

## 1. Aktor

Aktor mempresentasikan seseorang atau sesuatu (seperti perangkat atau sistem lain) yang berinteraksi dengan sistem yang bersangkutan. Pada sistem ini, terdapat 3 aktor yang dapat menggunakan sistem, yaitu:

- **Admin Operasional**

Admin Operasional merupakan admin yang dapat melakukan:

- Login
- Menambahkan, menampilkan, mengubah, dan menghapus data Tenaga Ahli Daerah
- Menambahkan, menampilkan, mengubah, dan menghapus data Peserta Daerah
- Menambahkan, menampilkan, mengubah, dan menghapus data Kegiatan
- Menampilkan data Administrasi Kegiatan
- Menampilkan data Buku Saku
- Menampilkan data SKA
- Menampilkan data Modul

- **Admin Utama**

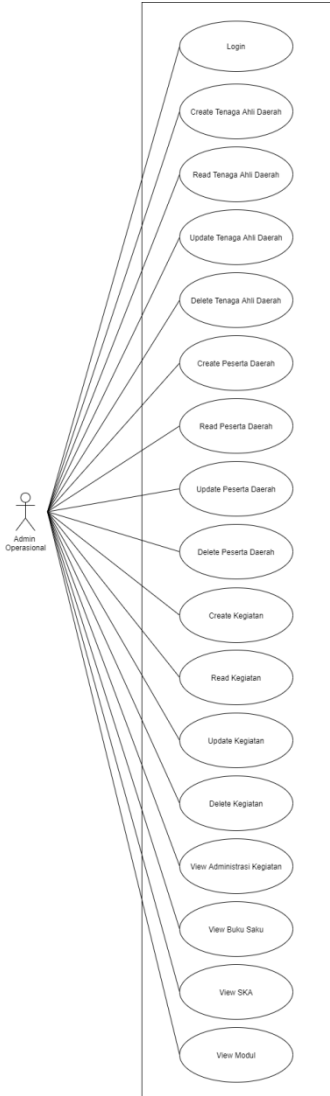
Admin Utama merupakan aktor yang bisa melakukan:

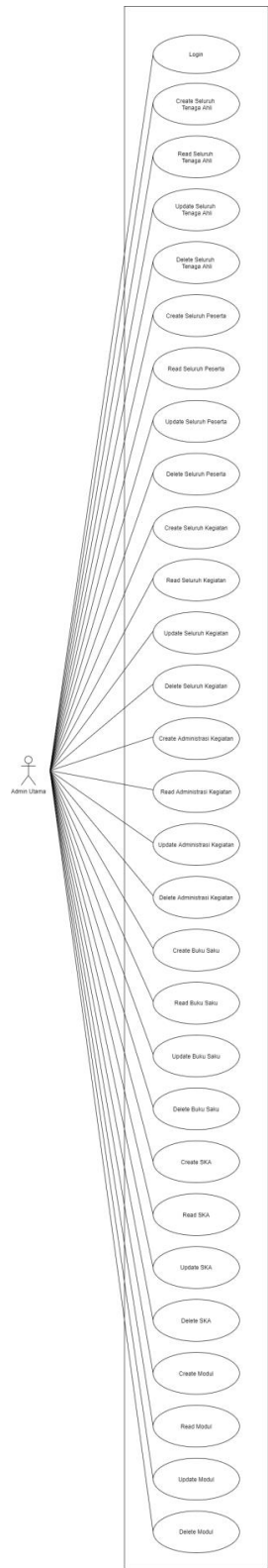
- Login
- Menambahkan, menampilkan, mengubah, dan menghapus seluruh data Tenaga Ahli Daerah
- Menambahkan, menampilkan, mengubah, dan menghapus seluruh data Peserta Daerah
- Menambahkan, menampilkan, mengubah, dan menghapus seluruh data Kegiatan
- Menambahkan, menampilkan, mengubah, dan menghapus seluruh data Administrasi Kegiatan
- Menambahkan, menampilkan, mengubah, dan menghapus seluruh data Buku Saku
- Menambahkan, menampilkan, mengubah, dan menghapus seluruh data SKA

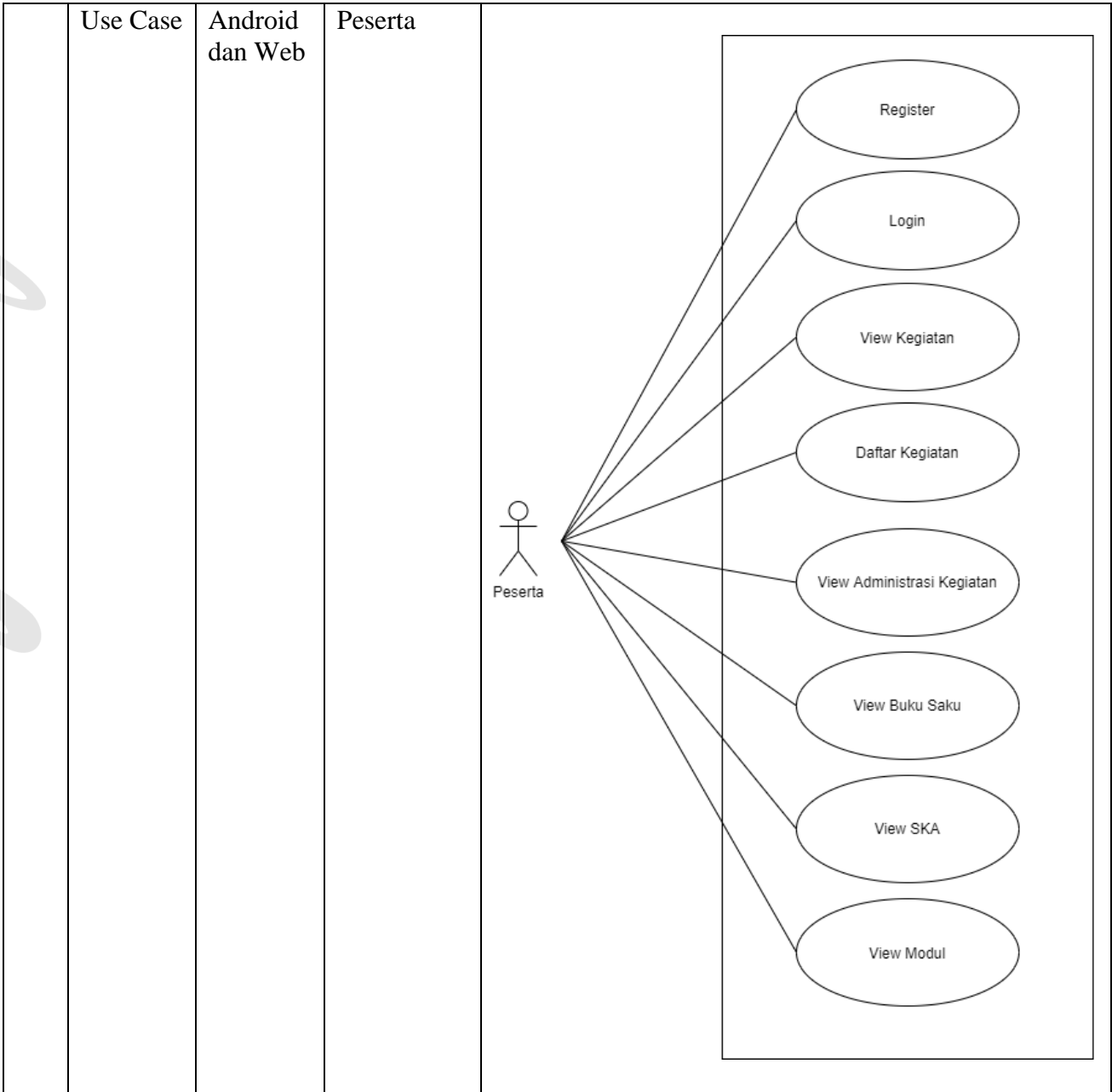
- Menambahkan, menampilkan, mengubah, dan menghapus seluruh data Modul
- **Peserta**  
Peserta dapat melakukan:
  - Register
  - Login
  - Menampilkan dan mendaftar Kegiatan
  - Menampilkan data Administrasi Kegiatan
  - Menampilkan data Buku Saku
  - Menampilkan data SKA
  - Menampilkan data Modul

## 2. Use Case Diagram

Use Case merupakan sebuah teknik yang digunakan dalam pengembangan sebuah software atau sistem informasi untuk menangkap kebutuhan fungsional dari sistem yang bersangkutan, Use Case menjelaskan interaksi yang terjadi antara ‘aktor’— inisiator dari interaksi sistem itu sendiri dengan sistem yang ada, sebuah Use Case dipresentasikan dengan urutan langkah yang sederhana.

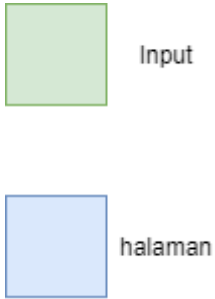
No.	Jenis Diagram	Platform	Jenis Pengguna	Gambar
1.	Use Case	Android dan Web	Admin Utama	 <p>The diagram shows a stick figure actor labeled 'Admin Operasional' connected to 20 use cases. The use cases are arranged in a vertical column and include: Login, Create Tenaga Ahli Daerah, Read Tenaga Ahli Daerah, Update Tenaga Ahli Daerah, Delete Tenaga Ahli Daerah, Create Peserta Daerah, Read Peserta Daerah, Update Peserta Daerah, Delete Peserta Daerah, Create Kegiatan, Read Kegiatan, Update Kegiatan, Delete Kegiatan, View Administrasi Kegiatan, View Buku Saku, View SKA, and View Modul.</p>

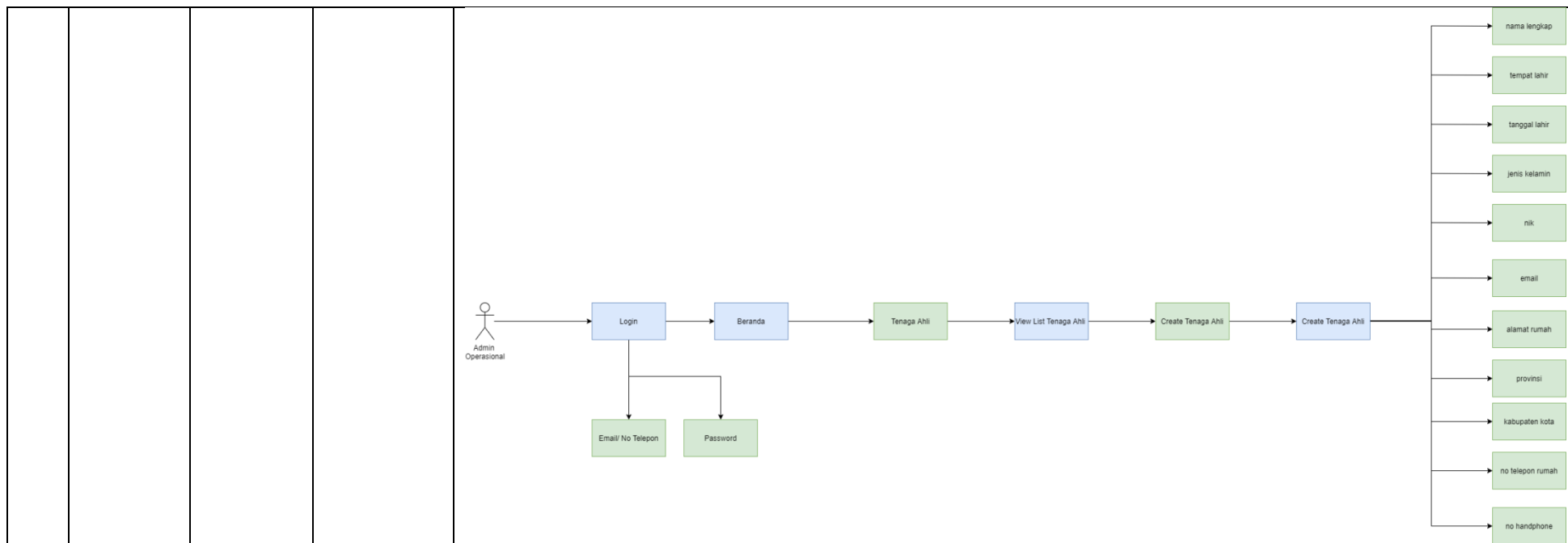
	Use Case	Android dan Web	Admin Operasional	 <p>The diagram shows a central actor labeled 'Admin Utama' connected to 40 use cases. The use cases are organized into five groups of eight, each enclosed in a rectangular box. The groups are: 1. Login, Create Seluruh Tenaga ahli, Read Seluruh Tenaga ahli, Update Seluruh Tenaga ahli, Delete Seluruh Tenaga ahli, Create Seluruh Peserta, Read Seluruh Peserta, Update Seluruh Peserta, Delete Seluruh Peserta. 2. Create Seluruh Kegiatan, Read Seluruh Kegiatan, Update Seluruh Kegiatan, Delete Seluruh Kegiatan, Create Administrasi Kegiatan, Read Administrasi Kegiatan, Update Administrasi Kegiatan, Delete Administrasi Kegiatan. 3. Create Buku Saku, Read Buku Saku, Update Buku Saku, Delete Buku Saku, Create SKA, Read SKA, Update SKA, Delete SKA. 4. Create Modul, Read Modul, Update Modul, Delete Modul.</p>
--	----------	-----------------	-------------------	---

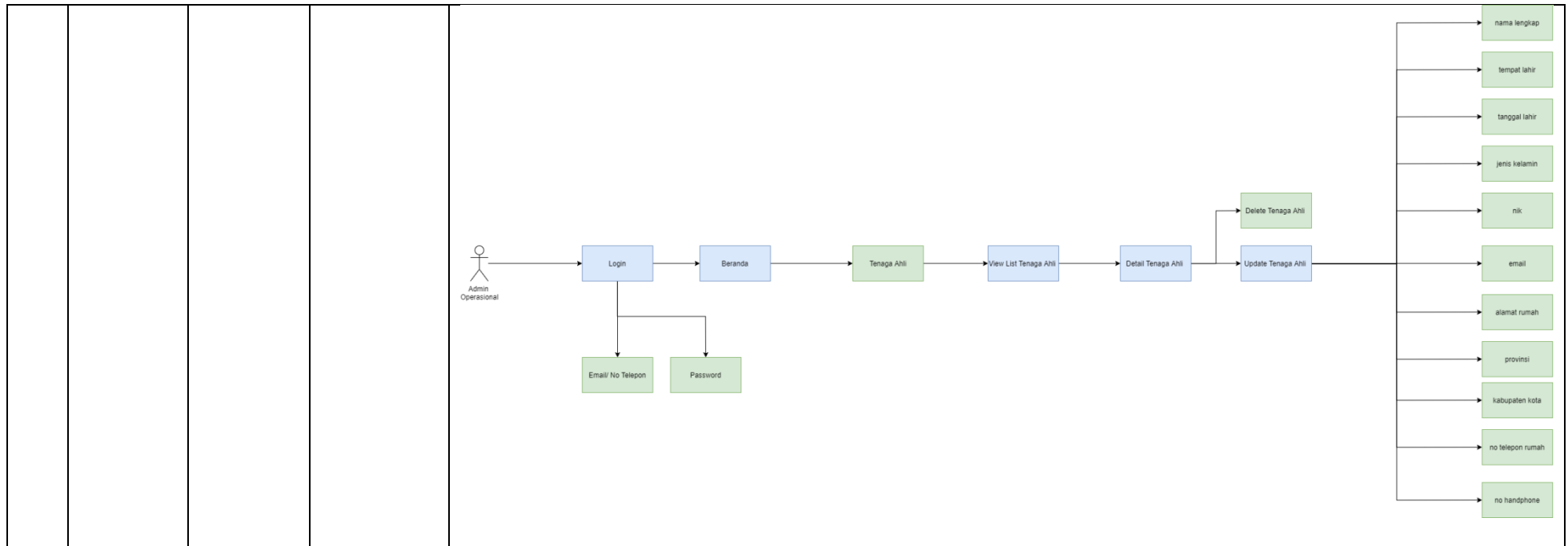


### 3. User Flow

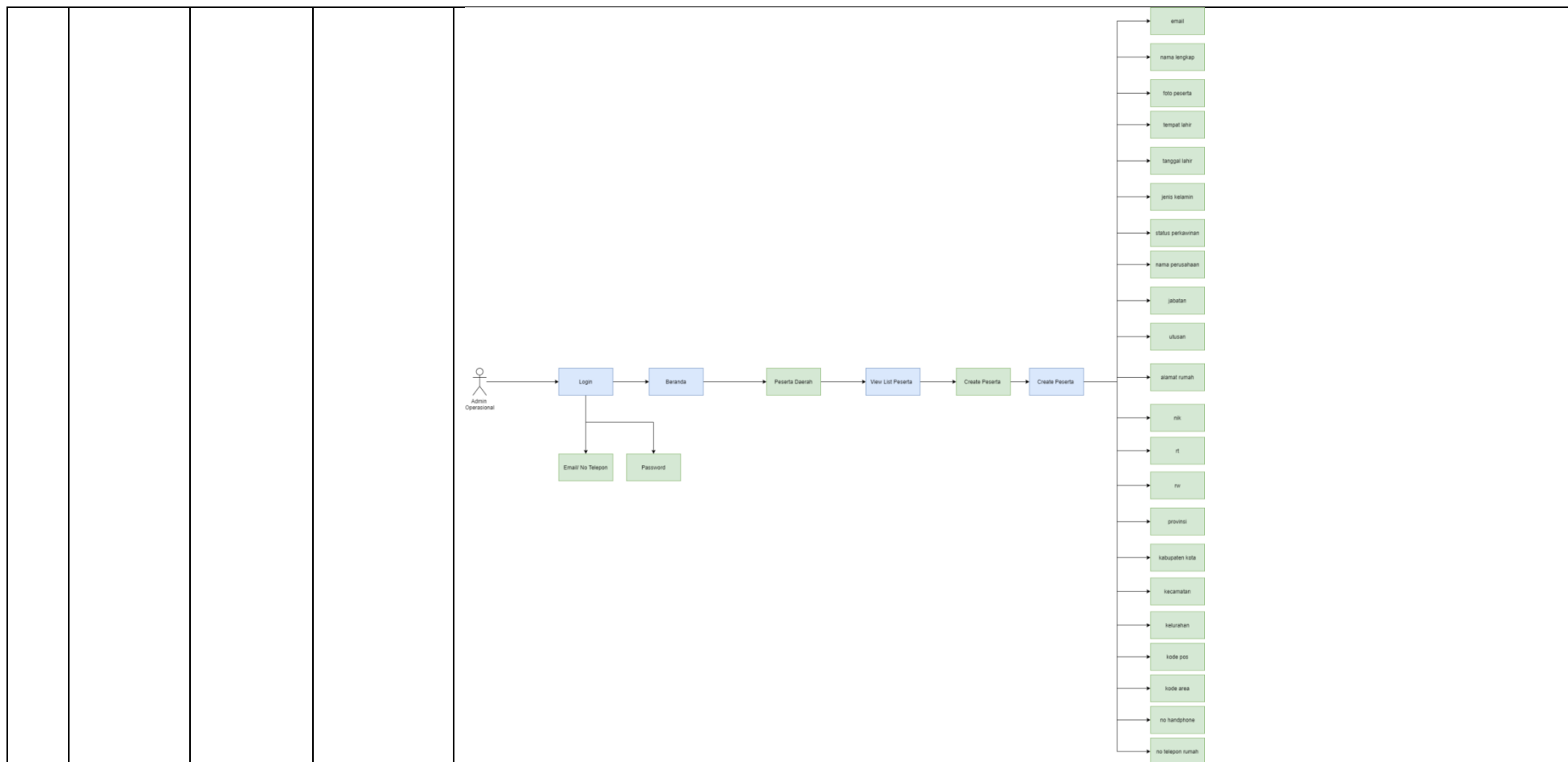
User Flow yang jika diartikan dalam bahasa Indonesia menjadi alur pengguna adalah jalur atau langkah-langkah yang diambil oleh pengguna di website atau software untuk menyelesaikan tugas dan tindakan. Alur pengguna membawa pengguna dari titik masuk (*input*) kemudian melalui serangkaian langkah menuju hasil yang sukses dan tindakan akhir, seperti membeli produk atau menginstall aplikasi serta menggunakan layanan dan lain sebagainya.

