

## BAB III METODOLOGI

### 3.1. Lokasi Penelitian

Penelitian mengenai perancangan aplikasi Sistem Pencatatan kehadiran mahasiswa berbasis *Android* yang menerapkan interaksi manusia komputer ini dilakukan di lingkungan Universitas Siliwangi. Fokus pembuatan aplikasi dilakukan terhadap mahasiswa dan dosen ketika melakukan kegiatan belajar mengajar tatap muka di kelas. Pengambilan data dan materi didasarkan pada pengalaman penulis selama menempuh jenjang pendidikan.

### 3.2. Bahan Dan Alat Penelitian

Dapat menjalankan program aplikasi yang penulis buat memerlukan perangkat keras dan perangkat lunak. Adapun bahan dan alat yang akan digunakan penulis dalam penelitian ini adalah:

#### 3.2.1. Bahan Penelitian

Bahan-bahan penelitian yang akan digunakan dalam penelitian melalui buku referensi *web* dan *android*.

#### 3.2.2. Alat Penelitian

Alat-alat penelitian yang akan digunakan dalam penelitian ini ialah sebagai berikut:

Tabel 3.1 Perangkat Keras dengan spesifikasi:

Proseso r	Intel Celeron 4205U dual-core 1,8GHz
--------------	--------------------------------------

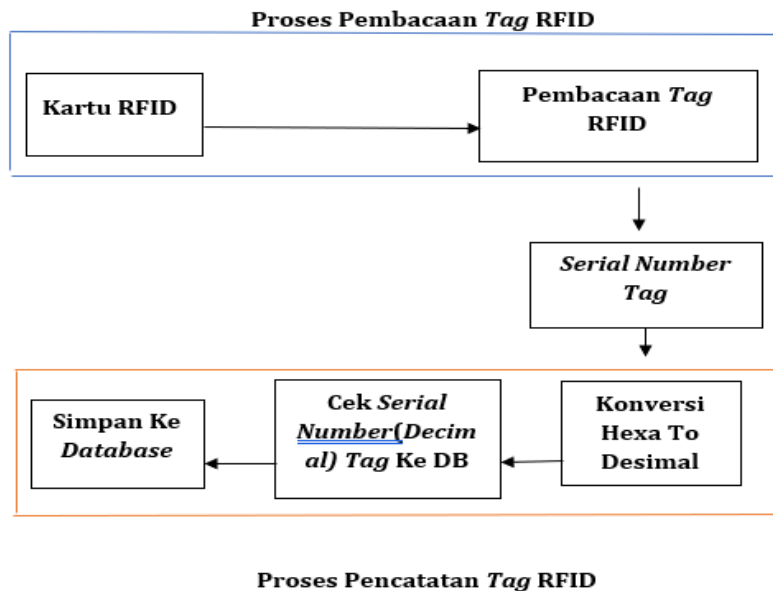
RAM	4 GB DDR4 2133MHz, upgradeable
Grafis Tipe	Intel UHD Graphics 610
Layar	14.0-inch HD SVA BrightView micro-edge WLED-backlit
HDD	512 GB SSD
RFID <i>Reader</i>	RC10C Mifare 13.56MHz – USB – <i>Plug &amp; Play</i>
RFID <i>Tag</i>	Kartu RFID PROXIMITY 125KHZ <i>CARD</i>

Tabel 3.2 Perangkat Lunak dengan spesifikasi:

Windows 10	64 bit
Android Studio	Versi 4.1
Mysql Server	Versi 8.0.12

### 3.3. Proses Pembacaan dan Pencatatan *Tag* RFID

Proses pembacaan dan pencatatan digunakan dalam menyusun skripsi ini terdiri dari langkah-langkah berikut dapat dilihat pada gambar 3.1.



Gambar 3.1 Proses Pembacaan dan Pencatatan *Tag* RFID  
(Ilman Hilmi Oriza, 2018)

Gambar 3.1 proses pertama dilakukan pembacaan *Tag* RFID dengan cara *Tag* RFID ditempelkan pada *Reader* untuk dilakukan pembacaan *Tag* RFID, hasil dari pembacaan akan didapatkan *Serial Number Tag* RFID. Proses kedua dilakukan pencatatan *Tag* RFID, sebelum dilakukan pencatatan *Serial Number* yang dihasilkan dari *Reader* dilakukan konversi *Hexa To Desimal* karena *Serial Tag* RFID harus berbentuk *Desimal*, karena pada umumnya *Tag* RFID berbentuk *Desimal*, kemudian Langkah selanjutnya dilakukan pengecekan *Serial Number Tag* ke *Database*, kemudian dilakukan penyimpanan ke *Database*.

### 3.4. Teknik Pengumpulan Data

Dalam mendapatkan informasi, data-data penunjang serta teori dalam penyusunan proposal skripsi ini, maka diperlukan teknik pengumpulan data.

Adapun teknik yang digunakan antara lain:

1. Pengamatan (Observasi)

Merupakan metode pengumpulan data yang diperoleh dengan melakukan pengamatan dan pencatatan data kelas sekitaran Universitas siliwangi.

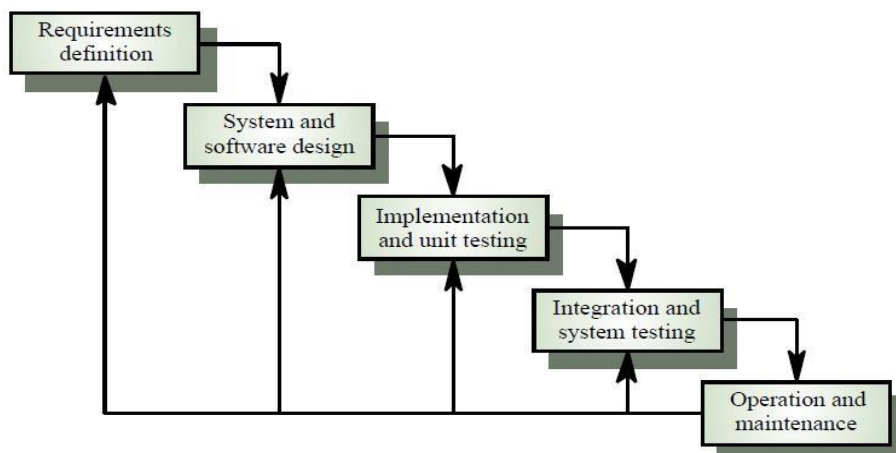
2. Studi Pustaka (Literatur)

Merupakan metode pencarian data buku, browsing internet atau literatur-literatur yang berkaitan dengan data yang dibutuhkan dalam penelitian dan pembuatan aplikasi berbasis Android.

### 3.5. Teknik Pengolahan dan Analisis Data

Penelitian ini, metode perencanaan aplikasi yang digunakan adalah waterfall yang merupakan salah satu metode dalam *System Development Live Cycle* (SDLC) yang mempunyai ciri khas pengerjaan setiap fase dalam *waterfall* harus diselesaikan terlebih dahulu sebelum melanjutkan ke fase selanjutnya. Metode *waterfall* adalah pengerjaan dari suatu sistem dilakukan secara berurutan atau secara *linear*.

