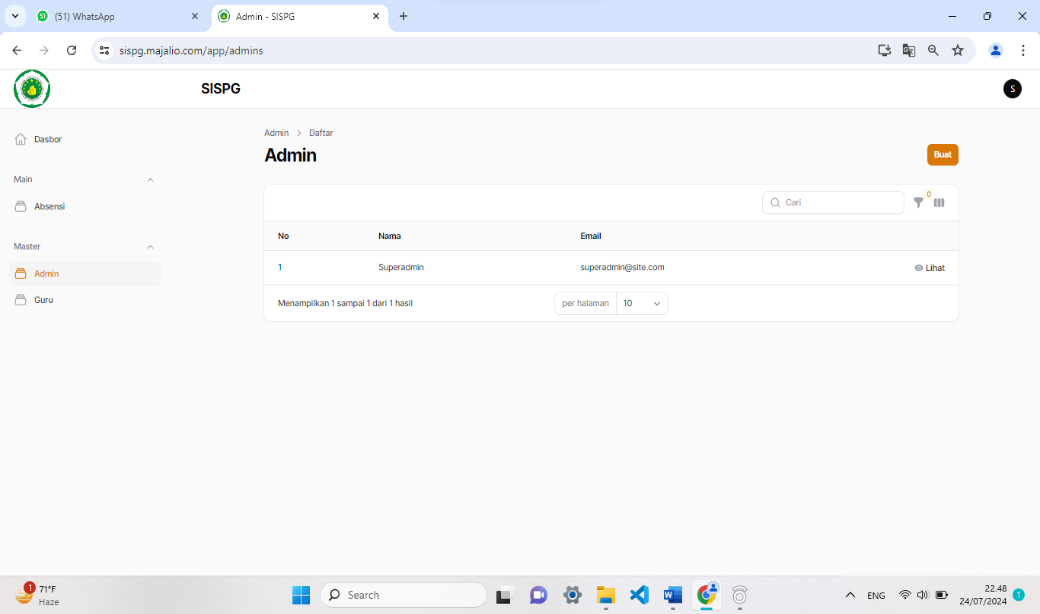


Gambar 4. 22 Halaman Absensi

Halaman absensi merupakan halaman yang digunakan untuk menambah, mengubah serta menghapus data user yang ada dalam aplikasi presensi guru.

1. Halaman Admin

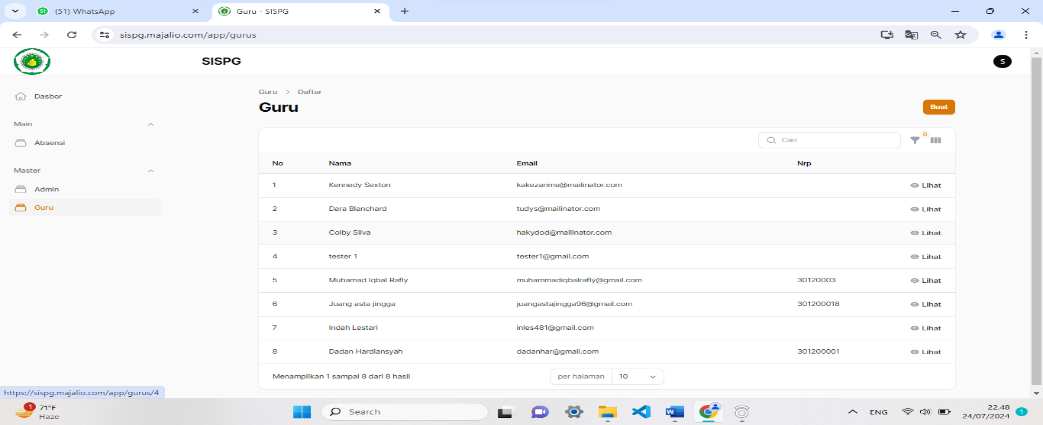
Halaman admin merupakan halaman yang digunakan untuk meninjau menambah, mengubah serta menghapus data admin jika sewaktu-waktu terjadi perubahan bagian staf oprator.



Gambar 4. 23 Halaman Admin

1. Halaman Guru

Halaman guru merupakan halaman yang digunakan untuk menambah, mengubah serta menghapus data guru yang ada dalam aplikasi presensi.



Gambar 4. 24 Halaman Guru

Gambar 7.1 Halaman Guru

1. Halaman Login User

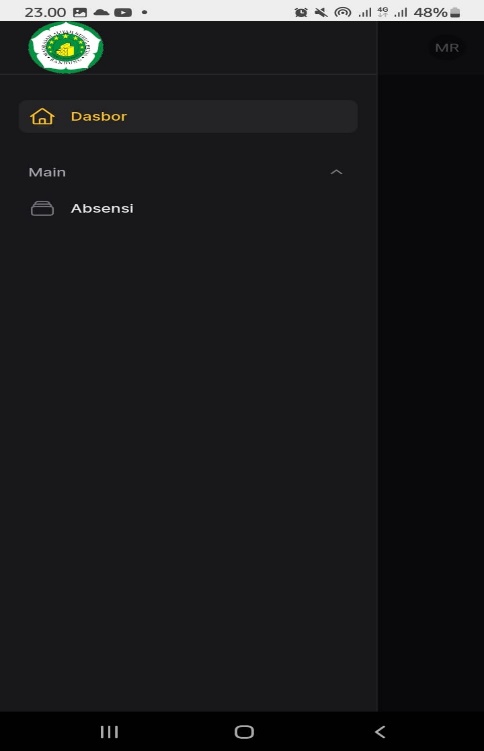
Halaman login user digunakan untuk mengakses ke halaman utama aplikasi.



Gambar 4. 25 Halaman Login User

1. Halaman *Dashboard User*

Setelah berhasil melakukan akses login, user dapat memasuki halaman dashboard *user* seperti yang sedang berjalan di bawah ini.



