

## Rancang Bangun Aplikasi Presensi Instruktur Berbasis Android Menggunakan Framework Flutter Pada LKP Bright School Lampung Timur

Rohim Kurniawan<sup>1</sup>, Didi Rosiyadi<sup>1,2</sup>, Nila Hardi<sup>3</sup>

<sup>1,3</sup>Universitas Bina Sarana Informatika

<sup>2</sup>Pusat Riset Kecerdasan Artifisial dan Keamanan Siber, BRIN

<sup>1,3</sup>Jl. Kramat Raya No.98 Kec. Senen, Kota Jakarta Pusat, Daerah Khusus Ibukota Jakarta, Indonesia

<sup>2</sup>Jl. Sangkuriang, Dago, Kecamatan Coblong, Kota Bandung, Jawa Barat

email: <sup>1</sup>[rohmkurniawan456@gmail.com](mailto:rohmkurniawan456@gmail.com), <sup>2</sup>[didi.rosiyadi@bsi.ac.id](mailto:didi.rosiyadi@bsi.ac.id), <sup>3</sup>[nila.nad@bsi.ac.id](mailto:nila.nad@bsi.ac.id)

**Abstrak** - Perkembangan teknologi saat ini telah memberikan dampak positif dalam berbagai bidang, termasuk dalam dunia pendidikan. Salah satu perkembangan yang dapat dimanfaatkan adalah penggunaan aplikasi mobile sebagai media pembelajaran. Sudah selanjutnya teknologi informasi diterapkan di instansi pendidikan salah satu contohnya aplikasi mobile dapat menjadi alat yang efektif bagi para instruktur untuk melakukan presensi. Penelitian ini bertujuan untuk mengembangkan sebuah aplikasi berbasis android presensi instruktur menggunakan teknologi *Flutter*. *Flutter* adalah sebuah *framework* open-source yang dikembangkan oleh Google untuk membangun aplikasi mobile *multi-platform*. Metode penelitian dan pengembangan yang digunakan adalah *Agile Methodology* dengan *framework Scrum*. Tahap analisis dilakukan untuk mengidentifikasi kebutuhan pengguna, tahap desain dilakukan untuk merancang antarmuka dan fitur aplikasi, tahap implementasi dilakukan untuk membangun aplikasi, dan tahap evaluasi dilakukan untuk menilai kelayakan aplikasi. Hasil penelitian menunjukkan bahwa aplikasi instruktur yang dikembangkan dengan menggunakan *framework Flutter* dapat berjalan dengan baik pada perangkat mobile berbasis Android. Aplikasi ini menyediakan fitur-fitur yang memudahkan instruktur pada lembaga kursus dalam melakukan presensi kehadiran dan bagi admin lembaga kursus dan pelatihan *Bright School* memudahkan dalam memantau rekam presensi serta mengolah data presensi.

Kata Kunci: *Flutter*, Android, Presensi

**Abstract** - The current development of technology has had a positive impact on various fields, including education. One of the developments that can be utilized is the use of mobile applications as a learning medium. Information technology should be applied in educational institutions, one example being that mobile applications can serve as an effective tool for instructors to manage attendance. This study aims to develop an Android-based attendance application for instructors using *Flutter* technology. *Flutter* is an open-source framework developed by Google for building multi-platform mobile applications. The research and development methodology used is *Agile Methodology* with the *Scrum* framework. The analysis phase was carried out to identify user needs, the design phase was conducted to design the interface and features of the application, the implementation phase was executed to build the application, and the evaluation phase was done to assess the feasibility of the application. The results of the study indicate that the instructor attendance application developed using the *Flutter* framework can run well on Android-based mobile devices. This application provides features that facilitate instructors at training institutions in managing attendance, and for the admin of *Bright School* training institutions, it simplifies monitoring attendance records and processing attendance data.

**Keywords:** *flutter, android, attendance*



## PEBDahuluan

Dengan berkembangnya teknologi informasi, pasti akan berdampak positif pada kehidupan manusia dalam bidang pendidikan, ekonomi, politik, dan budaya. Hal-hal yang akan terdampak tidak jauh dari sistem kesehatan, sistem sosial budaya dan tentu saja pada sistem pendidikan (Anggraeni et al., 2023). Maka dari itu, sudah selayaknya teknologi informasi mulai diimplementasikan pada tiap institusi pendidikan supaya mempermudah manajemen maupun operasional dalam jalannya pendidikan. Salah satunya yaitu dalam manajemen presensi kehadiran siswa maupun pengajar. Menurut Ulumudin, sistem presensi sangat penting dalam kehidupan sehari-hari, terutama di tempat-tempat formal seperti universitas, perkantoran, sekolah, dan tempat lain di mana presensi adalah sebuah kewajiban (Ulumudin et al., 2023). Presensi merupakan proses mencatat dan mengelola data kehadiran yang dilakukan sehari-hari pada waktu jam kerja pada setiap perusahaan (Fazrin et al., 2021). Contohnya, presensi digunakan oleh perusahaan untuk menunjukkan bahwa karyawan hadir di kantor dan digunakan oleh departemen keuangan sebagai faktor penting dalam menentukan upah karyawan.

Banyak bisnis, terutama perusahaan kecil, terus menggunakan presensi manual. Hal ini disayangkan di era digital saat ini karena metode manual mudah dimanipulasi oleh karyawan dan memakan waktu, sehingga tidak efisien (Arafat Febriandirza, 2019).