Secara garis besar metode *waterfall* mempunyai langkah-langkah sebagai berikut:



Gambar 3.2 Model *Waterfall*

Tahapan-tahapan dari metode *waterfall* adalah sebagai berikut:

- a. *Requirements Definition*, seluruh kebutuhan *software* harus bisa didapatkan dalam fase ini, termasuk didalamnya kegunaan *software* yang diharapkan pengguna dan batasan *software*.
- b. *Sistem & Software Design*, tahap ini dilakukan sebelum melakukan *coding*. tahap ini bertujuan untuk memberikan gambaran apa yang seharusnya dikerjakan dan bagaimana tampilannya.
- c. *Implementation & Unit Testing*, dalam tahap ini dilakukan pemrograman. Pembuatan *software* dipecah menjadi modul-modul kecil yang nantinya akan digabungkan dalam tahap berikutnya.

d. *Integration & Sistem Testing*, di tahap ini dilakukan penggabungan modul-modul yang sudah dibuat dan dilakukan pengujian ini untuk mengetahui apakah *software* yang dibuat telah sesuai dengan desainnya dan masih terdapat kesalahan atau tidak.

e. *Operation & Maintenance*, ini merupakan tahap terakhir dalam model *waterfall*. *Software* yang sudah jadi dijalankan serta dilakukan pemeliharaan. Pemeliharaan termasuk dalam memperbaiki kesalahan yang tidak ditemukan pada langkah sebelumnya.

### **3.6. Teknik Pengujian Sistem**

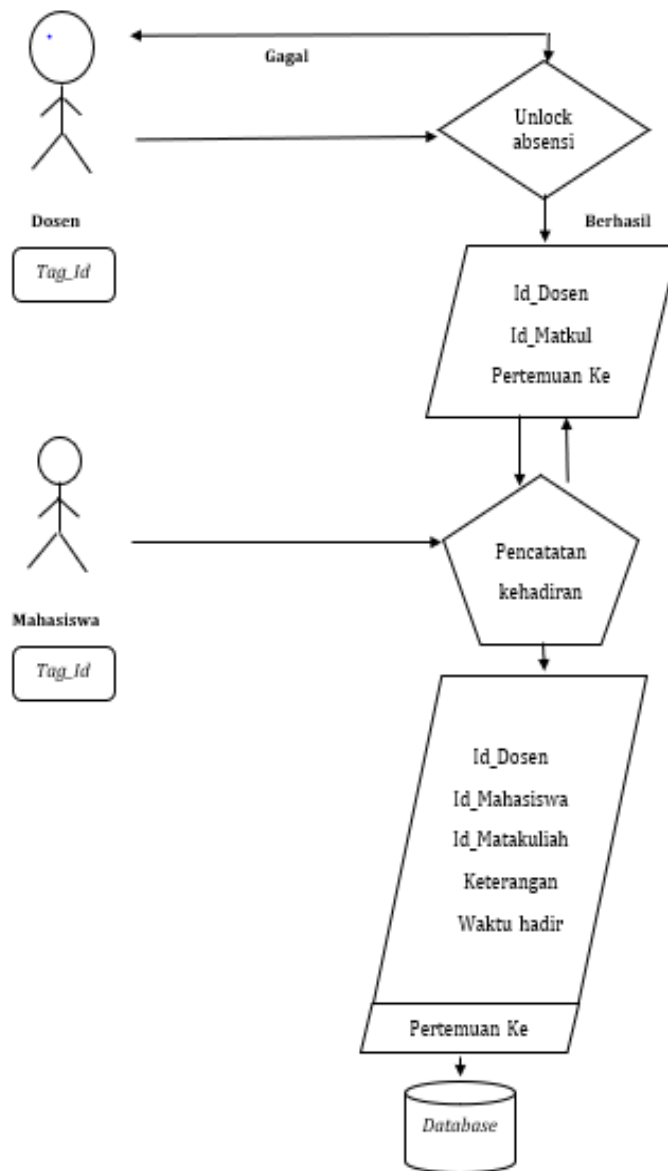
Pengujian sistem yang digunakan pada penelitian ini adalah metode pengujian langsung yaitu dengan menggunakan pengujian *BlackBox*. *Blackbox testing* yaitu menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian dimaksud untuk mengetahui apakah fungsi-fungsi, masukan dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan.

### **3.7. Perancangan Aplikasi**

Perancangan sistem dan proses dimulai dengan perancangan *Use Case Diagram*, rancangan arsitektur aplikasi, dan rancangan *interface* Aplikasi. Perancangan dan desain aplikasi dimulai dengan perancangan detail dari halaman yang akan ditampilkan dalam aplikasi.

#### **1. Use Case Diagram**

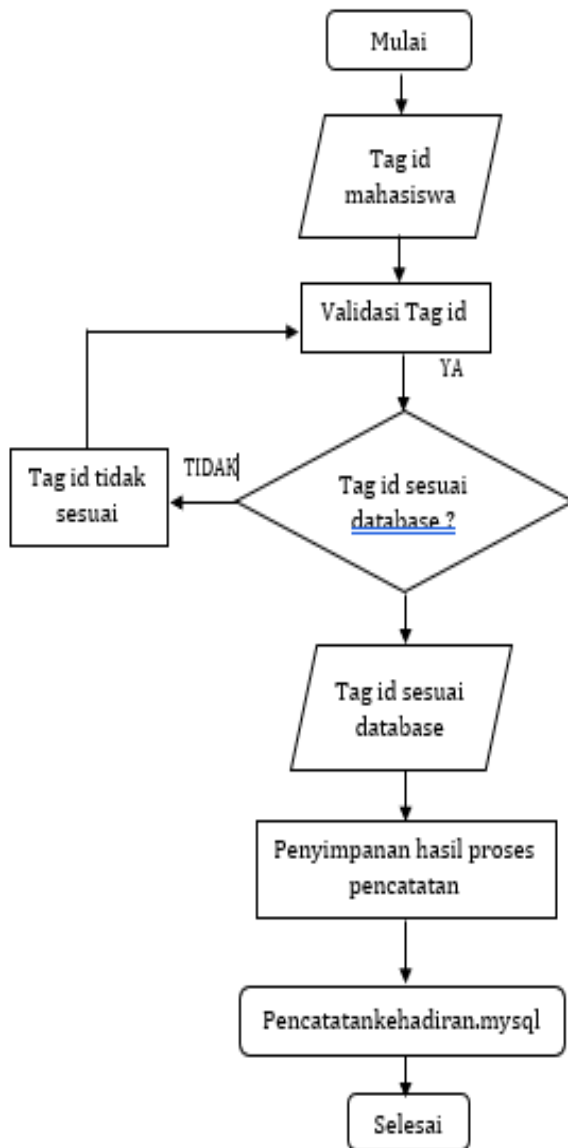
*Use case diagram* menjelaskan alur dosen dan mahasiswa untuk pengujian dengan diawali dosen meng-*unlock* sampai mahasiswa melakukan pencatatan dan di simpan ke *database* bisa dilihat pada gambar 3.3



Gambar 3.3 Use Case Diagram

## 2. Flowchart

*Flowchart* adalah suatu bagan dengan simbol-simbol tertentu yang menggambarkan urutan proses secara mendetail dan hubungan antara suatu proses (intruksi) dengan proses lainnya dalam suatu sistem.