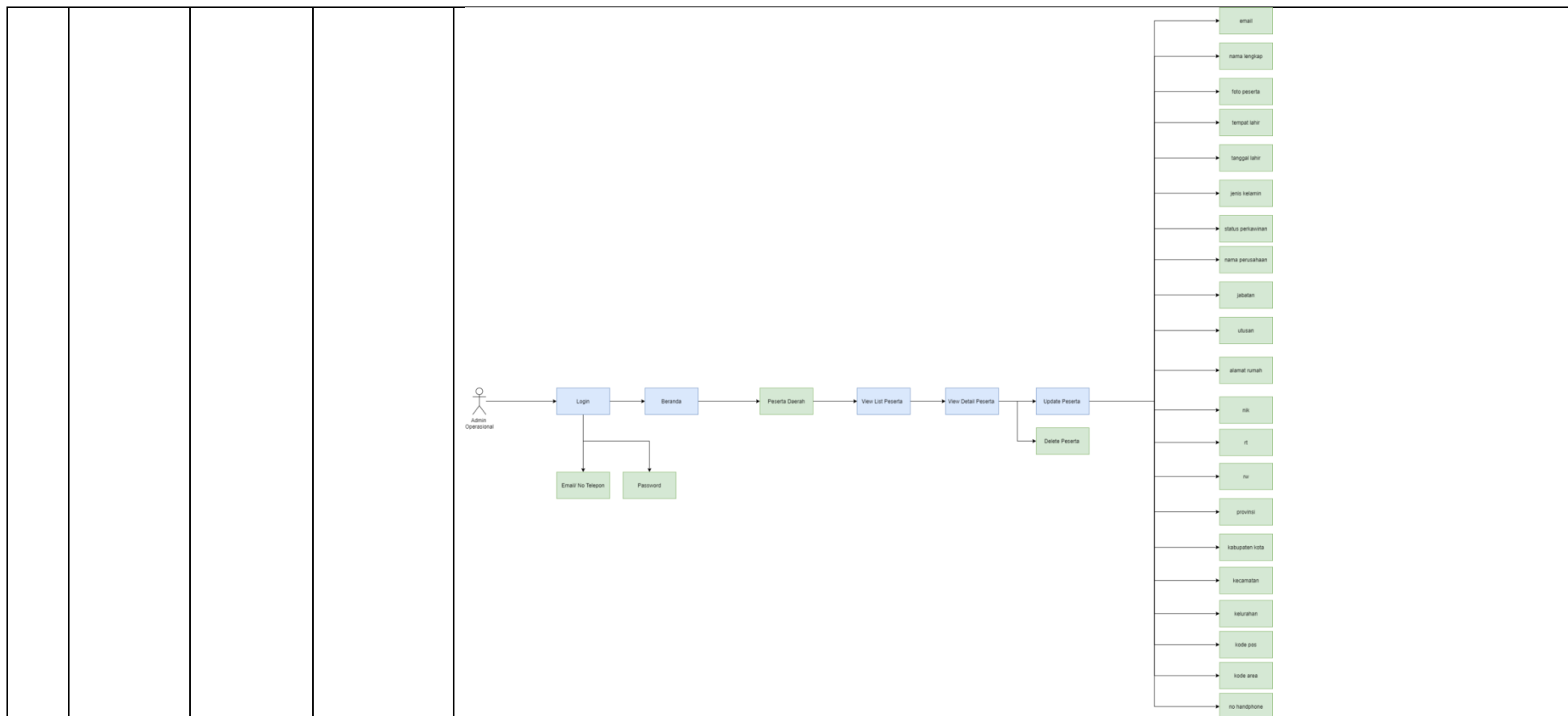
No.	Jenis Diagram	Platform	Jenis Pengguna	Gambar
1.	Flow Board	Android dan Web	Admin Profesional	

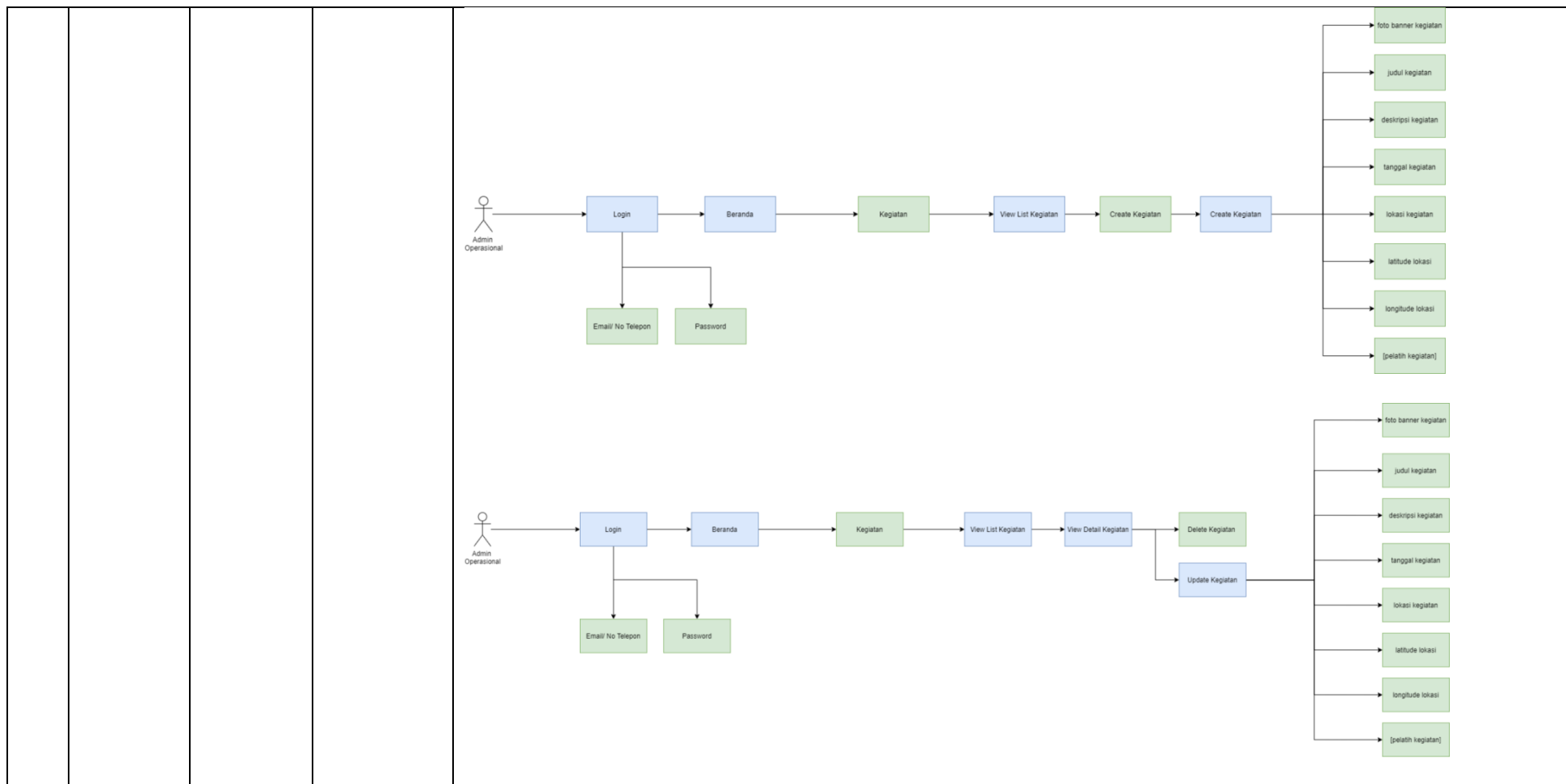


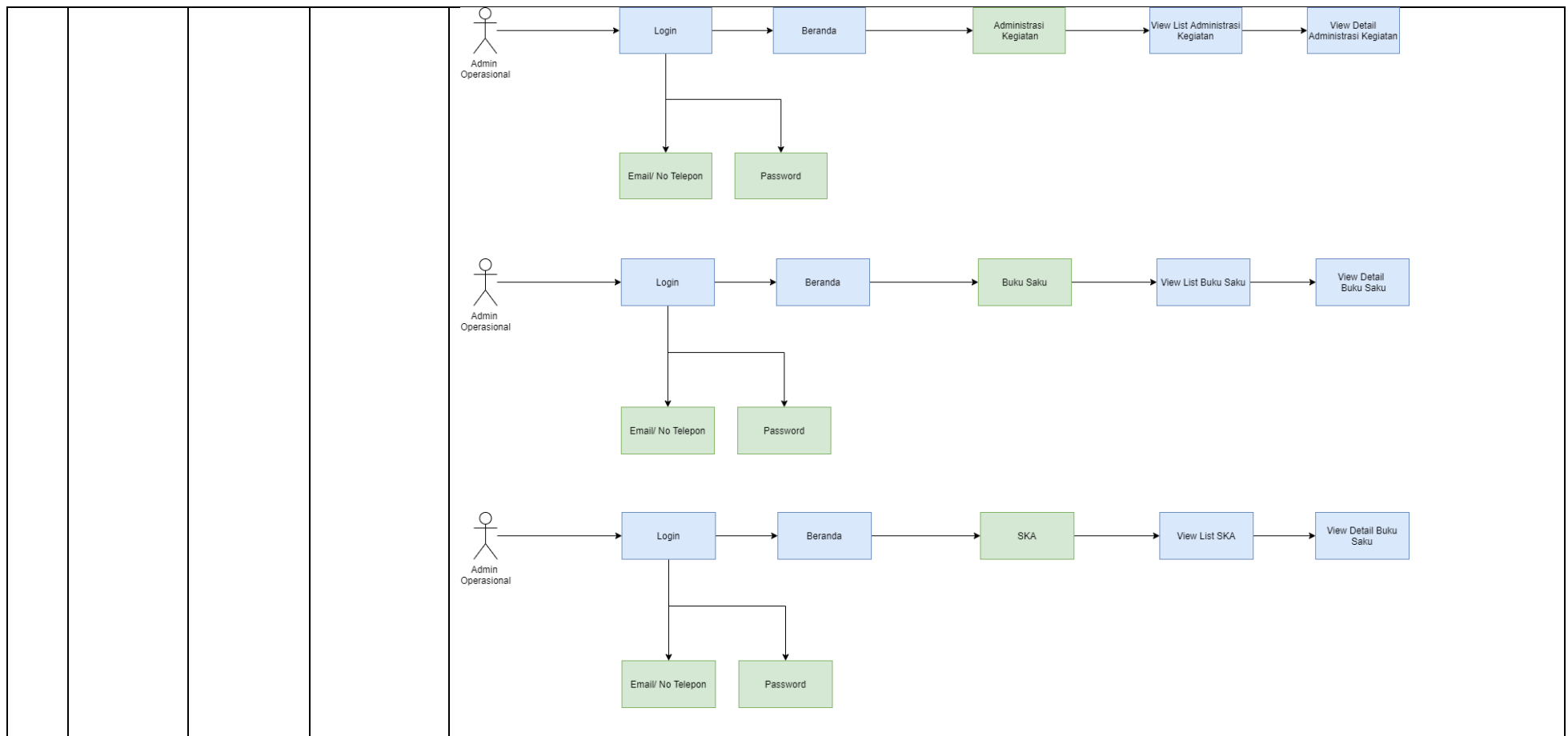




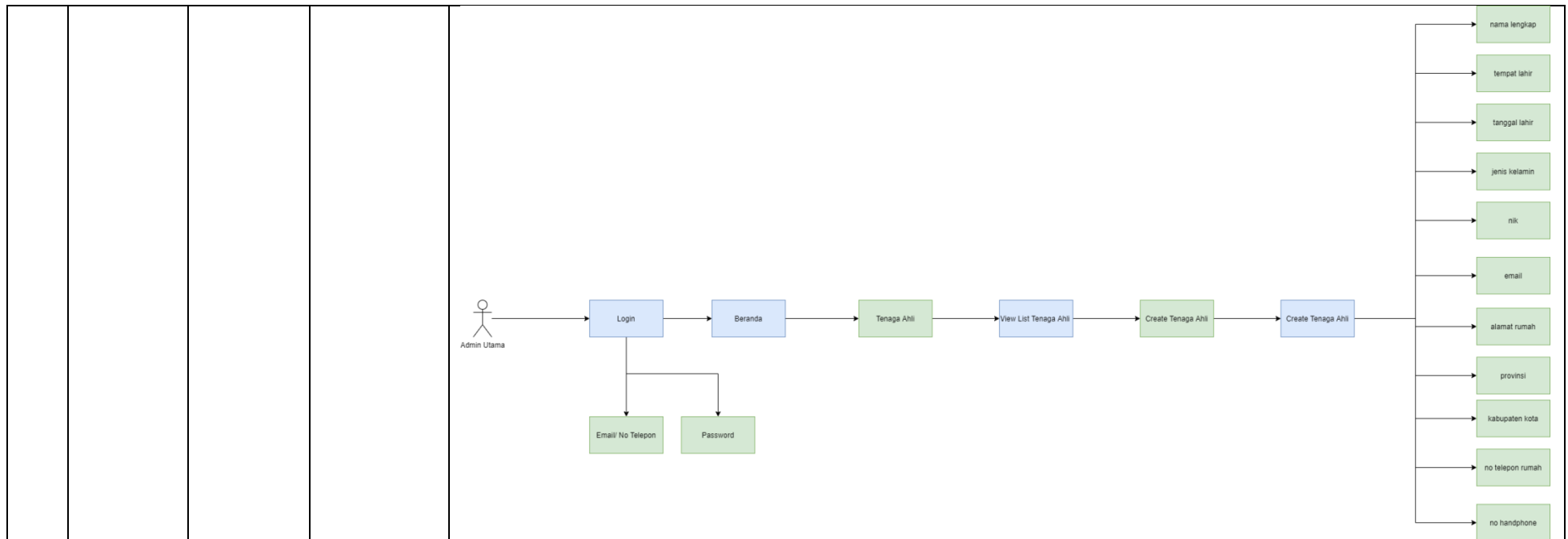


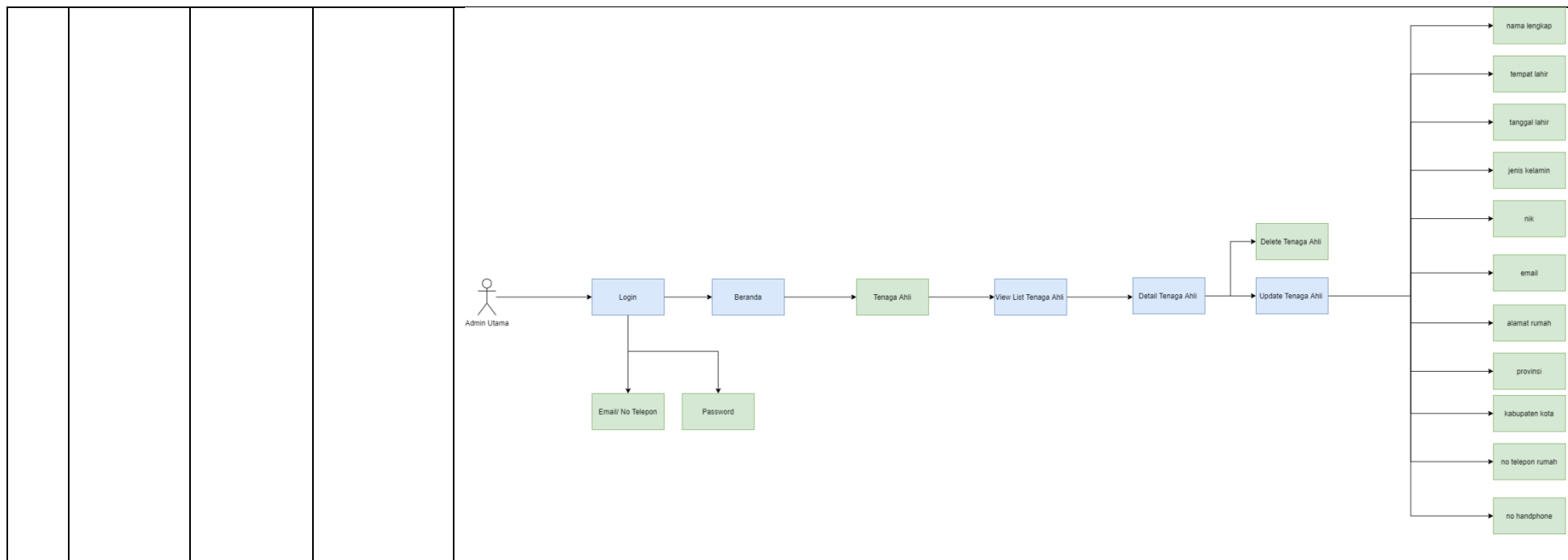


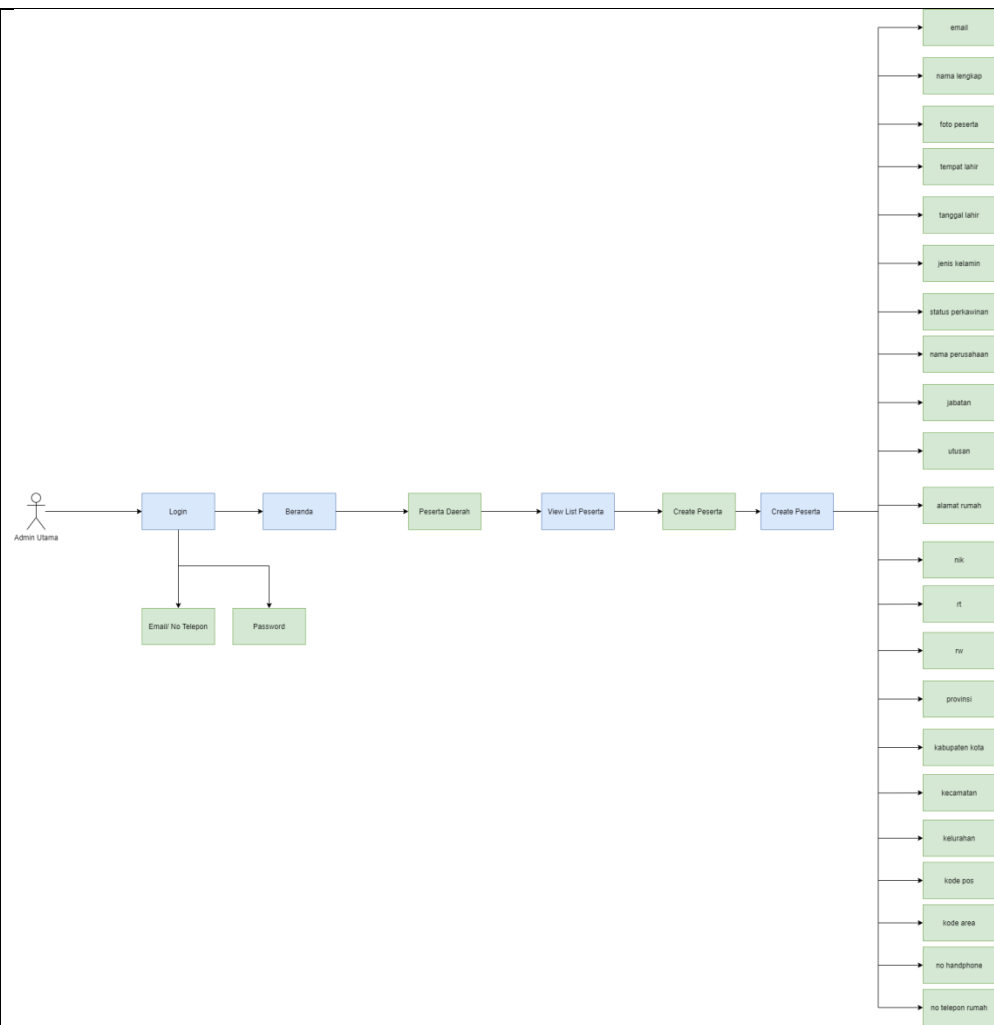




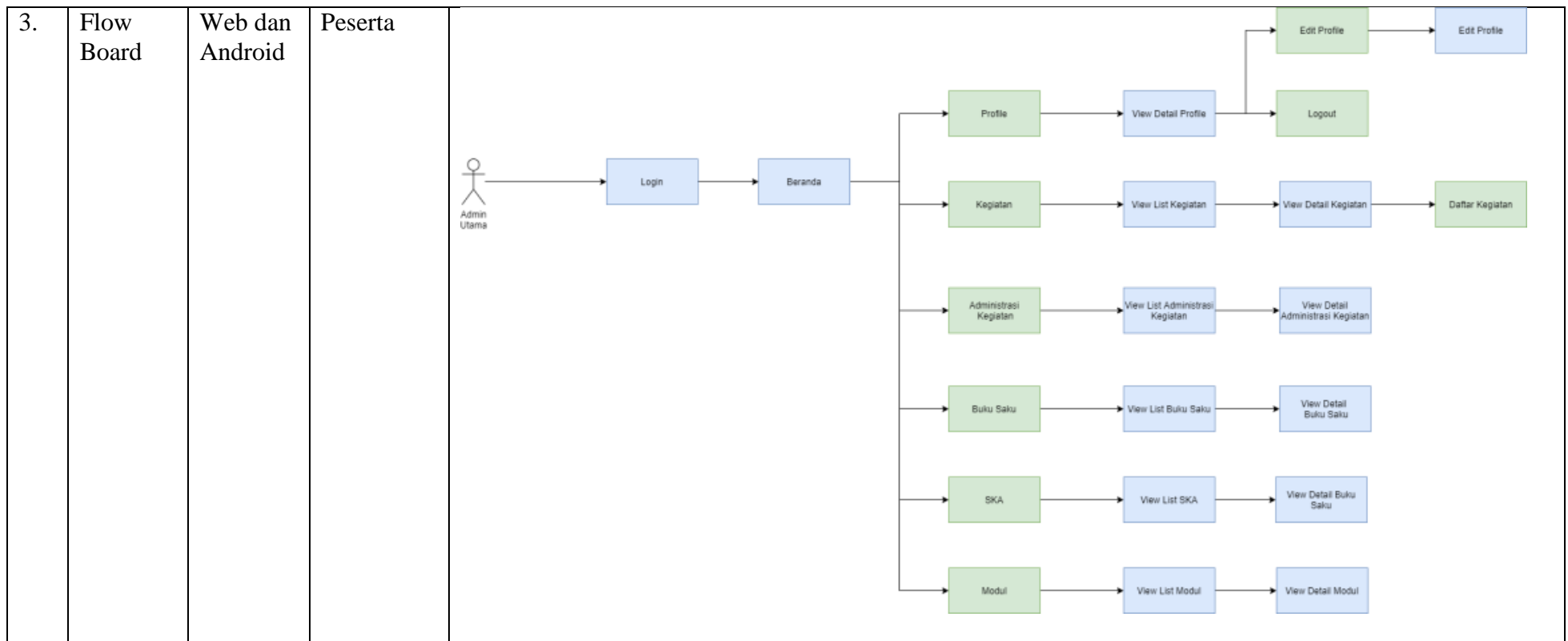
				<pre> graph LR     AdminOperasional[Admin Operasional] --&gt; Login[Login]     Login --&gt; EmailNoTelepon[Email/ No Telepon]     Login --&gt; Password[Password]     EmailNoTelepon --&gt; Beranda[Beranda]     Password --&gt; Beranda[Beranda]     Beranda --&gt; Modul[Modul]     Modul --&gt; ViewListModul[View List Modul]     ViewListModul --&gt; ViewDetailModul[View Detail Modul] </pre>
2.	Flow Board	Web dan Android	Admin Utama	<div> <div></div> Input </div> <div> <div></div> halaman </div>

