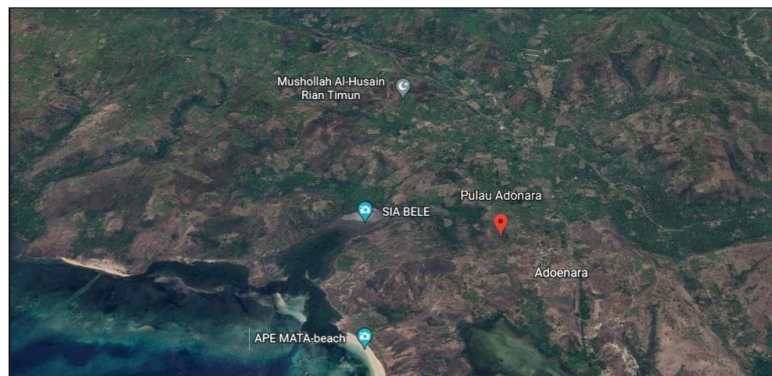


BAB III METODE PENELITIAN

3.1 Lokasi Penelitian

Penelitian ini dilakukan di Kecamatan Adonara yang berada di Kabupaten Flores Timur, Nusa Tenggara Timur. Adonara adalah sebuah pulau yang terletak di Kepulauan Nusa Tenggara, yakni di sebelah timur Pulau Flores. Luas wilayahnya 509 km², dan titik tertingginya 1.676 m. Pulau ini dibatasi oleh Laut Flores di sebelah utara, Selat Solor di selatan (memisahkan dengan Pulau Solor), serta Selat Lowotobi di barat (memisahkan dengan Pulau Flores)



Gambar 3. 1 Lokasi Penelitian

3.2 Alat dan Bahan

Dalam pengolahan data pada penelitian ini dibutuhkan data dan bahan yang mendukung proses pelaksanaannya. Adapun data dan bahan yang digunakan adalah sebagai berikut:

1. Alat

Tabel 3. 1 Peralatan

No	Perangkat lunak (<i>Software</i>)	Perangkat keras
1	<i>Xampp</i>	<i>GPS Handheld</i>
2	<i>Android Studio</i>	<i>Laptop</i>
3	<i>MySQL</i>	<i>Smartphone Android</i>

4	<i>Microsoft office excel 2010</i>	
5	<i>Microsoft Office Word 2010</i>	
6	<i>Google Maps API</i>	
7	<i>Android Studio</i>	
8	<i>Visual Studio Code</i>	
9	Bahasa pemrograman <i>flutter</i>	

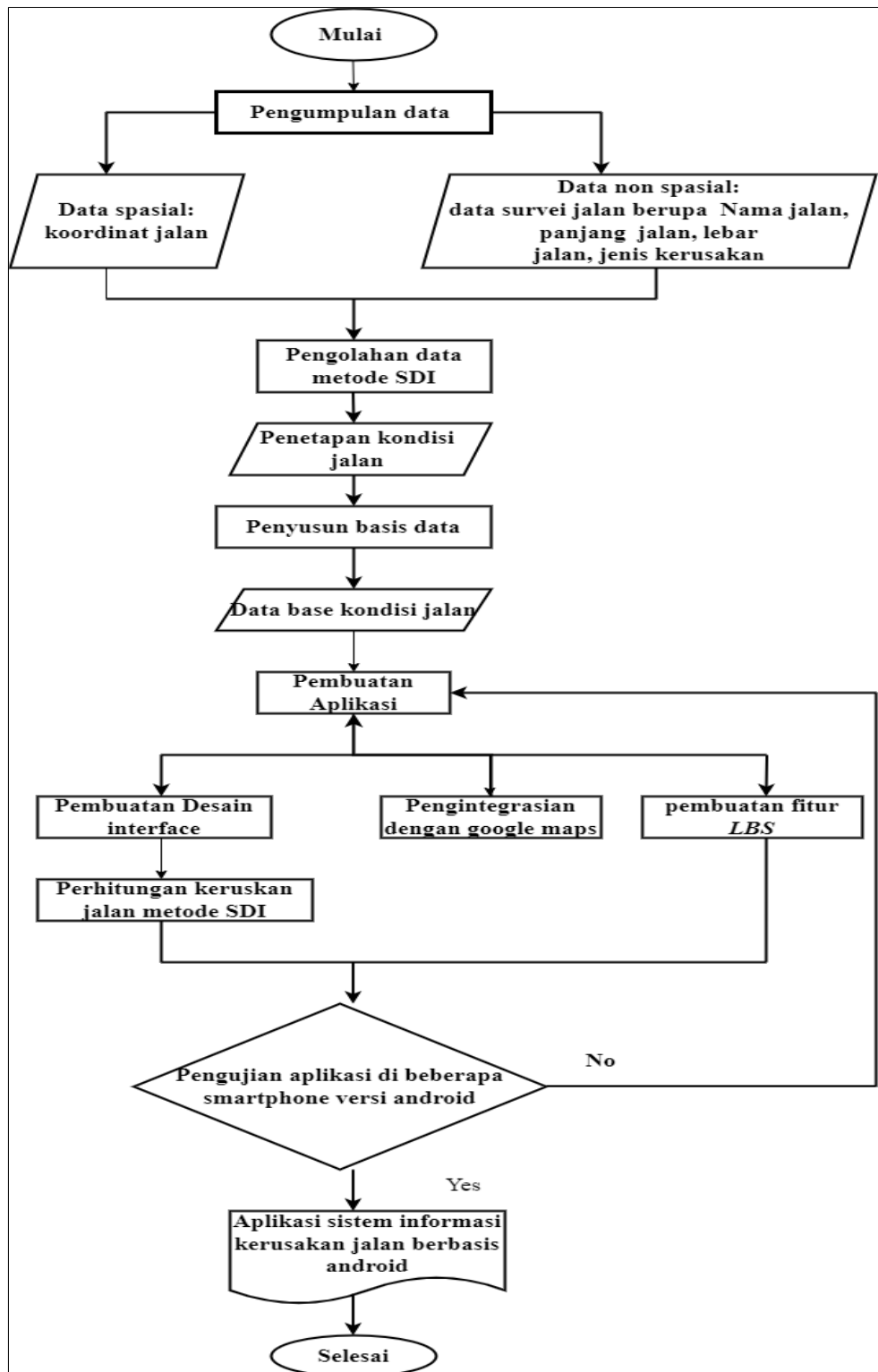
2. Bahan

Adapun beberapa data dari penelitian yang digunakan sebagai proses pembuatan aplikasi mobile GIS berbasis android untuk perhitungan kondisi perkerasan jalan di Kota Malang, agar terbentuk hasil output yang diinginkan. Beberapa data dalam penelitian tersebut adalah:

- a. Data koordinat jalan berupa koordinat *latitude* dan *longitude*.
- b. Data hasil survei kerusakan jalan berupa panjang, lebar, kondisi, jenis penanganan

3.3 Diagram Alir

Pada proses pelaksanaan penelitian akan memiliki alur penelitian yang mana bertujuan untuk menggambarkan dan menjelaskan tahap-tahap proses penelitian secara garis besar dimulai dari awal hingga selesai. Tahapan yang akan dilaksanakan dalam penelitian skripsi Pemanfaatan Teknologi Mobile SIG untuk penilain kondisi kerusakan jalan menggunakan metode *Surface Distress Index* sebagai berikut:



Gambar 3. 2 Diagram Alir

Adapun penjelasan dari diagram alir di atas adalah sebagai berikut:

1. Persiapan, pada tahap ini berisikan kegiatan persiapan peralatan dan perlengkapan yang akan digunakan dalam penelitian seperti laptop dengan spesifikasi yang memadai serta menginstall *software* untuk pembuatan aplikasi *android* seperti *android studio*, XAMPP, dan *visual studio code*.
2. Pada tahap ini dilakukan kegiatan pengumpulan data spasial dan non spasial. Untuk data Spasial berupa koordinat jalan dan untuk data non spasial berupa informasi jalan yang di dapat dari lapangan berupa nama ruas, panjang dan lebar jalan dan jenis kerusakan.
3. Pengolahan data metode SDI
Pengolahan metode SDI ini dilakukan sebagai proses perhitungan data-data lapangan yang telah di survei dengan beberapa proses pengolahan seperti perhitungan luas retak, lebar retak, jumlah lubang, dan bekas roda.
4. Penetapan kondisi jalan
Tahapan ini berisikan nilai kondisi jalan sesuai dengan perhitung sebelumnya yang menggunakan metode SDI
5. Penyusunan basis data. Pada tahap ini akan dilakukan proses penyimpanan basis data spasial dan basis data non spasial dalam bentuk DBMS menggunakan MySQL sehingga akan menghasilkan *database* jalan
6. Pembuatan aplikasi pada tahap ini dilakukan proses pembuatan aplikasi android berdasarkan rancangan yang telah dibuat sebelumnya yang dibagi menjadi tiga tahapan yaitu:
 - a. *Desain Interface* Aplikasi
Merupakan proses pembuatan tampilan antarmuka dari aplikasi mulai dari tampilan menu utama, deskripsi, dan perhitungan nilai SDI untuk mendapatkan kondisi penetapan jalan setelah perhitungan nilai SDI
 - b) Mengintegrasikan aplikasi dengan peta *Google Maps* proses menghubungkan *google maps* dengan aplikasi agar dapat menampilkan basemap berupa *google maps* secara *online* pada *android*.
7. *Upload Hosting*.
Aplikasi Akan di-upload ke dalam sebuah server hosting agar bisa

- bekerja secara online dan dapat diakses di manapun dan kapanpun
8. Pengujian sistem pada tahap ini Akan dilakukan pengujian dari sistem yangtelah dibuat sebelumnya. Terdapat 2 tahap pengujian yang dilakukan yaitu uji perangkat dan uji *usability*
 - a. Pengujian perangkat dilakukan untuk mengetahui apakah aplikasi yang dibuat dapat berjalan dengan baik di berbagai versi android yangberbeda menggunakan beberapa *smartphone*. Aplikasi akan diuji dengan melihat keberhasilan dalam menjalankan fungsi-fungsi yang sudah dirancang.
 - b. Pengujian *usability* dilakukan dengan cara menyebarkan kuesioner kepada beberapa responden untuk penilaian terkait kepuasan & tanggapan user terhadap aplikasi. Hasil kuesioner inilah yang digunakan untuk menetapkan tingkat kelayakan aplikasi.
 9. Penyajian hasil aplikasi adalah tahapan terakhir yang merupakan penyajian aplikasi kepada pengguna agar dapat dapat dipergunakan dimanfaatkan dengan baik.