Gambar 4. 26 Halaman Dashboard User

1. Halaman absensi *User*

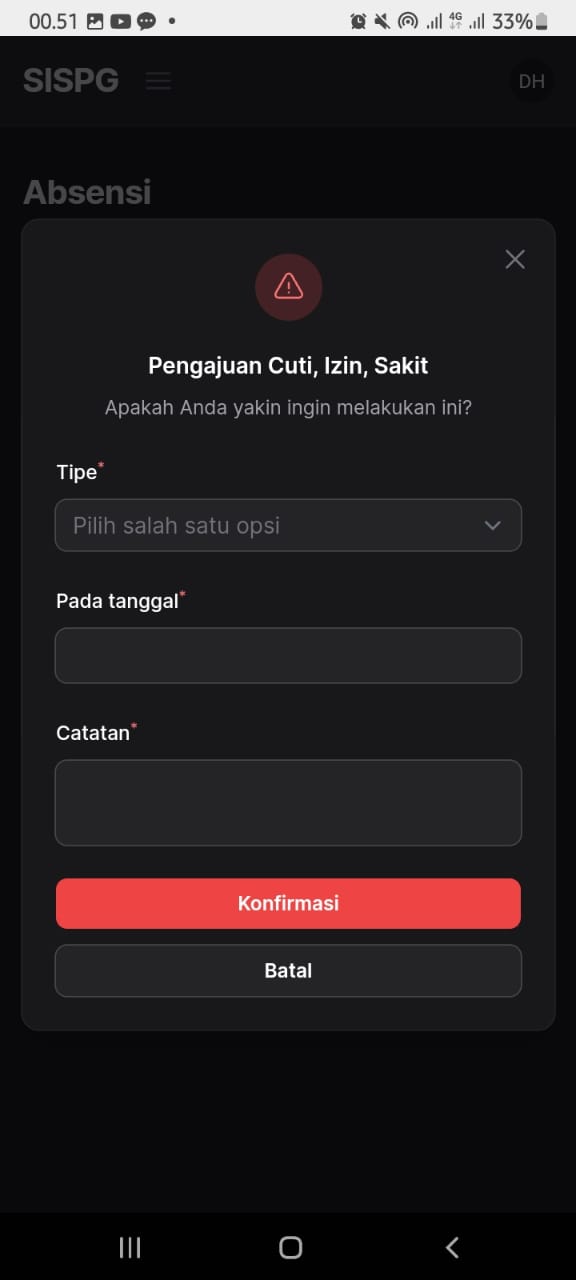
Pada desain halaman absensi *user* akan menampilkan dua kategori menu tambahan di dalamnya sebagai akses menuju proses abseni *user*, beberapa diantaranya terdapat menu perizinan dan *clock-In*/*clock-Out* sebagai proses absensi.



Gambar 4. 27 Halaman Absensi User

1. Halaman Perizinan *User*

Pada halaman perizinan akan menampilkan halaman yang menampilkan tipe atau opsi jenis izin yang ungin user pergunakan, lalu tanggal perizinan dan catatan sebagai keterangan alsan melakukan perizinan.



Gambar 4. 28 Halaman Perizinan User

## 5.2 Pengujian

Setelah dilakukan tahap implementasi pada rancang bangun aplikasi, maka tahapan selanjutnya yang dilakukan adalah tahap pengujian. Pada tahap ini bertujuan untuk melihat apakah aplikasi yang dibuat dengan analisis serta perancangan yang sebelumnya sudah sesuai dan berfungsi sebagaimana mestinya atau belum.

Tabel 5. 3 Hasil Pengujian Aplikasi Dengan Metode Blackbox

| No | Item uji | Skenario Pengujian | Hasil Yang Diharapkan | Hasil | |
| --- | --- | --- | --- | --- | --- |
| Sesuai | Tidak Sesuai |
| 1. | Halaman *Register* | *Register* dengan *username* dan *password* yang *valid* | Sistem akan menampilkan halaman *Register* sesuai hak akses | Sesuai |  |
| 2. | Halaman Login | Masuk ke halaman Login | Sistem akan menampilkan halaman Login | Sesuai |  |
| 3. | Halaman *Dashboard* | Masuk ke halaman *Dashboard* | Sistem akan menampilkan halaman *Dahboard* | Sesuai |  |
| 4. | Halaman absensi | Masuk ke halaman absensi | Sistem akan menampilkan halaman absensi | Sesuai |  |
| ­5. | Halaman admin | Masuk ke halaman admin | Sistem akan menampilkan halaman daftar admin | Sesuai |  |
| 6. | Halaman Guru | Masuk ke halaman Guru | Sistem akan menampilkan halaman daftar Guru | Sesuai |  |
| 7. | Halaman absensi User | Masuk ke halaman absensi User | Sistem akan menampilkan halaman daftar absensi User | Sesuai |  |
| 8. | *Logout* | *Logout* dari sistem | Dapat *logout* dari sistem | Sesuai |  |

Berdasarkan tabel 5.3 yang menampilkan hasil pengujian aplikasi dengan metode *blackbox*, maka dapat disimpulkan bahwa hasil pengujian aplikasi sudah berhasil dan sesuai dengan perancangan dan analisis yang dilakukan, semua fitur berjalan dengan baik sesuai dengan fungsi yang seharusnya dan penulis inginkan.

# BAB VI KESIMPULAN

## Kesimpulan

Berdasarkan penelitian yang telah dilakukan penulis melalui beberapa tahapan yang dilakukan pada bab-bab sebelumnya, maka penulis mengambil kesimpulan bahwa:

Adanya aplikasi presensi guru untuk mengelola data presensi ini diharapkan dapat membantu sekolah yayasan MA Nurul Huda Pacet dalam pengelolaan data presensi dengan baik untuk meningkatkan efisiensi dalam mengelola presensi guru di MA Nurul Huda Pacet.

Adanya aplikasi presensi guru ini diharapkan dapat meningkatkan efisiensi dalam mengecek data absensi seperti aktivitas kehadiran guru di MA Nurul Huda Pacet, dan dapat membuat pelaporan dengan cepat.

Adanya aplikasi presensi guru ini diharapkan dapat meningkatkan akurasi data seperti informasi mengenai kehadiran guru yang lebih terakurasi.

Aplikasi presensi guru berbasis android ini dapat diakses oleh admin dan user atau petugas staf administrasi yang sudah diberi akses oleh admin selaku pengelola aplikasi presensi guru di MA Nurul Huda Pacet.

## Saran

Untuk memaksimalkan hasil dari penggunaan aplikasi Presensi Guru Berbasis Android, beberapa saran berikut diharapkan dapat membantu meningkatkan kinerja dan efektivitas aplikasi:

Pelatihan Pengguna (Aplikasi Presensi Guru) disarankan untuk mengadakan pelatihan bagi para pengguna aplikasi, terutama pengelola Presensi Guru di MA Nurul Huda Pacet. Pelatihan ini bertujuan agar guru dapat mengoperasikan aplikasi Presensi guru berbasis android dengan lebih baik. Dengan pelatihan yang memadai, pengguna akan lebih memahami fitur-fitur yang ada dan mampu memanfaatkannya secara maksimal, sehingga aktivitas absensi menjadi lebih efisien.

Peningkatan keamanan sistem aplikasi merupakan aspek yang sangat penting. Disarankan untuk menambahkan lapisan keamanan tambahan, seperti fitur login yang lebih aman dan metode autentikasi lainnya. Langkah ini bertujuan untuk melindungi data pengguna dan mencegah akses dari pihak yang tidak berwenang, sehingga integritas dan kerahasiaan data tetap terjaga.

Evaluasi dan pengembangan berkala agar aplikasi Presensi Guru selalu relevan dan memenuhi kebutuhan pengguna, diperlukan evaluasi dan pengembangan secara berkala. Melalui evaluasi rutin, pengembang dapat mengidentifikasi area yang perlu ditingkatkan dan memperbaiki bug atau masalah yang ada. Selain itu, pengembangan fitur baru yang sesuai dengan kebutuhan pengguna juga akan meningkatkan kualitas dan layanan aplikasi.