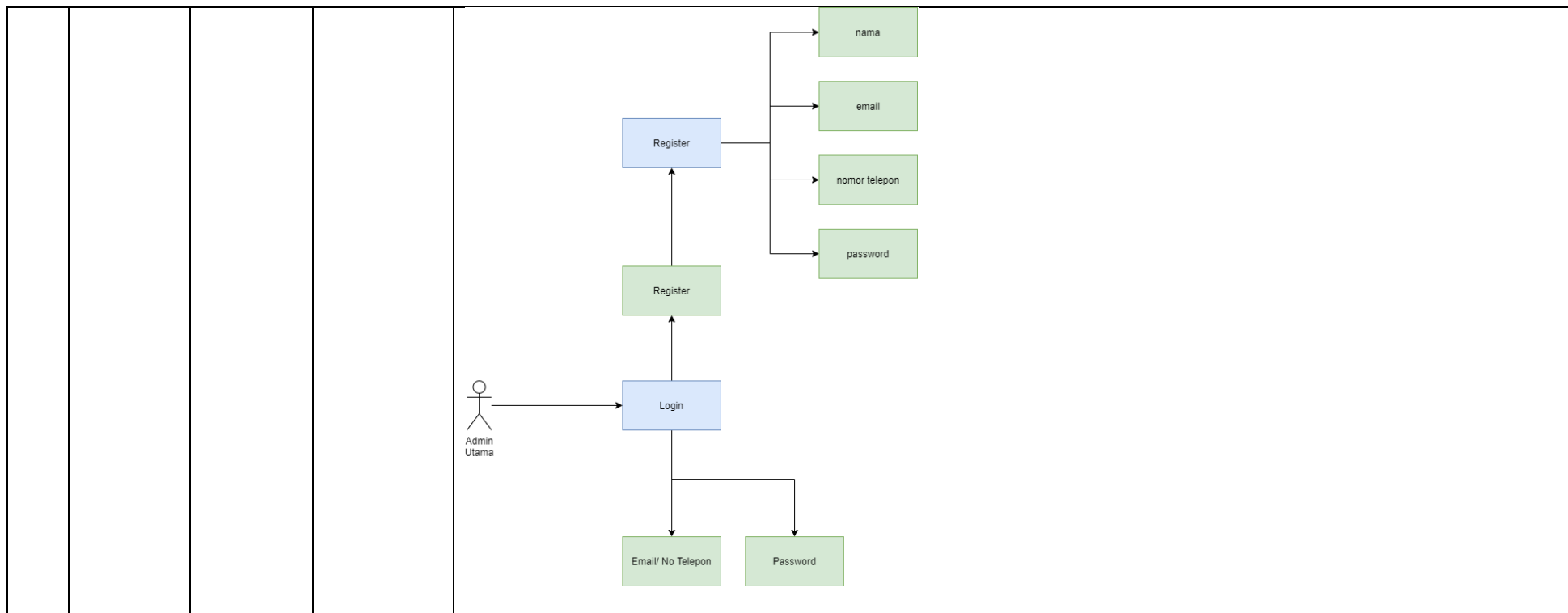






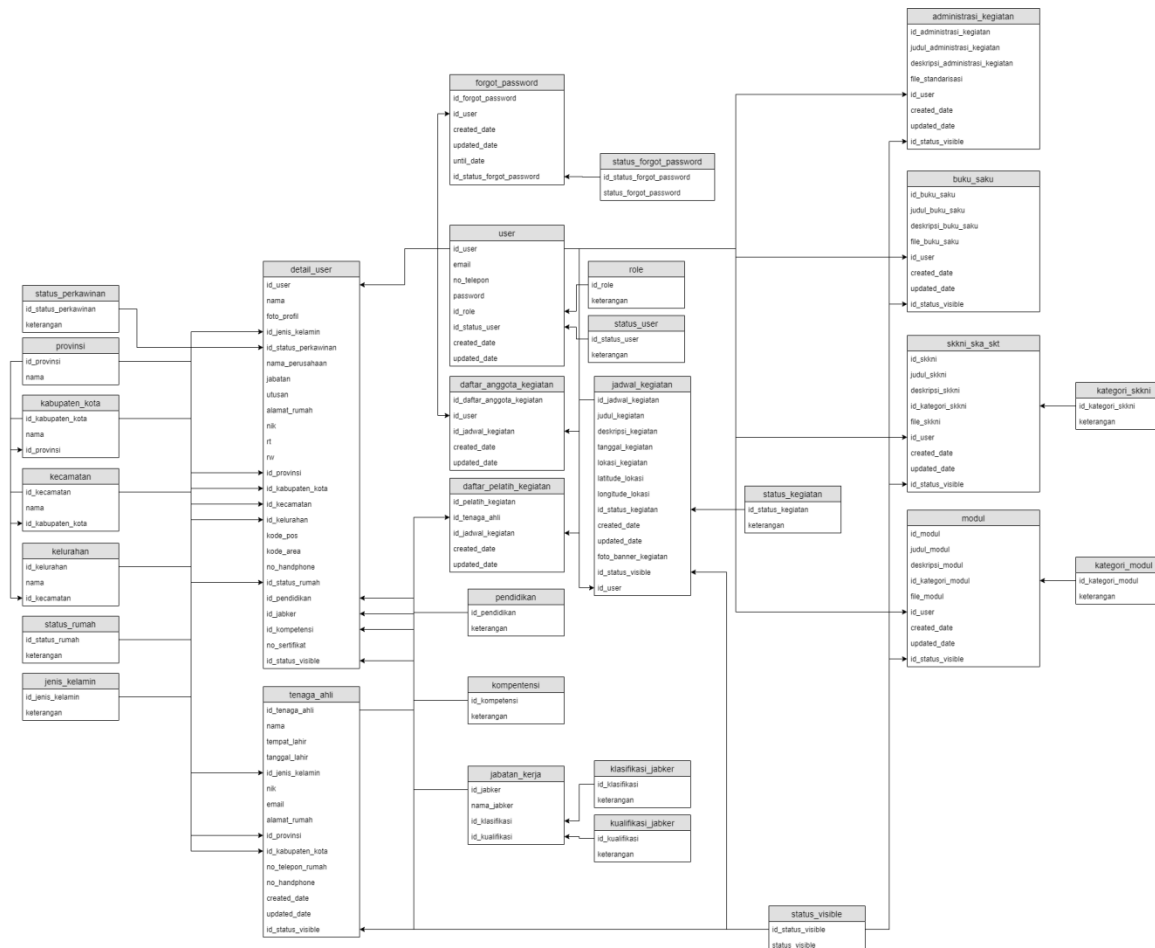






#### 4. Entity Relationship Diagram (ERD)

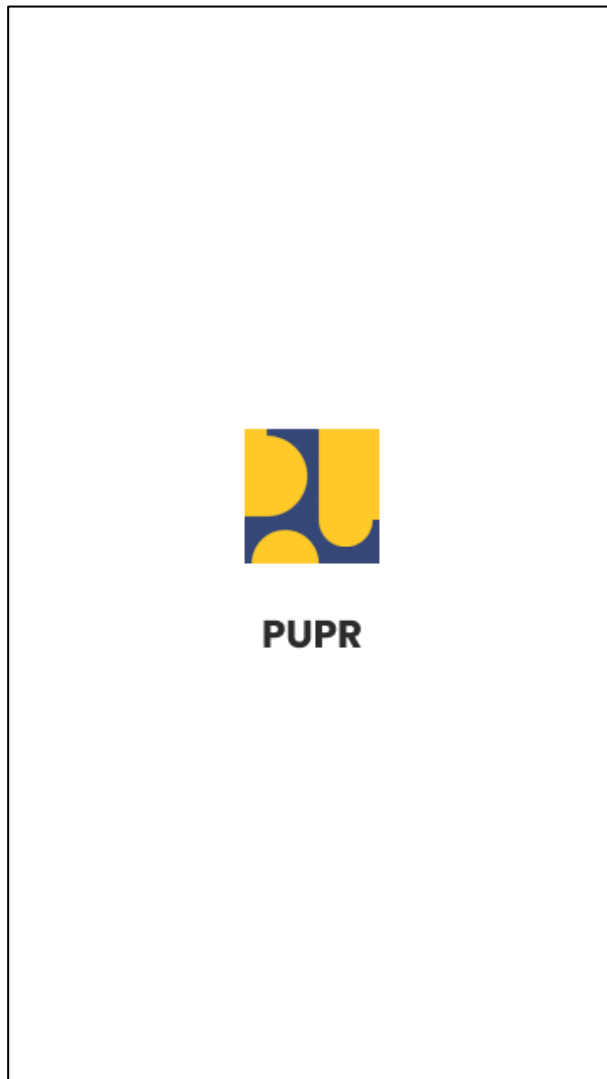
Entity Relationship Diagram (ERD) adalah suatu diagram yang digunakan untuk merancang suatu basis data, dipergunakan untuk memperlihatkan hubungan atau relasi antar entitas atau objek yang terlihat beserta atributnya. ERD dari sistem ini adalah sebagai berikut:



## 5. Tampilan Android

### a. Admin Utama

- Halaman Awal Saat Membuka Aplikasi



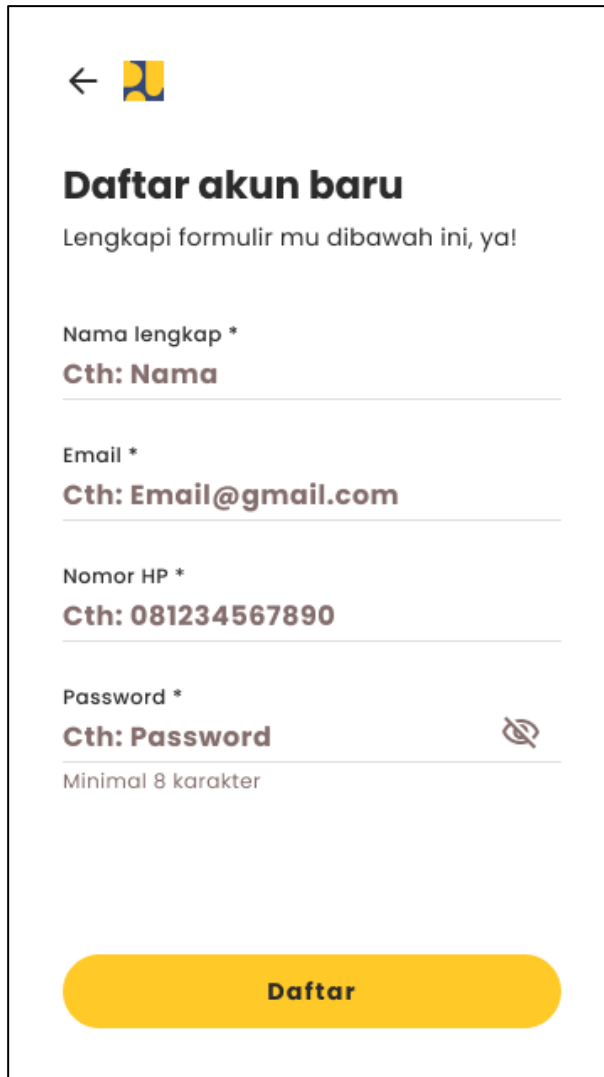
- Halaman Memilih Login atau Register



Tahapan Penggunaan:

1. Sesaat setelah tampilan halaman awal, pengguna akan diarahkan ke halaman selanjutnya dimana terdapat tombol untuk memilih 'Masuk' atau 'Daftar'.
2. Apabila pengguna sudah mempunyai akun, maka pengguna akan menekan tombol 'Masuk'.
3. Sebaliknya, jika pengguna belum mempunyai akun, pengguna harus menekan tombol 'Daftar' terlebih dahulu.

- Halaman Daftar atau *Register*

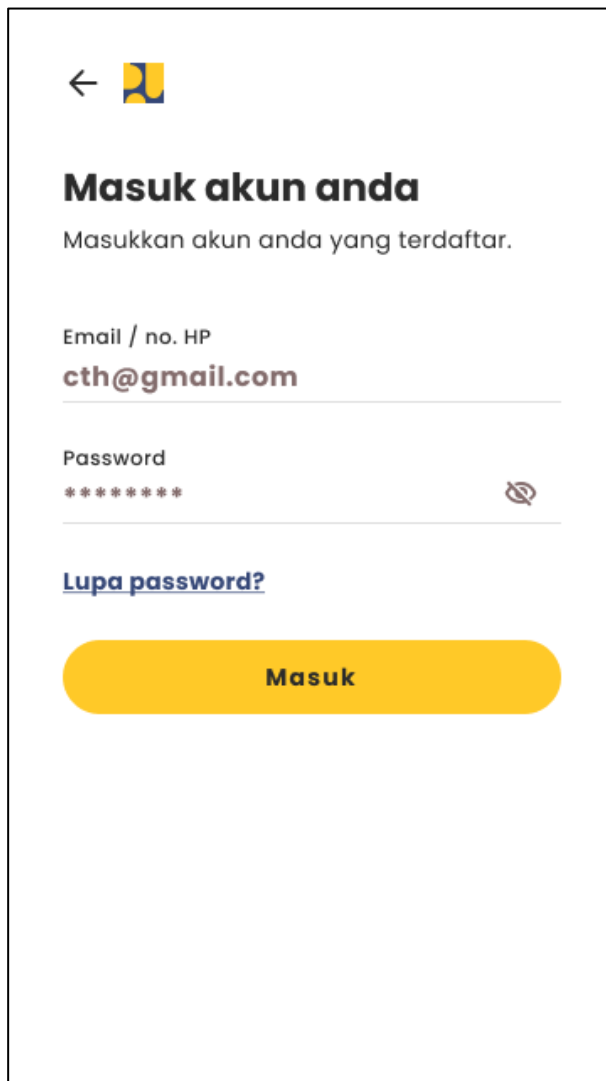



The screenshot shows a mobile application interface for a new account registration. At the top left, there is a back arrow and a profile icon. The title 'Daftar akun baru' is centered, followed by the instruction 'Lengkapi formulir mu dibawah ini, ya!'. Below this are four input fields: 'Nama lengkap \*' with the example 'Cth: Nama', 'Email \*' with 'Cth: Email@gmail.com', 'Nomor HP \*' with 'Cth: 081234567890', and 'Password \*' with 'Cth: Password' and a toggle icon. A note 'Minimal 8 karakter' is below the password field. At the bottom is a large yellow button labeled 'Daftar'.

Tahapan Penggunaan:

1. Bagi pengguna yang belum memiliki akun, dan telah menekan tombol 'Daftar' pada halaman sebelumnya, pengguna akan diarahkan ke halaman Daftar untuk pendaftaran akun baru.
2. Pada halaman 'Daftar', pengguna harus mengisi formulir pendaftaran akun berupa Nama, Email, Nomor HP, dan *Password*. Pengguna diharapkan untuk mengingat data-data yang sudah didaftarkan, karena data-data tersebut akan digunakan pada saat proses *Login* kedepannya.

3. Setelah formulir terisi semua, pengguna bisa langsung memproses pendaftaran dengan menekan tombol 'Daftar'.
- Halaman Masuk atau *Login*




← 

## Masuk akun anda

Masukkan akun anda yang terdaftar.

Email / no. HP  
**cth@gmail.com**

Password  
\*\*\*\*\* 

[Lupa password?](#)

**Masuk**

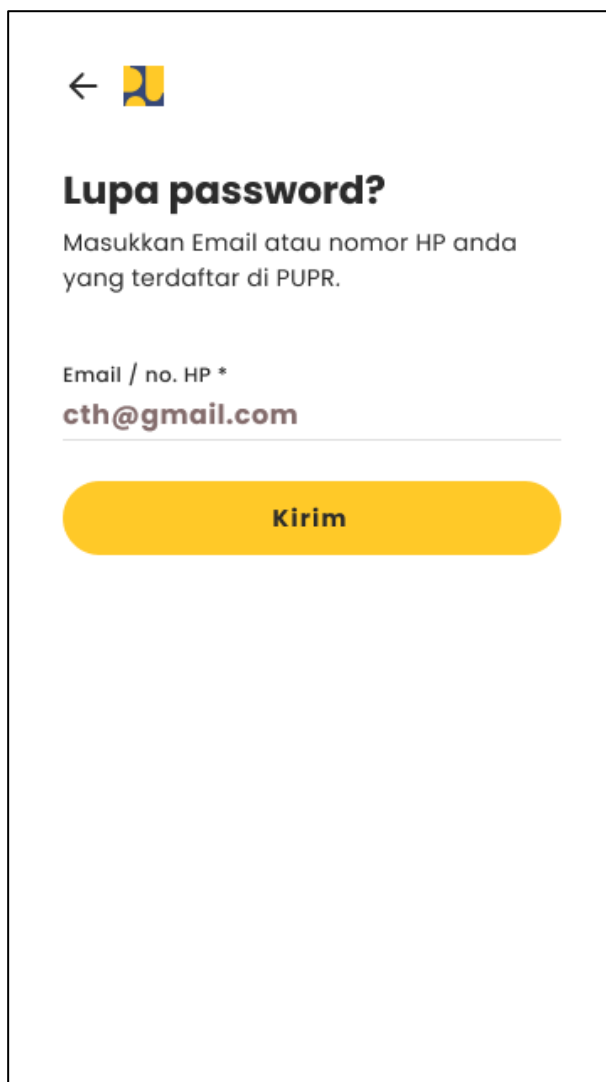
#### Tahapan Penggunaan:


1. Bagi pengguna yang sudah memiliki akun, dan telah menekan tombol 'Masuk' pada halaman 'Memilih Masuk atau Daftar', pengguna akan diarahkan ke halaman untuk masuk ke akun seperti yang terlihat di atas.
2. Di halaman 'Masuk', pengguna diminta mengisi data diri yang sudah didaftarkan sebelumnya yaitu Email atau Nomor HP dan juga *Password*.

Setelah itu, pengguna bisa langsung menekan tombol 'Masuk' untuk diarahkan ke halaman selanjutnya.

3. Apabila terdapat pengguna yang lupa akan *Password* yang terdaftar, pengguna dapat menekan tombol 'Lupa *Password*' untuk proses pergantian kata sandi atau *Password*.

- Halaman Lupa Password



← 

### Lupa password?

Masukkan Email atau nomor HP anda yang terdaftar di PUPR.

Email / no. HP \*

cth@gmail.com

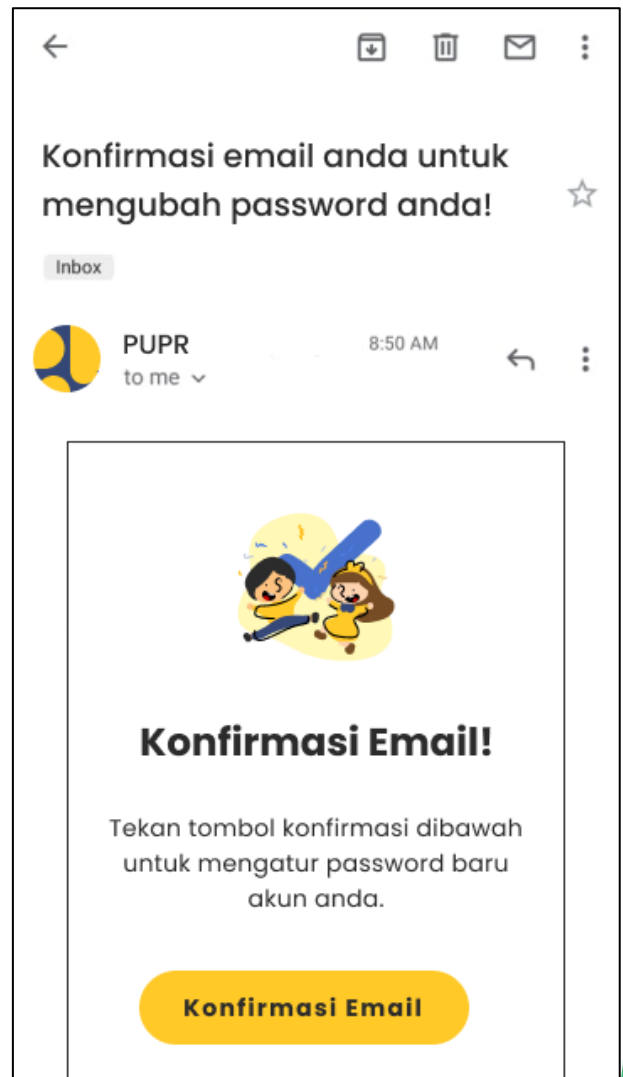
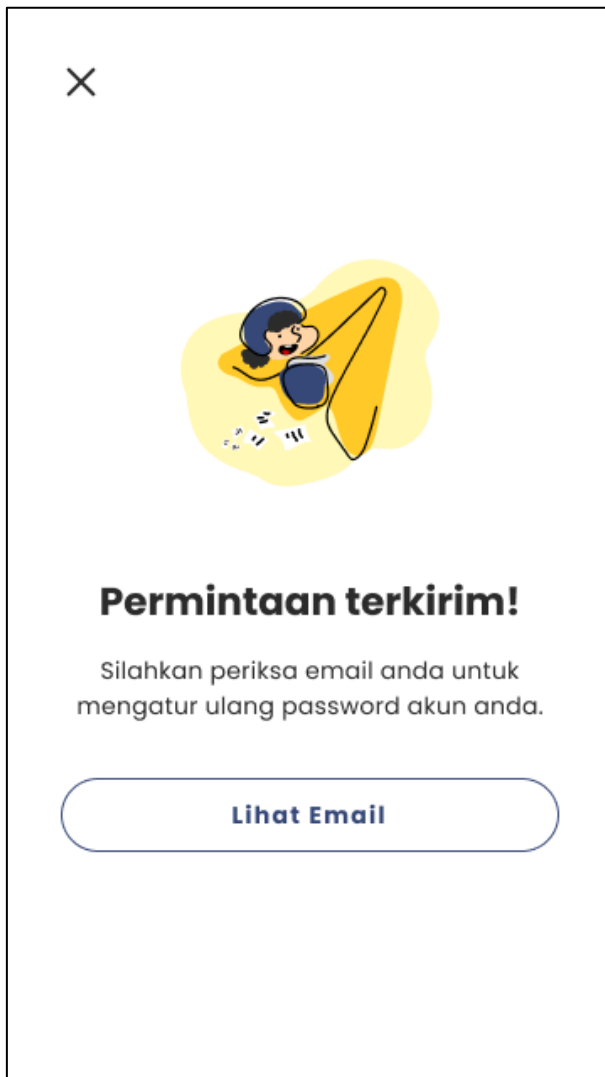
Kirim

Tahapan Penggunaan:



1. Pengguna yang menekan tombol 'Lupa Password' akan diarahkan ke halaman seperti yang terlihat di atas.



2. Pengguna harus memasukkan Email atau Nomor HP terdaftar, kemudian tekan tombol 'Kirim' dan pengguna akan diminta untuk memeriksa Email untuk bisa melakukan proses pengubahan *Password*.



3. Pengguna akan menerima pesan baru di kotak masuk Email yang meminta pengguna untuk melakukan konfirmasi Email.
4. Agar bisa melanjutkan proses pengubahan *Password*, pengguna bisa menekan tombol 'Konfirmasi Email'.



### Atur password baru!


Silahkan atur password baru untuk akun anda.