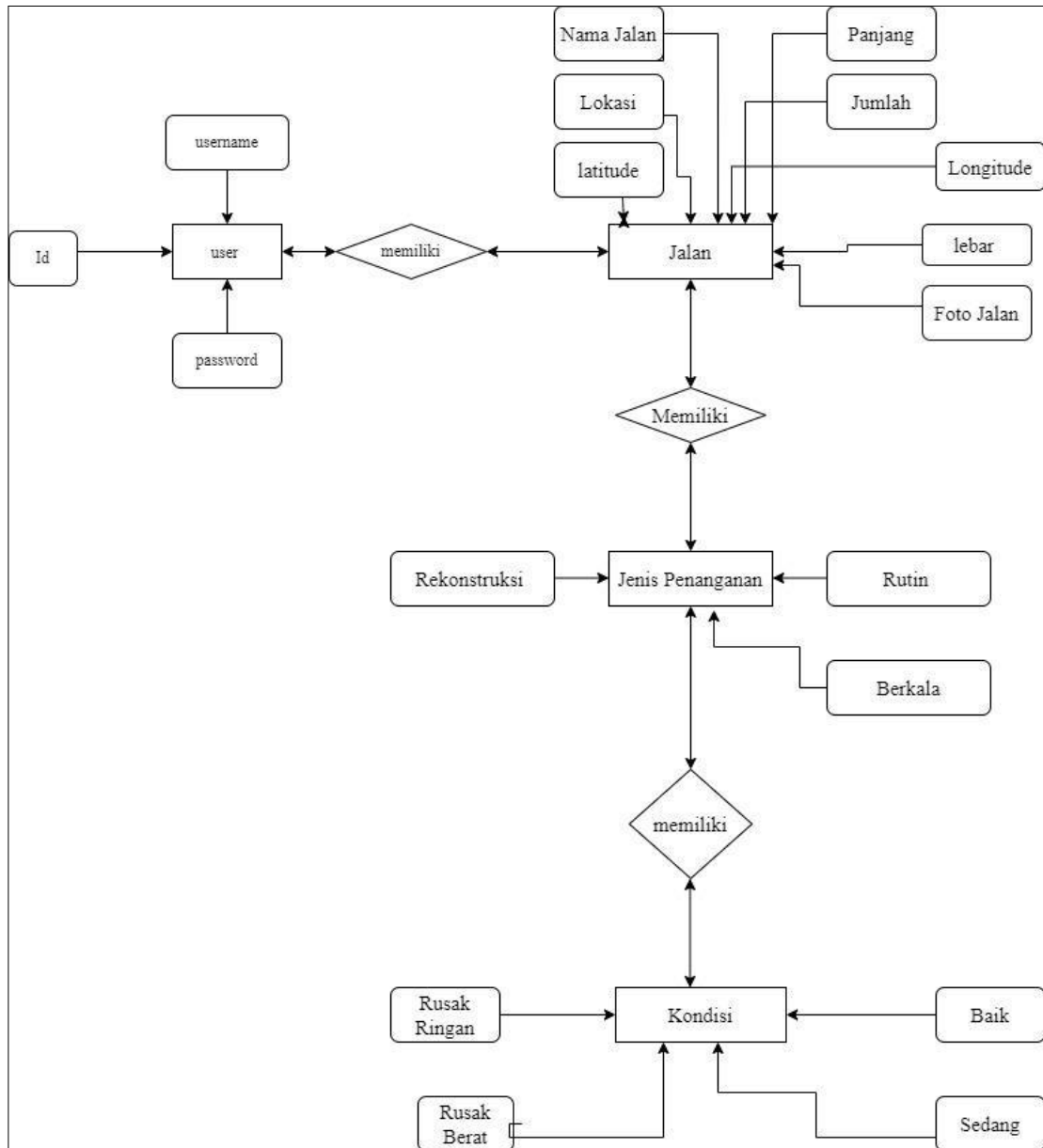
3.4 Perancangan Entity Relationship (ER)

Entity Relationship (ER) adalah model konseptual yang mendeskripsikan hubungan antara entitas. *Entity Relationship (ER)* digunakan untuk memodelkan struktur data dan hubungan antara data pada sistem informasi jaringan jalan kecamatan Adonara. Pada tahap ini dilakukan perancangan basis data berupa rancangan tabel dan atribut yang dibutuhkan untuk menyimpan berbagai data yang terkait dengan aplikasi kondisi jalan. Berikut desain basis data jalan:

1. Menentukan Entitas
 - a) Data administrasi
 - b) Jalan
 - c) Perhitungan nilai SDI
 - d) Kondisi jalan
2. Membuat *Enterprise Rule*
 - a) Satu wilayah administrasi memiliki satu ruas jalan
 - b) Satu ruas jalan terdapat beberapa perhitungan nilai SDI

c) Beberapa perhitungan memiliki kondisi jalan

3. Menentukan Tingkat *Obigatoring*



Gambar 3. 3 Diagram Er

Pada skema *Entity Relationship Diagram (ERD)* di atas terdapat 4 entitas yang tercantum di dalamnya yaitu user, ruas Jalan, kondisi, jenis penanganan. Tiap tiap entitas memiliki atribut yang berbeda-beda.

3.5 Perancangan Table

1. Perancangan user

Perancangan administrasi adalah tabel yang digunakan untuk

menyimpan data administrasi pada sistem basis data.

Tabel 3. 2 Tabel *User*

Nama	Tipe Data	Key
<i>ID_User</i>	<i>Varchar</i>	<i>Primary Key</i>
Nama	<i>Varchar</i>	
<i>Password</i>	<i>varchar</i>	

2. Perancangan Tabel jalan

Tabel jalan adalah tabel yang digunakan untuk menyimpan data jalan pada sistem basis data.

Tabel 3. 3 Perancangan Tabel Jalan

Nama	Type Data	Key
Nama jalan	<i>Varchar</i>	<i>Primary Key</i>
Lokasi jalan	<i>Varchar</i>	
Koordinat x	<i>Double</i>	
Koordiant Y	<i>Double</i>	
Panjang jalan	<i>Int</i>	
Kondisi jalan	<i>Varchar</i>	

3. Perancangan Tabel perhitungan nilai SDI

Tabel pengguna adalah tabel yang digunakan untuk menyimpan data pengguna pada sistem basis

Tabel 3. 4 Tabel Perhitungan Nilai SDI

Nama	Tipe Data	Key
Retak luas	<i>int</i>	
Retak Lebar	<i>Int</i>	
Jumlah lubang	<i>int</i>	
Bekas roda	<i>Int</i>	

4. Perancang Tabel Kondisi Jalan

Tabel pengguna adalah tabel yang digunakan untuk menyimpan data pengguna pada sistem basis data

Tabel 3. 5 Perancang Tabel Kondisi Jalan

Nama	Tipe Data	Key
Baik	<i>int</i>	
Sedang	<i>Int</i>	
Rusak ringan	<i>int</i>	
Rusak berat	<i>Int</i>	

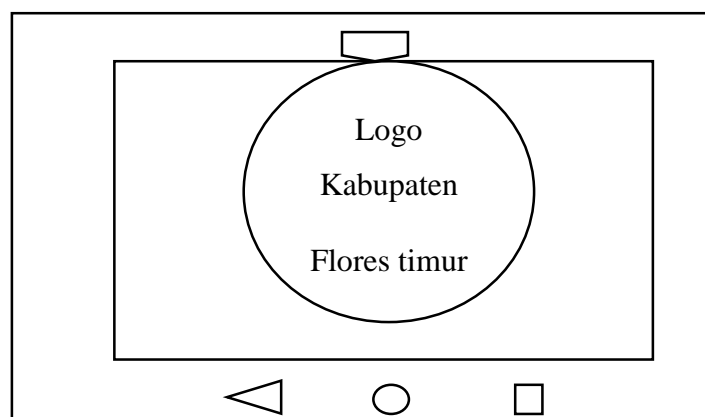
5. Perancang tabel kemantapan

Tabel pengguna adalah tabel yang digunakan untuk menyimpan data pengguna pada sistem basis data