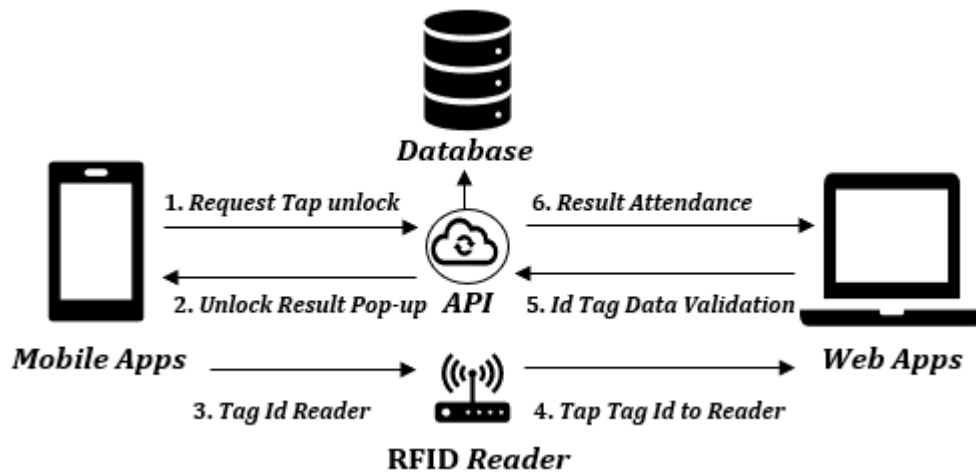


Gambar 3.4 Flowchart

### 3.8. Sistem Integrasi Pencatatan Kehadiran

Sistem integrasi pencatatan kehadiran adalah arsitektur sistem yang mengintegrasikan *Mobile Apps*, *RFID Reader*, *Web Apps*, *Database* menggunakan teknologi API (*Application Programming Interface*).

### 3.8.1. Arsitektur Sistem



Gambar 3.5 Arsitektur Sistem

(Ilman hilmi oriza, 2018)

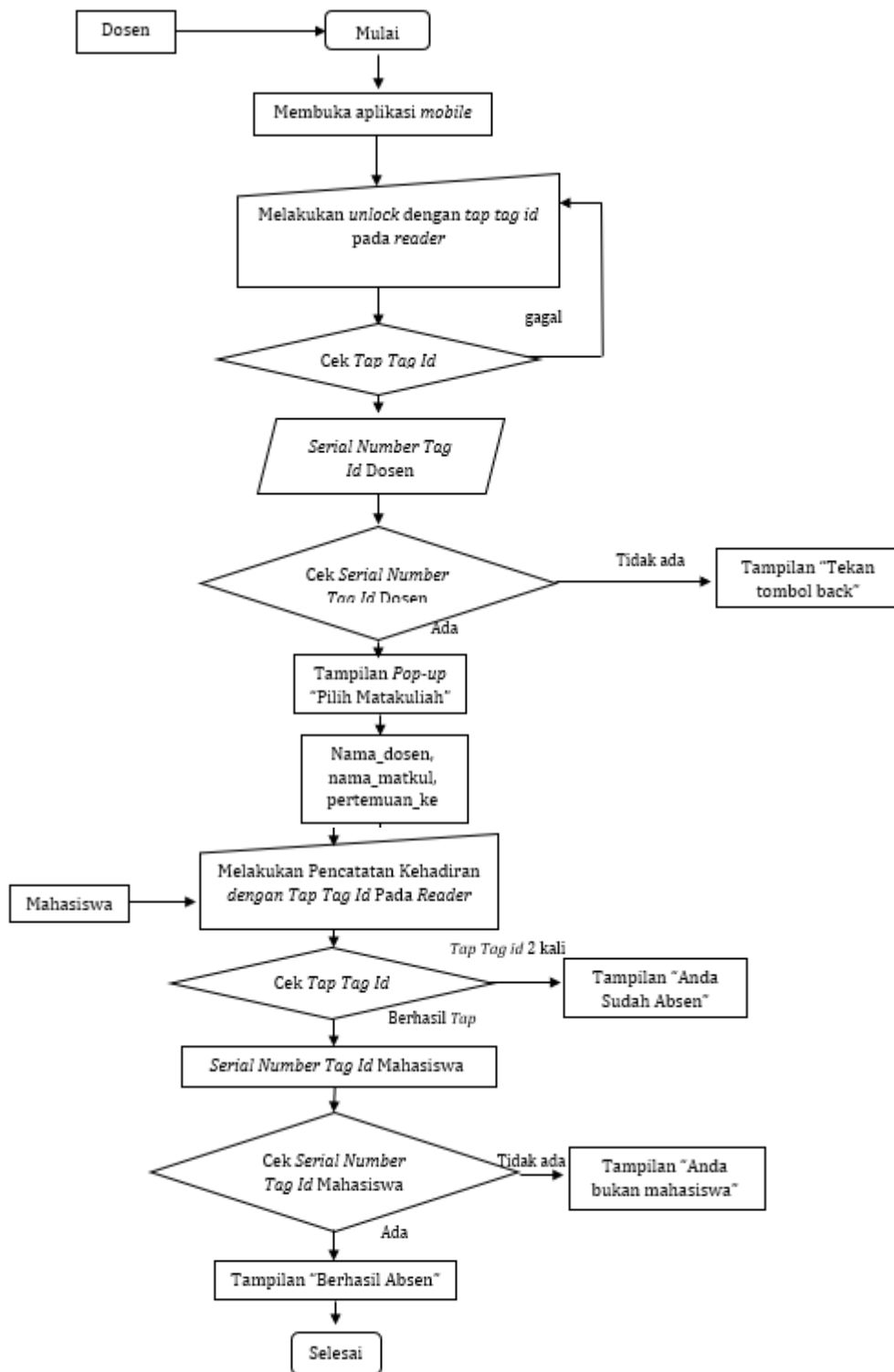
Gambar 3.5 merupakan arsitektur sistem integrasi pencatatan kehadiran RFID, untuk menghubungkan teknologi RFID dan melakukan pertukaran data pada aplikasi *mobile* dengan aplikasi *web* digunakan sebuah teknologi *Application Programming Interface (API)*, kemudian API tersebut akan melakukan pengecekan ke *database*. Prosesnya bisa dilihat pada Gambar 3.5.

### 3.8.2. Bagan Alir Sistem

#### A. Bagan Alir Sistem Penggunaan Absensi RFID

Bagan alir sistem yang dirancang dalam proses penggunaan absensi RFID dapat dilihat pada gambar 3.6.