Jurnal yang ditulis oleh Bayu Adytia Permana pada tahun 2022, menjelaskan bahwa Presensi online adalah sistem pencatatan kehadiran di tempat kerja secara online untuk mempermudah pengelolaan dan pembuatan laporan kehadiran kerja. Sistem ini efisien dan dapat diakses dari mana saja. Sistem ini memanfaatkan perangkat *smartphone* dengan sistem operasi Android, karena Android memiliki penggunaan yang luas dan platform terbuka untuk pengembangan aplikasi di berbagai *smartphone* (Permana, 2022).

Namun masih banyak lembaga, organisasi, atau perusahaan yang masih menggunakan prosedur manual dalam mengelola presensi. Hal ini rentan terhadap kesalahan manusia, seperti kesalahan pencatatan atau manipulasi presensi. Dengan Menyediakan sistem yang otomatis dan akurat, aplikasi presensi dapat menjadi solusi untuk masalah ini.

Salah satu contoh lembaga yang masih menggunakan prosedur manual adalah Lembaga Kursus dan Pelatihan *Bright School*. LKP *Bright School* didirikan di Desa Srimenanti Kecamatan Bandar Sribhawono Kabupaten Lampung Timur pada tahun 2020. Lembaga ini didirikan untuk meningkatkan Sumber Daya Manusia, khususnya dalam bidang Pendidikan Keterampilan Komputer, Bahasa Inggris, dan Pendidikan Nonformal lainnya. LKP *Bright School* mengalami tantangan dalam manajemen presensi kehadiran instruktur, yang

semula dilakukan melalui tanda tangan mandiri, namun pada saat ini telah beralih ke sistem satu orang yang melakukan absensi. Meskipun demikian masih sering terjadi kesalahan dalam pendataan presensi, menumpuknya kertas bekas presensi, dan biaya percetakan kertas yang tidak efisien.

Dengan demikian, perancangan aplikasi presensi menjadi suatu pilihan yang baik bagi LKP *Bright School* Lampung Timur untuk meningkatkan efisiensi dan akurasi dalam manajemen kehadiran instruktur.

## METODE PENELITIAN

### 1. Teknik Pengumpulan Data

#### a. Observasi

Penulis melakukan observasi dengan cara mengamati secara langsung sistem yang dipakai saat ini di LKP *Bright School* dan mengamati beberapa sistem presensi online yang telah digunakan instansi maupun organisasi yang ada.

#### b. Wawancara

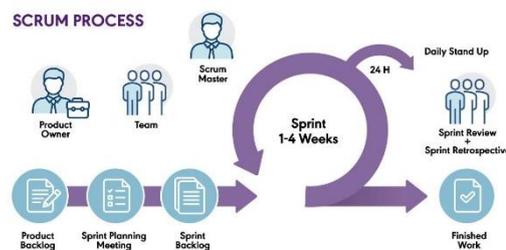
Penulis mewawancarai pemilik LKP *Bright School* untuk mendapatkan data yang mendukung penelitian, dengan data tersebut penulis akan membuat aplikasi yang sesuai dengan kebutuhan lembaga.

#### c. Studi Pustaka

Pada metode ini penulis melakukan studi kepustakaan dengan cara mempelajari, mengumpulkan, serta mengkaji data teoritis maupun praktis dari berbagai sumber referensi seperti buku, dokumentasi, artikel ilmiah, dan jurnal yang terkait dalam penelitian ini. Data yang sudah terkumpulkan akan menjadi dasar dari pengembangan sistem presensi instruktur.

### 2. Metode Pengembangan Perangkat Lunak

Metode yang akan digunakan penulis dalam membuat aplikasi presensi instruktur adalah metode *Agile* dengan penerapan *scrum*. *Agile* merupakan salah satu metode dalam SDLC yang menekankan pada fleksibilitas, responsibilitas terhadap perubahan, dan pengiriman hasil secara iteratif serta inkremental. Metode *scrum* memberikan fleksibilitas dalam pengaturan dan manajemen kebutuhan pengembangan dan pengembangan perangkat lunak (Rizky & Sugiarti, 2022).



Sumber: (Ghani & Andrian, 2023)  
Gambar 1. Scrum

a. *Product Backlog*

Pada tahap awal, tim mengumpulkan semua kebutuhan dan fitur yang diinginkan oleh pemangku kepentingan dan pengguna akhir dalam bentuk *Product Backlog*. Daftar ini berisi elemen-elemen prioritas yang akan dikembangkan sepanjang proses pengembangan (Mustika, 2024).

Dalam Tahapan ini penulis membagi sebuah fitur menjadi beberapa bagian lalu menentukan waktu yang harus dikerjakan untuk menyelesaikan bagian tersebut. Misalnya, dalam aplikasi presensi instruktur terdapat fitur autentikasi. Proses membangun fitur ini dapat dibagi menjadi beberapa tugas, seperti menentukan spesifikasi untuk login yang akurat, merancang tampilan login, dan menulis *code* untuk halaman login. Dari 3 tugas itu kita tentukan berapa lama waktu untuk mengerjakannya.

b. *Sprint Planning*

Pada tahapan *sprint planning*, penulis menentukan *backlog item* mana saja yang akan dimasukkan ke dalam tahapan *Sprint*. Misal dalam *sprint* pertama penulis memasukkan Desain Login Page, Development Login Page, dan *Back-end* untuk login.

c. *Sprint*

Dalam metodologi *scrum*, *sprint* adalah periode waktu yang singkat di mana tim *scrum* bekerja untuk menyelesaikan sejumlah tugas. Penulis menetapkan 2 minggu sebagai durasi sebuah *sprint* dalam penelitian ini. Pada tahapan ini penulis mengerjakan tugas-tugas atau *backlog item* yang sudah ditentukan pada tahapan *Sprint Planning*.

d. *Sprint Review*

*Sprint Review* merupakan tahapan untuk mempresentasikan mengenai tugas yang sudah dikerjakan dan menunjukkan kemajuan pengembangan produk yang dikerjakan. Pada tahapan ini penulis hanya melakukan testing pada fitur yang sudah dibuat.

e. *Retrospective*

*Sprint retrospective* adalah tahap yang dilakukan setelah *sprint* selesai, yang bertujuan untuk mengevaluasi kinerja tim secara keseluruhan (Amarta & Anugrah, 2021).