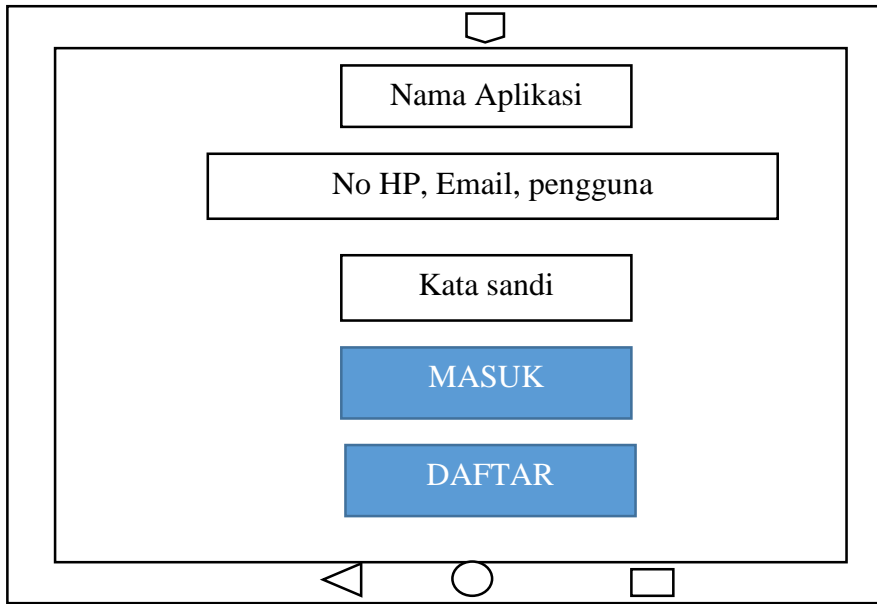
Tabel 3. 6 Perancang tabel kemantapan

Nama	Tipe Data		Key
Mantap	Int		
Tidak mantap	Int		

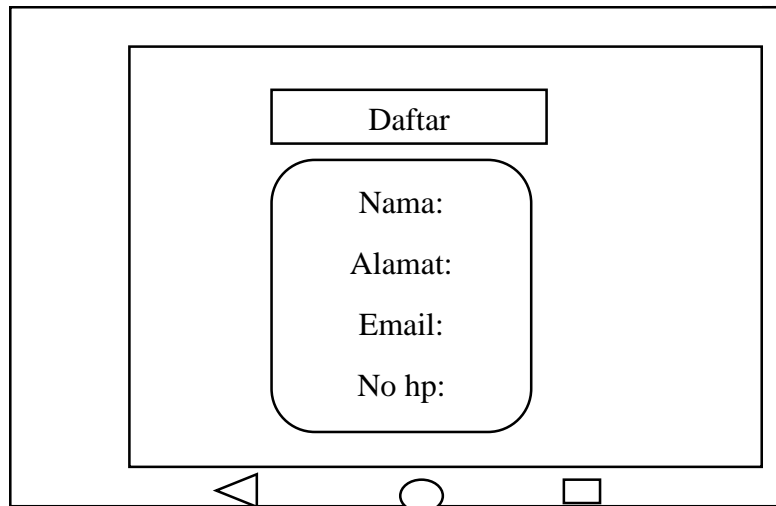
3.6 Perancangan Deesain Antarmuka Aplikasi (Desain Interface)



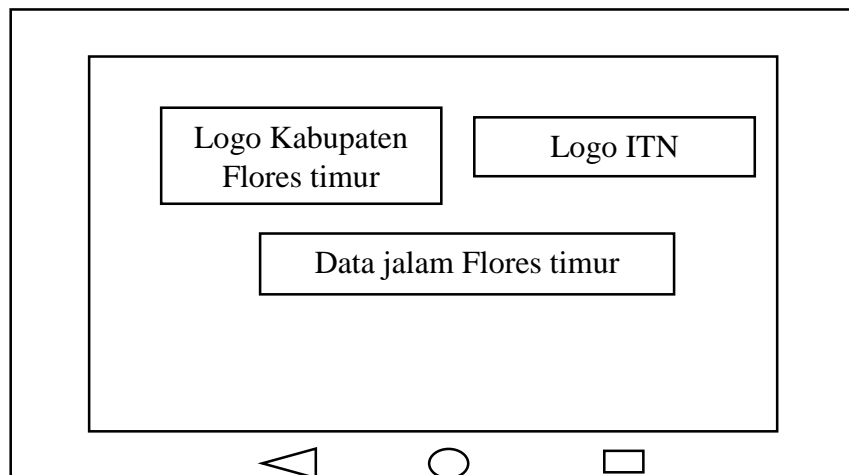
Gambar 3. 4 Tampilan Awal



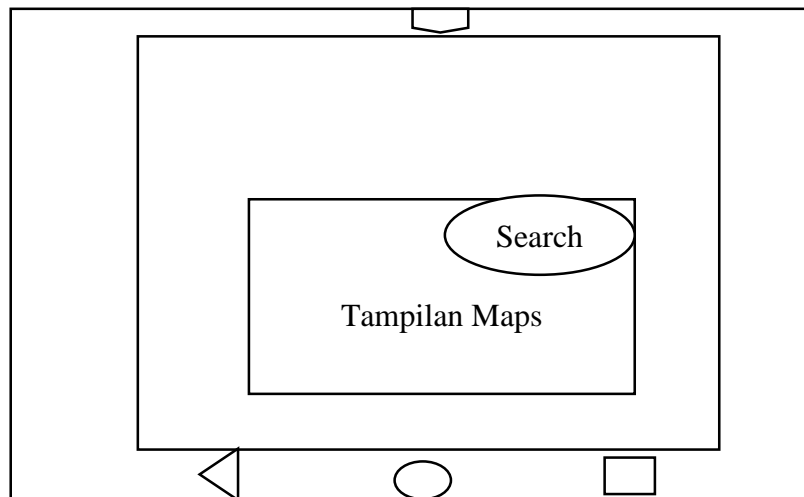
Gambar 3. 5 Tampilan Login



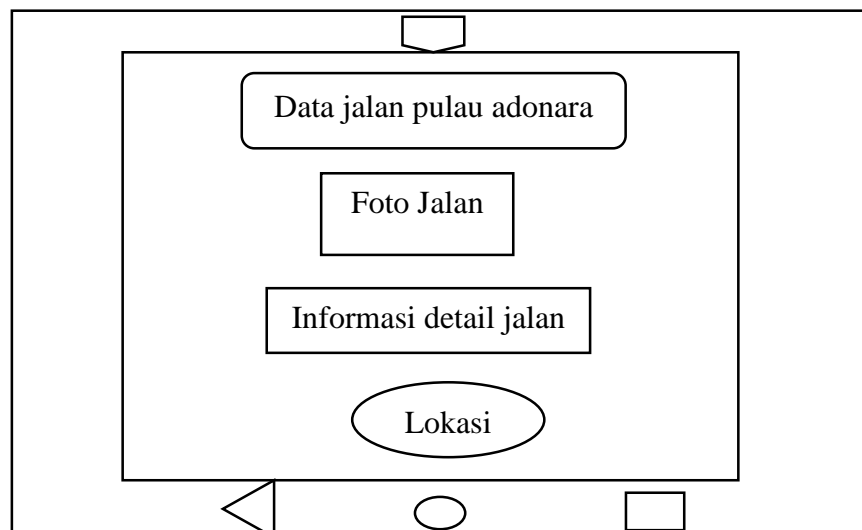
Gambar 3. 6 Tampilan Daftar Akun



Gambar 3. 7 Tampilan Beranda



Gambar 3. 8 Tampilan Peta



Gambar 3. 9 Tampilan Data Jalan

Adapun penjelasan dari tampilan *desain interface* di atas adalah sebagai berikut:

1. Tampilan *Login*: berisi menu yang digunakan oleh *user* untuk melakukan *login* sebelum mengakses dan berinteraksi pada aplikasi dengan memasukkan *email* dan *password* yang sudah didaftarkan sebelumnya
2. Tampilan Daftar Akun: berisi menu yang digunakan untuk melakukan pendaftaran akun bagi *user* yang belum memiliki akun untuk menggunakan aplikasi
3. Tampilan *Home*/Beranda: berisi tampilan layar depan aplikasi beserta

menu-menu utama yang tersedia di dalamnya, seperti menu peta jalan flores timur, menu data jalan Flores Timur, perhitungan kerusakan jalan dan menu tentang aplikasi

4. Tampilan Data Jalan: berisi menu yang digunakan untuk menampilkan daftar data jalan dari hasil survei yang dilakukan. Pada menu data jalan akan ditampilkan foto kondisi jalan, informasi kerusakan jalan (Nama jalan, panjang jalan , *longitude*, *latitude*, dll) serta lokasi titik jalan

3.7 Tahap Pelaksanaan

Pada pelaksanaan penelitian untuk pembuatan aplikasi *Mobile Geographic Information Systems (GIS)* untuk penilaian kondisi jalan dengan metode *SDI* terjadi beberapa proses meliputi pengumpulan data, perancangan dan penyimpanan basis data, dan pembuatan aplikasi hingga menjadi hasil akhir berupa suatu aplikasi.

3.7.1 Pengumpulan data

Pada tahap ini akan dilakukan pengumpulan data dengan cara pengambilam koordinat jalan, kemudian melakukan penelitian kerusakan jalan menggunakan metode *survey distress index*. Data- data yang terkumpul terdiri dari data primer dan data sekunder. Data primer berisi hasil *tracking GPS* berupa titik koordinat setiap ruas jalan yang telah terpasang berupa latitude dan longitude, data hasil survey setiap jalan dan dokumentasi terkait kondisi jalan.

3.7.2 Perhitungan Nilai *Survey Distress Index (SDI)*

1. Melakukan pendataan ruas jalan dengan mengisi form yang tersedia. Di bawah contoh salah satu perhitungan nilai SDI pada ruas jalan Lewoduli-Dokeng. Sedangkan perhitungan ruas jalan yang lain akan di lampirkan di bagian akhir.
2. Selanjutnya menghitung nilai SDI pada salah satu ruas jalan Lewoduli-Dokeng Perhitungan nilai SDI berdasarkan luas total retak.