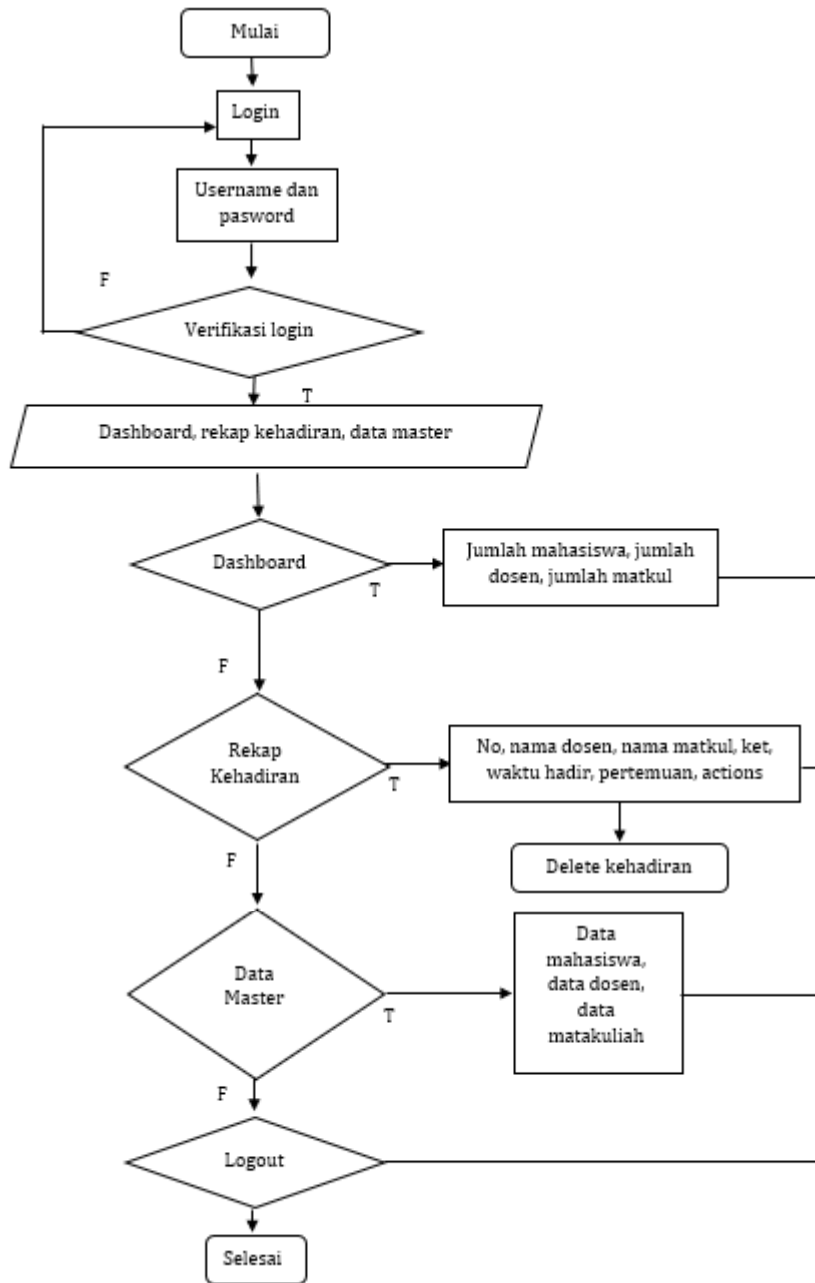


Gambar 3.6 Bagan Alir Sistem Penggunaan Absensi RFID

yang merupakan bagan alir sistem yang berjalan pada aplikasi mobile yang digunakan sebagai proses penggunaan aplikasi Absensi RFID.

#### B. Bagan Alir Sistem Pembacaan Absensi RFID

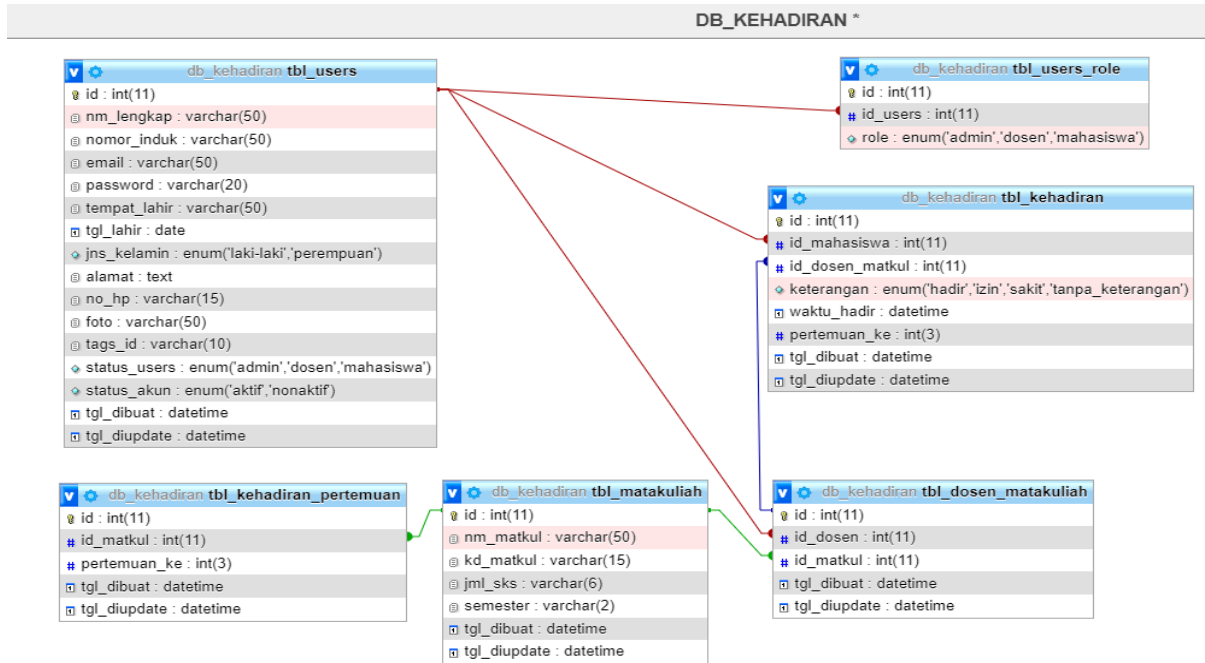
Bagan alir system yang dirancang dalam proses pembacaan Absensi RFID dapat dilihat pada gambar 3.7.



Gambar 3.7 Bagan Alir Sistem Pembacaan Absensi RFID yang merupakan bagan alir sistem yang berjalan pada *web mysql* yang digunakan sebagai proses pembacaan Absensi RFID.