Pada dasarnya tahap ini, development team akan merefleksikan pekerjaan yang telah dilakukan pada *sprint* sebelumnya. *Scrum master* akan membantu menunjukkan masalah dan improvement yang perlu dilakukan pada *sprint* berikutnya. Namun, karena penulis membangun aplikasi presensi instruktur ini sendirian, penulis hanya memikirkan hal-hal apa yang harus diperbaiki.

**HASIL DAN PEMBAHASAN**

**1. Analisis Kebutuhan Fungsional**

Analisis kebutuhan fungsional sangat penting untuk memahami tujuan dari sebuah aplikasi dan menggambarkan fungsi atau layanan spesifik yang harus disediakan oleh sistem aplikasi. Dalam pengembangan aplikasi Android, analisis fungsional melibatkan identifikasi dan dokumentasi fungsi-

fungsi spesifik untuk kepuasan pengguna dan manfaat bisnis. Berikut ini adalah kebutuhan - kebutuhan fungsional dari perancangan aplikasi presensi instruktur yang akan penulis buat:

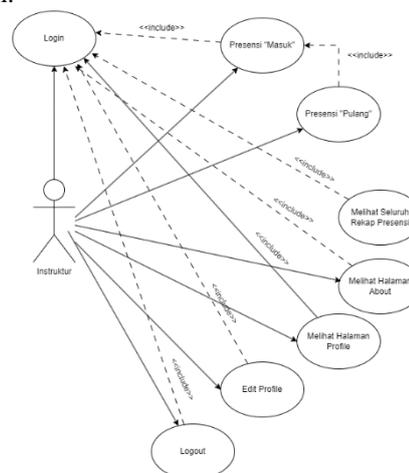
a. Halaman Instruktur

- 1) Instruktur dapat masuk/login kedalam aplikasi dengan account yang sudah diberikan lembaga.
- 2) Instruktur bisa melakukan presensi Masuk.
- 3) Instruktur dapat melakukan presensi Pulang.
- 4) Instruktur dapat melihat data riwayat absensi atau rekap.
- 5) Instruktur dapat melihat informasi account.
- 6) Instruktur dapat mengubah data pada informasi account.
- 7) Instruktur dapat melihat halaman About.
- 8) Instruktur dapat melakukan logout dari aplikasi

b. Halaman Admin

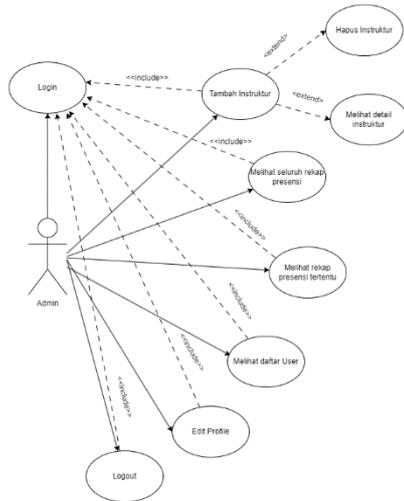
- 1) Admin dapat melakukan login kedalam aplikasi
- 2) Admin dapat melihat rekap presensi seluruh instruktur
- 3) Admin dapat melihat data presensi instruktur tertentu
- 4) Admin dapat melakukan search untuk mencari data yang diinginkan
- 5) Admin dapat melihat daftar seluruh user/instruktur
- 6) Admin bisa menghapus account instruktur
- 7) Admin dapat membuat/menambah account instruktur
- 8) Admin dapat melihat detail instruktur

Berdasarkan poin-poin kebutuhan fungsional tersebut, penulis telah membuat diagram use case untuk masing-masing aktor. Metode pemodelan yang disebut "*Use Case*" digunakan untuk menggambarkan fungsi yang perlu dilakukan oleh sebuah sistem baru (Adhella & Firdonsyah, 2021). Berikut merupakan *Use Case* Instruktur dan *Use Case* Admin.



Sumber: Peneliti (2024)

Gambar 2. Use Case Instruktur



Sumber: Peneliti (2024)

Gambar 3. Use Case Admin

## 2. Analisis Kebutuhan Non-Fungsional

Kebutuhan non-fungsional membahas aspek-aspek yang tidak terkait dengan sistem, memastikan sistem beroperasi dengan baik dalam kondisi tertentu dan memenuhi standar kualitas yang ditentukan. Analisis ini berfokus pada karakteristik aplikasi yang tidak terkait dengan fungsi-fungsinya, melainkan bagaimana aplikasi tersebut akan berperilaku dan berinteraksi dengan penggunanya dan lingkungannya.