(i) = Lebar lajur segment jalan

(p) = Panjang segment jalan

(L) = Luas segment jalan

$$L = p \times i$$

(c) = total luas retak

$$\text{Presentase luas retak: } \frac{c}{L} \times 100\%$$

Contoh perhitungan salah satu ruas jalan Lewoduli-dokeng dari STA

$$0+00 - 0+200$$

Panjang 1 segment = 200 m, lebar jalur (i) = 3,4 m, Panjang jalan

$$(p) = 200.$$

$$\text{luas segment jalan (L) = } p \times i = 200 \times 3,4 = 680 \text{ M}^2$$

➤ Tahap 1

Perhitungan nilai SDI berdasarkan **luas retak**

$$\text{Total luas retak (c) = } 20 \text{ m}^2$$

$$\text{Presentasi luas retak} = \frac{c}{L} \times 100\% = \frac{20}{680} \times 100\% = 2,94\%$$

Berdasarkan kategori luas retak pada table 2.1 maka Presentase total luas retak: 2,94% < 10 % Nilai SDI1: 5

➤ Tahap 2

Perhitungan akumulasi nilai SDI2 berdasarkan lebar rata-rata dari retak. Lebar retak rata-rata pada ruas lewoduli dokeng = 3 mm

Berdasarkan table 2.2 kategori lebar retak adalah sedang jadi nilai SDI adalah 0 (karena 3 nilai SDI nya 0)

Jadi lebar retak = 0

➤ Tahap 3

Perhitungan SDI3 yaitu perhitungan yang juga merupakan akumulasi dari nilai SDI2 yang nilainya ditentukan berdasarkan jumlah lubang yang ada

$$\text{Jumlah lubang} = 1$$

Maka berdasarkan table 2.3 dapat di jabarkan nilai SDI3 sebagai berikut

$$\text{Nilai SDI3} = \text{SDI2} + 15$$

$$0 + 15 = 15$$

➤ Tahap 4

Nilai SDI4 juga merupakan perhitungan akumulatif dari SDI3 yang nilainya ditentukan berdasarkan penurunan akibat bekas roda.

Penurunan bekas roda = 1 cm

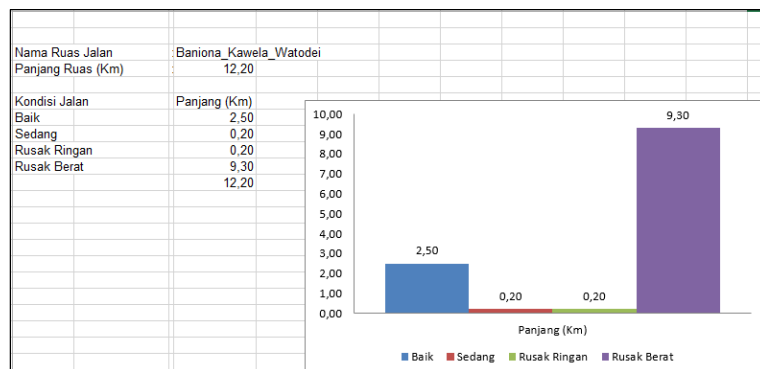
Nilai SDI4 = SDI3+(5x2)

$$15 + 10 = 20$$

Jadi nilai SDI4 atau total keseluruhan nilai SDI adalah 20

Berdasarkan table 2.5 tentang kategori kondisi maka kondisi jalan Lewoduli - Dokeng di kategori kan baik

3. Dari hasil perhitungan nilai SDI total maka didapat nilai untuk total kerusakan jalan untuk menentukan kondisi kerusakan jalan.

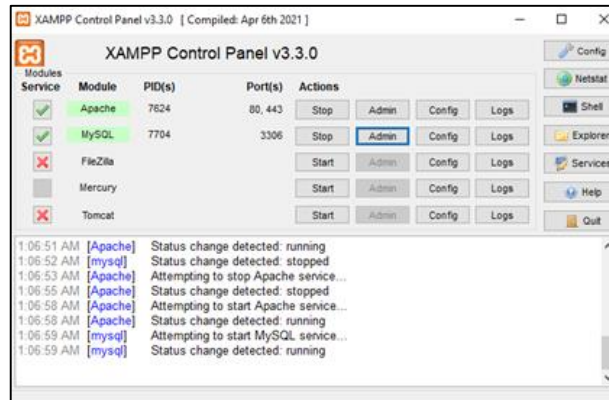


Gambar 3. 10 Grafik kondisi kerusakan jalan

3.7.3 Pembuatan Basis Data

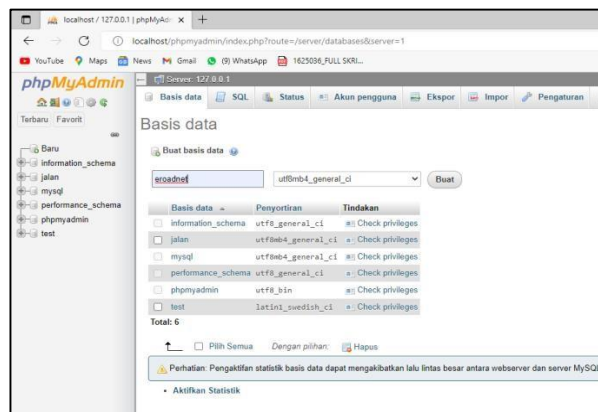
Pada tahap ini Pada tahap ini Akan dilakukan pembuatan basis data dengan menggunakan sistem basis data MySQL. Basis data menggunakan nama “db_eroadnet” dengan langkah-langkah sebagai berikut:

1. Langkah pertama adalah buka software XAMPP Control Panel yang sudah terinstall sebelumnya, kemudian mengaktifkan fasilitas Apache dan MySQL terlebih dahulu pada XAMPP Control Panel dengan cara menekan tombol start lalu klik admin.



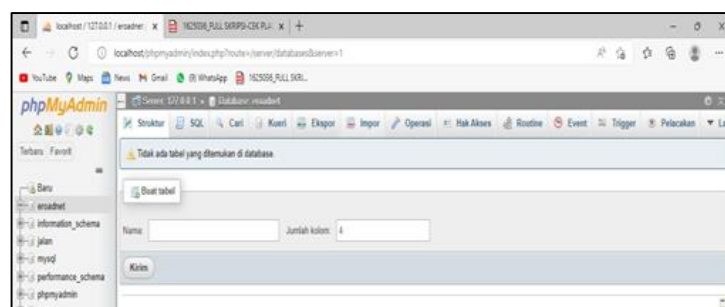
Gambar 3. 10 Tampilan Kotak *Dialog* Xampp

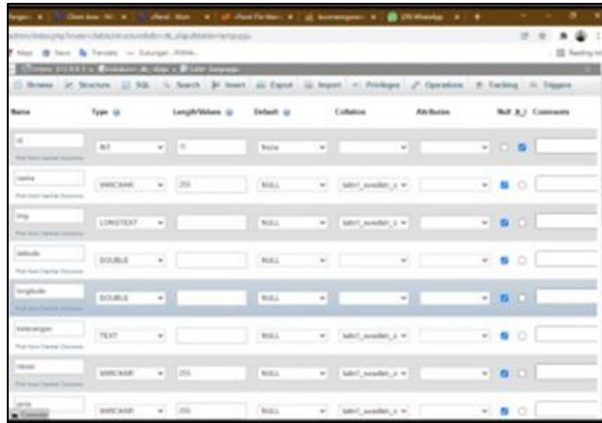
- Setelah itu akan masuk ke halaman *localhost* phpMyadmin <http://localhost/phpmyadmin/>. Untuk membuat basis data baru dengan cara klik menu database lalu tuliskan Nama basis data yang akan di buat pada kolom *create dabatase* dan klik *create*.



Gambar 3. 11 Tampilan *Create Database*

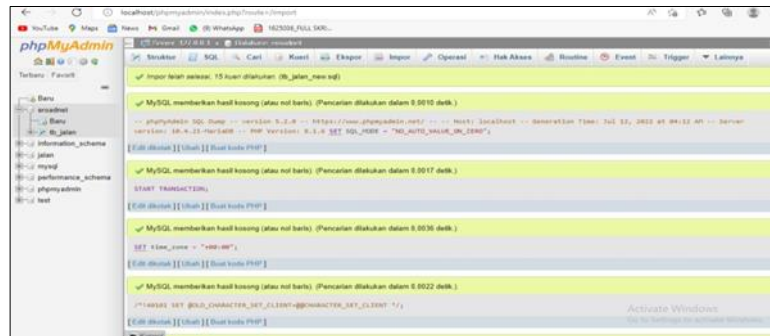
- Setelah membuat *create database* dengan nama *eroadnet* maka akan muncul tampilan seperti di bawah ini. Langkah selanjutnya klik impor untuk *mengimport database* yang telah di buat di *notepad*.



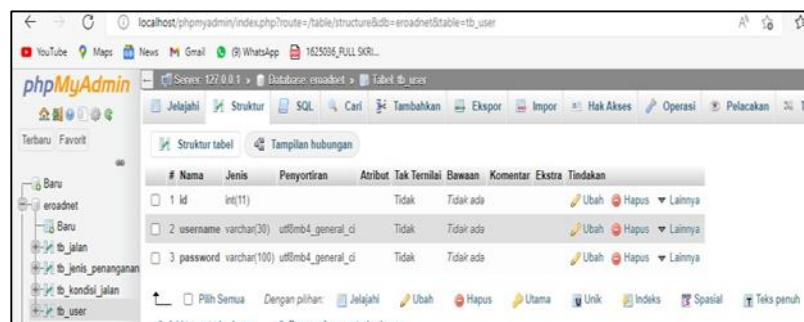


Gambar 3. 12 *Create Database Baru*

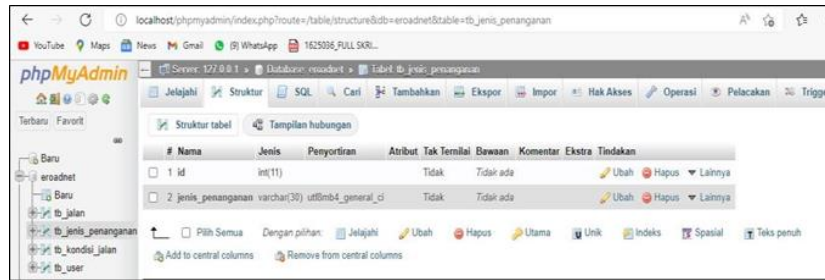
- Setelah itu memasukkan atribut pada setiap tabel seperti *Name, Type, Length/Values, Default, Collection, Indeks*, dll lalu klik *save*.