### 3.9. Struktur Tabel

#### a. Relasi Tabel



Gambar 3.8 Relasi Tabel

## b. Struktur Tabel

Tabel 3.1 Users

Tabel Users		
Nama Atribut	Tipe Data	Nilai Data
Id	Integer	11
nama_lengkap	Varchar	50
nomor_induk	Varchar	50
password	Varchar	20
tempat_lahir	Varchar	50
tanggal_lahir	Date	
jenis_kelamin	Enum	"laki-laki", "perempuan"
alamat	Text	
no_hp	Varchar	15
Foto	Varchar	50
tags_id	Varchar	10
status_users	Enum	"mahasiswa", "dosen"
status_akun	Enum	"aktif", "nonaktif"

tgl_di buat	d atetime	"current_timestamp"
tgl_up date	d atetime	"on Update current timestamp"

Tabel 3.2 User Role

<b>Tabel Users Role</b>		
<b>Nama Atribut</b>	<b>Tipe Data</b>	<b>Nilai Data</b>
Id	int	11
id_user	int	11
Role	int	11

Tabel 3.3 Matakuliah

<b>Tabel Matakuliah</b>		
<b>Nama Atribut</b>	<b>Tipe Data</b>	<b>Nilai Data</b>
Id	Int	11
kd_matkul	Varchar	15
nm_matkul	Varchar	50
jml_sks	Varchar	6
Semester	Varchar	2
tgl_dibuat	Datetime	"current_timestam p"
tgl_update	Datetime	"on Update current timestamp"

Tabel 3.4 Pengajar Matakuliah

<b>Tabel Pengajar Matakuliah</b>		
<b>Id</b>	<i>Int</i>	11
<b>id_dosen</b>	<i>Int</i>	11
<b>id_matkul</b>	<i>Int</i>	11
<b>jml_pertemuan</b>	<i>Int</i>	3
<b>tgl_dibuat</b>	<i>Datetime</i>	<i>"current_timestam p"</i>
<b>tgl_update</b>	<i>Datetime</i>	<i>"on Update current timestamp"</i>

Tabel 3.5 Kehadiran

<b>Tabel Kehadiran</b>		
<b>Nama Atribut</b>	<b>T ipe Data</b>	<b>Nilai Data</b>
<b>Id</b>	<i>i nt</i>	11
<b>id_ma hasiswa</b>	<i>i nt</i>	11

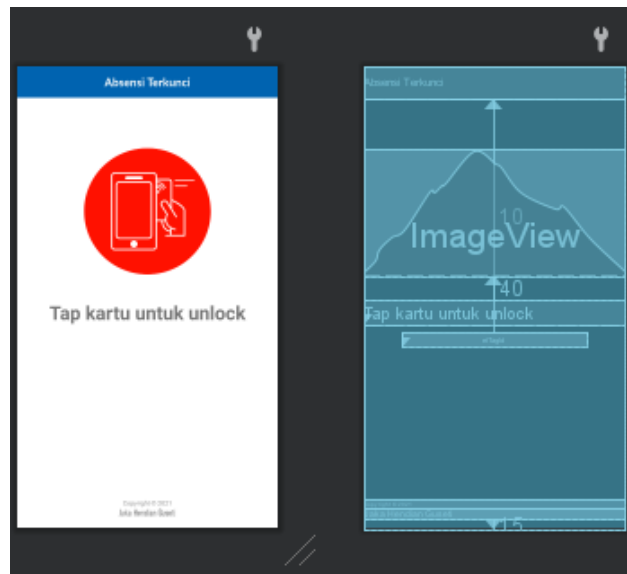
id_pen gajar	<i>i</i> <i>nt</i>	11
Ketera ngan	<i>e</i> <i>num</i>	"hadir","izin","sakit","tanpa_kete rangan"
waktu _hadir	<i>d</i> <i>atetime</i>	
tgl_dib uat	<i>d</i> <i>atetime</i>	"current_timestamp"
tgl_up date	<i>d</i> <i>atetime</i>	"on Update current timestamp"

Tabel 3.6 Kehadiran Pertemuan

<b>Tabel Kehadiran Pertemuan</b>		
<b>Nama</b> <b>Atribut</b>	<b>Ti</b> <b>pe Data</b>	<b>Nilai Data</b>
Id	<i>int</i>	11
id_matk ul	<i>int</i>	11
pertem uan_ke	<i>int</i>	3
tgl_dibu at	<i>da</i> <i>tetime</i>	"current_timestamp"
tgl_diup date	<i>da</i> <i>tetime</i>	"on Update current timestamp"

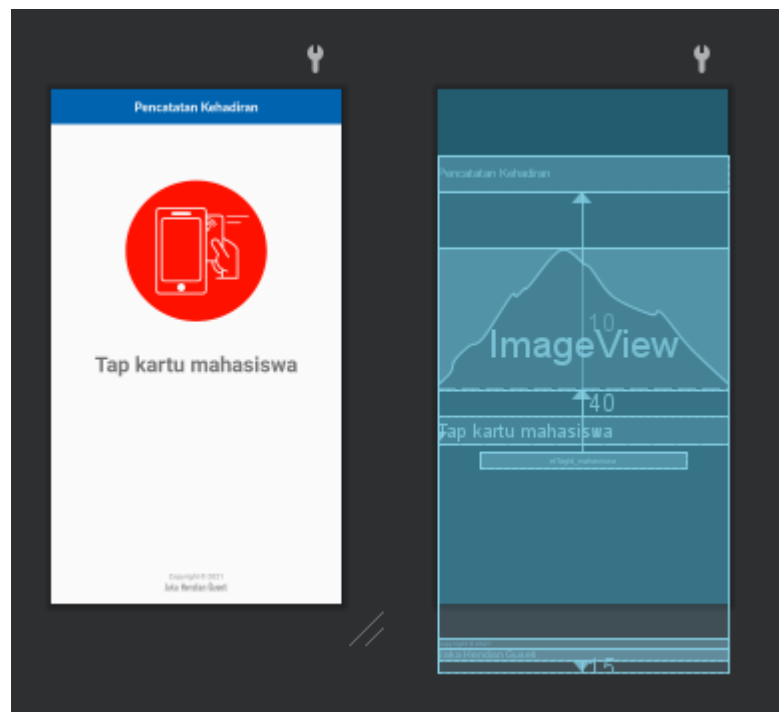
### 3.10. Perancangan *Interface*

Perancangan *interface* ada bagian yang penting dalam aplikasi, karena yang pertama kali dilihat dijalankan adalah tampilan antar muka(*interface*) aplikasi. Bagian bagian yang penting dapat dilihat pada gambar berikut:



Gambar 3.9 Rancangan *Interface* Tap kartu untuk unlock

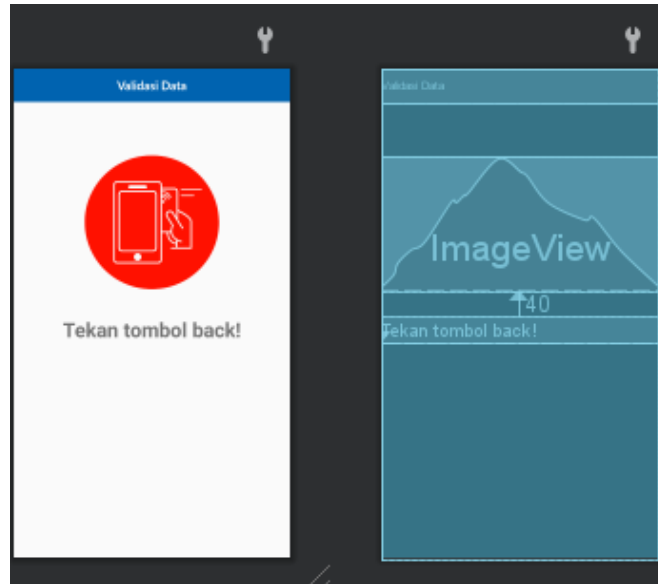
Rancangan *Interface* Tap kartu untuk unlock adalah rancangan *interface* awal dengan gambar dan tulisan “Tap kartu untuk unlock”.