Dengan menerapkan saran-saran tersebut, diharapkan aplikasi Presensi Guru dapat terus berkembang dan memberikan manfaat yang optimal bagi penggunanya. Pengelolaan absensi akan menjadi lebih efisien, aman, dan mudah diakses, sehingga mendukung kelancaran bagi sarana dan prasarana terutama pihak staf administrasi secara keseluruhan.

# DAFTAR PUSTAKA

Atmaja, A. P., & Aziz, A. (2019). “Sistem Informasi Terintegrasi Evaluasi Kegiatan Mengajar Dosen Sebagai Implementasi Sistem Penjaminan Mutu Internal.” Jurnal Manajemen Teknologi Dan Informatika, 9(1), 1-6.

Emanual Pabianan, Crishtine Dewi. 2023. Perancangan Global Positioning System (GPS) pada Presensi Online Berbasis Website Menggunakan Metode Waterfall . Jurnal Prendidikan Informasi (JUKANTI) Vol (6) No (2) Nopember 2023-elSSN : 2621-1467

Mustaqbal, M. S., Firdaus, R. F., & Rahmadi, H. (2015). PENGUJIAN APLIKASI MENGGUNAKAN BLACK BOX TESTING BOUNDARY VALUE ANALYSIS (Studi Kasus: Aplikasi Prediksi Kelulusan SNPTN). I(3), 31-36.

Prasetyo, B., & Trisyanti, D. (2019). Prosiding SEMATEKSOS 3 “Strategi Pembangunan Nasional MenghadapiRevolusiIndustri 4.0” REVOLUSI INDUSTRI 4.0. Revolusi Industri 4.0 Dan Tantangan Perubahan Sosial, 22–27.

Pradnyana, G. A., Arthana, K. R., Putrama, I. M., Pradnyana, I. M. A., & Sindu, I. G. P. (2017). Pembuatan dan Pelatihan Pengelolaan Website serta Email Resmi untuk Organisasi Pemuda Peduli Lingkungan Bali (PPLB) Kabupaten Buleleng. Prosiding Seminar Nasional Pengabdian Kepada Masyarakat (SENADIMAS) 2, 134–143.

Arhandi, P.P., Arief, S.N. and Firdausi, A.T. (2022) ‘Pengembangan Website Pendukung Mastery Based Learning Untuk Pembelajaran Mahasiswa’, *Jurnal Informatika Polinema*, 9(1), pp. 51–58. Available at: https://doi.org/10.33795/jip.v9i1.966.

Herlabang. S. P. 2011. “PEMBACAAN POSISI KOORDINAT DENGAN GPS SEBAGAI PENGENDALI PALANG PINTU REL KERETA API SECARA OTOMATIS UNTUK PENAMBAHAN APLIKASI MODUL PRAKTIK MIKROKONTROLER.” 182 Jurnal Pendidikan Teknologi dan Kejuruan. Vol 20, No 2, Oktober 2011.

Taufiq, M., Amalia, A. V., Parmi, P., and Laviana, A. (2016). “Design of science mobile learning of eclipse phenomena with conversation insight androide-based app inventor 2. Jurnal Pendidikan IPA Indonesia, 5(2), 291-298. https://doi.org/10.15294/jpii.5i3.7375

Arianto, O.D. and Susetyo, Y.A. (2022) ‘Penerapan Restful Web Service Dengan Framework Laravel Untuk Pembangunan Sistem Informasi Manajemen Sumber Daya Manusia’, *JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, 7(2), pp. 522–532. Available at: https://doi.org/10.29100/jipi.v7i2.2870.

Hariyanto, D. *et al.* (2021) ‘Implementasi Metode Rapid Application Development Pada Sistem Informasi Perpustakaan’, *Jurnal JUPITER*, 13(1), pp. 110–117.

Setya Budi, A. *et al.* (2022) ‘Pengenalan Dan Penggunaan Dbms (Database Management System) Di Smk Tunas Media Kota Depok’, *Abdi Jurnal Publikasi*, 1(2), pp. 52–57.

Titi Komalasari, R. (2021) ‘Menggunakan Metode White Box Testing dan Black Box Testing’, *Jurnal Teknologi Informasi*, 7(1), pp. 50–57.

Widiarta, I.M., Mulyanto, Y. and Sutrianto, A. (2023) ‘Rancang Bangun Sistem Informasi Inventory Menggunakan Metode Agile Software Development (Studi Kasus Toko Nada)’, *Digital Transformation Technology (Digitech)*, 3(Maret), p. 20.

Fachriyan, M., Dharmayanti, D. and Bandung, J. D. (2019) ‘UNTUK PENYANDANG TUNANETRA MENGGUNAKAN MLKIT DAN TEXT TO VOICE BERBASIS ANDROID Teknik Informatika – Universitas Komputer Indonesia’.

Afriansyah, A., & Syaripudin, A. (2022). Perancangan Sistem Informasi Absensi Dewan Guru Tenaga Harian Lepas Berbasis Web Pada Sekolah Dasar Negeri Kunciran 6 Kota Tangerang. *Biner: Jurnal Ilmiah Informatika dan Komputer*, *1*(1), 17-25.

Putra, Y. W. S., & Adhim, M. F. (2022). Sistem Informasi Presensi Online Menggunakan Teknologi Face Recognition dan GPS. *Jurnal Tekno Kompak*, *16*(1), 149-161.

Rohman, D. N., & Kusyadi, I. (2022). Perancangan Sistem Informasi Presensi Guru pada MTS Ar-Rasyidiyyah Berbasis Web. *Scientia Sacra: Jurnal Sains, Teknologi dan Masyarakat*, *2*(1), 89-94.

INFORMATIKA, C., & yudi herdiana. (2020). MEMBUAT APLIKASI BIMBINGAN SKRIPSI ONLINE UNTUK MEMPERMUDAH BIMBINGAN SKRIPSI (STUDI KASUS PRODI TEKNIK INFORMATIKA FTI UNIBBA) . COMPUTING | Jurnal Informatika, 7(02).Retrieved from

https://ejournal.unibba.ac.id/index.php/computing/article/view/853

INFORMATIKA, C., Iyus A muslimin, & Denny Rusdianto. (2022). PEMBUATAN APLIKASI PENJUALAN BINATANG TERNAK BERBASIS ANDROID DI PETERNAKAN BEBEK ALFALAH DESA

PADAULUN. COMPUTING | Jurnal Informatika, 9(02), 80– 89. Retrieved from x https://ejournal.unibba.ac.id/index.php/computing/article/view/102

INFORMATIKA, C., & Iyus A Muslimin. (2021). MEMBANGUN APLIKASI PEMBUATAN SURAT KETERANGAN KULIAH, SURAT PENGANTAR PENELITIAN DAN PENGAJUAN CUTI BERBASIS WEB. COMPUTING Jurnal Informatika, 8(02), 1–7. Retrieved from https://ejournal.unibba.ac.id/index.php/computing/article/view/698

INFORMATIKA, C., Khilda Nistrina, & Sukiman. (2022). MEMBANGUN APLIKASI SENSUS KEPENDUDUKAN BERBASIS WEB MENGGUNAKAN METODE WATERFALL DI DESA NEGLASARI. COMPUTING | Jurnal Informatika, 9(01), 44–50. Retrieved from https://ejournal.unibba.ac.id/index.php/computing/article/view/862