Gambar 3. 13 *Tampilan Setelah Data Di Import*



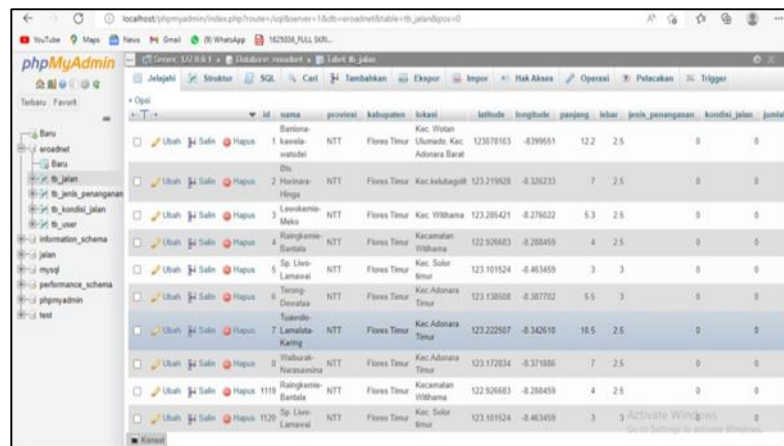
Gambar 3. 14 *Tampilan Input Attribute Table*



Gambar 3. 15 Tampilan Atribut Dari Tabel Jenis Penanganan



Gambar 3. 16 Tampilan Atribut Dari Tabel Kondisi Jalan



Gambar 3. 17 Tampilan Atribut Dari Tabel Jalan

5. Aplikasi ini menggunakan 4 buah tabel yaitu :” User”, “jalan”, “ jenis penanganan”, “kondisi”. Tabel user memuat data seperti: *Id*, *User name*, *password*. Tabel jalan memuat data seperti nama jalan, provinsi, kabupaten, lokasi, *latiude*, *longitude*, panjang ruas, lebar jalan. Tabel jenis penanganan memuat data pemeliharaan rutin, pemeliharaan berkala, rekonstruksi. Tabel kondisi memuat baik, sedang, rusak ringan, rusak berat.

The screenshot shows the phpMyAdmin interface for a database named 'eroadnet'. The main area displays a table structure with the following data:

Tabel	Tindakan	Baris	Jenis	Penyortiran	Ukuran	Beban
<input type="checkbox"/> tb_jalan	☆ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	10	InnoDB	utf8mb4_general_ci	16.0 KB	-
<input type="checkbox"/> tb_jenis_penanganan	☆ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	3	InnoDB	utf8mb4_general_ci	16.0 KB	-
<input type="checkbox"/> tb_kondisi_jalan	☆ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	4	InnoDB	utf8mb4_general_ci	16.0 KB	-
<input type="checkbox"/> tb_user	☆ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	0	InnoDB	utf8mb4_general_ci	16.0 KB	-
4 tabel	Jumlah	17	InnoDB	utf8mb4_general_ci	64.0 KB	0 B

Gambar 3. 18 Basis Data

3.7.4 Pembuatan Desain Antara muka (*Desain Interface*)

Pada tahap ini dilakukan pembuatan desain antarmuka aplikasi menggunakan bahasa pemrograman *dart* pada *framework flutter*. Berikut ini tampilan *source code* bahasa pemrograman *dart*-nya:

a) Tampilan Menu Beranda/*Home*

Menu ini *beranda/home* merupakan tampilan antarmuka awal dari aplikasi yang akan muncul setelah proses *login*. Dimana pada menu ini berfungsi sebagai navigasi pengguna dalam menggunakan aplikasi. Terdapat beberapa bagian dari menu lainnya yang bisa dipilih pengguna seperti menu logo kabupaten flores timur, menu maps, menu roads, menu logout, menu jurnal, dan menu tentang. Berikut ini adalah *source code* untuk menjalankan menu *beranda/home* dan untuk lebih lengkapnya ada di lampiran.

```
import 'package:flutter/material.dart';
import 'package:get/get.dart';
import
'package:jalan_apps/apps/controllers/dashboard.controllers.dart';
import
'package:jalan_apps/apps/resources/bridges/listbridge.dart';
import
'package:jalan_apps/apps/resources/dashboard/bottomsheet/searcheroadnet.dart';
```

```

import
'package:jalan_apps/apps/resources/mainpage.dart';
import
'package:jalan_apps/apps/resources/maps/maps.dart';
import 'package:
jalan_apps/apps/resources/sites/sitegoverment.dart';
import 'package: jalan_apps/component/textlabel.dart';
import 'package: jalan_apps/ui/Uicolor.dart';

class HomePage extends StatelessWidget {
  final controllers = DashboardControllers.to;
  List<String> layanan = [
    "MAPS",
    'ROADS',

```

Gambar 3. 19 *Source Code* Tampilan Menu Beranda/Home

b) Tampilan Menu Daftar Akun

Pada menu ini akan menampilkan data daftar akun. Berikut ini adalah *source code* untuk menjalankan menu daftar akun dan untuk lebih lengkapnya ada di lampiran.

```

import
'package:flutter/material.dart'
;import 'package:get/get.dart';
import 'package:mapsapps/helper/session.models.dart';
import 'package:mapsapps/modules/jalan/data
jalan/controllers/daftarlampu.pju.controllers.dart';
import
'package:mapsapps/modules/pju/daftarlampu/modelsjalan..models.dart';
import
'package:mapsapps/modules/eroadnet/kondisi/views/add.lampu.pju.views.dart'
;import
'package:mapsapps/modules/pju/daftarlampu/views/detaillampu.pju.views.dar
t'; import
'package:mapsapps/modules/pju/daftarlampu/views/filterlampu.pju.views.dar
t'; import 'package:search_page/search_page.dart';

Class DaftarPjuViews extends
StatefulWidget {@override
_DaftarPjuViewsState createState () => _DaftarjalanViewsState ();
}

Class _DaftarPjuViewsState extends
State<DaftarPjuViews> {DaftarLampuPjuControllers
controllers;

Color hexToColor (String code) {
return new Color(int.parse(code.substring(1, 7), radix: 16) +
0xFF000000);
}
@override
void initState() {
controllers = new
DaftarLampuPjuControllers();
controllers.getData();
super.initState();
}
@override
Widget build(BuildContext
context) {return SafeArea(
child: Scaffold(
backgroundColor:
Colors.white, appBar:

```

Gambar 3. 20 *Source Code* Tampilan Menu login

c) Tampilan *logout*

Pada menu ini akan menampilkan menu logout. Berikut ini adalah *source code* untuk menjalankan *menu logout* dan untuk lebih lengkapnya ada di lampiran.

```

Future<void> signOut() async {
    SharedPreferences sharedPreferences = await
    SharedPreferences.getInstance();
    sharedPreferences.clear();
}

```

Gambar 3. 21 *Source Code* Menu Logout

d) Tampilan detail jalan

Pada menu ini akan menampilkan detail informasi dari jalan sesuai yang dipilih. Pengguna dapat melihat kondisi kerusakan jalan. Berikut ini adalah *source code* untuk menjalankan menu detail jalan

```
import 'package:flutter/material.dart';
import 'package:get/get.dart';
import
'package:jalan_apps/apps/controllers/roads.controllers.dart';
import
'package:jalan_apps/apps/resources/roads/bottomsheet/searchroadslist.dart';
import
'package:jalan_apps/apps/resources/roads/bottomsheet/sortbridgeroads.dart';
import
'package:jalan_apps/apps/resources/roads/detailbridge.dart';
import
'package:jalan_apps/component/textlabel.dart';
import 'package:jalan_apps/ui/Uicolor.dart';
class ListroadsPage extends StatefulWidget {
  const ListroadsPage({ Key? key }) : super(key:
key);

  @override
  State<ListBridgePage> createState() =>
_ListBridgePageState();
}
```

Gambar 3. 22 *Source Code* Tampilan Menu Detail jalan

e) Tampilan Menu Peta jalan

Pada menu ini akan terhubung langsung dengan *google maps* yang akan

menampilkan persebaran lokasi kondisi jalan yang sudah terpasang dengan simbol titik sesuai dengan *latitude* dan yang ada pada *database*. Berikut ini adalah *source code* untuk menjalankan menu peta jalan

```
import 'package:flutter/material.dart';
import 'package:get/get.dart';
import
'package:google_maps_flutter/google_maps_flutter.da
rt';
import
'package:jalan_apps/apps/controllers/maps.controlle
rs.dart';
import
'package:jalan_apps/apps/resources/maps/bottomsheet
/searchroadsmat.dart';
import
'package:roads_apps/component/textlabel.dart';
import 'package:roads_apps/ui/Uicolor.dart';

class MapsPage extends StatelessWidget {
  final controllers = Get.put(MapControllers());

  @override
  Widget build(BuildContext context) {
    controllers.getDatajalan();
    return Scaffold(
      body: Container(
        child: SingleChildScrollView(
          child: Column(
            children: [
              Container(
                child: Stack(
                  children: [
                    Container(
                      height:
```

```
MediaQuery.of(context).size.height,  
                width: double.infinity,  
                decoration: BoxDecoration(  
                    color:  
Uicolor.hexToColor(Uicolor.blueTransparant),  
                    borderRadius:  
BorderRadius.circular(8),  
                ),
```