### a. Perangkat Keras

Dalam pengerjaan penelitian ini, penulis menggunakan beberapa perangkat keras untuk mendukung proses penelitian dan pembuatan aplikasi presensi instruktur. Berikut merupakan perangkat keras yang penulis gunakan:

Tabel 1. Perangkat Keras yang Digunakan

Nama	Spesifikasi	Fungsi
Laptop HP	- Processor AMD Ryzen 3 2200U - OS Windows 11 - RAM 8GB - System Type 64bit - Penyimpanan 256gb	Untuk membuat jurnal, membuat diagram, dan untuk membuat laporan lainnya.
Macbook Pro	- Processor Intel i7 - macOS Sonoma 14.5 - RAM 16gb - Penyimpanan 256gb	Untuk menulis kode program, melakukan <i>debugging</i> , membuat desain aplikasi, dan pengembangan sistem.
Asus Zenfone MI	- Version Android 9 - RAM 3gb - Penyimpanan 32gb	Untuk melakukan pengujian aplikasi secara langsung di <i>real-device</i> .

Sumber: Peneliti (2024)

### b. Perangkat Lunak

Dalam pembuatan aplikasi presensi instruktur, penulis menggunakan beberapa perangkat lunak yang mendukung dalam proses implementasi. Berikut perangkat lunak yang penulis pakai:

Tabel 2. Perangkat Lunak yang Digunakan

Nama	Versi	Fungsi
Android Studio	Jellfish   2023.3.1	Untuk membuat aplikasi, alat penulisan kode, dan untuk menjalankan emulator.
Visual Studio Code	1.90.0	Untuk pengembangan aplikasi dan penulisan kode program.
Postman	11.1.0	Untuk pengujian API yang telah dibuat, dengan method POST, PUT, GET, dan DELETE.
MAMP	6.9	Berperan sebagai <i>local server</i> dan menjalankan database.
Flutter	3.1.4	Sebagai <i>framework</i> yang digunakan untuk pengembangan aplikasi dan sebagai <i>front-end</i> .
Dart	3.2.3	Sebagai bahasa pemrograman yang digunakan untuk bagian <i>front-end</i>
Laravel	8.83.27	Sebagai <i>framework</i> untuk pembuatan API dan sebagai back-end,
PHP	8.3.4	Sebagai bahasa pemrograman untuk pembuatan API.
MySql	-	Sebagai database yang digunakan.
Figma	-	Digunakan untuk membuat desain aplikasi dan pembuatan <i>mock-up</i> aplikasi.
Chrome	-	Sebagai alat untuk mencari informasi yang membantu dalam pengembangan aplikasi dan pembuatan laporan skripsi.

Sumber: Peneliti (2024)

## 3. Analisis Kebutuhan Pengguna

Analisa kebutuhan pengguna atau sumber daya manusia adalah proses menentukan apa saja yang akan dibutuhkan oleh pembuat aplikasi maupun pengguna akhir. Sumber daya manusia itu sebagai berikut.

### a. Sistem Analis

Sistem analis bertanggung jawab untuk menganalisis sistem yang akan dibuat dan melakukan studi kelayakan pada desain yang sudah dirancang. Dalam tahapan ini, penulis sendiri yang menjadi sistem analis. Tugas sistem analis diantaranya adalah:

- 1) Menentukan kebutuhan fungsional dan non-fungsional untuk aplikasi atau sistem yang akan dibuat.
- 2) Mengevaluasi alur kerja sistem yang ada dan menentukan apa saja yang harus ditambah atau diperbaiki.
- 3) Menentukan spesifikasi teknis dan persyaratan sistem yang harus dipenuhi.

### b. Programmer

Dalam penelitian ini, Programmer bertanggung jawab untuk membuat kode program, membuat desain, dan menentukan perangkat lunak yang akan digunakan. Berikut tugas utama programmer:

- 1) Membuat kode dan menguji kode aplikasi presensi instruktur berbasis Android.
- 2) Membuat fitur yang sudah dianalisis dan disetujui.
- 3) Melakukan perawatan sistem.

#### c. Pengguna (*End-User*)

Pengguna atau end-user pada penelitian ini adalah sebuah lembaga kursus dan pendidikan yang bernama *Bright School*. Pengguna utama aplikasi ini meliputi:

- 1) Admin yang bertugas untuk mengelola data instruktur dan mengelola data presensi.
- 2) Instruktur berperan sebagai pengguna yang dapat melakukan presensi dan dapat melihat rekap dari presensi yang dilakukan.

### 4. Skenario Program

Dalam aplikasi presensi instruktur, penulis membuat beberapa fitur yang dirancang untuk memberikan pengalaman yang optimal, memudahkan akses, dan mempercepat proses presensi serta pengolahan data. Berikut adalah skenario program yang penulis buat, untuk memberikan penjelasan yang menyeluruh mengenai fitur-fitur tersebut.

#### a. Fitur Presensi

Fitur ini memungkinkan instruktur melakukan presensi tanpa harus melakukan tanda tangan sebagai bukti kehadiran atau tanpa harus melakukan proses presensi manual lainnya. Instruktur cukup membuka aplikasi presensi instruktur (*ToBright*) lalu tekan satu dua kali, maka presensi sudah dilakukan.

#### b. Fitur Rekap

Dalam aplikasi presensi ini, terdapat fitur rekap yang akan membantu instruktur maupun admin untuk memantau Riwayat presensi. Bagi admin fitur ini bukan hanya untuk mengecek riwayat saja tapi data yang ada dapat diolah sebagai bahan acuan untuk melakukan pembayaran gaji.