INFORMATIKA, C., & Khilda Nistrina. (2021). RANCANG BANGUN GAME EDUKASI SEBAGAI MEDIA PEMBELAJARAN BERBASIS ANDROID MENGGUNAKAN ALGORITMA FISHER YATES. COMPUTING | Jurnal Informatika, 8(02), 8–13. Retrieved from https://ejournal.unibba.ac.id/index.php/computing/article/view/699

INFORMATIKA, C., & Nurul Imamah. (2021). PERANCANGAN SISTEM MONITORING DAN PENGENDALIAN LAMPU MENGGUNAKAN SENSOR GERAK DAN SENSOR CAHAYA DILENGKAPI INTERNET OF THINGS (IOT). COMPUTING | Jurnal Informatika, 8(02), 14–21. https://ejournal.unibba.ac.id/index.php/computing/article/view/700

INFORMATIKA, C., Nurul Imamah, sutiyono WP, & Ari Reynaldi. (2022). PERANCANGAN DAN IMPLEMENTASI SISTEM MONITORING KEAMANAN TOKO BERBASIS INTERNET OF THINGS (IOT) DENGAN MENGGUNAKAN MIKROKONTROLER ESP32-CAM (STUDI KASUS DI TOKO STELIOS AQUATIC). COMPUTING | Jurnal Informatika, 9(02),70–79.Retrievedfromhttps://ejournal.unibba.ac.id/index.php/computing/article/view/102 5

INFORMATIKA, C., & Nurul Imamah. (2021). PERANCANGAN SISTEM MONITORING DAN PENGENDALIAN LAMPU MENGGUNAKAN SENSOR GERAK DAN SENSOR CAHAYA DILENGKAPI INTERNET OF THINGS (IOT). COMPUTING | Jurnal Informatika, 8(02), 14–21. Retrieved from https://ejournal.unibba.ac.id/index.php/computing/article/view/700 xi

INFORMATIKA, C., & Nurul Imamah. (2021). PERBANDINGAN ALGORITMA SEQUENTIAL SEARCH DAN ALGORITMA BINARY SEARCH PADA APLIKASI KAMUS BAHASA INDONESIA MENGGUNAKAN PHP DAN JQUERY . COMPUTING | Jurnal Informatika, 8(01), 1–6. Retrieved from https://ejournal.unibba.ac.id/index.php/computing/article/view/564

INFORMATIKA, C., & Rosmalina. (2022). SISTEM INFORMASI AKADEMIK BERBASIS WEB MENGGUNAKAN BOOTSTRAP DI SMP AMS PAMEUNGPEUK: SISTEM INFORMASI AKADEMIK BERBASIS WEB MENGGUNAKAN BOOTSTRAP DI SMP AMS PAMEUNGPEUK. COMPUTING | Jurnal Informatika, 9(01), 21– 25. Retrieved from https://ejournal.unibba.ac.id/index.php/computing/article/view/857

INFORMATIKA, C., & Rosmalina. (2021). PENERAPAN ALGORITMA SAW (SIMPLE ADDITIVE WEIGHTING) PADA MODUL APLIKASI SPK DI MA AL-AZHAR MAJALAYA BERBASIS WEB. COMPUTING | Jurnal Informatika, 8(01), 7–15. Retrieved from https://ejournal.unibba.ac.id/index.php/computing/article/view/565

INFORMATIKA, C., & Rosmalina. (2021). APLIKASI PENGELOLAAN KAMAR INDEKOS BERBASIS WEB DI KAMAR INDEKOS ALNAJAH CIPARAY. COMPUTING | Jurnal Informatika, 8(02), 22– 28. Retrieved from https://ejournal.unibba.ac.id/index.php/computing/article/view/701

INFORMATIKA, C., & Rustiyana. (2022). IMPLEMENTASI WEBSITE SEBAGAI MEDIA MEMBUAT TEMPLATE KONFIGURASI PERANGKAT JARINGANTELEKOMUNIKASI DI PT ICON+. COMPUTING | Jurnal Informatika, 9(01), 26–30. Retrieved from https://ejournal.unibba.ac.id/index.php/computing/article/view/858

INFORMATIKA, C., & Rustiyana. (2021). IMPLEMENTASI ALGORITMA FISHER YATES SHUFFLE DALAM PEMBUATAN UJIAN ONLINE BERBASIS WEB . COMPUTING | Jurnal Informatika, 8(01), 16–21. Retrieved from https://ejournal.unibba.ac.id/index.php/computing/article/view/566

INFORMATIKA, C., Rustiyana, Khilda Nistrina, Sukiman, & Sandhy Dwi A. (2023). PERANCANGAN DAN IMPLEMENTASI TEMPAT SAMPAH OTOMATIS BERBASIS ARDUINO UNO MENGGUNAKAN ALGORITMA FUZZY LOGIC. COMPUTING | Jurnal Informatika, 10(01), 21–25. Retrieved from https://ejournal.unibba.ac.id/index.php/computing/article/view/114 6

# LAMPIRAN

**Lampiran: 1: Wawancara**

Hari, Tanggal : Kamis, 25 April 2024

Tempat : MA Nurul Huda Pacet

Narasumber : Adeng Hamdani, S.Pd.I

Pewawancara : Muhamad Iqbal Rafly

Peneliti : Siapa yang mengelola absensi di MA Nurul Huda Pacet?

Narasumber : Nama Saya Adeng Hamdani, saya sebagai pengelola absensi guru di sekolah ini.

Peneliti : MA Nurul Huda Pacet beralamat dimana?

Narasumber : Tepatnya ada di jalan raya Pacet Rt.02 Rw.15 Desa Pangauban, Kecamatan Pacet, Kabupaten Bandung

Peneliti : Apakah ada kendala yang di alami pada saat ini dalam pengelolaan data presensi?

Narasumber : Ada beberapa yang dialami dalam pengelolaan aktivitas absensi, Pengelolaan absen disini masih menggunakan manual dan seringnya, dan kesulitan dalam pencarian dokumen absensi lama, yang sewaktu-waktu tak jarang sering dibutuhkan sebagai laporan.

Peneliti : Fitur apa yang anda inginkan dari aplikasi yang akan dibuat?

Narasumber : Fitur yang di inginkan ialah fitur absensi melalui ponsel guru dan membuat laporan dengan cepat.

Peneliti : Bagaimana dampak dari pencatatan data barang secara manual menggunakan buku catatan?

Narasumber : Sering menyebabkan ketidak sesuaian antara kebutuhan dan implementasi akibat penggunaan yang masih manual. Hal tersebut mengakibatkan kurangnya efektivitas dalam pembuatan laporan yang sedang berjalan secara manual ini.

Peneliti : Apa yang Bapak harapkan jika aplikasi ini sudah dibuat untuk keberlangsungkan anda?

Narasumber : Harapan saya dengan adanya aplikasi tersebut semoga bisa menjadi solusi untuk permasalahan yang saya hadapi seperti presensi guru yang masih manual ini, mengalami pembaharuan sehingga dapat membuat pencatatan atau laporan dengan efisien, mengecek cepat, serta dengan adanya pengelolaan data presensi yang lebih ter autentikasi.