Password \*

**Cth: Password**

Minimal 8 karakter

**Simpan**





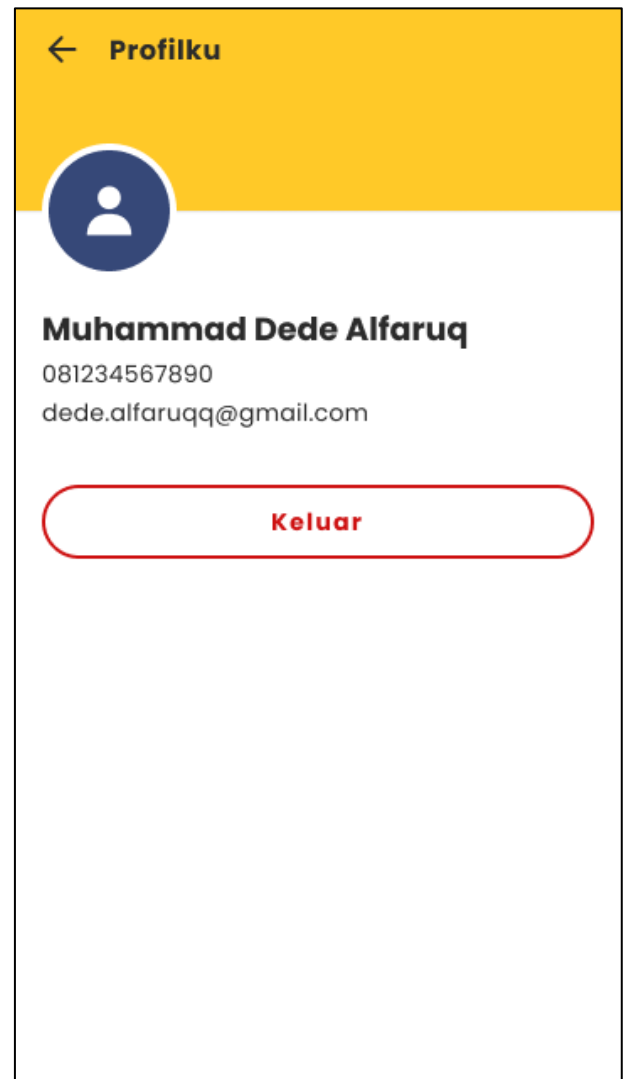
### Ganti password berhasil!

Password anda berhasil di ganti, silahkan masuk ke akun anda!

**Masuk akun**

- Pengguna lalu diarahkan ke halaman ‘Atur Password Baru’ dan memasukkan *Password* terbaru untuk akunnya dan mengakhiri proses dengan menekan tombol ‘Simpan’.
- Apabila pengaturan ulang *Password* berhasil, pengguna akan diberitahu oleh tampilan di atas. Selanjutnya, pengguna bisa melakukan proses ‘Masuk Akun’ dengan menggunakan *Password* terbaru.

## 7. Halaman *Home* dan Profil



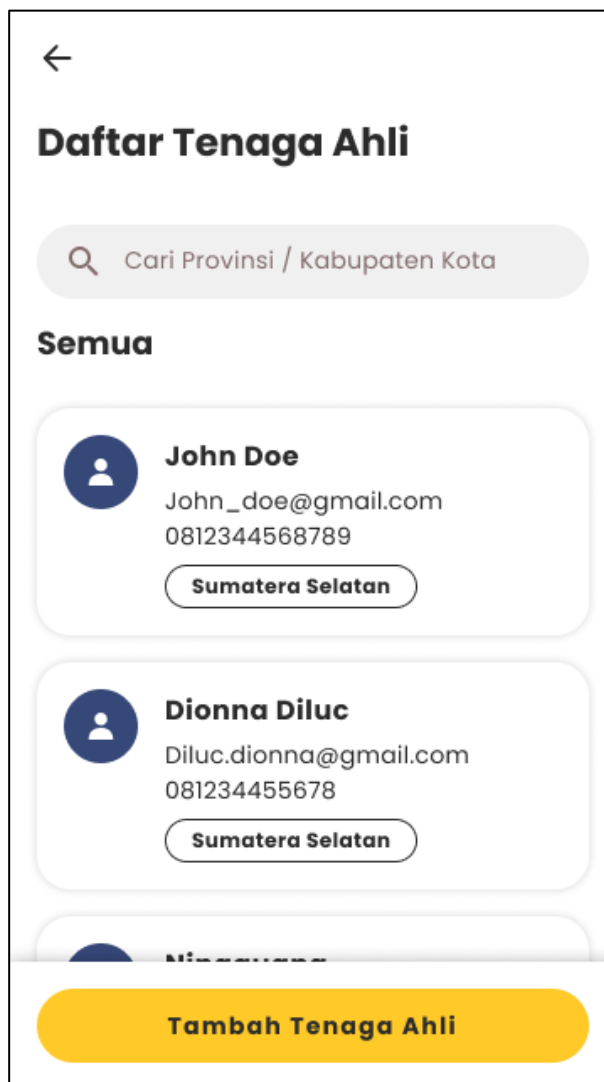
#### Tahapan Penggunaan:

1. Setelah proses Login atau Masuk, pengguna diarahkan ke halaman Home seperti yang terlihat di atas. Di halaman ini, pengguna atau admin dapat mengakses berbagai hal yang tersedia seperti *Dashboard*, *Kegiatan*, *Tenaga Ahli*, dan lain-lain.
  2. Pengguna juga bisa melihat halaman profil dengan cara menekan nama pengguna yang tertera di halaman Home tersebut.
- Halaman *Dashboard*



#### Tahapan Penggunaan:

1. Apabila pengguna menekan tombol *Dashboard* pada halaman *Home*, pengguna dapat melihat halaman seperti tampilan di atas. Di sana, pengguna akan disugahi data total pendaftar dan jumlah peserta tahun ini. Pengguna juga bisa menerapkan filter grafik sesuai dengan yang diinginkan.
- Halaman Tenaga Ahli
  - Daftar Tenaga Ahli



### Tahapan Penggunaan:

1. Apabila pengguna menekan tombol Tenaga Ahli pada halaman *Home*, pengguna dapat melihat halaman seperti tampilan di atas. Pada halaman tersebut terdapat daftar Tenaga Ahli yang bisa dilihat detailnya dengan cara langsung menekan pada salah satu di daftar, lalu pengguna juga bisa menambah daftar tersebut dengan menekan tombol ‘Tambah Tenaga Ahli’.
- Tambah, Lihat Detail, Ubah, dan Hapus Tenaga Ahli

**Tambah Tenaga Ahli**  
Lengkapi formulir dibawah ini, ya!

**Data diri**  
Nama Lengkap \*  
Cth: John Doe

Jenis Kelamin \*  
☐ Laki - laki ☐ Perempuan

Tempat Lahir \*  
Pilih tempat lahir

Tanggal Lahir \*  
Pilih tanggal lahir

**Data Kontak**  
Email \*  
Cth: Johndoe@gmail.com

Nomor HP \*  
Cth: 081234567890

Nomor Telepon Rumah \*  
Cth: 071112345

NIK \*  
Cth: 1671070123123124

**Data Alamat**  
Alamat \*  
Cth: Jalan Gajah Madah II

Provinsi \*  
Pilih provinsi

Kabupaten / kota \*  
Pilih kabupaten / kota

**Tambah**

**Batal**

**Profil Tenaga Ahli**

**John Doe**  
John\_doe@gmail.com  
0812344568789 / 071172352

**Jenis Kelamin**  
 Laki - laki

**NIK**  
 16711412030120140

**Tempat & Tanggal Lahir**  
 Palembang, 03 Januari 1987

**Alamat**  
 Jalan Demang Lebar Daun, nomor 100 C  
Kecamatan Demang Lebar Daun.

**Provinsi**  
 Sumatera Selatan

**Kabupaten Kota**  
 Palembang

**Ubah**

**Hapus**



## Ubah Data Tenaga Ahli

Lengkapi formulir dibawah ini, ya!

### Data diri

Nama Lengkap \*

John Doe

Jenis Kelamin \*

Laki - laki

Perempuan

Tempat Lahir \*

Jakarta

Tanggal Lahir \*

03-11-1987

### Data Kontak

Email \*

John\_doe@gmail.com

Nomor HP \*

0812344568789

Nomor Telepon Rumah \*

071172352

NIK \*

16711412030120140

### Data Alamat

Alamat \*

Jalan Demang Lebar Daun, nomor 100  
C Kecamatan Demang Lebar.

Provinsi \*

Sumatera Selatan

Kabupaten / kota \*

Palembang

Simpan

Batal



## Profil Tenaga Ahli



### John Doe

John\_doe@gmail.com

0812344568789 / 071172352

### Jenis Kelamin



Laki - laki

### NIK



16711412030120140

### Tempat & Tanggal Lahir



Palembang, 03 Januari 1987

### Alamat



Jalan Demang Lebar Daun, nomor 100 C  
Kecamatan Demang Lebar Daun.

### Provinsi

## Hapus Tenaga Ahli?

Pastikan dengan benar bahwa tenaga  
ahli ini ingin anda hapus!

Batal

Hapus

#### Tahapan Penggunaan:

1. Bila ingin menambahkan data Tenaga Ahli, pengguna harus terlebih dahulu mengisi formulir Data Diri, Data Kontak, dan Data Alamat dari Tenaga Ahli yang bersangkutan lalu meng-klik tombol 'Tambah' untuk memproses.
  2. Di halaman daftar Tenaga Ahli sebelumnya, pengguna bisa melihat detail Tenaga Ahli serta bisa melakukan perubahan dan menghapus data Tenaga Ahli dengan menekan tombol 'Ubah' atau 'Hapus' pada halaman detail.
- Cari Tenaga Ahli

**Daftar Tenaga Ahli**

Sulawesi Selatan

Semua

**Daftar Tenaga Ahli  
Masih Kosong!**

Tenaga ahli belum ada satupun,  
silahkan tambah tenaga ahli terlebih  
dahulu.

**Tambah Tenaga Ahli**

**Daftar Tenaga Ahli**

Jawa Barat

Semua

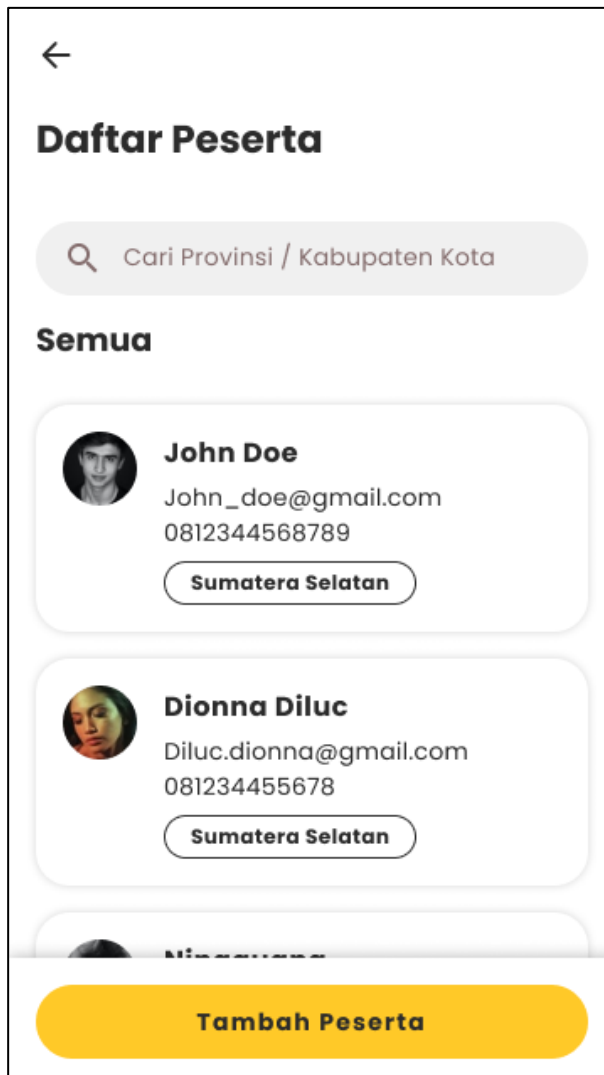
**Maulana**  
Maul\_ana@gmail.com  
0812344568789  
**Jawa Barat**

**Tambah Tenaga Ahli**



Tahapan Penggunaan:

1. Pengguna juga bisa melakukan pencarian terhadap data Tenaga Ahli dengan cara memasukkan provinsi asalnya.
  2. Jika Tenaga Ahli yang dicari tidak ada, maka daftar tidak akan menampilkan satu pun Tenaga Ahli.
  3. Sebaliknya, jika Tenaga Ahli yang dicari ada, daftar akan menampilkan data Tenaga Ahli tersebut.
- Halaman Peserta Daerah
  - Daftar Peserta Daerah




←


## Daftar Peserta

🔍 Cari Provinsi / Kabupaten Kota


Semua



**John Doe**  
John\_doe@gmail.com  
0812344568789  
Sumatera Selatan



**Dionna Diluc**  
Diluc.dionna@gmail.com  
081234455678  
Sumatera Selatan



**Ningsun**

**Tambah Peserta**

### Tahapan Penggunaan:

1. Apabila pengguna menekan tombol Peserta Daerah pada halaman *Home*, pengguna dapat melihat halaman seperti tampilan di atas. Pada halaman tersebut terdapat daftar Peserta Daerah yang bisa dilihat detailnya dengan cara langsung menekan pada salah satu di daftar, lalu pengguna juga bisa menambah daftar tersebut dengan menekan tombol 'Tambah Peserta'.
- Tambah, Lihat Detail, Ubah, dan Hapus Peserta Daerah

←

Tambah Peserta

Lengkapi formulir dibawah ini, ya!

Data diri

Foto profil \*

Nama Lengkap \*

Cth: John Doe

Jenis Kelamin \*

Laki - Laki

Pemilihan

Tempat Lahir \*

Pilih tempat lahir

Tanggal Lahir \*

Pilih tanggal lahir

Status Perkawinan \*

Mendaki

Belum Mendaki

Data Pekerjaan

Nama Perusahaan \*

Cth: PT. Digital Creative

Jabatan \*

Cth: CEO

Urutan \*

Cth: Pengawas

Data Pendidikan

Pendidikan \*

Cth: SI

Jurusan \*

Cth: CEO

Kompetensi \*

Cth: Sertifikasi Ilmu Sipil

Nomor Sertifikat \*

Cth: sipil/00/2020

Data Kontak

Email \*

Cth: johndoe@gmail.com

Nomor HP \*

Cth: 081234567890

Nomor Telepon Rumah \*

Cth: 07112345

NK \*

Cth: 1671070123123124

Data Tempat Tinggal

Alamat \*

Cth: Jalan Gajah Madah II

Provinsi \*

Pilih provinsi

Kabupaten / kota \*

Pilih kabupaten / kota

Kecamatan \*

Pilih Kecamatan

Kelurahan \*

Pilih Kelurahan

RT \*

Cth: 38

RW \*

Cth: II

Kode Pos \*

Cth: 30196

Kode Area \*

Cth: 0766

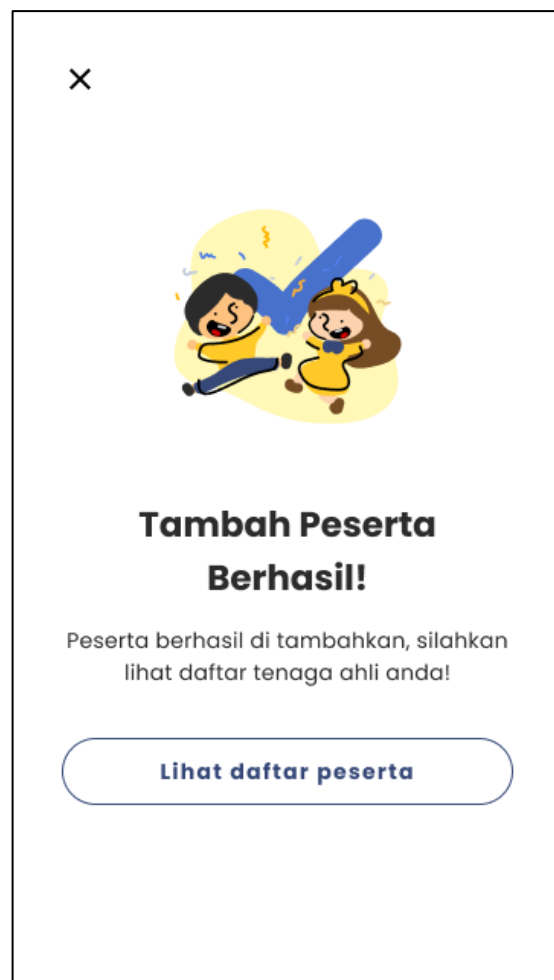
Status Rumah \*

Milik Sendiri

Kontrak

Tambah

Batal



←

Profil Peserta

**Michaela**  
Michaela@gmail.com  
081234456789 / 07112352

NIK

1671070324235

Jenis Kelamin

Laki - laki

Tempat & Tanggal Lahir

Palembang, 03 Januari 1987

Status Perkawinan

Belum Menikah

Data Pekerjaan

Nama Perusahaan

Digital Creative

Jabatan

Vice President

Utusan

Pengawas

Data Pendidikan

Pendidikan

S1 Ilmu Komputer

Jabker

Vice President

Kompetensi

Android Developer Certified

Nomor Sertifikat

AAD/100

Data Kontak

Email

Michaela@gmail.com

Nomor HP

0812344567890

Nomor Telepon Rumah

0711234235

Data Alamat

Alamat

Jalan Demang Lebar Daun, nomor 100 C  
Kecamatan Demang Lebar Daun.

Provinsi

Sumatera Selatan

Kabupaten Kota

Palembang

Kecamatan

Bukit Besar

Kelurahan

Demang Lebar Daun

RT / RW

11 / 24

Kode Pos

30154

Kode Area

0766

Status Rumah

Milik Sendiri

Ubah

Hapus

←

Ubah Data Peserta

Lengkapi formulir dibawah ini, ya!

Data diri

Foto profil \*

Nama Lengkap \*

Michaela

Jenis Kelamin \*

Laki - laki

Perempuan

Tempat Lahir \*

Palembang

Tanggal Lahir \*

03 - 01 - 1987

Status Perkawinan \*

Menikah

Belum Menikah

Data Pekerjaan

Nama Perusahaan \*

Digital Creative

Jabatan \*

Vice President

Utusan \*

Pengawas

Data Pendidikan

Pendidikan \*

S1 Ilmu Komputer

Jabker \*

Vice President

Kompetensi \*

Android Developer Certified

Nomor Sertifikat \*

AAD/100

Data Kontak

Email \*

Michaela@gmail.com

Nomor HP \*

0812344567890

Nomor Telepon Rumah \*

07112345

NIK \*

1671070123123124

Data Alamat

Alamat \*

Jalan Demang Lebar Daun, nomor 100  
C Kecamatan Demang Lebar Daun.

Provinsi \*

Sumatera Selatan

Kabupaten / kota \*

Palembang

Kecamatan \*

Bukit Kecil

Kelurahan \*

Demang Lebar Daun

RT \*

11

RW \*

24

Kode Pos \*

30156

Kode Area \*

0766

Status Rumah \*


Milik Sendiri

Kontrak

Simpan

Batal

← Profil Peserta



**Michaela**  
Michaela@gmail.com  
081234456789 / 071172352

**NIK**  
1671070324235

**Jenis Kelamin**  
Laki - laki

**Tempat & Tanggal Lahir**  
Palembang, 03 Januari 1987

**Status Perkawinan**  
Belum Menikah

**Data Pekerjaan**

**Nama Perusahaan**  
Digital Creative

**Jabatan**  
Vice President

**Utusan**  
Pengawas

**Data Pendidikan**

**Pendidikan**  
SI Ilmu Komputer

**Jabker**  
Vice President

**Kompetensi**  
Android Developer Certified

**Nomor Sertifikat**  
AAD/100

**Data Kontak**

**Email**  
Michaela@gmail.com

**Nomor HP**  
0812344567890

**Nomor Telepon Rumah**  
0711234235

**Data Alamat**

**Alamat**  
Jalan Demang Lebar Daun, nomor 100 C  
Kecamatan Demang Lebar Daun,

**Provinsi**  
Sumatera Selatan

**Kabupaten Kota**  
Palembang

**Kecamatan**  
Bukit Besar

**Kelurahan**  
Demang Lebar Daun

**RT / RW**  
11 / 24

**Kode Pos**  
30154


**Kode Area**

**Hapus Peserta?**  
Pastikan dengan benar bahwa peserta ini ingin anda hapus!

Batal

Hapus

×




**Hapus Peserta**  
**Berhasil!**

Peserta berhasil dihapus dari daftar Peserta


Lihat daftar peserta

#### Tahapan Penggunaan:

1. Bila ingin menambahkan data Peserta Daerah, pengguna harus terlebih dahulu mengisi formulir Data Diri, Data Pekerjaan, Data Pendidikan, Data Kontak, dan Data Tempat Tinggal dari Peserta Daerah yang bersangkutan lalu meng-klik tombol 'Tambah' untuk memproses.
  2. Di halaman daftar Peserta Daerah sebelumnya, pengguna bisa melihat detail Peserta Daerah serta bisa melakukan perubahan dan menghapus data Peserta Daerah dengan menekan tombol 'Ubah' atau 'Hapus' pada halaman detail.
- Cari Peserta Daerah



**Daftar Peserta**


 Sulawesi Selatan

**Semua**


**Daftar Peserta Masih Kosong!**

Peserta belum ada satupun, silahkan tambah Peserta terlebih dahulu.


**Tambah Peserta**



**Daftar Peserta**

 Jawa Barat

**Semua**



**Maulana**

Maul\_ana@gmail.com  
0812344568789


**Jawa Barat**

**Tambah Peserta**

#### Tahapan Penggunaan:

1. Pengguna juga bisa melakukan pencarian terhadap data Peserta Daerah dengan cara memasukkan provinsi asalnya.
  2. Jika Peserta Daerah yang dicari tidak ada, maka daftar tidak akan menampilkan satu pun Peserta Daerah.
  3. Sebaliknya, jika Peserta Daerah yang dicari ada, daftar akan menampilkan data Peserta Daerah tersebut.
- Halaman Kegiatan
    - Daftar Kegiatan

← Kegiatan

<  November, 2020 >

Se	Sel	Ra	Ka	Ju	Sa	Mi
1	2	3	4	5	6	7
8	9	10	11 Pelatihan	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	1	2	3	4	5
6	7	8	9	10	11	12

+

←

Daftar Kegiatan

**11 November, 2020**

**Pelatihan Pemrograman Android**

 10:00 – 16:00 WIB

 Jalan Demang Lebar Daun, nomor 100 C Kecamatan Demang Lebar Daun.