#### c. Fitur Login

Fitur ini sangat berperan penting, karena jika instruktur dan admin tidak dapat login maka fitur-fitur yang tersedia tidak dapat di akses. Ini merupakan salah satu sistem keamanan aplikasi karena jika seseorang dari luar instansi mencoba untuk menggunakan aplikasi ini dikhawatirkan akan menyalahgunakan aplikasi. Jadi hanya orang yang mendapatkan akun dari LKP *Bright School* saja yang dapat mengakses aplikasi ini.

#### d. Fitur Edit Profile

Fitur Edit Profile memungkinkan instruktur mengubah data yang sudah ada atau data *default* yang dibuat admin. Data *default* yang sudah diisi berupa nama dan role sisanya dapat diperbaharui langsung oleh instruktur.

#### e. Fitur Tambah User

Fitur ini hanya bisa diakses oleh admin, karena aplikasi merupakan sistem yang dibangun khusus untuk LKP *Bright School*, maka tidak ada fitur registrasi di bagian awal aplikasi. *User* dapat mengakses aplikasi jika sudah menerima akun dari admin, akun admin didapat dengan menggunakan *seed* di laravel.

#### f. Fitur Hapus User

Hapus *user* merupakan fitur khusus untuk admin, sama halnya dengan fitur tambah. *User* yang sudah dijelaskan tapi fitur ini memiliki fungsi sebaliknya yaitu untuk menghapus akun.

#### g. Fitur Logout

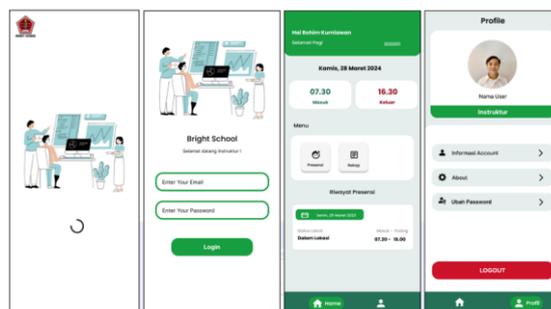
Dalam sebuah aplikasi yang memiliki fitur login pasti akan ada fitur logout, ini merupakan sebuah kewajiban karena tugas dari fungsi logout yaitu untuk mengeluarkan akun dari sebuah aplikasi dan menghapus sesi yang sudah tersimpan dipenyimpanan lokal user.

### 5. Perancangan dan Implementasi Aplikasi

#### A. Perancangan Aplikasi

Salah satu tahap penting dalam pengembangan aplikasi mobile adalah perancangan program aplikasi, dimana pengembang dan desainer bekerja sama untuk membuat rencana awal dan prototipe tampilan dari awal hingga akhir aplikasi. Pada penelitian ini penulis merancang sendiri desain dari aplikasi presensi instruktur menggunakan *Figma*. Berikut merupakan rancangan desain yang sudah penulis buat.

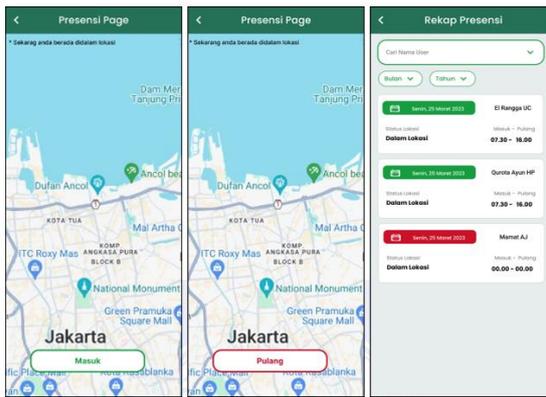
#### a. Desain Halaman *Splash Screen*, *Login*, *Home*, dan *Profil* Pada Tampilan Instruktur



Sumber: Peneliti (2024)

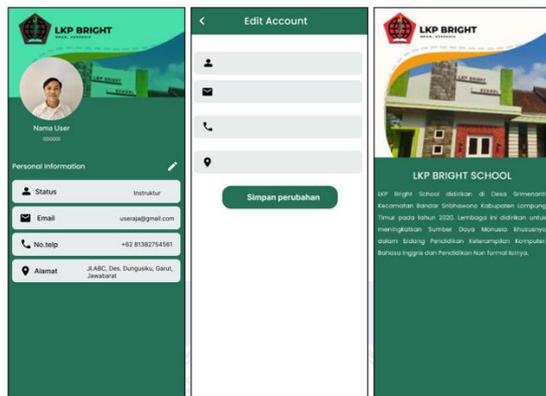
Gambar 4. Desain *Splash Screen*, *Login*, *Home*, dan *Profile*

#### b. Desain Halaman Presensi Masuk, Pulang, dan Rekap Presensi Pada Tampilan Instruktur



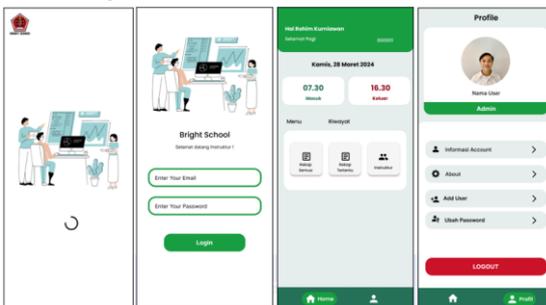
Sumber: Peneliti (2024)  
Gambar 5. Desain Presensi Masuk, Pulang, dan Rekap Presensi