Bandung, 25 April 2024

Staf Tatausaha

Adeng Hamdani, S.Pd.I

**Lampiran: 2 : ToR (Term Of Reference)**

Sebelum Melaksanakan penelitian skripsi penulis melakukan beberapa metode penelitian yaitu observasi, wawancara, dan studi pustaka. Setelah mengamati dan mempelajari lokasi penelitian yang telah ditentukan dan disetujui oleh Yayasan MA Nurul Huda Pacet, yang terletak di jalan raya Pacet Rt.02 Rw.15 Desa Pangauban, Kecamatan Pacet, Kabupaten Bandung, Jawa Barat. Menghadapi permasalahan yang saat ini sedang dihadapi MA Nurul Huda Pacet tersebut yaitu presensi masih secara manual menggunakan kertas tulis, yang menyebabkan berbagai kesulitan dalam pencarian dan pembuatan laporan yang cepat kepada kepala sekolah. Sistem manual ini juga menyebabkan ketidaksesuaian antara kebutuhan dan implementasi karena terkendala oleh waktu dan antrian permintaan data yang sulit dicari. Penelitian ini bertujuan untuk membangun aplikasi presensi yang terdigitalisasi agar lebih efisien. Dengan memanfaatkan *framework* *Laravel* untuk mengelola data barang. Bahasa pemrograman yang digunakan yaitu PHP, XAMPP, dan database MYSQL, dengan penerapan *framework laravel* Untuk memastikan fokus penelitian adapun batasan masalah yang ditetapkan adalah sebagai berikut:

1. Fokus pada pembangunan aplikasi pengelolaan presensi guru berbasis android.
2. Fitur yang tersedia berupa . *Register*, *Login*, *Dashboard*, Data Admin dan Guru, Data absensi, perizinan, dan *LogOut*.
3. Pengujian aplikasi dilakukan secara simulasi atau pada lingkungan uji coba terbatas.
4. Menggunakan framework Laravel dalam pengembangan aplikasinya.*.*

|  |  |
| --- | --- |
| **Pewawancara**  **Muhamad Iqbal Rafly** | **Narasumber**  **Adeng Hamdani, S.Pd.I** |

**Lampiran:3 : Dokumentasi Wawancara**





Lampiran: 4 **: Listing Program**

1. . *Listing* Program *Attendance Resource*

|  |
| --- |
| \AttendanceResource\Pages;  use App\Models\Attendance;  use Filament\Forms;  use Filament\Forms\Components\DatePicker;  use Filament\Forms\Form;  use Filament\Resources\Resource;  use Filament\Tables;  use Filament\Tables\Table;  use Illuminate\Database\Eloquent\Builder;  use Illuminate\Database\Eloquent\SoftDeletingScope;  use function Filament\Support\get\_model\_label;  class AttendanceResource extends Resource  {  protected static ?string $model = Attendance::class;  protected static ?string $navigationIcon = 'heroicon-o-rectangle-stack';  public static function getNavigationGroup(): ?string  {  return trans('admin.menu.main');  }  public static function getNavigationLabel(): string  {  return trans('admin.resources.attendance\_report');  }  public static function getNavigationBadge(): ?string  {  return null;  }  public static function getNavigationBadgeTooltip(): ?string  {  return static::$navigationBadgeTooltip;  }  public static function getNavigationBadgeColor(): string | array | null  {  return null;  }  public static function shouldRegisterNavigation(): bool  {  // return auth()->user()->type == UserType::Admin;  return true;  }  public static function getModelLabel(): string  {  return trans('admin.resources.attendance');  }  public static function form(Form $form): Form  {  return $form  ->schema([  Forms\Components\Select::make('user\_id')  ->relationship('guru', 'name')  ->searchable()  ->preload()  ->required(),  Forms\Components\Select::make('type')  ->options(fn () => AttendanceType::class)  ->searchable()  ->preload()  ->required(),  Forms\Components\Select::make('status')  ->options(fn () => AttendanceStatus::class)  ->searchable()  ->preload()  ->required(),  Forms\Components\DateTimePicker::make('recorded\_at')  ->label(trans('admin.fields.recorded\_at'))  ->translateLabel()  ->required(),  Forms\Components\Textarea::make('notes')  ->label(trans('admin.fields.notes'))  ->translateLabel()  ->columnSpanFull()  ->hidden(fn ($record) => empty($record->notes))  ->visibleOn(['view']),  Forms\Components\FileUpload::make('image\_clockin')  ->label(trans('admin.fields.image\_clockin'))  ->translateLabel()  ->disabled()  ->hidden(fn ($record) => empty($record->image\_clockin))  ->visibleOn(['view']),  Forms\Components\FileUpload::make('image\_clockout')  ->label(trans('admin.fields.image\_clockout'))  ->translateLabel()  ->disabled()  ->hidden(fn ($record) => empty($record->image\_clockout))  ->visibleOn(['view']),  ]);  }  public static function table(Table $table): Table  {  $user = auth()->user();  return $table  ->modifyQueryUsing(fn ($query) => $query  ->when(  $user->type != UserType::Admin,  fn ($query) => $query->where('user\_id', $user->id)  )  )  ->defaultSort('recorded\_at', 'desc')  ->columns([  Tables\Columns\TextColumn::make('no')  ->rowIndex(),  Tables\Columns\TextColumn::make('guru.name')  ->description(fn ($record) => $record->guru?->nrp)  ->sortable(),  Tables\Columns\TextColumn::make('type')  ->formatStateUsing(fn ($state) => str($state->value)->headline())  ->label(trans('admin.fields.type'))  ->translateLabel()  ->searchable()  ->badge()  ->toggleable(isToggledHiddenByDefault: true),  Tables\Columns\TextColumn::make('status')  ->formatStateUsing(fn ($state) => str($state->value)->headline())  ->badge()  ->searchable(),  Tables\Columns\TextColumn::make('recorded\_at')  ->label(trans('admin.fields.recorded\_at'))  ->translateLabel()  ->dateTime()  ->sortable(),  Tables\Columns\ImageColumn::make('image\_clockin')  ->label(trans('admin.fields.image\_clockin'))  ->translateLabel()  ->toggleable(isToggledHiddenByDefault: true),  Tables\Columns\ImageColumn::make('image\_clockout')  ->label(trans('admin.fields.image\_clockout'))  ->translateLabel()  ->toggleable(isToggledHiddenByDefault: true),  Tables\Columns\TextColumn::make('deleted\_at')  ->dateTime()  ->sortable()  ->toggleable(isToggledHiddenByDefault: true),  ])  ->filters([  Tables\Filters\TrashedFilter::make(),  Tables\Filters\SelectFilter::make('status')  ->options(fn () => AttendanceStatus::class)  ->searchable()  ->multiple()  ->preload(),  Tables\Filters\Filter::make('recorded\_at')  ->form([  DatePicker::make('recorded\_from')  ->label(trans('admin.fields.recorded\_from'))  ->translateLabel(),  DatePicker::make('recorded\_until')  ->label(trans('admin.fields.recorded\_until'))  ->translateLabel(),  ])  ->query(function (Builder $query, array $data): Builder {  return $query  ->when(  $data['recorded\_from'],  fn (Builder $query, $date): Builder => $query->whereDate('recorded\_at', '>=', $date),  )  ->when(  $data['recorded\_until'],  fn (Builder $query, $date): Builder => $query->whereDate('recorded\_at', '<=', $date),  );  }),  ])  ->actions([  Tables\Actions\ViewAction::make(),  ])  ->bulkActions([  //  ]);  }  public static function getRelations(): array  {  return [  //  ];  }  public static function getPages(): array  {  return [  'index' => Pages\ListAttendances::route('/'),  'create' => Pages\CreateAttendance::route('/create'),  'view' => Pages\ViewAttendance::route('/{record}'),  'edit' => Pages\EditAttendance::route('/{record}/edit'),  ];  }  public static function getEloquentQuery(): Builder  {  return parent::getEloquentQuery()  ->withoutGlobalScopes([  SoftDeletingScope::class,  ]);  }  } |
| ?php  namespace App\Filament\Pages;  use Filament\Facades\Filament;  use Filament\Pages\Page;  use Filament\Support\Facades\FilamentIcon;  use Filament\Widgets\Widget;  use Filament\Widgets\WidgetConfiguration;  use Illuminate\Contracts\Support\Htmlable;  class Dashboard extends Page  {  protected static string $routePath = '/';  protected static ?int $navigationSort = -2;  /\*\*  @var view-string  \*/  protected static string $view = 'filament-panels::pages.dashboard';  public static function getNavigationLabel(): string  {  return static::$navigationLabel ??  static::$title ??  \_\_('filament-panels::pages/dashboard.title');  }  public static function getNavigationIcon(): string | Htmlable | null  {  return static::$navigationIcon  ?? FilamentIcon::resolve('panels::pages.dashboard.navigation-item')  ?? (Filament::hasTopNavigation() ? 'heroicon-m-home' : 'heroicon-o-home');  }  public static function getRoutePath(): string  {  return static::$routePath;  }  /\*\*  @return array<class-string<Widget> | WidgetConfiguration>  \*/  public function getWidgets(): array  {  return Filament::getWidgets();  }  /\*\*  @return array<class-string<Widget> | WidgetConfiguration>  \*/  public function getVisibleWidgets(): array  {  return $this->filterVisibleWidgets($this->getWidgets());  }  /\*\*  @return int | string | array<string, int | string | null>  \*/  public function getColumns(): int | string | array  {  return 2;  }  public function getTitle(): string | Htmlable  {  return static::$title ?? \_\_('filament-panels::pages/dashboard.title');  }  }<?php  namespace App\Filament\Resources;  use App\Enums\AttendanceStatus;  use App\Enums\AttendanceType;  use App\Enums\UserType;  use App\Filament\Resources |