 Sudah dimulai

#### Tahapan Penggunaan:

1. Apabila pengguna menekan tombol Kegiatan pada halaman *Home*, pengguna dapat melihat halaman seperti tampilan di atas. Pada halaman tersebut terdapat daftar Kegiatan yang ditampilkan dalam bentuk kalender.
2. Kegiatan dilihat detailnya dengan cara langsung menekan pada salah satu tanggal, lalu pengguna juga bisa menambah daftar kegiatan tersebut dengan menekan tombol ‘(+)’ di pojok bawah kanan.
  - Tambah, Lihat Detail, Ubah, dan Hapus Kegiatan



←

### Tambah Kegiatan

Lengkapi formulir dibawah ini, ya!

Banner Kegiatan \*

Nama Kegiatan \*

**Cth: Seminar nasional**

Deskripsi Kegiatan \*

**Cth: Kegiatan dilaksanakan pukul 10.00 Wib secara online**

Tanggal & Waktu Kegiatan \*

**Pilih tanggal dan waktu**

Lokasi Kegiatan \*

**Tulis lokasi**

Status Kegiatan \*

**Pilih status**

Pelatih Kegiatan \*

**Pilih pelatih**

**Tambah**

**Batal**

×



## Tambah kegiatan berhasil!

Kegiatan baru berhasil di tambahkan, silahkan lihat daftar kegiatan anda!

[Lihat daftar kegiatan](#)



## Pelatihan pemrograman Android



### Deskripsi

Android merupakan salah satu sistem operasi untuk perangkat mobile yang paling banyak digunakan di dunia saat ini, pelatihan ini akan mengajarkan peserta untuk membuat program untuk platform Mobile.

### Pelatih



John Doe



Dionna Diluc



### Tanggal & Waktu



11 November, 2020

10:00 - 16:00 WIB

### Lokasi



Jalan Demang Lebar Daun, nomor 100 C  
Kecamatan Demang Lebar Daun.

[Buka di Map](#)

### Status Kegiatan

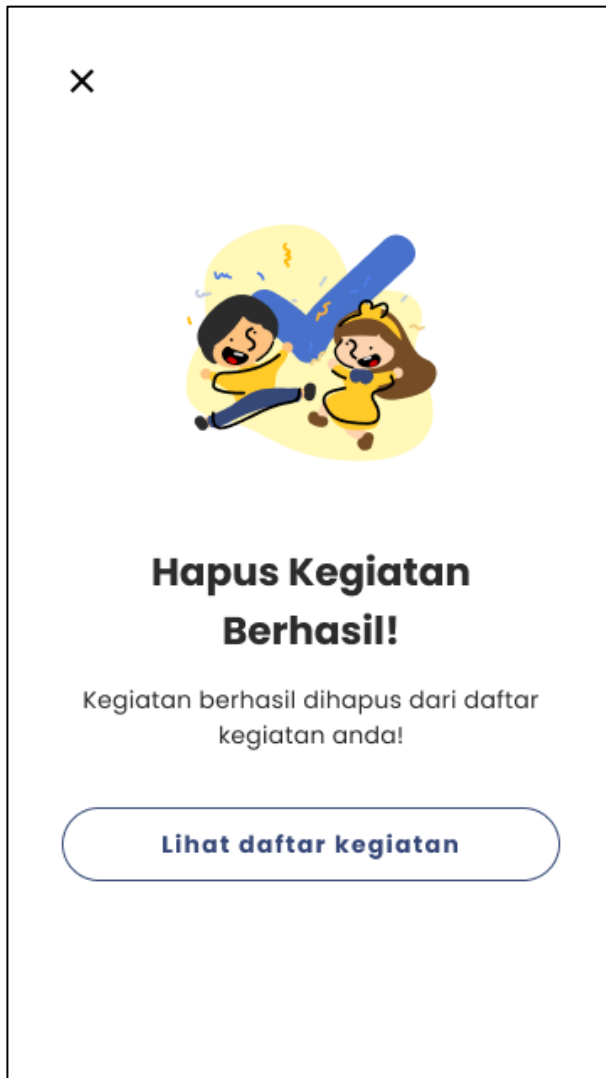


Sudah dimulai

Ubah

Hapus

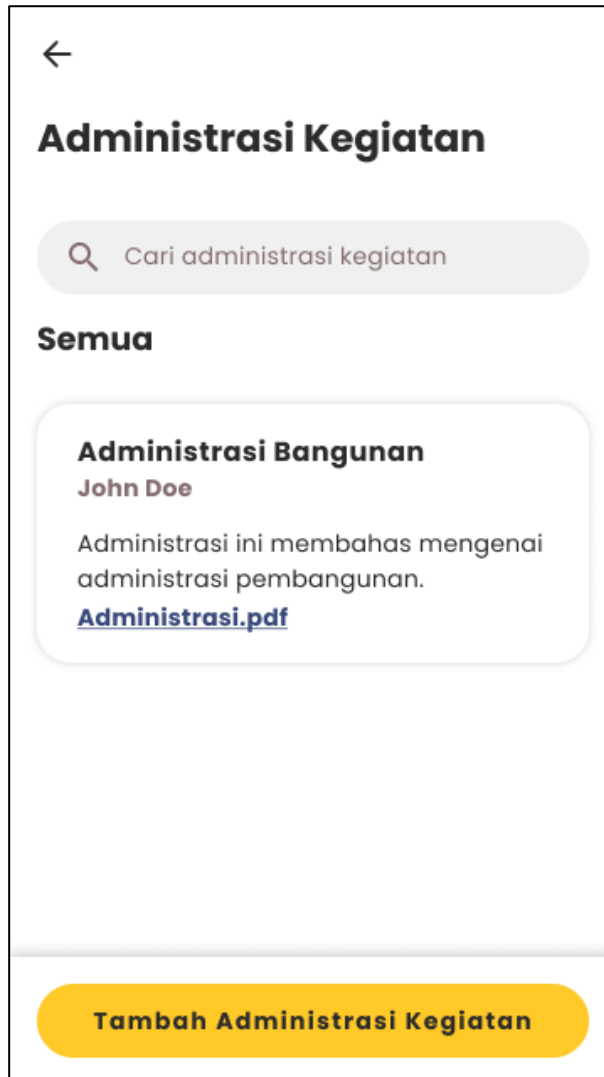




Tahapan Penggunaan:

1. Bila ingin menambahkan data Kegiatan, pengguna harus terlebih dahulu mengisi formulir Tambah Kegiatan lalu meng-klik tombol 'Tambah' untuk memproses.
2. Di halaman daftar Kegiatan sebelumnya, pengguna bisa melihat detail Kegiatan serta bisa melakukan perubahan dan menghapus data Peserta Daerah dengan menekan tombol 'Ubah' atau 'Hapus' pada halaman detail.


- Halaman Administrasi Kegiatan
  - Daftar Administrasi Kegiatan



Tahapan Penggunaan:

1. Apabila pengguna menekan tombol Administrasi Kegiatan pada halaman *Home*, pengguna dapat melihat halaman seperti tampilan di atas. Pada halaman tersebut terdapat daftar Administrasi Kegiatan yang bisa dilihat detailnya dengan cara langsung menekan pada salah satu di daftar, lalu pengguna juga bisa menambah daftar tersebut dengan menekan tombol 'Tambah Administrasi Kegiatan'.

- Tambah, Lihat Detail, Ubah, dan Hapus Administrasi Kegiatan



## Tambah Administrasi Kegiatan

Lengkapi formulir dibawah ini, ya!


Judul Administrasi Kegiatan \*

**Cth: Administrasi Kegiatan**

Deskripsi Administrasi Kegiatan \*

**Cth: Ini deskripsi administrasi kegiatan**

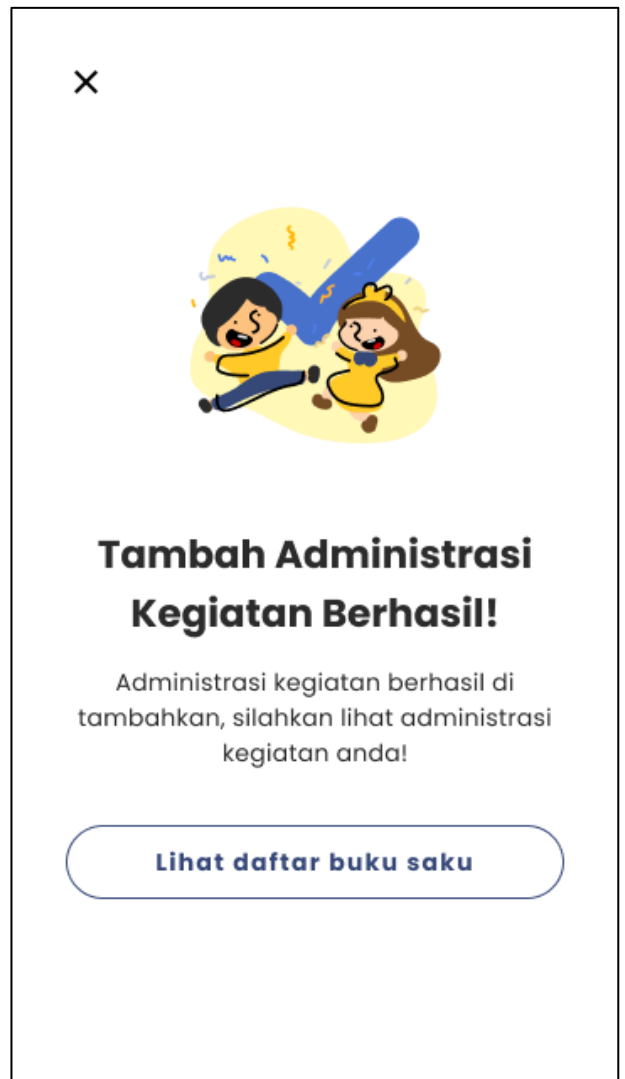
File Administrasi Kegiatan \*



Upload file

**Tambah**

**Batal**



← Detail Administrasi Kegiatan



## Administrasi Bangunan

### Deskripsi Buku



Administrasi ini membahas mengenai administrasi pembangunan.

### File Buku



[Administrasi.pdf](#)

### Pengirim



Mauladi

Ubah

Hapus

←

## Ubah Data Administrasi Kegiatan

Lengkapi formulir dibawah ini, ya!

Judul Buku \*

Administrasi Bangunan

Deskripsi Buku \*

Administrasi ini membahas mengenai administrasi pembangunan.

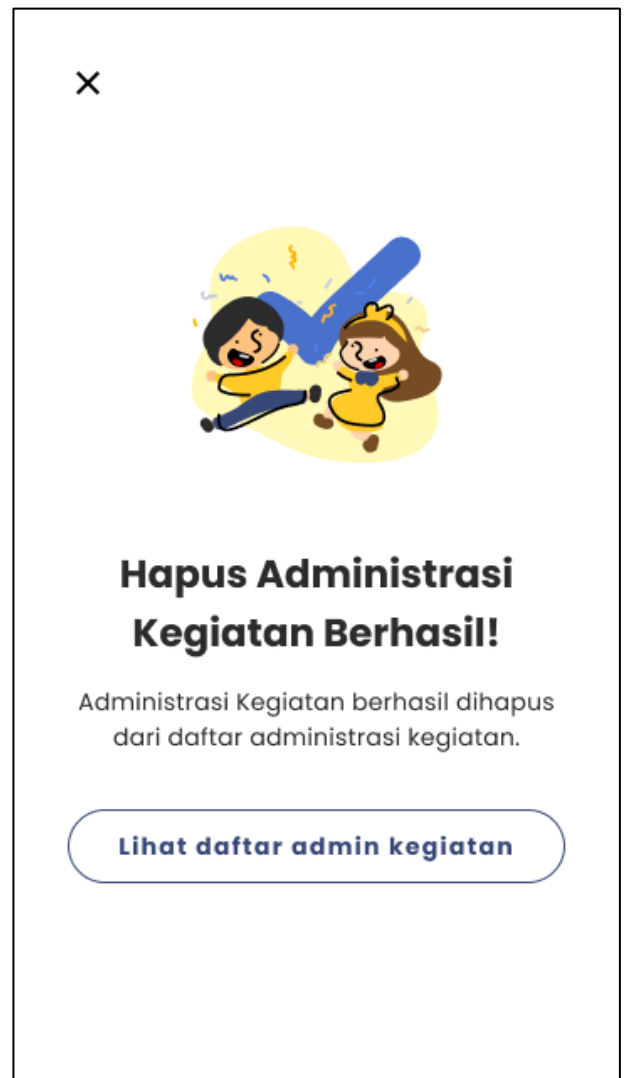
File Buku \*



Administrasi.pdf

Simpan

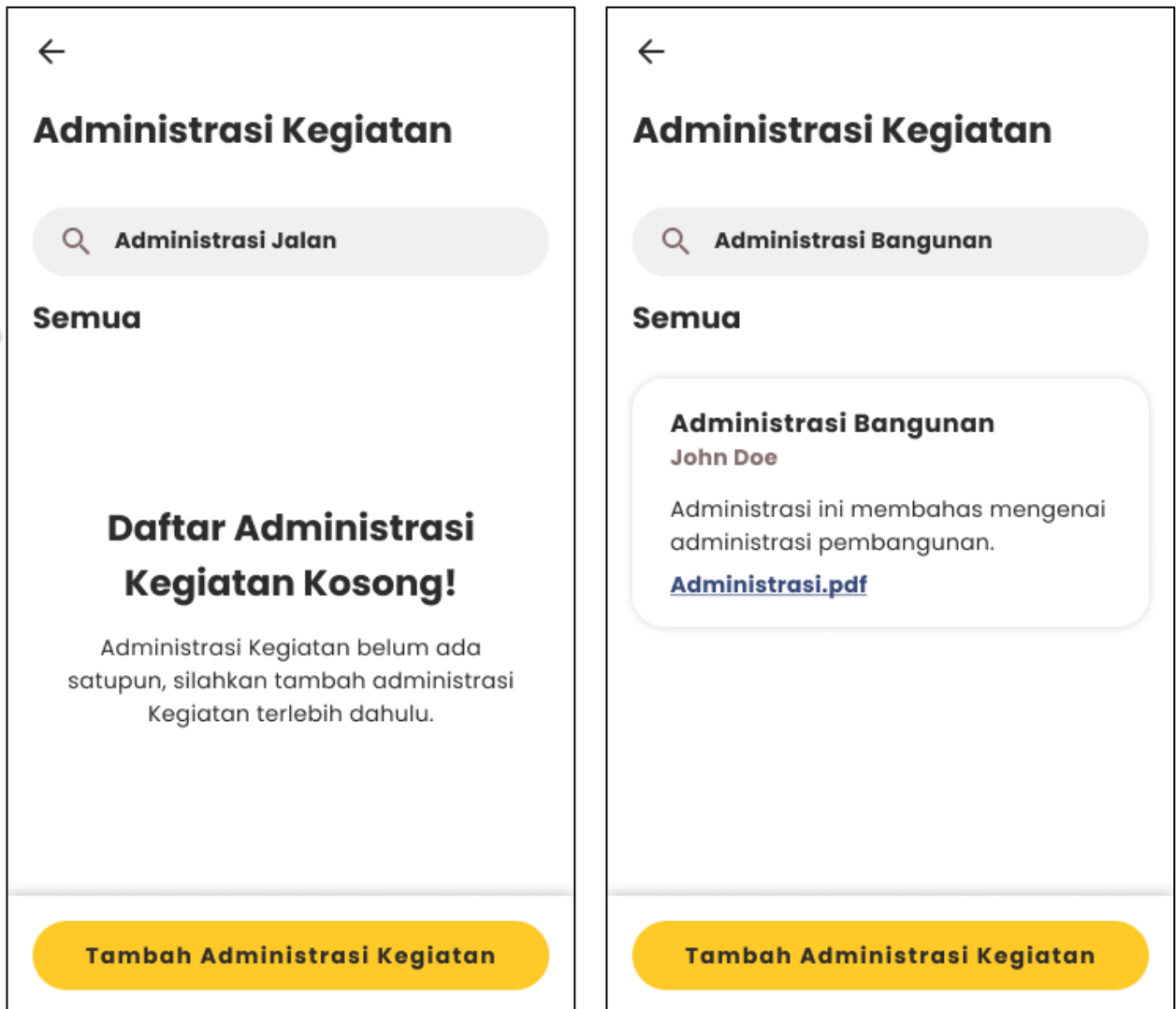
Batal



Tahapan Penggunaan:

1. Bila ingin menambahkan data Administrasi Kegiatan, pengguna harus terlebih dahulu mengisi formulir Tambah Administrasi Kegiatan lalu meng-klik tombol 'Tambah' untuk memproses.
2. Di halaman daftar Administrasi Kegiatan sebelumnya, pengguna bisa melihat detail Administrasi Kegiatan serta bisa melakukan perubahan dan menghapus data Administrasi Kegiatan dengan menekan tombol 'Ubah' atau 'Hapus' pada halaman detail.

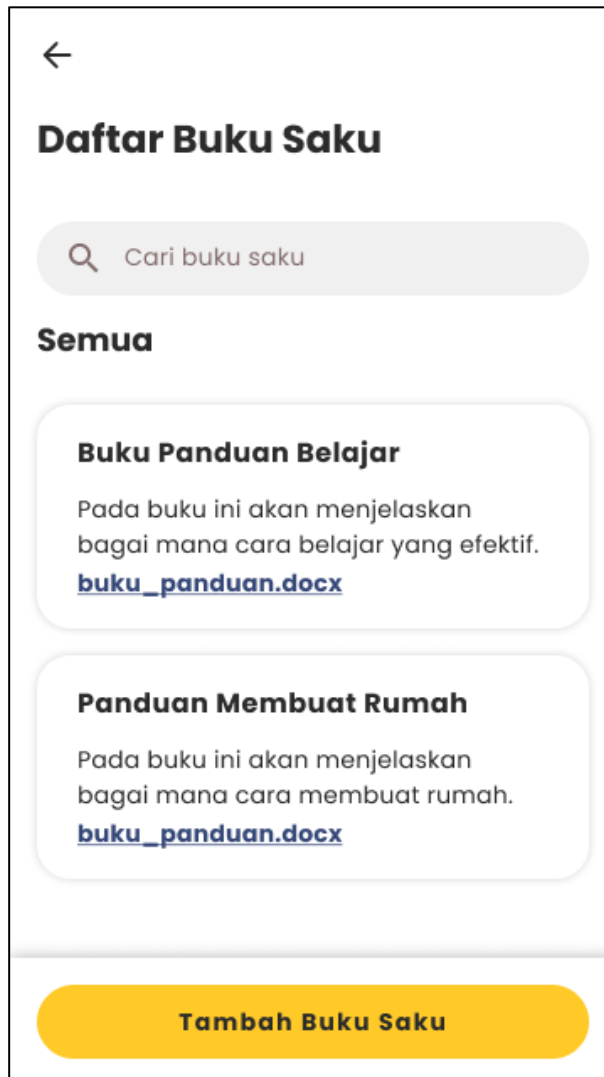
- Cari Administrasi Kegiatan



Tahapan Penggunaan:

1. Pengguna juga bisa melakukan pencarian terhadap data Administrasi Kegiatan dengan cara memasukkan Judul Buku.
2. Jika Administrasi Kegiatan yang dicari tidak ada, maka daftar tidak akan menampilkan satu pun Administrasi Kegiatan.
3. Sebaliknya, jika Administrasi Kegiatan yang dicari ada, daftar akan menampilkan data Administrasi Kegiatan tersebut.

- Halaman Buku Saku
  - Daftar Buku Saku



Tahapan Penggunaan:

1. Apabila pengguna menekan tombol Buku Saku pada halaman *Home*, pengguna dapat melihat halaman seperti tampilan di atas. Pada halaman tersebut terdapat daftar Buku Saku yang bisa dilihat detailnya dengan cara langsung menekan pada salah satu di daftar, lalu pengguna juga bisa menambah daftar tersebut dengan menekan tombol 'Tambah Buku Saku'.

- Tambah, Lihat Detail, Ubah, dan Hapus Buku Saku

### Tambah Buku Saku

Lengkapi formulir dibawah ini, ya!

Judul Buku \*

**Cth: Buku Panduan Belajar Mengajar**

Deskripsi Buku \*

**Cth: Ini deskripsi buku**

File Buku \*

**Upload file**

**Tambah**

**Batal**

### Tambah Buku Saku Berhasil!

Buku saku berhasil di tambahkan, silahkan lihat daftar buku saku anda!

**Lihat daftar buku saku**



← Detail Buku Saku



## Buku Panduan Belajar

### Deskripsi Buku



Pada buku ini akan menjelaskan bagaimana cara belajar yang efektif.

### File Buku



[Buku\\_panduan.docx](#)

### Pengirim



Maulana

Ubah

Hapus

←

## Ubah Data Buku Saku

Lengkapi formulir dibawah ini, ya!