c. Desain Halaman Informasi Akun, Edit Akun, dan About Pada Tampilan Instruktur



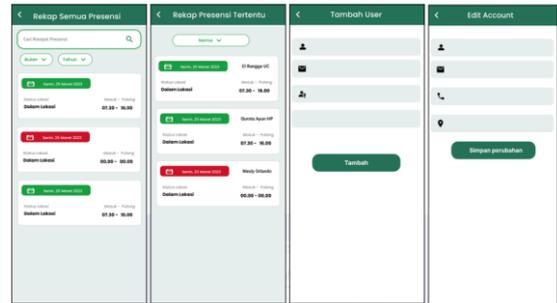
Sumber: Peneliti (2024)  
Gambar 6. Desain Informasi Akun, Edit Akun, dan About

d. Desain Halaman Splash, Login, Home, dan Profil Pada Tampilan Admin



Sumber: Peneliti (2024)  
Gambar 7. Desain Halaman Splash, Login, Home, dan Profile pada tampilan Admin

e. Desain Halaman Rekap Semua, Rekap tertentu, Tambah User, dan Edit Akun Pada Tampilan Admin



Sumber: Peneliti (2024)  
Gambar 8. Desain Rekap Semua, Rekap Tertentu, Tambah User, dan Edit Akun

## B. Implementasi

Pada tahapan ini penulis akan menampilkan hasil implementasi dari desain perancangan yang sudah dibuat kedalam sebuah aplikasi android, mencakup berbagai langkah, mulai dari tampilan awal hingga tampilan *output* akhir dari aplikasi mobile.

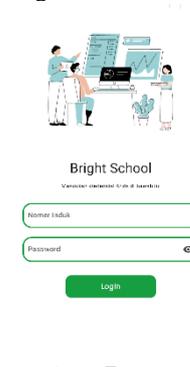
a. Implementasi *Splash Screen*



Sumber: Peneliti (2024)  
Gambar 9. Implementasi Halaman Splash

Gambar 9 merupakan tampilan dari halaman *Splash Screen*, halaman ini menampilkan 2 gambar dan circle loading icon yang akan tampil selama 2 detik. Ini merupakan halaman pertama yang akan dilihat user ketika membuka aplikasi.

b. Implementasi Login



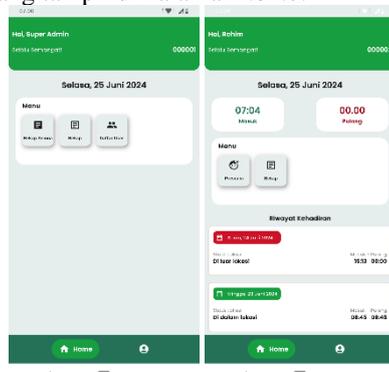
Sumber: Peneliti (2023)  
Gambar 10 Implementasi Halaman Login

Pada halaman login, user harus memasukan Nomer Induk dan Password yang sudah diberikan

oleh admin. Jika instruktur belum mendapatkan nomer induk, maka bisa meminta kepada bagian admin LKP *Bright School*. Ketika user menekan tombol login, sistem akan melakukan pengecekan terhadap nomer induk dan password melalui metode GET untuk mengakses API. Jika nomer induk dan password ada serta sesuai maka akan berpindah ke halaman Home. Jika gagal maka sistem akan menampilkan pesan melalui snackbar.

### c. Implementasi Halaman Home

Pada halaman *Home* sistem akan melakukan pengecekan pada role *User* yang login, Jika role *user* adalah admin maka tampilan yang muncul akan seperti gambar di sisi kiri, admin dapat mengakses tiga menu yang ada di halaman *Home*. Jika *user* rolenya instruktur maka halaman yang tampil seperti gambar sisi kanan, instruktur dapat mengakses 2 menu yang tampil di halaman *home*.



Sumber: Peneliti (2024)

Gambar 11. Implementasi Halaman Home

### d. Implementasi Presensi Masuk

Halaman presensi dapat diakses di menu *Home* dan juga menu ini hanya bisa diakses oleh Instruktur. Halaman ini akan menampilkan peta lokasi terkini *user*, jadi *user* harus memberikan akses lokasi dan menyalakan lokasi untuk dapat mengakses menu ini. Pada menu presensi akan melakukan pengecekan terhadap data presensi, bila data presensi untuk hari ini masih kosong maka tombol akan bertuliskan *Masuk*.



Sumber: Peneliti (2024)

Gambar 12. Implementasi Presensi Masuk

### e. Implementasi Presensi Pulang

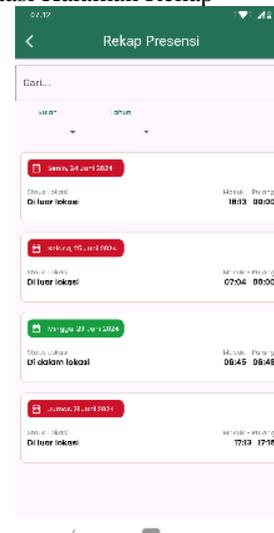
Halaman presensi pulang tidak jauh berbeda seperti halaman presensi masuk. Sebenarnya Presensi Masuk dan Presensi Pulang adalah satu halaman yang sama hanya saja pada halaman ini sistem melakukan pengecekan data presensi hari ini, jika data presensi masuk sudah ada nilai maka tombol akan berubah menjadi "Pulang".