1. *Listing* Program *Dashboard*

|  |
| --- |
| <?php  namespace App\Filament\Pages;  use Filament\Facades\Filament;  use Filament\Pages\Page;  use Filament\Support\Facades\FilamentIcon;  use Filament\Widgets\Widget;  use Filament\Widgets\WidgetConfiguration;  use Illuminate\Contracts\Support\Htmlable;  class Dashboard extends Page  {      protected static string $routePath = '/';      protected static ?int $navigationSort = -2;      /\*\*       \* @var view-string       \*/      protected static string $view = 'filament-panels::pages.dashboard';      public static function getNavigationLabel(): string      {          return static::$navigationLabel ??              static::$title ??              \_\_('filament-panels::pages/dashboard.title');      }      public static function getNavigationIcon(): string | Htmlable | null      {          return static::$navigationIcon              ?? FilamentIcon::resolve('panels::pages.dashboard.navigation-item')              ?? (Filament::hasTopNavigation() ? 'heroicon-m-home' : 'heroicon-o-home');      }      public static function getRoutePath(): string      {          return static::$routePath;      }      /\*\*       \* @return array<class-string<Widget> | WidgetConfiguration>       \*/      public function getWidgets(): array      {          return Filament::getWidgets();      }      /\*\*       \* @return array<class-string<Widget> | WidgetConfiguration>       \*/      public function getVisibleWidgets(): array      {          return $this->filterVisibleWidgets($this->getWidgets());      }      /\*\*       \* @return int | string | array<string, int | string | null>       \*/      public function getColumns(): int | string | array      {          return 2;      }      public function getTitle(): string | Htmlable      {          return static::$title ?? \_\_('filament-panels::pages/dashboard.title');      }  } |