Gambar 3.10 Rancangan *Interface* Tap kartu mahasiswa

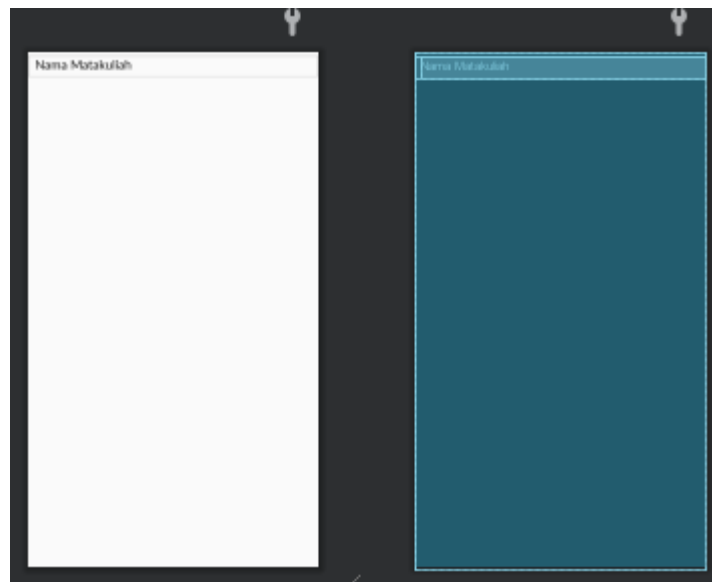
Rancangan *Interface* ini adalah untuk *interface* kedua setelah berhasil tap unlock dengan gambar dan tulisan “Tap kartu mahasiswa”.





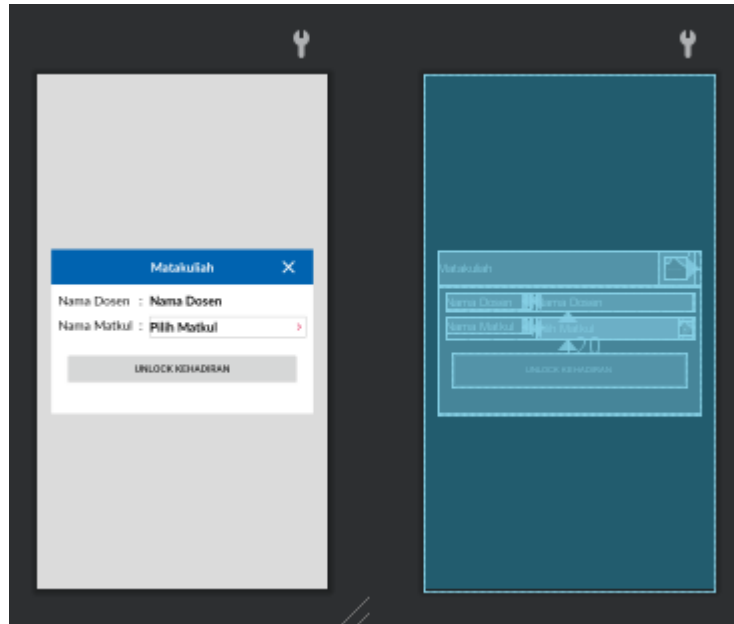
Gambar 3.11 Rancangan Interface Tekan tombol back

Rancangan Interface ini bergambar dan bertuliskan “Tekan tombol back!”.



Gambar 3.12 Rancangan Interface Mocklist Pilih Matkul

Rancangan ini adalah interface sesudah tap kartu untuk unlock.



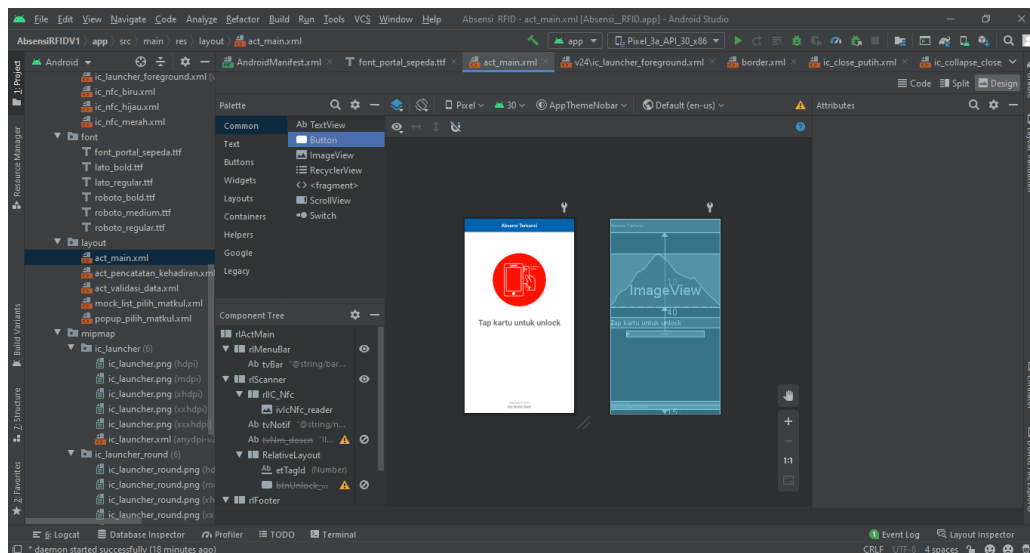
Gambar 3.13 Rancangan Interface Popup pilih matkul

Rancangan ini adalah interface untuk PopUp pilih matkul dengan memunculkan Matakuliah, Nama Dosen, Nama Matkul dan untuk Unlock Kehadiran nya.