Gambar 3. 23 Source Code Tampilan Menu Peta jalan

f) Tampilan Menu *Login*

Menu ini akan menampilkan proses *login* untuk mengakses pada aplikasi dengan memasukkan data *username* dan *password* yang telah didaftarkan sebelumnya. Berikut ini adalah *source code* untuk menjalankan menu *login*

```

Import 'package:
flutter/material.dar
t';import 'package:
get/get.dart';

import
'package:mapsapps/modules/jalan/mainmenu/views/ma
inmenu.jalan.views.dart';
import'package:mapsapps/modules/pju/masuk/controll
ers/masuk.pju.controllers.dart'; class
MasukjalanViews extends StatefulWidget {

  @override
  _MasukPjuViewState createState () =>
  _MasukPjuViewState ();
}

Class _MasukPjuViewState extends
State<MasukPjuViews>
{ontrollers;

Color hexToColor (String code) {
  Return new Color (int.parse (code.substring (1,
7), radix: 16) + 0xFF000000);
}

@override
Void initState () {
  Controllers = new
MasukjalanControllers ();
super.initState ();
}

@override
Widget
build(BuildCont
ext context) {
return
SafeArea(
  child:
  Scaffold(
  backgroundCo
lor:
Colors.white
, appBar:

```

Gambar 3. 24 *Source Code Tampilan Menu Login*

g) *Source code bottom navigation Jurnal*

Menu ini akan menampilkan jurnal yang di buat yang telah di *upload* di aplikasi. untuk *Berikut* ini adalah *source code* untuk menjalankan menu *navigation jurnal*

```
// Copyright 2014 The Flutter Authors. All rights reserved.
// Use of this source code is governed by a BSD-style license that can be
// found in the LICENSE file.

import 'package:flutter/foundation.dart';
import 'package:flutter/rendering.dart';

import 'basic.dart';
import 'framework.dart';
import 'image.dart';

// Examples can assume:
// late BuildContext context;

/// A widget that paints a [Decoration] either before or after its child paints.
///
/// [Container] insets its child by the widths of the borders; this widget does
/// not.
///
/// Commonly used with [BoxDecoration].
///
/// The [child] is not clipped. To clip a child to the shape of a particular
/// [ShapeDecoration], consider using a [ClipPath] widget.
```



```

///
/// {@tool snippet}
///
/// This sample shows a radial gradient that draws
a moon on a night sky:
///
/// ```dart
/// const DecoratedBox(
///   decoration: BoxDecoration(
///     gradient: RadialGradient(
///       center: Alignment(-0.5, -0.6),
///       radius: 0.15,
///       colors: <Color>[
///         Color(0xFFEEEEEE),

```

Gambar 3. 25 Source Code Bottom Navigation Jurnal

h) Source Code Bottom Navigation About

Pada menu ini akan menampilkan *navigation about* tentang sebuah aplikasi yang di buat. Berikut merupakan *source kode* nya.

```

import 'package:flutter/material.dart';
import
'package:jalan_apps/component/textlabel.dart';
import 'package:jalan_apps/ui/Uicolor.dart';

class TentangPage extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: Colors.white,
      body: Container(
        child: SingleChildScrollView(
          child: Column(
            children: [
              Stack(
                children: [

```

```

        Container(
            height: 197,
            width: double.infinity,
            child: Image(
                image:
AssetImage("bg_tentang.png"),
                fit: BoxFit.cover,
            ),
        ),
        Container(
            alignment: Alignment.center,
            margin: EdgeInsets.only(
                top: 50,
            ),
            child: Row(
                mainAxisAlignment:
MainAxisAlignment.center,
                children: [
                    Column(
                        crossAxisAlignment:
CrossAxisAlignment.start,
                        children: [
                            Container(
                                child:
TextLabelComponent.show(
}

```

Gambar 3. 26 *Source Code Bottom Navigation About*

i. Tampilan Menu Laporan

Pada menu ini berisikan daftar laporan kerusakan kerusakan jalan yang Akan di hitung di lapangan. Berikut ini adalah source code untuk menjalankan menu laporan dan dan untuk lebih lengkapnya ada di lampiran

```

Import 'dart: convert';

Import 'package: flutter/material.dart'; Import
'package: get/get.dart';
Import 'package:
jalan_apps/apps/controllers/bridge.controllers.dart
'; Import 'package:
jalan_apps/apps/controllers/laporan.controllers.dar
t'; Import 'package:
jalan_apps/apps/resources/eroadnet/bottomsheet/sear
cheroadnetlist.dart'; Import 'package:
jalan_apps/apps/resources/eroadnet/bottomsheet/sort
eroadnet.dart'; Import 'package:
jalan_apps/apps/resources/eroadnet/detaileroadnet.d
art'; Import 'package:
jalan_apps/apps/resources/infoapps/notfoundpage.dar
t'; Import 'package:
jalan_apps/apps/resources/laporan/addlaporan.dart';
Import 'package:
jalan_apps/apps/resources/laporan/detaillaporan.

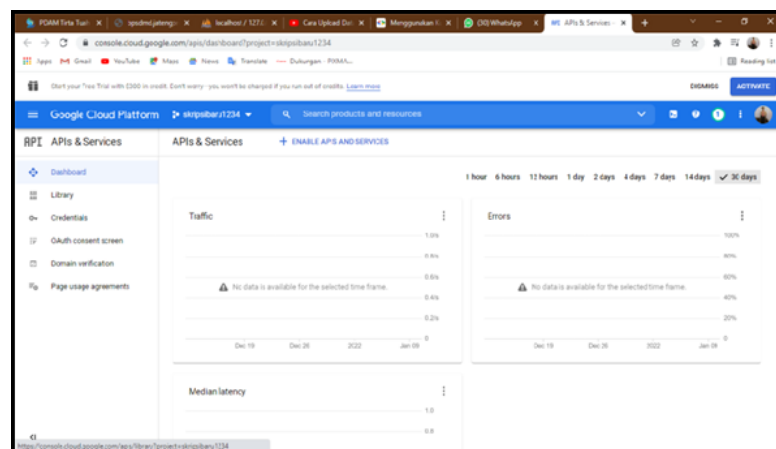
```

Gambar 3. 27 Tampilan source code laporan

3.7.5 Mengintegrasikan Aplikasi Dengan Peta Google Maps

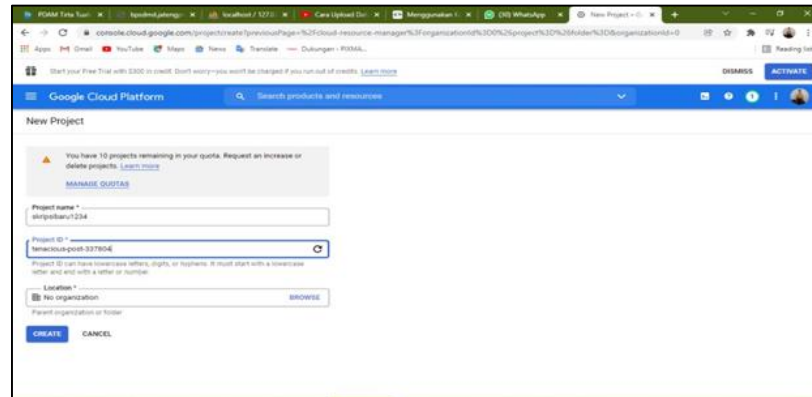
Agar aplikasi dapat terhubung dengan Google Map, maka aplikasi memerlukan akses dari Google Maps API untuk menampilkan basemap berupa Google Maps secara online pada aplikasi android. Adapun langkah-langkah untuk mendapatkan Google Maps API Key sebagai berikut:

1. Buka website <http://console.developers.google.com> pada browser maka akan muncul halaman dashboard APIs & Services



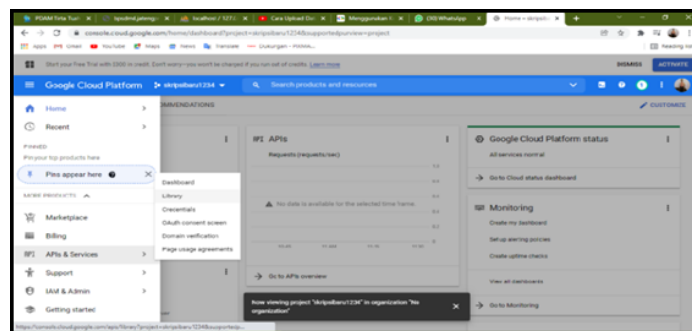
Gambar 3. 28 Web Console Google Maps API

2. Buat *project* baru dengan cara klik tombol *new project* maka akan muncul halaman *new project*, lengkapi informasi yang wajib diisi seperti *project name* dan *location*. Kemudian tekan tombol *create*



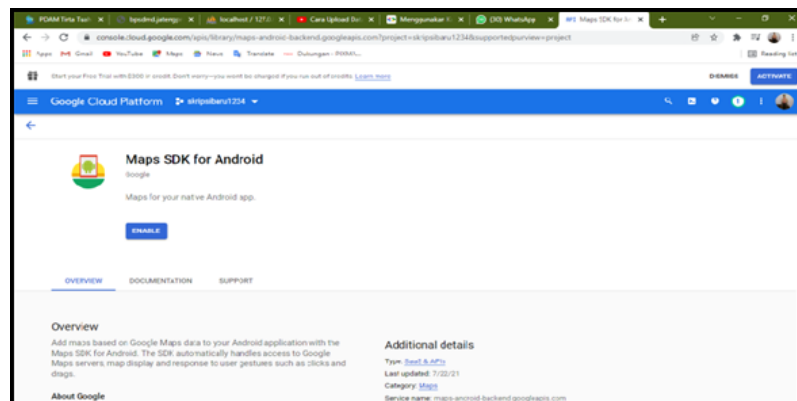
Gambar 3. 29 Project Name And Location

3. Selanjutnya klik *navigation* menu yang ada di pojok kiri atas kemudian pilih menu *APIs & services* > klik *library*.