Judul Buku \*

Buku Panduan Belajar

Deskripsi Buku \*

**Pada buku ini akan menjelaskan bagaimana cara belajar yang efektif.**

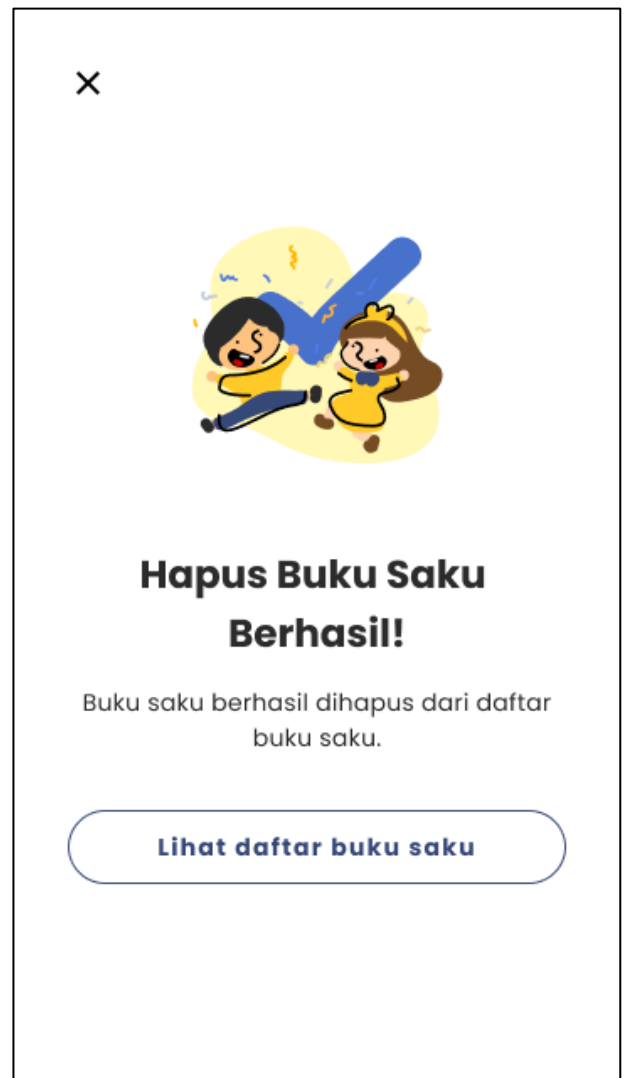
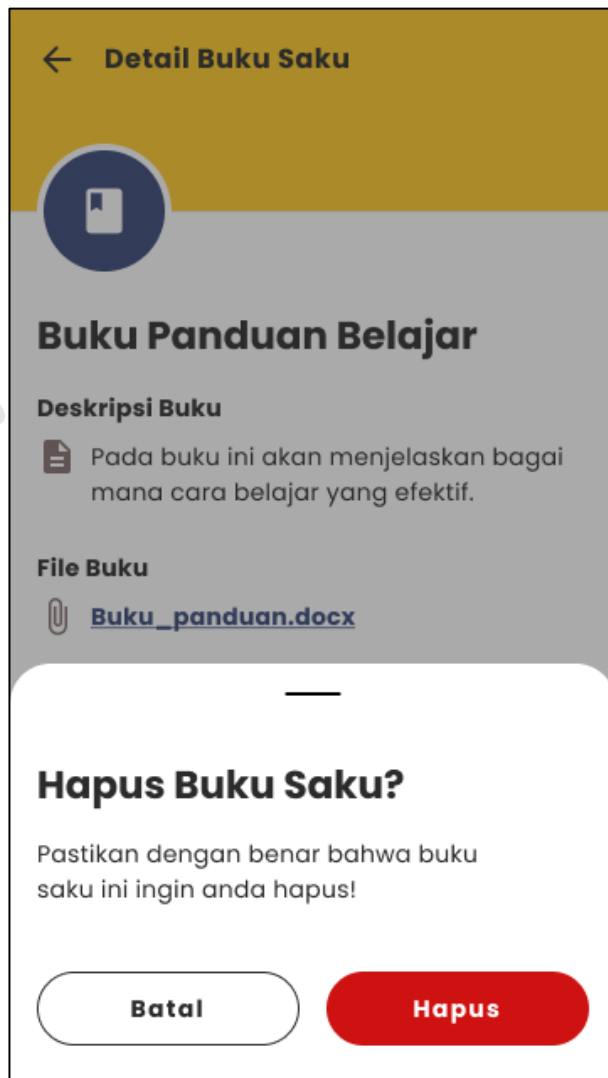
File Buku \*



Buku\_panduan.docx

Simpan

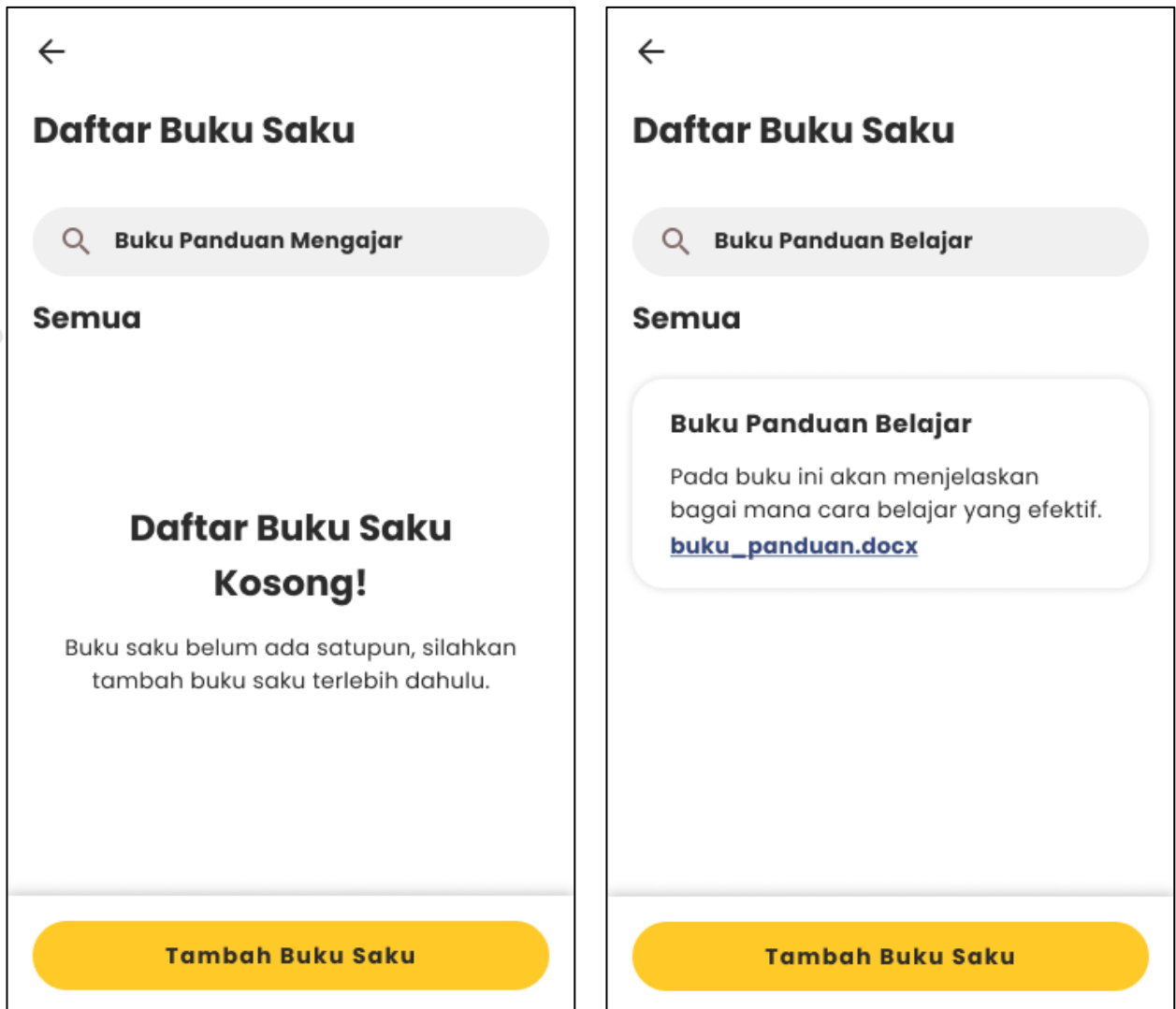
Batal



Tahapan Penggunaan:

1. Bila ingin menambahkan data Buku Saku, pengguna harus terlebih dahulu mengisi formulir Tambah Buku Saku lalu meng-klik tombol 'Tambah' untuk memproses.
2. Di halaman daftar Buku Saku sebelumnya, pengguna bisa melihat detail Buku Saku serta bisa melakukan perubahan dan menghapus data Buku Saku menekan tombol 'Ubah' atau 'Hapus' pada halaman detail.

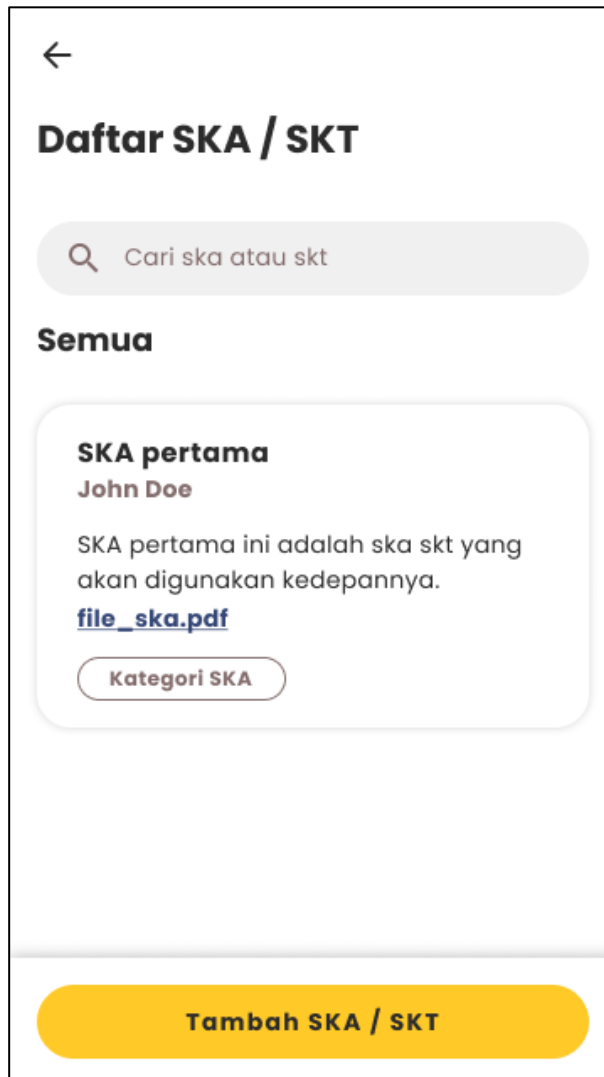
- Cari Buku Saku



Tahapan Penggunaan:

1. Pengguna juga bisa melakukan pencarian terhadap data Buku Saku dengan cara memasukkan Judul Buku.
2. Jika Buku Saku yang dicari tidak ada, maka daftar tidak akan menampilkan satu pun Buku Saku.
3. Sebaliknya, jika Buku Saku yang dicari ada, daftar akan menampilkan data Buku Saku tersebut.

- Halaman SKA/SKT
  - Daftar SKA/SKT



Tahapan Penggunaan:

1. Apabila pengguna menekan tombol SKA/SKT pada halaman *Home*, pengguna dapat melihat halaman seperti tampilan di atas. Pada halaman tersebut terdapat daftar SKA/SKT yang bisa dilihat detailnya dengan cara langsung menekan pada salah satu di daftar, lalu pengguna juga bisa menambah daftar tersebut dengan menekan tombol 'Tambah SKA/SKT'.

- Tambah, Lihat Detail, Ubah, dan Hapus SKA

**Tambah SKA / SKT**  
Lengkapi formulir dibawah ini, ya!

Judul SKA / SKT \*

**Cth: Judul SKA**

Deskripsi SKA / SKT \*

**Cth: Ini merupakan deskripsi detail dari SKA / SKT**

Kategori SKA / SKT \*

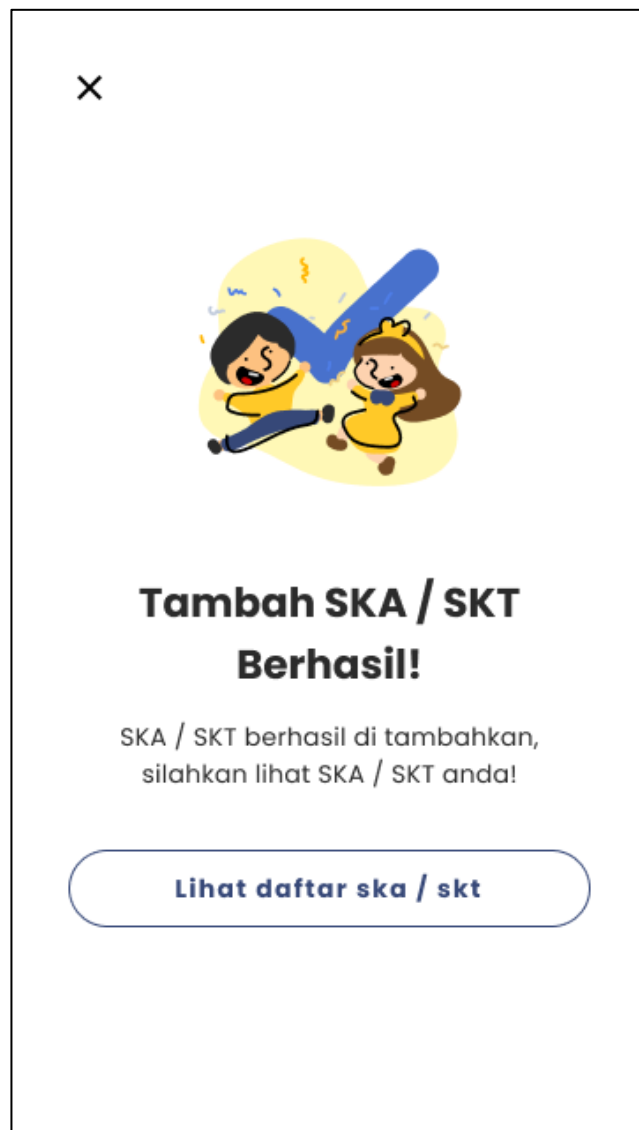
**Pilih Kategori**

File SKA / SKT \*

**Upload file**

**Tambah**

**Batal**



← Detail SKA / SKT



## SKT Pertama

### Deskripsi SKA / SKT



SKA pertama ini adalah ska skt yang akan digunakan kedepannya.

### File SKA / SKT



[file\\_ska.pdf](#)

### Kategori SKA / SKT



Kategori SKA

### Pengirim



Munira

Ubah

Hapus

←

## Ubah Data SKA / SKT

Lengkapi formulir dibawah ini, ya!

Judul SKA / SKT \*

SKT Pertama

Deskripsi SKA / SKT \*

SKA pertama ini adalah ska skt yang akan digunakan kedepannya.

Kategori SKA / SKT \*

Kategori SKA



File SKA / SKT \*



file\_ska.pdf

Simpan

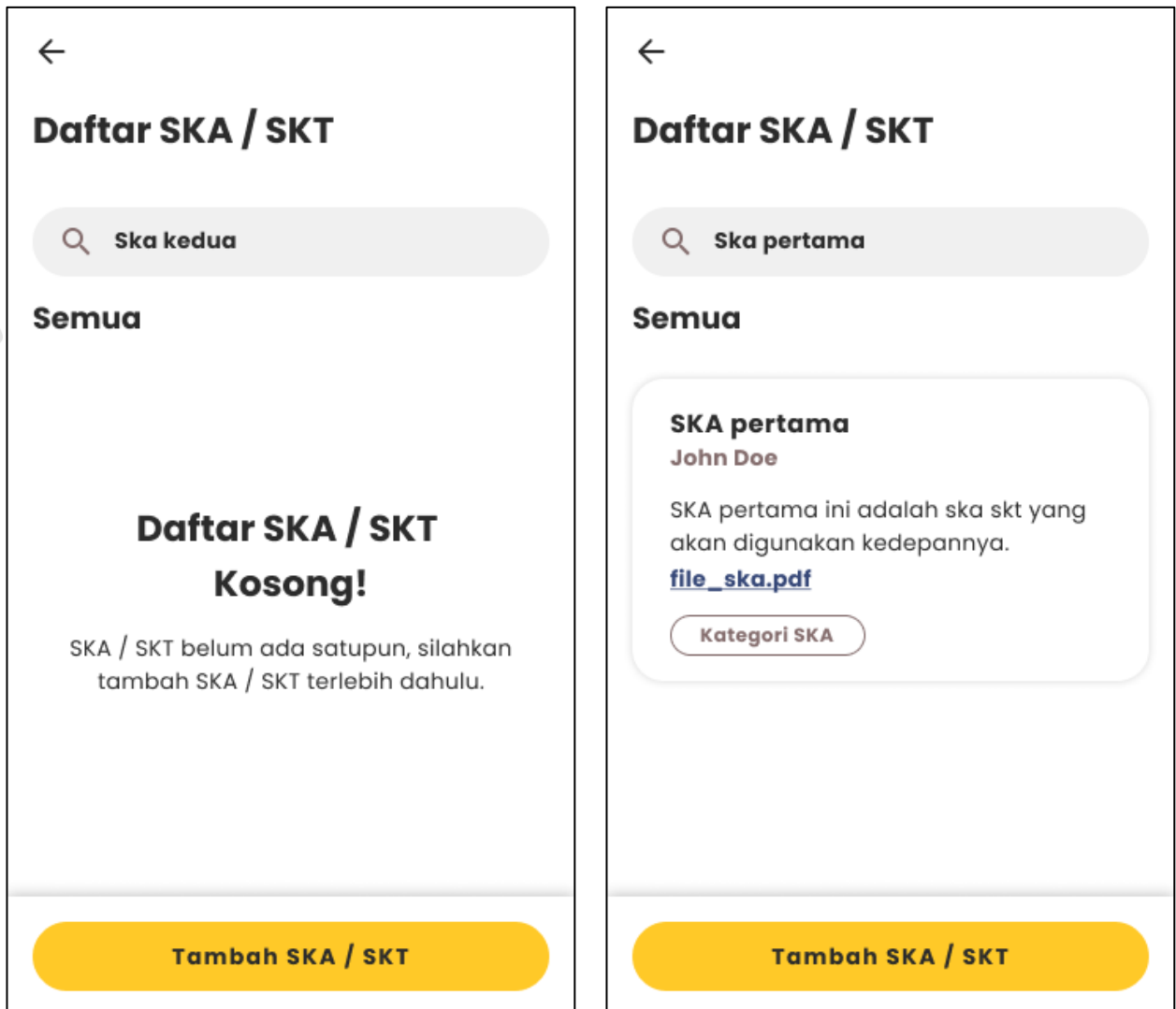
Batal



Tahapan Penggunaan:

1. Bila ingin menambahkan data SKA/SKT, pengguna harus terlebih dahulu mengisi formulir Tambah SKA/SKT lalu meng-klik tombol 'Tambah' untuk memproses.
2. Di halaman daftar SKA/SKT sebelumnya, pengguna bisa melihat detail SKA/SKT serta bisa melakukan perubahan dan menghapus data SKA/SKT menekan tombol 'Ubah' atau 'Hapus' pada halaman detail.

- Cari SKA

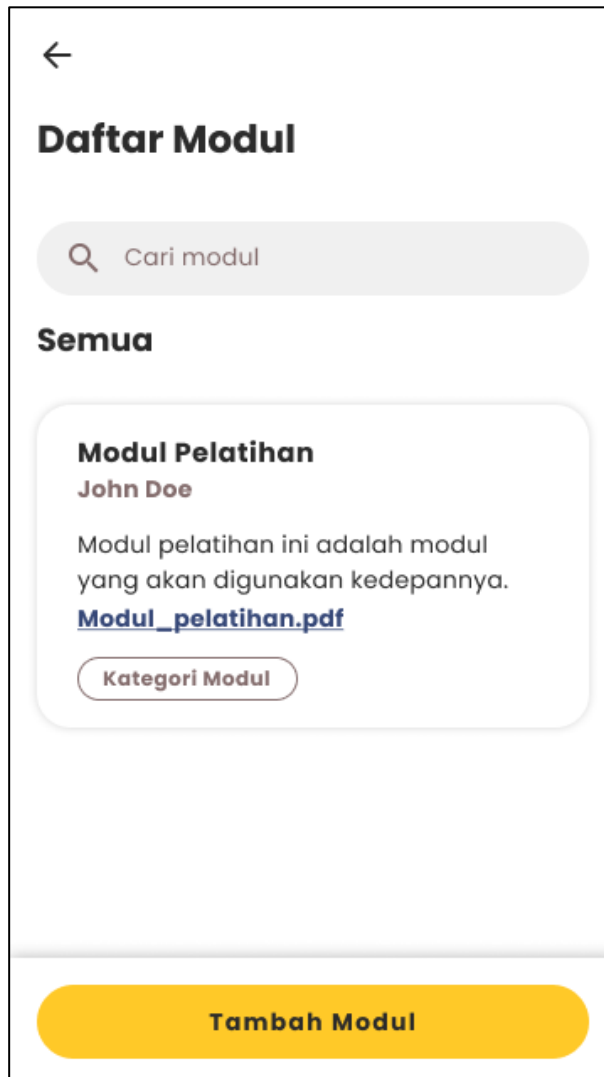


Tahapan Penggunaan:

1. Pengguna juga bisa melakukan pencarian terhadap data SKA/SKT dengan cara memasukkan Judul SKA/SKT.
2. Jika SKA/SKT yang dicari tidak ada, maka daftar tidak akan menampilkan satu pun Buku Saku.
3. Sebaliknya, jika SKA/SKT yang dicari ada, daftar akan menampilkan data SKA/SKT tersebut.



- Halaman Modul
  - Daftar Modul



Tahapan Penggunaan:

1. Apabila pengguna menekan tombol Modul pada halaman *Home*, pengguna dapat melihat halaman seperti tampilan di atas. Pada halaman tersebut terdapat daftar Modul yang bisa dilihat detailnya dengan cara langsung menekan pada salah satu di daftar, lalu pengguna juga bisa menambah daftar tersebut dengan menekan tombol 'Tambah Modul'.

- Tambah, Lihat Detail, Ubah, dan Hapus Modul

### Tambah Modul

Lengkapi formulir dibawah ini, ya!

Judul Modul \*

**Cth: Judul Modul**

---

Deskripsi Modul \*

**Cth: Ini merupakan deskripsi detail dari Modul**

---

Kategori Modul \*

**Pilih Kategori**

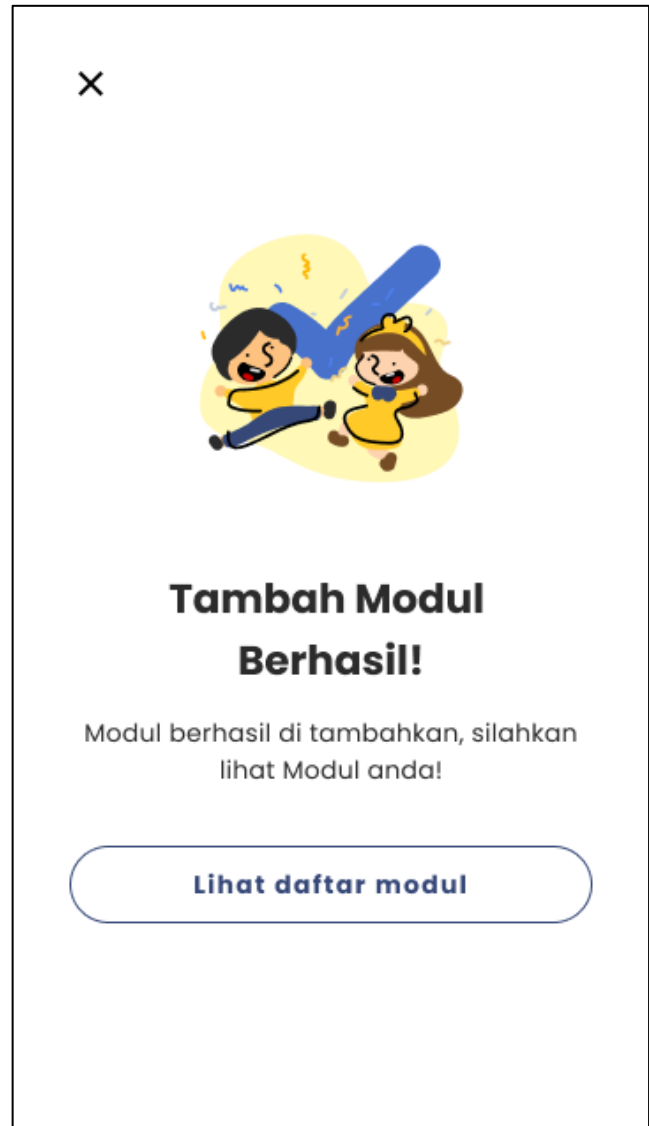
---

File Modul \*

**Upload file**

**Tambah**

**Batal**



← Detail Modul



## Modul Pelatihan

### Deskripsi Modul



Modul pelatihan ini adalah modul yang akan digunakan kedepannya.