1. *Listing* Program *List* *Attendance*

|  |
| --- |
| <?php  namespace App\Filament\Resources\AttendanceResource\Pages;  use App\Enums\AttendanceStatus;  use App\Enums\AttendanceTimeOff;  use App\Enums\AttendanceType;  use App\Enums\UserType;  use App\Filament\Resources\AttendanceResource;  use App\Models\Attendance;  use Filament\Actions;  use Filament\Actions\Action;  use Filament\Forms\Components\DatePicker;  use Filament\Forms\Components\Select;  use Filament\Forms\Components\Textarea;  use Filament\Notifications\Notification;  use Filament\Resources\Pages\ListRecords;  use Illuminate\Support\Facades\DB;  use Illuminate\Support\Facades\Log;  use pxlrbt\FilamentExcel\Actions\Pages\ExportAction;  use pxlrbt\FilamentExcel\Exports\ExcelExport;  class ListAttendances extends ListRecords  {      protected static string $resource = AttendanceResource::class;      protected static string $view = 'filament.resources.attendance-resource.pages.list-records';      public $imageDataUrl;      public $latitude;      public $longitude;      public $hasClockin;      public $hasClockout;      public function mount(): void      {          parent::mount();          $user = auth()->user();          $clockin = Attendance::query()              ->whereDate('recorded\_at', now()->format('Y-m-d'))              ->where('user\_id', $user->id)              ->where('type', AttendanceType::ClockIn->value)              ->where('status', AttendanceStatus::ClockIn->value)              ->exists();          $clockout = Attendance::query()              ->whereDate('recorded\_at', now()->format('Y-m-d'))              ->where('user\_id', $user->id)              ->where('type', AttendanceType::ClockOut->value)              ->where('status', AttendanceStatus::ClockOut->value)              ->exists();          $this->hasClockin = $clockin;          $this->hasClockout = $clockout;      }      protected function getHeaderActions(): array      {          return [              // Actions\CreateAction::make(),              ExportAction::make()                  ->exports([                      ExcelExport::make('table')->fromTable(),                  ])                  ->label('Unduh')              ->hidden(fn () => auth()->user()->type != UserType::Admin),              Action::make('requestTimeoff')                  ->label(trans('admin.actions.request\_timeoff'))                  ->translateLabel()                  ->color('danger')                  ->modalIconColor('danger')                  ->modalSubmitActionLabel('Submit')                  ->requiresConfirmation()                  ->hidden(fn () => !in\_array(auth()->user()->type, [UserType::Guru]))                  ->form([                      Select::make('type')                          ->label(trans('admin.fields.type'))                          ->translateLabel()                          ->options(fn () => AttendanceTimeOff::class)                          ->searchable()                          ->preload()                          ->required(),                      DatePicker::make('recorded\_at')                          ->label(trans('admin.fields.recorded\_at'))                          ->translateLabel()                          ->required(),                      Textarea::make('notes')                          ->label(trans('admin.fields.notes'))                          ->translateLabel()                          ->required(),                  ])                  ->action(function ($data) {                      switch($data['type']) {                          case AttendanceTimeOff::Sick->value:                              $status = AttendanceStatus::SickPending;                              break;                          case AttendanceTimeOff::Leave->value:                              $status = AttendanceStatus::LeavePending;                              break;                          case AttendanceTimeOff::Permit->value:                              $status = AttendanceStatus::PermitPending;                              break;                          default:                              $status = AttendanceStatus::Absent;                              break;                      }                        DB::beginTransaction();                      try {                            $userId = auth()->user()->id;                          $attendance = Attendance::create([                              'user\_id' => $userId,                              'type' => $data['type'],                              'status' => $status,                              'recorded\_at' => $data['recorded\_at'],                              'notes' => $data['notes'],                          ]);                          DB::commit();                          Notification::make()                              ->title('Request submitted')                              ->success()                              ->send();                          return;                      } catch (\Throwable $throw) {                          DB::rollBack();                          Notification::make()                              ->title($throw->getMessage())                              ->danger()                              ->send();                          Log::error($throw);                          return;                      }                  })          ];      }      public function clockinSave()      {          if(empty($this->latitude) || empty($this->longitude)) {              Notification::make()                  ->title('Your GPS location cannot be located')                  ->danger()                  ->send();              return $this->redirectRoute('filament.admin.resources.attendances.index');          }          DB::beginTransaction();          try {              $userId = auth()->user()->id;              $attendance = Attendance::create([                  'user\_id' => $userId,                  'type' => AttendanceType::ClockIn,                  'status' => AttendanceStatus::ClockIn,                  'latitude' => $this->latitude,                  'longitude' => $this->longitude,                  'recorded\_at' => now(),              ]);              $filename = $userId . '-clockin-' . now()->timestamp . '.jpg';              $attendance->addMediaFromBase64($this->imageDataUrl)                  ->usingFileName($filename)                  ->toMediaCollection('clockin');              DB::commit();              Notification::make()                  ->title('Clockin Sukses')                  ->success()                  ->send();              return $this->redirectRoute('filament.admin.resources.attendances.index');          } catch (\Throwable $throw) {              DB::rollBack();              Notification::make()                  ->title($throw->getMessage())                  ->danger()                  ->send();              Log::error($throw);              return $this->redirectRoute('filament.admin.resources.attendances.index');          }      }      public function clockoutSave()      {          if(empty($this->latitude) || empty($this->longitude)) {              Notification::make()                  ->title('Your GPS location cannot be located')                  ->danger()                  ->send();              return $this->redirectRoute('filament.admin.resources.attendances.index');          }          DB::beginTransaction();          try {              $userId = auth()->user()->id;              $attendance = Attendance::create([                  'user\_id' => $userId,                  'type' => AttendanceType::ClockOut,                  'status' => AttendanceStatus::ClockOut,                  'recorded\_at' => now(),                  'latitude' => $this->latitude,                  'longitude' => $this->longitude,              ]);              $filename = $userId . '-clockout-' . now()->timestamp . '.jpg';              $attendance->addMediaFromBase64($this->imageDataUrl)                  ->usingFileName($filename)                  ->toMediaCollection('clockout');              DB::commit();              Notification::make()                  ->title('Clockout Sukses')                  ->success()                  ->send();              return $this->redirectRoute('filament.admin.resources.attendances.index');          } catch (\Throwable $throw) {              DB::rollBack();              Notification::make()                  ->title($throw->getMessage())                  ->danger()                  ->send();              Log::error($throw);              return $this->redirectRoute('filament.admin.resources.attendances.index');          }      }  } |

1. *Listing* Program *Admin* *Resource*

|  |
| --- |
| <?php  namespace App\Filament\Resources;  use App\Enums\UserType;  use App\Filament\Resources\AdminResource\Pages;  use App\Filament\Resources\AdminResource\RelationManagers;  use App\Models\Admin;  use Filament\Forms;  use Filament\Forms\Form;  use Filament\Resources\Resource;  use Filament\Tables;  use Filament\Tables\Table;  use Illuminate\Database\Eloquent\Builder;  use Illuminate\Database\Eloquent\SoftDeletingScope;  use function Filament\Support\get\_model\_label;  class AdminResource extends Resource  {      protected static ?string $model = Admin::class;      protected static ?string $navigationIcon = 'heroicon-o-rectangle-stack';      public static function canViewAny(): bool      {          return auth()->user()->type == UserType::Admin;      }      public static function getNavigationGroup(): ?string      {          return trans('admin.menu.master');      }      public static function getNavigationLabel(): string      {          return static::$navigationLabel ?? static::getTitleCasePluralModelLabel();      }      public static function getNavigationBadge(): ?string      {          return null;      }      public static function getNavigationBadgeTooltip(): ?string      {          return static::$navigationBadgeTooltip;      }      public static function getNavigationBadgeColor(): string | array | null      {          return null;      }      public static function shouldRegisterNavigation(): bool      {          return auth()->user()->type == UserType::Admin;      }      public static function getModelLabel(): string      {          return static::$modelLabel ?? static::getLabel() ?? get\_model\_label(static::getModel());      }      public static function form(Form $form): Form      {          return $form              ->schema([                  Forms\Components\TextInput::make('type')                      ->label(trans('admin.fields.type'))                      ->translateLabel()                      ->maxLength(255)                      ->default('admin'),                  Forms\Components\TextInput::make('name')                      ->label(trans('admin.fields.name'))                      ->translateLabel()                      ->required()                      ->maxLength(255),                  Forms\Components\TextInput::make('email')                      ->email()                      ->required()                      ->maxLength(255),                  Forms\Components\TextInput::make('nrp')                      ->maxLength(255),                  Forms\Components\TextInput::make('birth\_place')                      ->label(trans('admin.fields.birth\_place'))                      ->translateLabel()                      ->maxLength(255),                  Forms\Components\DatePicker::make('birth\_date')                      ->label(trans('admin.fields.birth\_date'))                      ->translateLabel(),                  Forms\Components\TextInput::make('latitude')                      ->maxLength(255),                  Forms\Components\TextInput::make('longitude')                      ->maxLength(255),                  Forms\Components\DateTimePicker::make('email\_verified\_at')                      ->label(trans('admin.fields.email\_verified\_at'))                      ->translateLabel(),                  Forms\Components\TextInput::make('password')                      ->label(trans('admin.fields.password'))                      ->translateLabel()                      ->password()                      ->required()                      ->maxLength(255),              ]);      }      public static function table(Table $table): Table      {          return $table              ->columns([                  Tables\Columns\TextColumn::make('no')                      ->rowIndex(),                  Tables\Columns\TextColumn::make('name')                      ->label(trans('admin.fields.name'))                      ->translateLabel()                      ->searchable(),                  Tables\Columns\TextColumn::make('email')                      ->searchable(),                  Tables\Columns\TextColumn::make('email\_verified\_at')                      ->label(trans('admin.fields.email\_verified\_at'))                      ->translateLabel()                      ->dateTime()                      ->sortable()                      ->toggleable(isToggledHiddenByDefault: true),                  Tables\Columns\TextColumn::make('created\_at')                      ->dateTime()                      ->sortable()                      ->toggleable(isToggledHiddenByDefault: true),                  Tables\Columns\TextColumn::make('updated\_at')                      ->dateTime()                      ->sortable()                      ->toggleable(isToggledHiddenByDefault: true),                  Tables\Columns\TextColumn::make('deleted\_at')                      ->dateTime()                      ->sortable()                      ->toggleable(isToggledHiddenByDefault: true),              ])              ->filters([                  Tables\Filters\TrashedFilter::make(),              ])              ->actions([                  Tables\Actions\ViewAction::make(),              ])              ->bulkActions([                  //              ]);      }      public static function getRelations(): array      {          return [              //          ];      }      public static function getPages(): array      {          return [              'index' => Pages\ListAdmins::route('/'),              'create' => Pages\CreateAdmin::route('/create'),              'view' => Pages\ViewAdmin::route('/{record}'),              'edit' => Pages\EditAdmin::route('/{record}/edit'),          ];      }      public static function getEloquentQuery(): Builder      {          return parent::getEloquentQuery()              ->withoutGlobalScopes([                  SoftDeletingScope::class,              ]);      }  } |