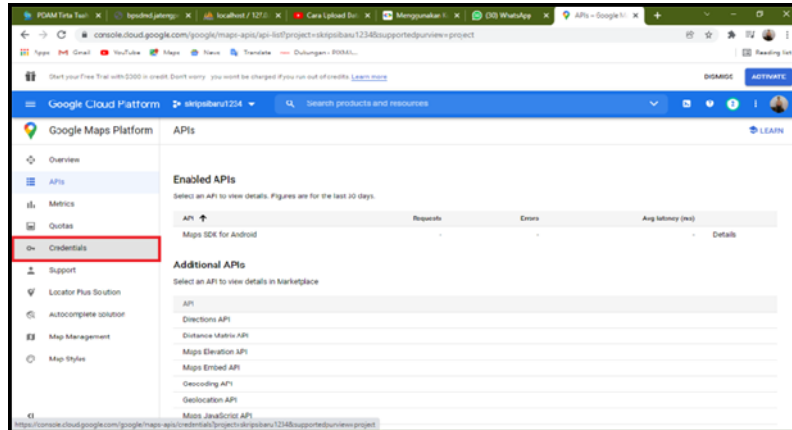
Gambar 3. 30 Menu Apis & Service

4. Setelah itu, pilih menu *Maps SDK For Android* > klik tombol *enable* dan *manage* untuk mengaktifkan SDK yang akan digunakan dengan *project*.



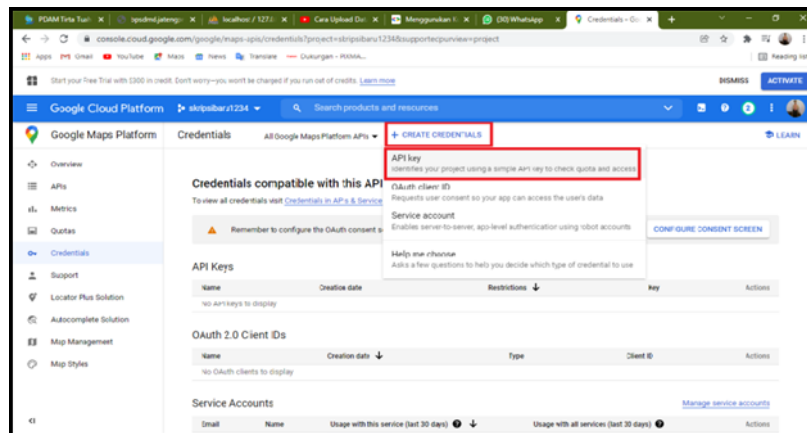
Gambar 3. 31 Menu Google Maps Enable Sdk

5. Kemudian pilih menu *credentials* yang berlogo kunci untuk membuat *API Key* pada *Maps SDK for Android* dan menambahkannya ke *project* aplikasi yang menggunakan *SDK* tersebut.



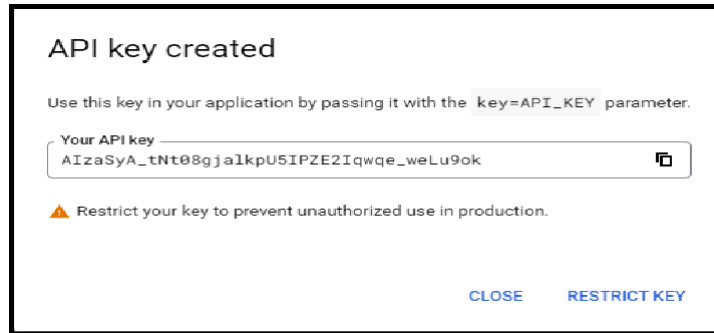
Gambar 3. 32 Menu *Credentials*

6. Pada halaman *credentials* untuk membuat *API-Key* dengan cara klik *create credentials* > pilih *API-Key*



Gambar 3. 33 Create Credentials & Get Api-Key

7. Maka akan muncul tampilan kotak dialog *API key created*. Untuk menambahkan *API Key* ke aplikasi, *Copy API key* agar dimasukkan ke dalam *source code* yang dibuat di *Sublime Text* menggunakan bahasa pemrograman *dart*



Gambar 3. 34 Menu *API Key Created*

```

<manifest
xmlns:android="http://schemas.android.com/apk/res/a
ndroid"
    xmlns:tools="http://schemas.android.com/tools"
    package="com.apps.jembatan_apps">
    <uses-permission
android:name="android.permission.ACCESS_FINE_LOCATI
ON"/>
    <uses-permission
android:name="android.permission.INTERNET" />
    <application
        android:label="Road Net"
        android:name=".${applicationName}"

android:networkSecurityConfig="@xml/network_securit
y_config"
        tools:replace="android:icon"
        android:icon="@drawable/logo">
        <activity
            android:name=".MainActivity"
            android:exported="true"
            android:launchMode="singleTop"
            android:theme="@style/LaunchTheme"

            android:configChanges="orientation|keyboardHidden|k
eyboard|screenSize|smallestScreenSize|locale|layout
Direction|fontScale|screenLayout|density|uiMode"

```

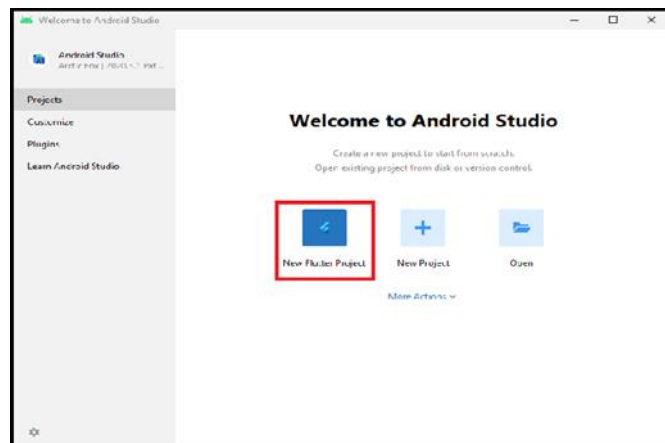
```
android:hardwareAccelerated="true"
android:usesCleartextTraffic="true"
android:largeHeap="true"
```

Gambar 3. 35 *Source Code* Menu API Key

3.7.6 Pembuatan Project Pada Android Studio

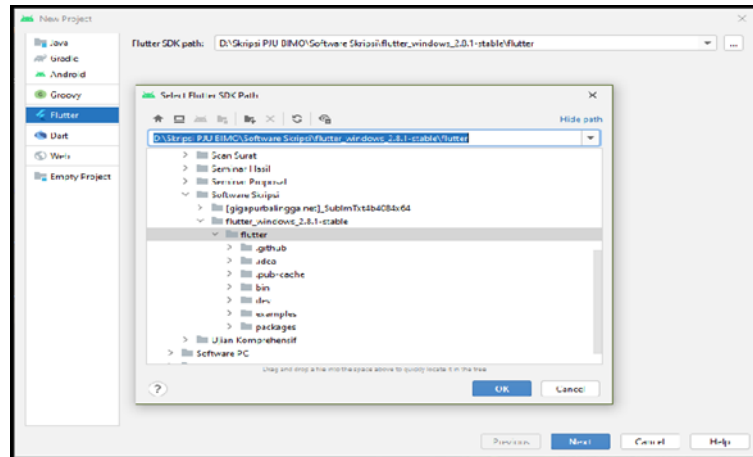
Pada tahap ini akan dilakukan pembuatan *project* pada android studio dengan langkah-langkah sebagai berikut:

1. Langkah pertama adalah membuka *software android studio*. Kemudian klik *new flutter project* untuk membuat *project* baru menggunakan bahasa pemrograman *dart*



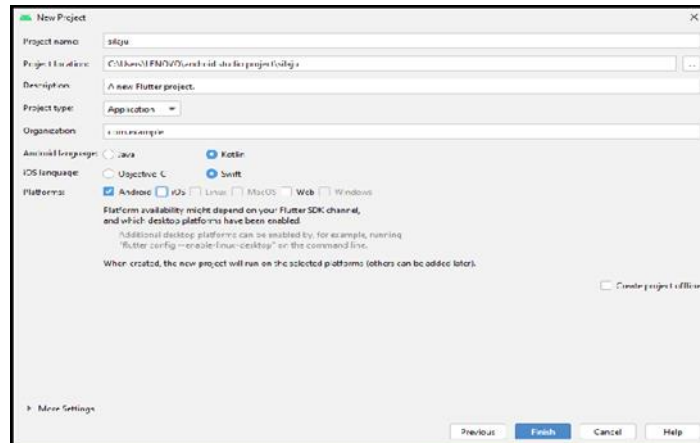
Gambar 3. 36 *Create New Flutter Project*

2. Setelah itu cari lokasi penyimpanan *flutter sdk path* yang sudah di *download* sebelumnya ➡ klik ok dan *next*.



Gambar 3. 37 Flutter Sdk Path

3. Maka akan muncul jendela *new project*, kemudian beri nama *project* aplikasi yang akan dibuat pada kolom *project name*, setelah itu pilih tempat dimana penyimpanan *project* tersebut dan memilih *project typenya* berupa *application*, selanjutnya pilih bahasa pemrograman yang digunakan dan pilih pada *platform* apa *project* tersebut akan di-*install*, lalu klik *finish*