### 3.11. Perancangan Pembuatan Aplikasi

#### 1. Membuat Tampilan Tap Kartu Untuk Unlock

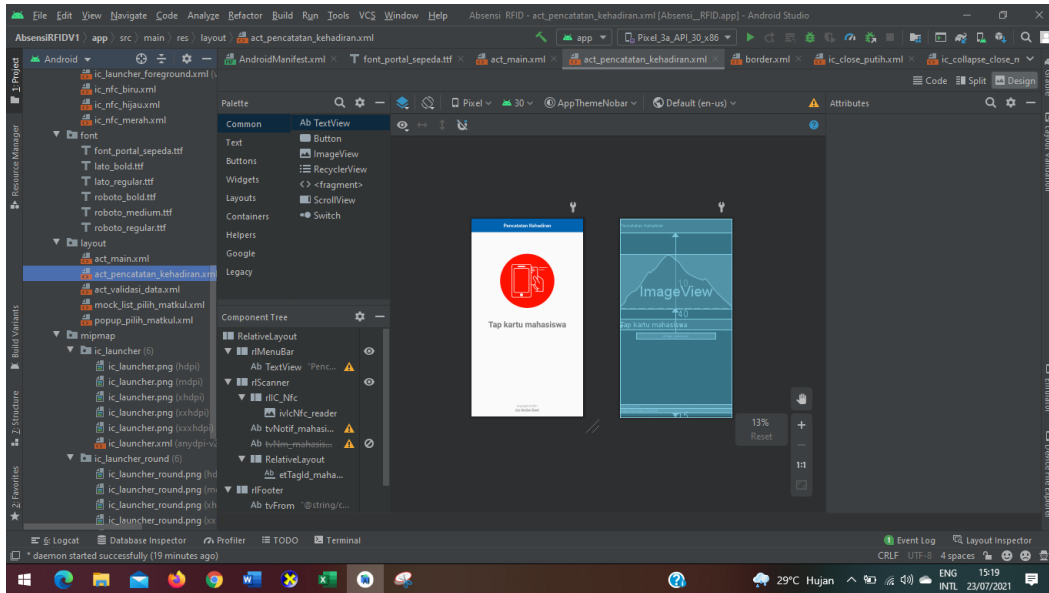
Membuat tampilan ini menggunakan Aplikasi software Android Studio versi 4.1 yang bertujuan untuk mengunlock oleh Dosen.



Gambar 3.14 Pembuatan Tampilan Tap Kartu Untuk Unlock

#### 3. Membuat Tampilan Tap Kartu Mahasiswa

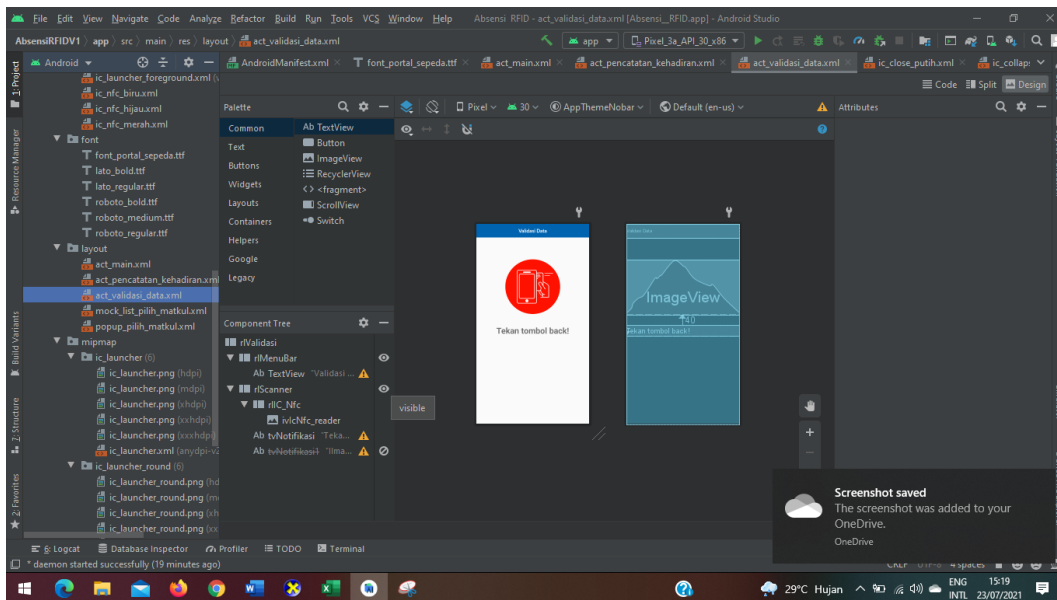
Tampilan Tap Kartu Mahasiswa untuk para mahasiswa untuk men Tap.



Gambar 3.15 Pembuatan Tampilan Tap Kartu Mahasiswa

#### 4. Membuat Tampilan Tekan Tombol Back

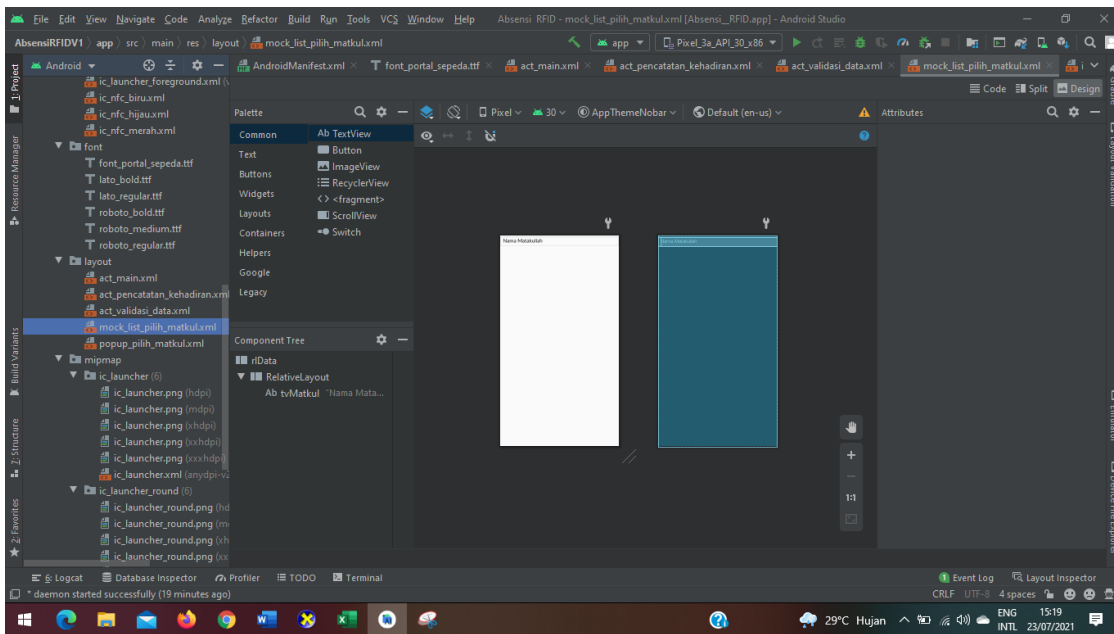
Membuat tampilan tekan tombol *back* untuk terjadi kesalahan jaringan.