Sumber: Peneliti (2024)

Gambar 13. Implementasi Presensi Pulang

### f. Implementasi Halaman Rekap

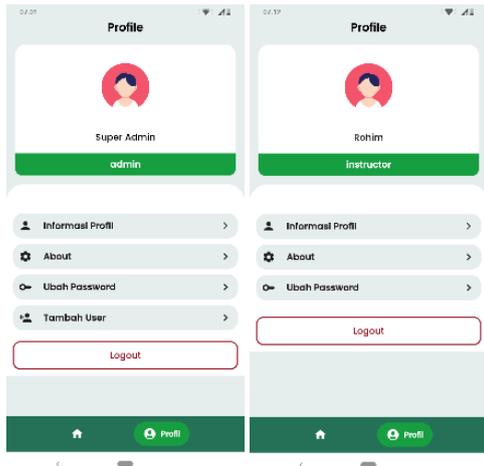


Sumber: Peneliti (2024)

Gambar 14. Implementasi Halaman Rekap

Halaman ini merupakan halaman yang hanya dapat diakses oleh Instruktur, halaman ini menampilkan semua data presensi yang sudah dilakukan instruktur.

### g. Halaman Profile



Sumber: Peneliti (2024)  
Gambar 15. Implementasi Halaman Profile

Pada halaman Profil sistem akan menampilkan *user* sesuai dengan data login. Pada halaman ini jika *user* memiliki role *Admin* maka tampilan akan seperti gambar pada sisi kiri dan jika role instruktur akan seperti gambar pada sisi kanan.

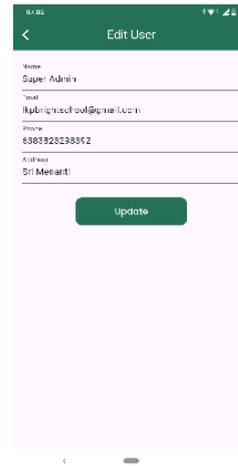
h. Implementasi Informasi Profile



Sumber: Peneliti (2024)  
Gambar 16. Implementasi Informasi Profile

Halaman Informasi Profile dapat diakses ketika *user* menekan menu informasi profile pada halaman Profil. Halaman informasi akun akan menampilkan data detail *user*, datanya itu berupa nomer induk, nama, role, email, nomer telepon, dan alamat.

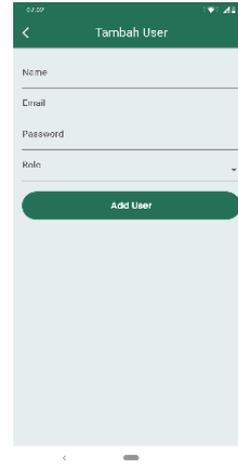
i. Implementasi Edit Profile



Sumber: Peneliti (2024)  
Gambar 17. Implementasi Edit Profile

Halaman Edit Profile merupakan halaman yang dapat diakses di halaman informasi profile dengan cara mengklik *icon* edit. Disini *user* dapat mengedit informasi sesuai kebutuhan, jika salah satu kolom ada yang belum di isi data tidak akan dikirim, jadi *user* harus memastikan semua kolom sudah terisi dengan benar.

j. Implementasi Tambah User



Sumber: Peneliti (2024)  
Gambar 18. Implementasi Tambah User

Halaman *Tambah User* hanya dapat diakses oleh role admin, disini admin dapat menambahkan akun untuk instruktur yang bekerja di lembaga pelatihan dan kursus *Bright School*. Halaman ini dapat diakses dengan menekan menu *tambah user* pada halaman Profile.

k. Implementasi Halaman About

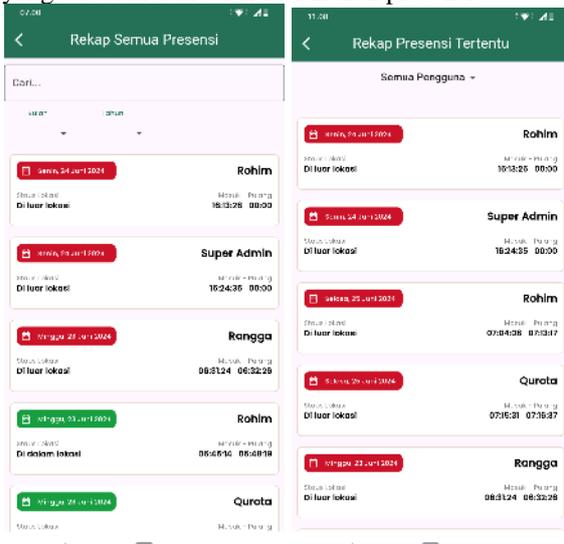


Sumber: Peneliti (2024)  
Gambar 19. Implementasi Halaman About

Halaman *About* adalah halaman yang menampilkan deskripsi tentang LKP *Bright School*. Halaman ini dapat diakses pada menu *about* yang ada pada halaman *Profile*

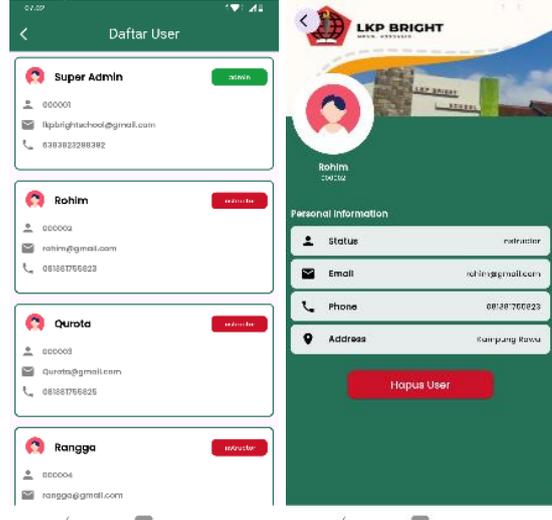
l. Implementasi Halaman Rekap Semua dan Tertentu

Halaman Rekap Semua merupakan halaman yang hanya dapat diakses oleh admin, halaman ini menampilkan seluruh data presensi instruktur yang terdaftar. Admin dapat memantau presensi di halaman ini, admin dapat menggunakan fitur pencarian maupun *filter* yang sudah tersedia. Berbeda dengan halaman Rekap Semua, halaman Rekap Tertentu menampilkan data sesuai dengan nama yang dipilih oleh admin pada tombol *dropdown*. Ketika tombol *dropdown* di klik akan muncul daftar nama instruktur yang terdaftar dan telah melakukan presensi.