1. *Listing* Program Guru *Resource*

|  |
| --- |
| <?php  namespace App\Filament\Resources;  use App\Enums\UserType;  use App\Filament\Resources\GuruResource\Pages;  use App\Filament\Resources\GuruResource\RelationManagers;  use App\Models\Guru;  use Filament\Forms;  use Filament\Forms\Form;  use Filament\Resources\Resource;  use Filament\Tables;  use Filament\Tables\Table;  use Illuminate\Database\Eloquent\Builder;  use Illuminate\Database\Eloquent\SoftDeletingScope;  use function Filament\Support\get\_model\_label;  class GuruResource extends Resource  {      protected static ?string $model = Guru::class;      protected static ?string $navigationIcon = 'heroicon-o-rectangle-stack';      public static function canViewAny(): bool      {          return auth()->user()->type == UserType::Admin;      }      public static function getNavigationGroup(): ?string      {          return trans('admin.menu.master');      }      public static function getNavigationLabel(): string      {          return static::$navigationLabel ?? static::getTitleCasePluralModelLabel();      }      public static function getNavigationBadge(): ?string      {          return null;      }      public static function getNavigationBadgeTooltip(): ?string      {          return static::$navigationBadgeTooltip;      }      public static function getNavigationBadgeColor(): string | array | null      {          return null;      }      public static function shouldRegisterNavigation(): bool      {          return auth()->user()->type == UserType::Admin;      }      public static function getModelLabel(): string      {          return static::$modelLabel ?? static::getLabel() ?? get\_model\_label(static::getModel());      }      public static function form(Form $form): Form      {          return $form              ->schema([                  Forms\Components\TextInput::make('name')                      ->label(trans('admin.fields.name'))                      ->translateLabel()                      ->required()                      ->maxLength(255),                  Forms\Components\TextInput::make('email')                      ->email()                      ->required()                      ->maxLength(255),                  Forms\Components\TextInput::make('nrp')                      ->maxLength(255),                  Forms\Components\TextInput::make('birth\_place')                      ->label(trans('admin.fields.birth\_place'))                      ->translateLabel()                      ->maxLength(255),                  Forms\Components\DatePicker::make('birth\_date')                      ->label(trans('admin.fields.birth\_date'))                      ->translateLabel(),                  Forms\Components\TextInput::make('latitude')                      ->maxLength(255),                  Forms\Components\TextInput::make('longitude')                      ->maxLength(255),                  Forms\Components\DateTimePicker::make('email\_verified\_at')                      ->label(trans('admin.fields.email\_verified\_at'))                      ->translateLabel(),                  Forms\Components\TextInput::make('password')                      ->label(trans('admin.fields.password'))                      ->translateLabel()                      ->password()                      ->required()                      ->maxLength(255),              ]);      }      public static function table(Table $table): Table      {          return $table              ->columns([                  Tables\Columns\TextColumn::make('no')                      ->rowIndex(),                  Tables\Columns\TextColumn::make('name')                      ->label(trans('admin.fields.name'))                      ->translateLabel()                      ->searchable(),                  Tables\Columns\TextColumn::make('email')                      ->searchable(),                  Tables\Columns\TextColumn::make('nrp')                      ->searchable(),                  Tables\Columns\TextColumn::make('birth\_place')                      ->label(trans('admin.fields.birth\_place'))                      ->translateLabel()                      ->searchable()                      ->toggleable(isToggledHiddenByDefault: true),                  Tables\Columns\TextColumn::make('birth\_date')                      ->label(trans('admin.fields.birth\_date'))                      ->translateLabel()                      ->date()                      ->sortable()                      ->toggleable(isToggledHiddenByDefault: true),                  Tables\Columns\TextColumn::make('latitude')                      ->searchable()                      ->toggleable(isToggledHiddenByDefault: true),                  Tables\Columns\TextColumn::make('longitude')                      ->searchable()                      ->toggleable(isToggledHiddenByDefault: true),                  Tables\Columns\TextColumn::make('email\_verified\_at')                      ->dateTime()                      ->sortable()                      ->toggleable(isToggledHiddenByDefault: true),                  Tables\Columns\TextColumn::make('created\_at')                      ->dateTime()                      ->sortable()                      ->toggleable(isToggledHiddenByDefault: true),                  Tables\Columns\TextColumn::make('updated\_at')                      ->dateTime()                      ->sortable()                      ->toggleable(isToggledHiddenByDefault: true),                  Tables\Columns\TextColumn::make('deleted\_at')                      ->dateTime()                      ->sortable()                      ->toggleable(isToggledHiddenByDefault: true),              ])              ->filters([                  Tables\Filters\TrashedFilter::make(),              ])              ->actions([                  Tables\Actions\ViewAction::make(),              ])              ->bulkActions([                  //              ]);      }      public static function getRelations(): array      {          return [              //          ];      }      public static function getPages(): array      {          return [              'index' => Pages\ListGurus::route('/'),              'create' => Pages\CreateGuru::route('/create'),              'view' => Pages\ViewGuru::route('/{record}'),              'edit' => Pages\EditGuru::route('/{record}/edit'),          ];      }      public static function getEloquentQuery(): Builder      {          return parent::getEloquentQuery()              ->withoutGlobalScopes([                  SoftDeletingScope::class,              ]);      }  } |

Lampiran: 5 **: Riwayat Hidup**