Gambar 3.16 Pembuatan Tampilan Tekan Tombol Back

#### 5. Membuat Tampilan *Mocklist* Pilih Matkul

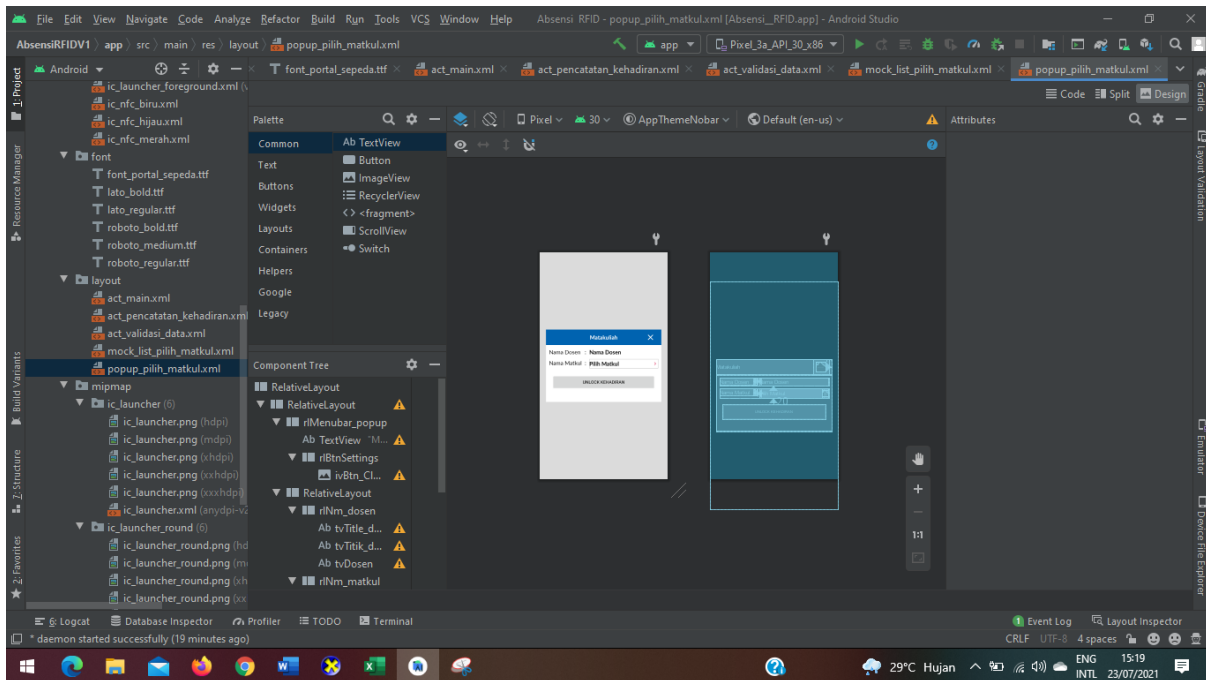
Tampilan ini dirancang untuk *Mocloist* Pilih Matkul *loading* saat *tap* kartu.



Gambar 3.17 Pembuatan Tampilan *Mocklist* Pilih Matkul

## 6. Membuat Tampilan *Pop-up* Pilih Matkul

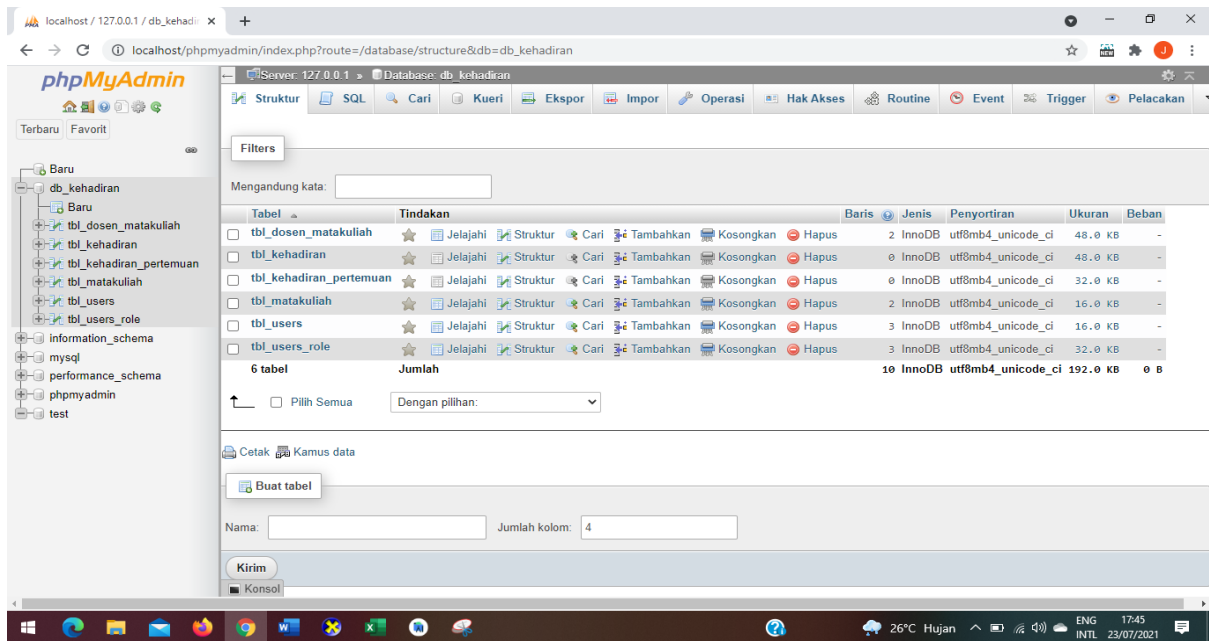
Tampilan ini dirancang untuk Pilihan Matakuliah, Nama Dosen, Nama Matkul.



Gambar 3.18 Pembuatan Tampilan *Pop-up* Pilih Matkul

### 3.12. Koneksi *Database Mysql*

Aplikasi Phpmysql adalah sebuah aplikasi manajemen *database Mysql* yang berbasis *web*. XAMPP menambahkan aplikasi ini untuk mempermudah pengaksesan database yang biasanya diakses menggunakan *console*, atau *CMD Windows*. Namun penulis tidak akan membahas cara penggunaan aplikasi phpMyAdmin pada tutorial kali ini. Kita hanya untuk mengecek koneksi dari aplikasi PHP ke Mysql.



Gambar 3.19 Database db\_kehadiran

Aplikasi Pencatatan kehadiran ini menggunakan database Mysql sehingga dibutuhkan sebuah koneksi antara aplikasi ke database Mysql cara mengkoneksikan database keaplikasi adalah sebagai berikut :

```

1 package id.orzdevs.absensirfidv1.controllers.koneksi;
2
3 public class Config {
4     private static final String LOCAL_URL = "https://absensirfidv1.orzdevs.com/api/";
5
6     public static final String URL_Get_Unlock_kehadiran = LOCAL_URL+"get_unlock_kehadiran/tags_id/";
7     public static final String URL_Get_Pertemuan_ke = LOCAL_URL+"get_pertemuan/id_matkul/";
8     public static final String URL_Post_Catat_kehadiran = LOCAL_URL+"post_add_kehadiran";
9
10 }

```

Gambar 3.20 Cara Koneksi Database db\_kehadiran

1. Masukan *private static final* mysql server ke aplikasi disini aplikasi android studio menggunakan Local URL "<https://absensirfidv1.com/api/>" dengan cara meng hosting mysql dan aplikasi sistem pencatatan nya.
2. Masukan *public static final* Local URL "get\_unlock\_kehadiran/tags\_id/".
3. Masukan *public static final* Local URL "get\_pertemuan/id\_matkul/".

Masukan *public static final* Local URL "post\_add\_kehadiran".