### File Modul



[Modul\\_Pelatihan.pdf](#)

### Kategori Modul



Kategori Modul

### Pengirim



Mukhlis

Ubah

Hapus

←

## Ubah Data Modul

Lengkapi formulir dibawah ini, ya!

Judul SKA / SKT \*

Modul Pelatihan

Deskripsi SKA / SKT \*

Modul pelatihan ini adalah modul yang akan digunakan kedepannya.

Kategori SKA / SKT \*

Kategori Modul



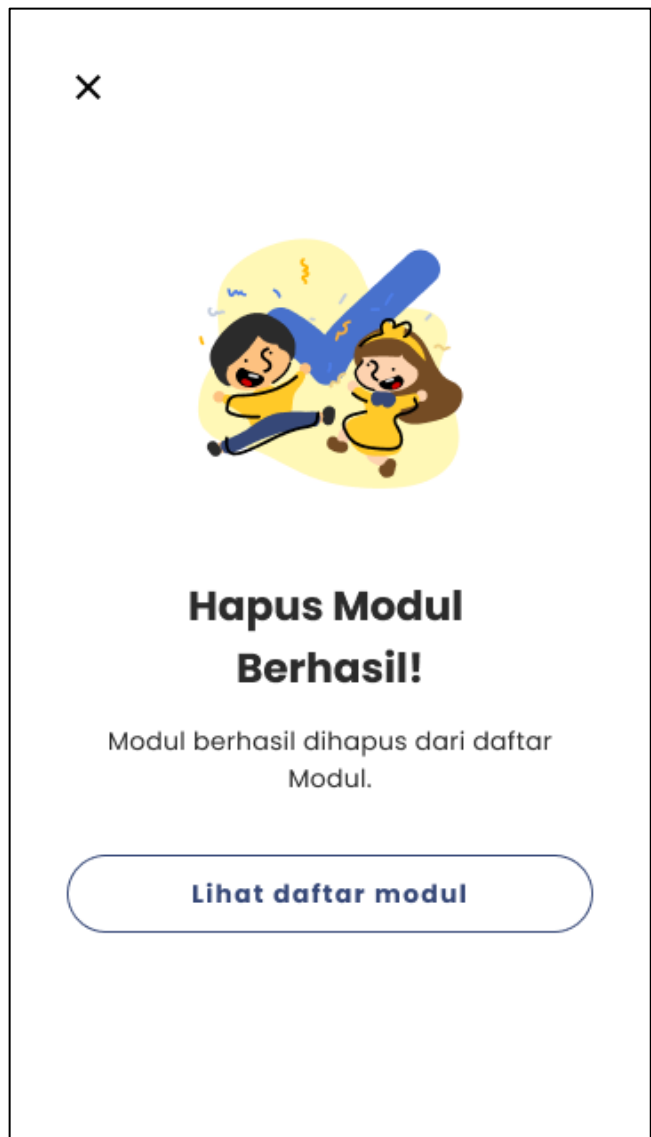
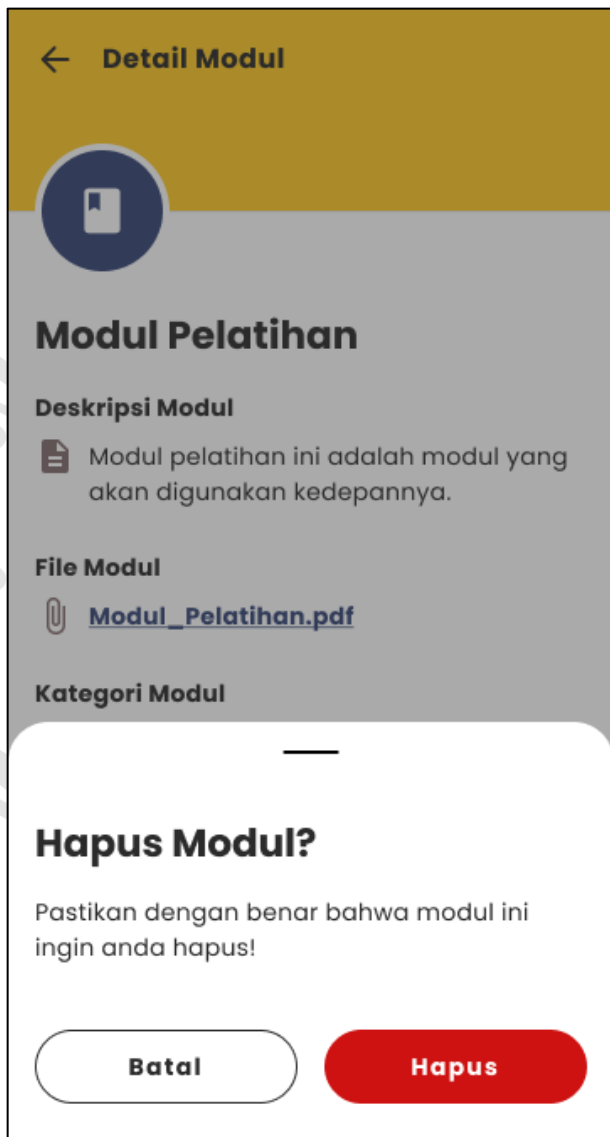
File SKA / SKT \*



Modul\_Pelatihan.pdf

Simpan

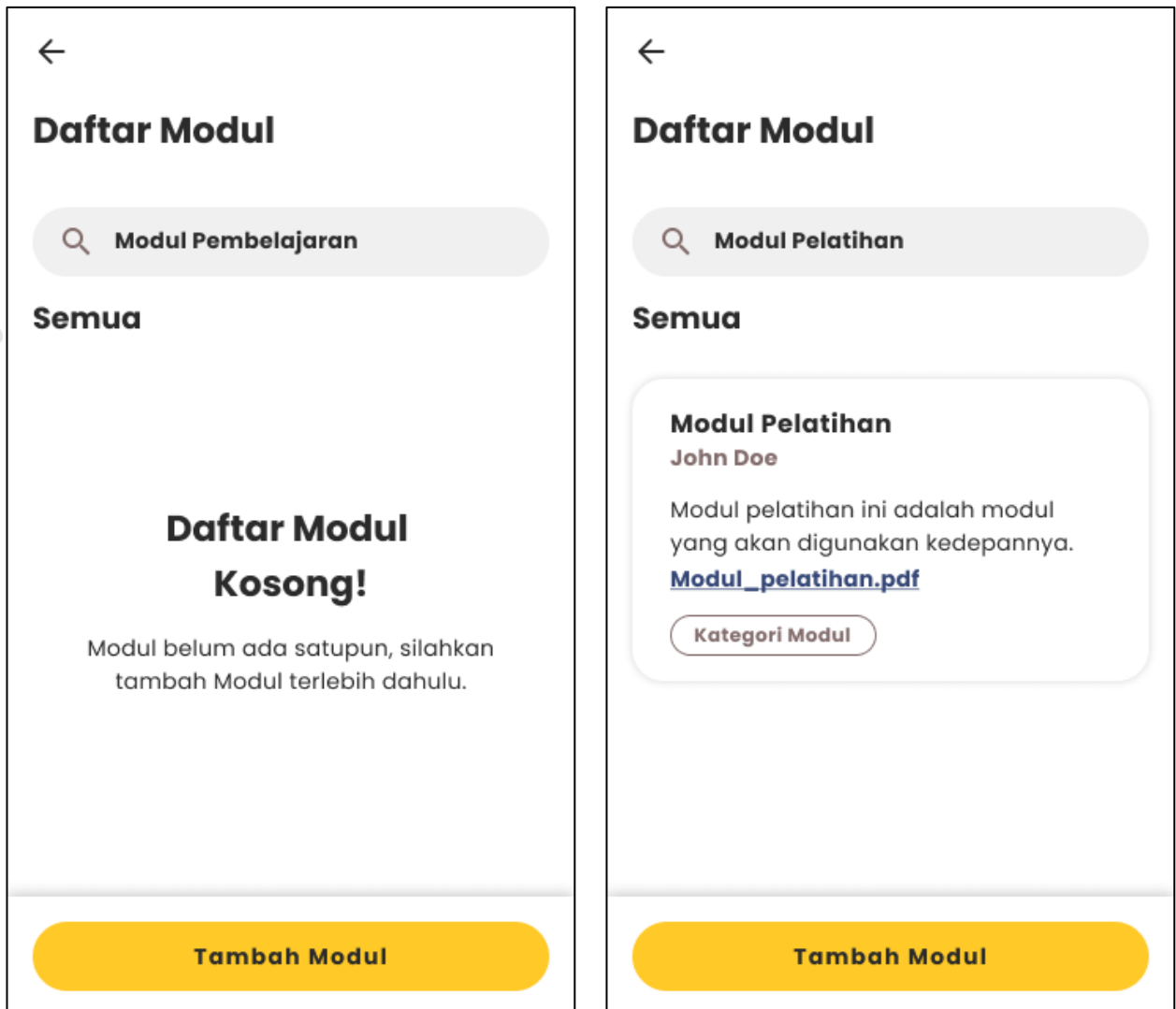
Batal



Tahapan Penggunaan:

1. Bila ingin menambahkan data Modul, pengguna harus terlebih dahulu mengisi formulir Tambah Modul lalu meng-klik tombol 'Tambah' untuk memproses.
2. Di halaman daftar Modul sebelumnya, pengguna bisa melihat detail Modul serta bisa melakukan perubahan dan menghapus data Modul menekan tombol 'Ubah' atau 'Hapus' pada halaman detail.

- Cari Modul



Tahapan Penggunaan:

1. Pengguna juga bisa melakukan pencarian terhadap data Modul dengan cara memasukkan Judul Modul.
2. Jika Modul yang dicari tidak ada, maka daftar tidak akan menampilkan satu pun Buku Saku.
3. Sebaliknya, jika Modul yang dicari ada, daftar akan menampilkan data Modul tersebut.

## 5. Rute API

	Base URL : APi Dev: <a href="#">api.classico.id/isell/</a>	Pattern	Type	Request Param	Request Body	Header	Res	Data Object	Function Query	Base Query	Ket.	Status
User Management	{{ API_URL }} /user											
Login	/login	REST	POST		{ email_no_telepon:"TEXT" password:"TEXT" }		{ status:"Success Error" message:"TEXT" data: { object } }	{ id_token:"TEXT" }	SELECT * FROM is_email_no_telpon_exist(email_no_telpon_param character varying) SELECT * FROM is_email_verified(email_no_telepon_param character varying) SELECT * FROM login(email_no_telpon_param varchar, password_digest_param varchar)			
Register	/register	REST	POST		{ nama:"TEXT" email:"TEXT" no_telpon:"TEXT" password:"TEXT" }		{ status:"Success Error" message:"TEXT" }	-	SELECT * FROM is_email_no_telpon_exist(email_no_telpon_param character varying) SELECT * FROM is_id_user_exist(id_user_param varchar) SELECT * FROM			

									register(id_user_param varchar, nama_param varchar, email_param varchar, no_telpon_param varchar, password_digest_param varchar)			
Email Verification	/email-verification/{id_user}	REST	POST				{ status:"Success Error" message:"TEXT" }	-	SELECT * FROM email_verification(id_user_param varchar)			
View User Detail	/	REST	GET			{ Authentication : Bearer Token }	{ status:"Success Error" message:"TEXT" data:{object} }	{ nama:"TEXT" email:"TEXT" no_telpon:"TEXT" provinsi:"TEXT" kota_kabupaten:"TEXT" kecamatan:"TEXT" kelurahan:"TEXT" jenis_kelamin:"TEXT" status_perkawinan:"TEXT" " nama_perusahaan:"TEXT" jabatan:"TEXT" utusan:"TEXT" alamat_rumah:"TEXT" nik:"TEXT" rt:"TEXT" rw:"TEXT" kode_pos:"TEXT" kode_area:"TEXT"	SELECT * FROM get_detail_user(id_user_param varchar)			

								no_telepon:"TEXT" no_handphone:"TEXT" status_rumah:"TEXT" pendidikan:"TEXT" jabker:"TEXT" kualifikasi_jabker:"TEXT" klasifikasi_jabker:"TEXT" kompetensi:"TEXT" no_sertifikat:"TEXT" }				
Edit User Detail	/	REST	POST		{ nama:"TEXT" email:"TEXT" no_telpon:"TEXT" id_provinsi:"INT" id_kota_kabupaten:"INT" id_kecamatan:"INT" id_kelurahan:"INT" id_jenis_kelamin:"INT" id_status_perkawinan:"INT" nama_perusahaan:"TEXT" jabatan:"TEXT" utusan:"TEXT" alamat_rumah:"TEXT" nik:"TEXT" rt:"TEXT" rw:"TEXT" }	{ Authentication : Bearer Token }	{ status:"Success Error" message:"TEXT" }	-	SELECT * FROM edit_user(id_user_param varchar, nama_param varchar, email_param varchar, no_telepon_param varchar, id_provinsi_param int, id_kabupaten_kota_param int, id_kecamatan_param int, id_kelurahan_param int, id_jenis_kelamin_param int, id_status_perkawinan_param int, nama_perusahaan_param varchar, jabatan_param			



					<pre>kode_pos:"TEXT" kode_area:"TEXT" no_handphone:"TEXT" id_status_rumah:"TEXT" id_pendidikan:"TEXT" id_jabker:"TEXT" id_kompetensi:"TEXT" no_sertifikat:"TEXT" file_foto_profil:"FILE" }</pre>				<pre>varchar, utusan_param varchar, alamat_rumah_param text, nik_param varchar, rt_param varchar, rw_param varchar, kode_pos_param varchar, kode_area_param varchar, no_handphone_param varchar, id_status_rumah_param int, id_pendidikan_param int, id_jabker_param int, id_kompetensi_param int, no_sertifikat_param varchar, foto_profile_param text)</pre>			
Forgot Password	/forgot-password	REST	POST		<pre>{ email_no_telepon:"TEXT" }</pre>		<pre>{ status:"Success Error" message:"TEXT" }</pre>	-	<pre>SELECT * FROM get_id_user_by_email_no_telepon(email_no_telepon_param varchar) SELECT * FROM add_forgot_password</pre>			

									(id_forgot_password_param varchar, id_user_param varchar)			
Change Password Forgot Password	/forgot-password/change-password/{id_forgot_password}	REST	PUT		{ new_password:"TEXT" }		{ status:"Success Error" message:"TEXT" }	-	SELECT * FROM is_forgot_password_available(id_forgot_password_param varchar) SELECT * FROM edit_password_forgot_password(id_forgot_password_param varchar, new_password_digest varchar)			
Peserta Management	{{API_URL}}/peserta											
Add Peserta	/	REST	POST		{ nama:"TEXT" email:"TEXT" no_telpn:"TEXT" id_provinsi:"INT" id_kota_kabupaten:"INT" id_kecamatan:"INT" id_kelurahan:"INT" id_jenis_kelamin:"INT" id_status_perkawinan:"INT" }	s	{ status:"Success Error" message:"TEXT" }	-	SELECT * FROM is_id_user_exist(id_user_param varchar)  SELECT * FROM is_user_admin(id_user_param varchar)  SELECT * FROM add_peserta(id_user_param varchar, nama_param varchar, email_param varchar,			

					nama_perusahaan:"TEXT" jabatan:"TEXT" utusan:"TEXT" alamat_rumah:"TEXT" nik:"TEXT" rt:"TEXT" rw:"TEXT" kode_pos:"TEXT" kode_area:"TEXT" no_handphone:"TEXT" id_status_rumah:"TEXT" id_pendidikan:"TEXT" id_jabker:"TEXT" id_kompetensi:"TEXT" no_sertifikat:"TEXT" file_foto_profil:"FILE" }				password_param varchar, no_telepon_param varchar, id_provinsi_param int, id_kabupaten_kota_param int, id_kecamatan_param int, id_kelurahan_param int, id_jenis_kelamin_param int, id_status_perkawinan_param int, nama_perusahaan_param varchar, jabatan_param varchar, utusan_param varchar, alamat_rumah_param text, nik_param varchar, rt_param varchar, rw_param varchar, kode_pos_param varchar, kode_area_param varchar, no_handphone_param varchar, id_status_rumah_param		
--	--	--	--	--	--	--	--	--	---	--	--

									m int, id_pendidikan_param int, id_jabker_param int, id_kompetensi_param int, no_sertifikat_param varchar, foto_profile_param text)			
Edit Peserta	/id_user_peserta}	REST	POST		<pre> {   nama:"TEXT"   email:"TEXT"   no_telpon:"TEXT"   id_provinsi:"INT"   id_kota_kabupaten:"INT"   id_kecamatan:"INT"   id_kelurahan:"INT"   id_jenis_kelamin:"INT"   id_status_perkawinan:"INT"   nama_perusahaan:"TEXT"   jabatan:"TEXT"   utusan:"TEXT"   alamat_rumah:"TEXT"   nik:"TEXT"   rt:"TEXT"   rw:"TEXT" </pre>	s	<pre> {   status:"Success Error"   message:"TEXT" } </pre>	-	SELECT * FROM is_user_admin(id_user_param varchar)  SELECT * FROM edit_user(id_user_param varchar, nama_param varchar, email_param varchar, no_telepon_param varchar, id_provinsi_param int, id_kabupaten_kota_param int, id_kecamatan_param int, id_kelurahan_param int, id_jenis_kelamin_param int, id_status_perkawinan_param int,			

					<pre>kode_pos:"TEXT" kode_area:"TEXT" no_handphone:"TEXT" id_status_rumah:"TEXT" id_pendidikan:"TEXT" id_jabker:"TEXT" id_kompetensi:"TEXT" no_sertifikat:"TEXT" file_foto_profil:"FILE" }</pre>			<pre>nama_perusahaan_param varchar, jabatan_param varchar, utusan_param varchar, alamat_rumah_param text, nik_param varchar, rt_param varchar, rw_param varchar, kode_pos_param varchar, kode_area_param varchar, no_handphone_param varchar, id_status_rumah_param int, id_pendidikan_param int, id_jabker_param int, id_kompetensi_param int, no_sertifikat_param varchar, foto_profile_param text)</pre>				
Delete Peserta	/id_user_peserta}	REST	DELETE			{ Authentication : Bearer Token }	{ status:"Success Error" message:"TEXT" }	-	<pre>SELECT * FROM is_user_admin(id_user_param varchar)  SELECT * FROM</pre>			

									delete_peserta(id_user _param varchar)			
View Peserta Keseluruhan	/	REST	GET			{ Authenticatio n : Bearer Token }	{ status:"Success Error" message:"TEXT" data:[{object}] }	{ nama:"TEXT" email:"TEXT" no_telpon:"TEXT" provinsi:"TEXT" kota_kabupaten:"TEXT" kecamatan:"TEXT" kelurahan:"TEXT" jenis_kelamin:"TEXT" status_perkawinan:"TEXT" " nama_perusahaan:"TEXT" jabatan:"TEXT" utusan:"TEXT" alamat_rumah:"TEXT" nik:"TEXT" rt:"TEXT" rw:"TEXT" kode_pos:"TEXT" kode_area:"TEXT" no_handphone:"TEXT" status_rumah:"TEXT" pendidikan:"TEXT" jabker:"TEXT" kompetensi:"TEXT" no_sertifikat:"TEXT" file_foto_profile:"TEXT" }	SELECT * FROM is_user_admin(id_user _param varchar) SELECT * FROM get_peserta()			
View Peserta	/provinsi/{id _provinsi}	REST	GET			{ Authenticatio	{ status:"Success Error"	{ nama:"TEXT"	SELECT * FROM is_user_admin(id_user			