Sumber: Peneliti (2024)  
Gambar 20. Implementasi Rekap Semua dan Tertentu

m. Implementasi Daftar dan Detail Instruktur



Sumber: Peneliti (2024)  
Gambar 21. Implementasi Halaman

Halaman *Daftar User* hanya dapat diakses oleh user yang memiliki role admin, halaman ini dapat diakses dengan memilih menu *daftar user* pada halaman *Home*. Terakhir adalah halaman *Detail Instruktur*, halaman ini dapat diakses ketika admin menekan nama yang tampil pada halaman *Daftar User*. Pada halaman ini sistem menampilkan detail data dari user yang dipilih dan ada tombol untuk menghapus tombol tersebut.

**KESIMPULAN**

Dengan adanya aplikasi presensi instruktur (ToBright), instruktur akan lebih mudah dalam melakukan presensi. Instruktur hanya perlu membuka aplikasi ToBright, mengakses fitur presensi, lalu mengklik tombol masuk. Data kehadiran akan langsung tersimpan. Selain itu, fitur lihat rekap dalam aplikasi ini membantu instruktur mengetahui jumlah kehadiran mereka, yang bisa dijadikan acuan dalam penerimaan gaji. Dan bagi LKP *Bright School*, aplikasi ToBright akan memudahkan admin dalam mengelola daftar instruktur dan kehadiran mereka. Aplikasi ini mempermudah pemantauan presensi instruktur setiap harinya. Admin dapat melihat lokasi instruktur saat melakukan presensi melalui data yang tersimpan. Data presensi ini akan menjadi acuan dalam proses penggajian instruktur.

**REFERENSI**

Adhella, R., & Firdonsyah, A. (2021). Design of School Information System Using Waterfall Method (Case study: MTsN 8 Bantul). *Procedia of Engineering and Life Science*, 1(2). <https://doi.org/10.21070/pels.v1i2.1020>

Amarta, A. A. F., & Anugrah, I. G. (2021). Implementasi Agile Scrum Dengan Menggunakan Trello Sebagai Manajemen Proyek Di PT Andromedia. *Jurnal Nasional Komputasi Dan Teknologi Informasi (JNKTI)*, 4(6), 528–534. <https://doi.org/10.32672/jnkti.v4i6.3702>

- Anggraeni, M. D., Mucharromah, R., Taqiyya, B. Z., Fadilah, R. E., Mahardika, I. K., & Yusmar, F. (2023). Perkembangan Teknologi Dan Komunikasi Dalam Pendidikan. *FKIP E-PROCEEDING*, 1–5.
- Arafat Febriandirza. (2019). Perancangan Aplikasi Absensi Online Dengan Menggunakan Bahasa Pemrograman Kotlin. *Pseudocode*, 6(1), 53–59.  
[www.ejournal.unib.ac.id/index.php/pseudocode](http://www.ejournal.unib.ac.id/index.php/pseudocode)  
e%0APERANCANGAN
- Fazrin, Q. E., Lisawati, T., Nurhayati, S., Satya, J. B., & Alamsyah, D. (2021). Penerapan Metode Pengembangan Sistem Extreme Programing (XP) Pada Aplikasi Presensi Karyawan dengan QR Code. *Building of Informatics, Technology and Science (BITS)*, 3(3), 164–170.  
<https://doi.org/10.47065/bits.v3i3.1018>
- Ghani, A. I. Al, & Andrian, R. (2023). Pengembangan Presensee: Aplikasi Presensi Mahasiswa Mobile Menggunakan Framework Flutter (Studi Kasus: Studi Independen Alterra Academy. *Jurnal Media Infotama*, 19(2), 447–453. <https://doi.org/10.37676/jmi.v19i2.4351>
- Mustika, A. (2024). Journal of Data Science and Information System (DIMIS) Permodelan Sistem Informasi Penjualan Barang Menggunakan Metode Scrum. *Journal of Data Science and Information System (DIMIS)*, 2(1), 1–8. <https://doi.org/10.58602/dimis.v2i1.97>
- Permana, B. A. (2022). Aplikasi Presensi Online Menggunakan Validasi Jarak Lokasi Pengguna Berbasis Android. *Jurnal Informatika Dan Rekayasa Perangkat Lunak*, 3(1), 86–92. <https://doi.org/10.33365/jatika.v3i1.1865>
- Rizky, M., & Sugiarti, Y. (2022). Penggunaan Metode Scrum Dalam Pengembangan Perangkat Lunak: Literature Review. *Journal of Computer Science and Engineering (JCSE)*, 3(1), 41–48. <https://doi.org/10.36596/jcse.v3i1.353>
- Ulumudin, I., Faizah, N., & Nurcahyo, W. (2023). Aplikasi Sistem Presensi Pegawai PT. Berkah Pena Ilmu dengan Metode Location Based Service (LBS) Berbasis Android Menggunakan Firebase. *Design Journal*, 1(1), 89–98. <https://doi.org/10.58477/dj.v1i1.61>