Daerah						n : Bearer Token }	message:"TEXT" data:[{object}] }	email:"TEXT" no_telpon:"TEXT" provinsi:"TEXT" kota_kabupaten:"TEXT" kecamatan:"TEXT" kelurahan:"TEXT" jenis_kelamin:"TEXT" status_perkawinan:"TEXT" " nama_perusahaan:"TEXT" jabatan:"TEXT" utusan:"TEXT" alamat_rumah:"TEXT" nik:"TEXT" rt:"TEXT" rw:"TEXT" kode_pos:"TEXT" kode_area:"TEXT" no_handphone:"TEXT" status_rumah:"TEXT" pendidikan:"TEXT" jabker:"TEXT" kompetensi:"TEXT" no_sertifikat:"TEXT" file_foto_profile:"TEXT" }	_param varchar) SELECT * FROM get_peserta_per_daera h(id_provinsi_param int)			
View Detail Peserta	/{{id_user_peserta}}	REST	GET			{ Authentication : Bearer Token }	{ status:"Success Error" message:"TEXT" data:{object} }	{ nama:"TEXT" email:"TEXT" no_telpon:"TEXT" provinsi:"TEXT" kota_kabupaten:"TEXT"	SELECT * FROM is_user_admin(id_user_param varchar) SELECT * FROM get_detail_peserta(id_user_param varchar)			

								kecamatan:"TEXT" kelurahan:"TEXT" jenis_kelamin:"TEXT" status_perkawinan:"TEXT" " nama_perusahaan:"TEXT" jabatan:"TEXT" utusan:"TEXT" alamat_rumah:"TEXT" nik:"TEXT" rt:"TEXT" rw:"TEXT" kode_pos:"TEXT" kode_area:"TEXT" no_handphone:"TEXT" status_rumah:"TEXT" pendidikan:"TEXT" jabker:"TEXT" kompetensi:"TEXT" no_sertifikat:"TEXT" file_foto_profile:"TEXT" }				
Tenaga Ahli Management	{{API_URL}}/tenaga-ahli											
Add Tenaga Ahli	/	REST	POST		{ nama_lengkap:"TEXT" " tempat_lahir:"TEXT" tanggal_lahir:"TEXT" id_jenis_kelamin:"TEXT" }	{ Authentication : Bearer Token }	{ status:"Success Error" message:"TEXT" }	-	SELECT * FROM is_tenaga_ahli_exist(id_tenaga_ahli_param varchar) SELECT * FROM is_user_admin(id_user_param varchar)			



					nik:"TEXT" email:"TEXT" alamat_rumah:"TEXT" " id_provinsi:"TEXT" id_kabupaten_kota:"TEXT" no_telepon_rumah:"TEXT" no_handphone:"TEXT" "				SELECT * FROM add_tenaga_ahli(id_tenaga_ahli_param varchar, nama_param varchar, tempat_lahir_param varchar, tanggal_lahir_param timestamp, id_jenis_kelamin_param int, nik_param varchar, email_param varchar, alamat_rumah_param text, id_provinsi_param int, id_kabupaten_kota_param int, no_telepon_param varchar, no_handphone_param varchar)			
Edit Tenaga Ahli	/id_tenaga_ahli}	REST	PUT		{ nama_lengkap:"TEXT" tempat_lahir:"TEXT" tanggal_lahir:"TEXT" " id_jenis_kelamin:"TEXT" nik:"TEXT" email:"TEXT"	{ Authentication : Bearer Token }	{ status:"Success Error" message:"TEXT" }	-	SELECT * FROM is_user_admin(id_user_param varchar) SELECT * FROM is_tenaga_ahli_exist(id_tenaga_ahli_param varchar) SELECT * FROM edit_tenaga_ahli(id_tenaga_ahli_param			

					alamat_rumah:"TEXT" id_provinsi:"TEXT" id_kabupaten_kota:"TEXT" no_telepon_rumah:"TEXT" no_handphone:"TEXT"				varchar, nama_param varchar, tempat_lahir_param varchar, tanggal_lahir_param timestamp, id_jenis_kelamin_param int, nik_param varchar, email_param varchar, alamat_rumah_param text, id_provinsi_param int, id_kabupaten_kota_param int, no_telepon_param varchar, no_handphone_param varchar)			
Delete Tenaga Ahli	/id_tenaga_ahli}	REST	DELETE			{ Authentication : Bearer Token }	{ status:"Success Error" message:"TEXT" }	-	SELECT * FROM is_user_admin(id_user_param varchar) SELECT * FROM is_tenaga_ahli_exist(id_tenaga_ahli_param varchar) SELECT * FROM delete_tenaga_ahli(id_tenaga_ahli_param varchar)			

View Tenaga Ahli Keseluruhan	/	REST	GET			{ Authentication : Bearer Token }	{ status:"Success Error" message:"TEXT" data:[{object}] }	{ id_tenaga_ahli:"TEXT" nama_lengkap:"TEXT" tempat_lahir:"TEXT" tanggal_lahir:"TEXT" jenis_kelamin:"TEXT" nik:"TEXT" email:"TEXT" alamat_rumah:"TEXT" provinsi:"TEXT" kabupaten_kota:"TEXT" no_telepon_rumah:"TEXT" no_handphone:"TEXT" }	SELECT * FROM get_tenaga_ahli() SELECT * FROM is_user_admin(id_use r_param varchar)			
View Tenaga Ahli Daerah	/{{id_provinsi}}	REST	GET			{ Authentication : Bearer Token }	{ status:"Success Error" message:"TEXT" data:[{object}] }	{ id_tenaga_ahli:"TEXT" nama_lengkap:"TEXT" tempat_lahir:"TEXT" tanggal_lahir:"TEXT" jenis_kelamin:"TEXT" nik:"TEXT" email:"TEXT" alamat_rumah:"TEXT" provinsi:"TEXT" kabupaten_kota:"TEXT" no_telepon_rumah:"TEXT" no_handphone:"TEXT" }	SELECT * FROM get_tenaga_ahli_per_ daerah(id_provinsi_p aram varchar) SELECT * FROM is_user_admin(id_use r_param varchar)			
View Detail Tenaga Ahli	/{{id_tenaga_ahli}}	REST	GET			{ Authentication : Bearer }	{ status:"Success Error" message:"TEXT" }	{ id_tenaga_ahli:"TEXT" nama_lengkap:"TEXT" }	SELECT * FROM get_detail_tenaga_ahli(id_tenaga_ahli_para			

						Token }	data:{ object } }	tempat_lahir:"TEXT" tanggal_lahir:"TEXT" jenis_kelamin:"TEXT" nik:"TEXT" email:"TEXT" alamat_rumah:"TEXT" provinsi:"TEXT" kabupaten_kota:"TEXT" no_telepon_rumah:"TEXT" " no_handphone:"TEXT" }	m varchar) SELECT * FROM is_user_admin(id_use r_param varchar)			
Kegiatan Management	{{ API_URL }}/kegiatan											
Add Kegiatan	/	REST	POST		{ judul_kegiatan:"TEX T" deskripsi_kegiatan:" TEXT" tanggal_kegiatan:"D ATE" lokasi_kegiatan:"TE XT" latitude_lokasi:"DOU BLE" longitude_lokasi:"D OUBLE" status_kegiatan:"TE XT" id_pelatih_kegiatan:[ "INT"] foto_banner_kegiatan	{ Authenticatio n : Bearer Token }	{ status:"Success Error" message:"TEXT" }	-	SELECT * FROM is_kegiatan_exist(id_ jadwal_kegiatan_para m varchar) SELECT * FROM is_user_admin(id_use r_param varchar) SELECT * FROM add_jadwal_kegiatan (id_kegiatan_param character varying, judul_kegiatan_para m character varying, deskripsi_kegiatan_p aram character varying, tanggal_kegiatan_par			

					:"FILE" }				am timestamp without time zone, lokasi_kegiatan_param character varying, latitude_lokasi_param numeric, longitude_lokasi_param numeric, foto_banner_kegiatan_param text, id_user_param character varying)  SELECT * FROM add_pelatih_kegiatan (id_kegiatan_param bigint, id_tenaga_ahli_param varchar)			
Edit Kegiatan	/id_kegiatan}	REST	PUT		{ judul_kegiatan:"TEXT" deskripsi_kegiatan:"TEXT" tanggal_kegiatan:"DATE" lokasi_kegiatan:"TEXT" latitude_lokasi:"DOUBLE" longitude_lokasi:"DOUBLE"	{ Authentication : Bearer Token }	{ status:"Success Error" message:"TEXT" }	-	SELECT * FROM is_kegiatan_exist(id_jadwal_kegiatan_param varchar)  SELECT * FROM edit_jadwal_kegiatan (id_kegiatan_param character varying, judul_kegiatan_param character varying, deskripsi_kegiatan_p			

					status_kegiatan:"TEXT" id_pelatih_kegiatan:[ "INT"] foto_banner_kegiatan: "FILE" }				aram character varying, tanggal_kegiatan_param timestamp without time zone, lokasi_kegiatan_param character varying, latitude_lokasi_param numeric, longitude_lokasi_param numeric, foto_banner_kegiatan_param text, id_user_param character varying)  SELECT * FROM is_user_admin(id_user_param varchar)			
Delete Kegiatan	/id_kegiatan}	REST	DELETE			{ Authentication : Bearer Token }	{ status:"Success Error" message:"TEXT" }	-	SELECT * FROM is_kegiatan_exist(id_jadwal_kegiatan_param varchar)  SELECT * FROM delete_jadwal_kegiatan(id_jadwal_kegiatan_param varchar)  SELECT * FROM is_user_admin(id_user_param varchar)			

View Kegiatan	/	REST	GET			{ Authentication : Bearer Token }	{ status:"Success Error" message:"TEXT" data:[{object}] }	{ id_kegiatan:"TEXT" judul_kegiatan:"TEXT" deskripsi_kegiatan:"TEXT" " tanggal_kegiatan:"DATE" lokasi_kegiatan:"TEXT" latitude_lokasi:"DOUBLE" " longitude_lokasi:"DOUBLE" " status_kegiatan:"TEXT" pelatih_kegiatan:["TEXT"] foto_banner_kegiatan:"TEXT" } SELECT * FROM get_jadwal_kegiatan( )			
View Kegiatan Perbulan	/	REST	GET			{ Authentication : Bearer Token }	{ status:"Success Error" message:"TEXT" data:[{object}] }	{ id_kegiatan:"TEXT" judul_kegiatan:"TEXT" deskripsi_kegiatan:"TEXT" " tanggal_kegiatan:"DATE" lokasi_kegiatan:"TEXT" latitude_lokasi:"DOUBLE" " longitude_lokasi:"DOUBLE" " status_kegiatan:"TEXT" pelatih_kegiatan:["TEXT"] foto_banner_kegiatan:"TEXT" } SELECT * FROM get_jadwal_kegiatan_ per_bulan(bulan_tahun _param timestamp)			

View Detail Kegiatan	/{{id_kegiatan}}	REST	GET			{ Authentication : Bearer Token }	{ status:"Success Error" message:"TEXT" data:[{object}] }	{ id_kegiatan:"TEXT" judul_kegiatan:"TEXT" deskripsi_kegiatan:"TEXT" " tanggal_kegiatan:"DATE" lokasi_kegiatan:"TEXT" latitude_lokasi:"DOUBLE" " longitude_lokasi:"DOUBLE" " status_kegiatan:"TEXT" pelatih_kegiatan:["TEXT"] foto_banner_kegiatan:"TEXT" }	SELECT * FROM get_detail_jadwal_kegiatan(id_jadwal_kegiatan_param varchar)			
Daftar Kegiatan	/daftar/	REST	POST		{ id_kegiatan:"TEXT" }	{ Authentication : Bearer Token }	{ status:"Success Error" message:"TEXT" }		SELECT * FROM is_kegiatan_exist(id_jadwal_kegiatan_param varchar)  SELECT * FROM register_kegiatan(id_user_param varchar, id_jadwal_kegiatan_param varchar)			
Administrasi Kegiatan Management	{{API_URL}}/administrasi-kegiatan											
Add	/	REST	POST		{	{	{	-				



Administrasi Kegiatan					judul_administrasi_kegiatan:"TEXT" deskripsi_administrasi_kegiatan:"TEXT" file_administrasi_kegiatan:"FILE" }	Authentication : Bearer Token }	status:"Success Error" message:"TEXT" }		SELECT * FROM is_user_admin(id_user_param varchar)  SELECT * FROM add_administrasi_kegiatan(judul_administrasi_kegiatan_param varchar, deskripsi_administrasi_kegiatan_param varchar, file_standarisasi_param text, id_user_param varchar)			
Edit Administrasi Kegiatan	/[{id_administrasi_kegiatan}]	REST	PUT		{ judul_administrasi_kegiatan:"TEXT" deskripsi_administrasi_kegiatan:"TEXT" file_administrasi_kegiatan:"FILE" }	{ Authentication : Bearer Token }	{ status:"Success Error" message:"TEXT" }	-	SELECT * FROM is_user_admin(id_user_param varchar)  SELECT * FROM is_administrasi_kegiatan_exists(id_administrasi_kegiatan_param bigint)  SELECT * FROM edit_administrasi_kegiatan(id_administrasi_kegiatan_param bigint,			

									judul_administrasi_kegiatan_param varchar, deskripsi_administrasi_kegiatan_param varchar, file_standarisasi_param text, id_user_param varchar)			
Delete Administrasi Kegiatan	/id_administrasi_kegiatan}	REST	DELETE			{ Authentication : Bearer Token }	{ status:"Success Error" message:"TEXT" }	-	SELECT * FROM is_user_admin(id_user_param varchar) SELECT * FROM is_administrasi_kegiatan_exist(id_administrasi_kegiatan_param bigint)  SELECT * FROM delete_administrasi_kegiatan(id_administrasi_kegiatan_param bigint)			
View Administrasi Kegiatan	/	REST	GET			{ Authentication : Bearer Token }	{ status:"Success Error" message:"TEXT" data:[{object}] }	{ id_administrasi_kegiatan:"TEXT" judul_administrasi_kegiatan:"TEXT" deskripsi_administrasi_kegiatan:"TEXT"	SELECT * FROM get_administrasi_kegiatan()			

								file_administrasi_kegiatan: "TEXT" pengirim:"TEXT" }				
View Detail Administrasi Kegiatan	/id_administrasi_kegiatan}	REST	GET			{ Authentication : Bearer Token }	{ status:"Success Error" message:"TEXT" data:{object} }	{ id_administrasi_kegiatan:"TEXT" judul_administrasi_kegiatan:"TEXT" deskripsi_administrasi_kegiatan:"TEXT" file_administrasi_kegiatan: "TEXT" pengirim:"TEXT" }	SELECT * FROM get_detail_administrasi_kegiatan(id_administrasi_kegiatan_param bigint)			
Buku Saku Management	{{API_URL}}/buku-saku											
Add Buku Saku	/	REST	POST		{ judul_buku_saku:"TEXT" deskripsi_buku_saku:"TEXT" file_buku_saku:"TEXT" }	{ Authentication : Bearer Token }	{ status:"Success Error" message:"TEXT" }	-	SELECT * FROM is_user_admin(id_user_param varchar) SELECT * FROM add_buku_saku(judul_buku_saku_param varchar, deskripsi_buku_saku_param varchar, file_buku_saku_param text, id_user_param			

									varchar)			
Edit Buku Saku	/id_buku_saku}	REST	PUT		{ judul_buku_saku:"TEXT" deskripsi_buku_saku:"TEXT" file_buku_saku:"TEXT" }	{ Authentication : Bearer Token }	{ status:"Success Error" message:"TEXT" }	-	SELECT * FROM is_user_admin(id_user_param varchar)  SELECT * FROM is_buku_saku_exist(id_buku_saku_param bigint)  SELECT * FROM edit_buku_saku(id_buku_saku_param bigint, judul_buku_saku_param varchar, deskripsi_buku_saku_param varchar, file_buku_saku_param text, id_user_param varchar)			
Delete Buku Saku	/id_buku_saku}	REST	DELETE			{ Authentication : Bearer Token }	{ status:"Success Error" message:"TEXT" }	-	SELECT * FROM is_user_admin(id_user_param varchar) SELECT * FROM is_buku_saku_exist(id_buku_saku_param bigint)			

									SELECT * FROM delete_buku_saku(id_buku_saku_param bigint)			
View Buku Saku	/	REST	GET			{ Authentication : Bearer Token }	{ status:"Success Error" message:"TEXT" data:[{object}] }	{ id_buku_saku:"TEXT" judul_buku_saku:"TEXT" deskripsi_buku_saku:"TEXT" file_buku_saku:"TEXT" pengirim:"TEXT" }	SELECT * FROM get_buku_saku()			
View Detail Buku Saku	/id_buku_saku}	REST	GET			{ Authentication : Bearer Token }	{ status:"Success Error" message:"TEXT" data:{object} }	{ id_buku_saku:"TEXT" judul_buku_saku:"TEXT" deskripsi_buku_saku:"TEXT" file_buku_saku:"TEXT" pengirim:"TEXT" }	SELECT * FROM get_detail_buku_saku(id_buku_saku_param bigint)			
SKA Management	{{API_URL}}/ska											
Add SKA	/	REST	POST		{ judul_ska:"TEXT" deskripsi_ska:"TEXT" id_kategori_ska:"INT" file_ska:"FILE" }	{ Authentication : Bearer Token }	{ status:"Success Error" message:"TEXT" }	-	SELECT * FROM add_ska(judul_ska_param varchar, deskripsi_ska_param varchar, id_kategori_ska_param int, file_ska_param text,			

									id_user_param varchar)  SELECT * FROM is_user_admin(id_use r_param varchar)			
Edit SKA	/{{id_ska}}	REST	PUT		{ judul_ska:"TEXT" deskripsi_ska:"TEXT" " id_kategori_ska:"INT" " file_ska:"FILE" }	{ Authenticatio n : Bearer Token }	{ status:"Success Error" message:"TEXT" }	-	SELECT * FROM is_ska_exist(id_ska_ param bigint)  SELECT * FROM edit_ska(id_ska_para m bigint, judul_ska_param varchar, deskripsi_ska_param varchar, id_kategori_ska_para m int, file_ska_param text, id_user_param varchar)  SELECT * FROM is_user_admin(id_use r_param varchar)			
Delete SKA	/{{id_ska}}	REST	DELETE			{ Authenticatio n : Bearer Token }	{ status:"Success Error" message:"TEXT" }	-	SELECT * FROM is_ska_exist(id_ska_ param bigint)  SELECT * FROM			

									delete_ska(id_ska_param bigint)  SELECT * FROM is_user_admin(id_user_param varchar)			
View SKA	/	REST	GET			{ Authentication : Bearer Token }	{ status:"Success Error" message:"TEXT" data:[{object}] }	{ id_ska:"TEXT" judul_ska:"TEXT" deskripsi_ska:"TEXT" kategori_ska:"TEXT" file_ska:"TEXT" pengirim:"TEXT" }	SELECT * FROM SELECT * FROM get_ska()			
View Detail SKA	/id_ska}	REST	GET			{ Authentication : Bearer Token }	{ status:"Success Error" message:"TEXT" data:{object} }	{ id_ska:"TEXT" judul_ska:"TEXT" deskripsi_ska:"TEXT" kategori_ska:"TEXT" file_ska:"TEXT" pengirim:"TEXT" }	SELECT * FROM get_detail_ska(id_ska_param bigint)			
Modul Management	{{ API_URL }}/modul											
Add Modul	/	REST	POST		{ judul_modul:"TEXT" deskripsi_modul:"TEXT" id_kategori_modul:"INT" file_modul:"FILE"	{ Authentication : Bearer Token }	{ status:"Success Error" message:"TEXT" }	-	SELECT * FROM add_modul(judul_modul_param varchar, deskripsi_modul_param varchar, id_kategori_modul_param int,			

					}				file_modul_param text, id_user_param varchar)  SELECT * FROM is_user_admin(id_use r_param varchar)			
Edit Modul	/{{id_modul}}	REST	PUT		{ judul_modul:"TEXT" deskripsi_modul:"TE XT" id_kategori_modul:"I NT" file_modul:"FILE" }	{ Authenticatio n : Bearer Token }	{ status:"Success Error" message:"TEXT" }	-	SELECT * FROM is_modul_exist(id_m odul_param bigint)  SELECT * FROM edit_modul(id_modul _param bigint, judul_modul_param varchar, deskripsi_modul_par am varchar, id_kategori_modul_p aram int, file_modul_param text, id_user_param varchar)  SELECT * FROM is_user_admin(id_use r_param varchar)			
Delete Modul	/{{id_modul}}	REST	DELETE			{ Authenticatio n : Bearer Token }	{ status:"Success Error" message:"TEXT" }	-	SELECT * FROM is_modul_exist(id_m odul_param bigint)			



									SELECT * FROM delete_modul(id_modul_param bigint)  SELECT * FROM is_user_admin(id_user_param varchar)			
View Modul	/	REST	GET			{ Authentication : Bearer Token }	{ status:"Success Error" message:"TEXT" data:[{object}] }	{ id_modul:"TEXT" judul_modul:"TEXT" deskripsi_modul:"TEXT" kategori_modul:"TEXT" file_modul:"TEXT" pengirim:"TEXT" }	SELECT * FROM get_modul()			
View Detail Modul	/id_modul	REST	GET			{ Authentication : Bearer Token }	{ status:"Success Error" message:"TEXT" data:{object} }	{ id_modul:"TEXT" judul_modul:"TEXT" deskripsi_modul:"TEXT" kategori_modul:"TEXT" file_modul:"TEXT" pengirim:"TEXT" }	SELECT * FROM get_detail_modul(id_modul_param bigint)			
Common	{{API_URL}}/common											
Jenis Kelamin	/jenis-kelamin	REST	GET			{ Authentication : Bearer Token }	{ status:"Success Error" message:"TEXT" data:[{object}] }	{ id_jenis_kelamin:"INT" jenis_kelamin:"TEXT" }	SELECT * FROM jenis_kelamin			

Status Perkawinan	/status-perkawinan	REST	GET			{ Authenticatio n : Bearer Token }	{ status:"Success Error" message:"TEXT" data:[{object}] }	{ id_status_perkawinan:"INT" status_perkawinan:"TEXT" }		SELECT * FROM status_pe rkawinan		
Status Rumah	/status-rumah	REST	GET			{ Authenticatio n : Bearer Token }	{ status:"Success Error" message:"TEXT" data:[{object}] }	{ id_status_rumah:"INT" status_rumah:"TEXT" }		SELECT * FROM status_ru mah		
Pendidikan	/pendidikan	REST	GET			{ Authenticatio n : Bearer Token }	{ status:"Success Error" message:"TEXT" data:[{object}] }	{ id_pendidikan:"INT" pendidikan:"TEXT" }		SELECT * FROM pendidik an		
Jabker	/jabker	REST	GET			{ Authenticatio n : Bearer Token }	{ status:"Success Error" message:"TEXT" data:[{object}] }	{ id_jabker:"INT" jabker:"TEXT" }		SELECT * FROM jabker		
Provinsi	/provinsi	REST	GET			{ Authenticatio n : Bearer Token }	{ status:"Success Error" message:"TEXT" data:[{object}] }	{ id_provinsi:"INT" provinsi:"TEXT" }		SELECT * FROM provinsi		
Kabupaten Kota	/kabupaten-kota	REST	GET			{ Authenticatio n : Bearer Token }	{ status:"Success Error" message:"TEXT" data:[{object}] }	{ id_kabupaten_kota:"INT" kabupaten_kota:"TEXT" }	SELECT * FROM get_kabupaten_kota(i d_provinsi_param int)			

Kecamatan	/kecamatan	REST	GET			{ Authentication : Bearer Token }	{ status:"Success Error" message:"TEXT" data:[{object}] }	{ id_kecamatan:"INT" kecamatan:"TEXT" }	SELECT * FROM get_kecamatan(id_kabupaten_kota_param int)			
Kelurahan Desa	/kelurahan- desa	REST	GET			{ Authentication : Bearer Token }	{ status:"Success Error" message:"TEXT" data:[{object}] }	{ id_kelurahan_desa:"INT" kelurahan_desa:"TEXT" }	SELECT * FROM get_kelurahan_desa(id_kecamatan_param int)			
Kategori SKKNI SKA	/skkni-ska	REST	GET			{ Authentication : Bearer Token }	{ status:"Success Error" message:"TEXT" data:[{object}] }	{ id_kategori_skkni:"INT" kategori_skkni:"TEXT" }		SELECT * FROM kategori_ skkni		
Kategori Modul	/modul	REST	GET			{ Authentication : Bearer Token }	{ status:"Success Error" message:"TEXT" data:[{object}] }	{ id_kategori_modul:"INT" kategori_modul:"TEXT" }		SELECT * FROM kategori_ modul		