Gambar 3. 38 Jendela New Project

4. Tunggu hingga proses *build gradle* selesai. Setelah itu, tulis *source code* yang ada di bawah ini pada *file Android Manifest.xml* untuk “*user permission*” yang nantinya akan digunakan setelah aplikasi di-*install* dan dijalankan pertama kali. Aplikasi ini memerlukan izin penggunaan seperti *GPS & location access*, *internet access*, dan *wifi access*. Serta untuk menentukan logo aplikasi, *splash screen* dan *google maps api key*

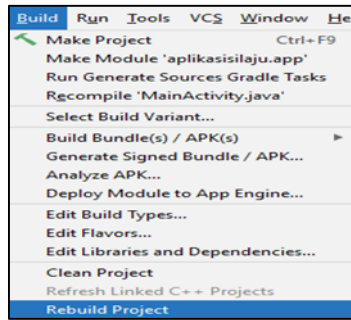

```

<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    package="com.apps.mapsapps">
<!-- io.flutter.app.FlutterApplication is an android.app.Application that
    calls FlutterMain.startInitialization(this); in its onCreate method.
    In most cases you can leave this as-is, but you if you want to provide
    additional functionality it is fine to subclass or reimplement
    FlutterApplication and put your custom class here. -->
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>
<uses-permission android:name="android.permission.FOREGROUND_SERVICE" />
<uses-permission android:name="android.permission.ACCESS_BACKGROUND_LOCATION"/>
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE"/>
    <application
        android:name="io.flutter.app.FlutterApplication"
        android:label="SILAJU"
        android:icon="@drawable/logo_silaju">
        <activity
            android:name=".MainActivity"
            android:launchMode="singleTop"
            android:theme="@style/LaunchTheme"
            android:configChanges="orientation|keyboardHidden|keyboard|screenSize|smallestScreenSize|locale|layoutDirection|fontScale|screenLayout|density|uiMode"
            android:hardwareAccelerated="true"
            android:windowSoftInputMode="adjustResize">
<meta-data
            android:name="io.flutter.embedding.android.NormalTheme"
            android:resource="@style/NormalTheme"
            />
        <meta-data
            android:name="io.flutter.embedding.android.SplashScreenDrawable"
            android:resource="@drawable/launch_background"
            /><intent-filter>
            <action android:name="android.intent.action.MAIN"/>
            <category android:name="android.intent.category.LAUNCHER"/>
        </intent-filter>
        </activity>
        <meta-data
            android:name="flutterEmbedding"
            android:value="2" />
        <meta-data android:name="com.google.android.geo.API_KEY"
            android:value="AIzaSyA_tNt08gjalpUSIPZE2Iqvwqe_wLu9ok"/>
        <meta-data android:name="com.mapbox.token"
            android:value="pk.eyJ1IjoiZGFwcmVhbnV0aW9ucyIsImEiOiJja2RiasdvZDQwY2w4MnNkdDR0e
            jVrcTB5In0.YCMYhheEcT0eB0aXmVd28w" />
        </application>
</manifest>

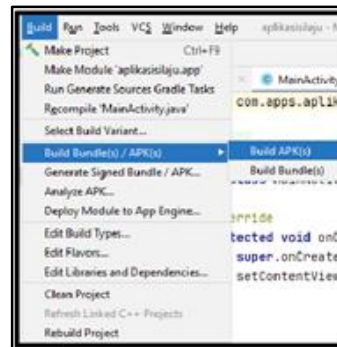
```

Gambar 3. 39 Membuat *Uses-Permission*

- Langkah terakhir yaitu membangun aplikasi dengan menggabungkan semua *source code* di *Android Studio* melalui menu *Build > rebuild project*. Setelah itu membuat *file* APK dengan menggunakan menu *Build > Build Bundle/APK > Build APK*. *File* APK tersebut nantinya akan digunakan untuk menginstall aplikasi pada *device android*



Gambar 3. 40 Membangun Aplikasi Dengan *Rebuild Project*



Gambar 3. 41 Membuat File Apk

3.7.7 Pembuatan Web Service Aplikasi

Web service adalah aplikasi yang tersedia melalui jaringan internet yang terdiri dari kumpulan protoko dan standar terbuka yang dirancang memiliki sistem pertukaran pesan atau infomrasi berbasis XML untuk bertukar data antar perangkat ke perangkat lainnya dalam format klien-server. Di bawah ini merupakan source code yang berfungsi untuk menyambungkan web service dengan server. Setelah itu hosting di upload agar data base bias di akses secara online dan menjadi halaman utama dari lingkungan kerja admin.

```

<?php

namespace App\Http\Controllers\ruasjalan;

use App\Http\Controllers\Controller;
use App\Model\RuasJalan\Jalan;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\DB;

class JalanController extends Controller
{
    /* untuk ambil data seluruh jalan*/
    public function getData(Request $request){
        $data = $request->all();
        $result['is_valid'] = false;
        $datadb = DB::connection('ruas')->table('tb_jalan')
        ->select([
            'tb_jalan.*',
            'thi.image'
        ])
        ->leftJoinSub("select min(id) as id, jalan
from tb_jalan_has_image group by jalan", 'min_img',
function($query){
            return $query->on('min_img.jalan',
'tb_jalan.id');
        })
        ->leftJoin('tb_jalan_has_image as thi',
'thi.id', 'min_img.id');
        if(isset($data['keyword'])){
            if(trim($data['keyword']) != ''){
                $datadb->where('tb_jalan.nama',
'like', '%'.$data['keyword'].'%');
            }
        }
    }
}

```

```

        if(isset($data['orderby'])) {
            if(trim($data['orderby']) ==
'longest'){
                $datadb-
>orderBy('tb_jalan.panjang', 'desc');
            }
            if(trim($data['orderby']) == 'best'){
                $datadb-
>orderBy('tb_jalan.kondisi', 'desc');

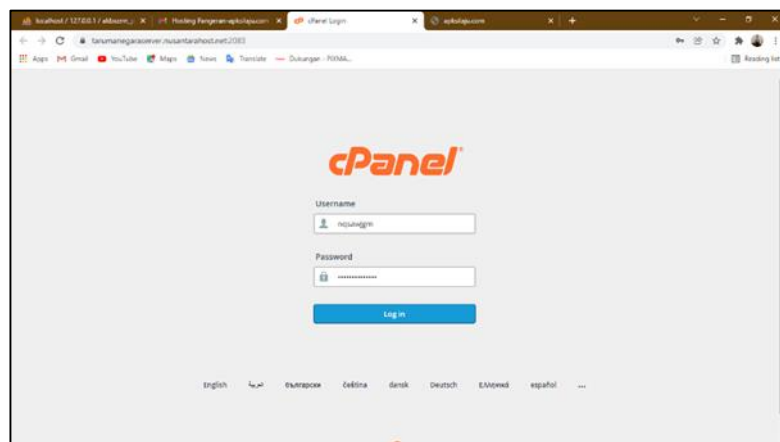
```

Gambar 3. 42 Source Code Koneksi Database.Php

3.7.8 Upload Hosting

Untuk men-*online*-kan atau membuat aplikasi dapat dibuka di *android* oleh pengguna di sembarang tempat secara bersamaan, maka yang harus dilakukan adalah mengunggah *file web service* dan basis data ke dalam *web hosting*. *Web Hosting* dapat diperoleh dengan membeli dari jasa *hosting*. Pengguna akan memperoleh *control panel* yang terproteksi dengan *username* dan *password* untuk mengontrol halaman *web*-nya. Berikut ini merupakan langkah-langkah dalam melakukan *hosting*.

1. Buka browser dan masuk ke dalam control panel web hosting di alamat <https://tarumanegaraserver.nusantarahost.net:2083> → login menggunakan *username* dan *password* yang sudah dimiliki

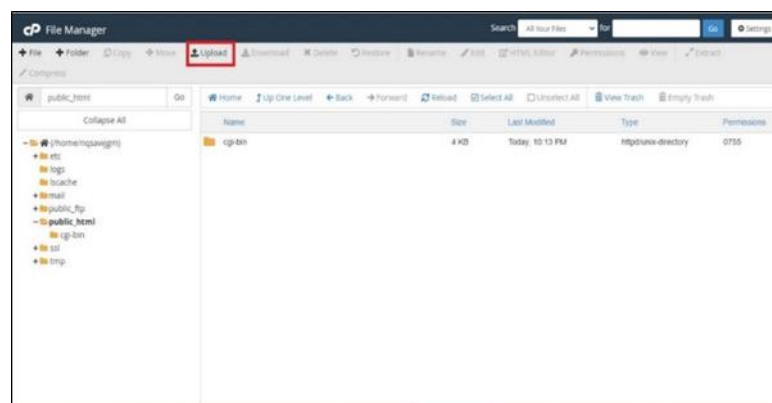


Gambar 3. 43 Login Ke Web Hosting

- Selanjutnya mengunggah *file web service* ke *web hosting* dengan cara memilih menu *file manager* → masuk ke folder *public_html*

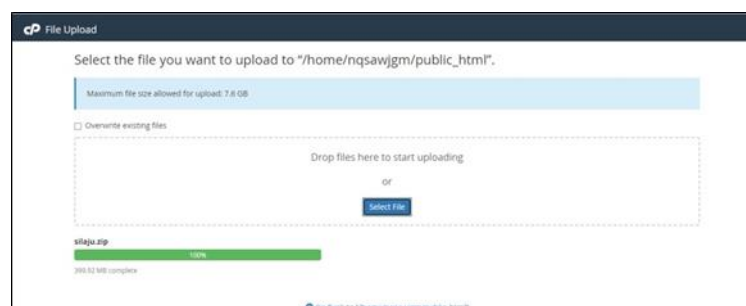


Gambar 3. 44 Menu *File Manger*



Gambar 3. 45 Menu *Public_Html*

- Pada menu di atas, klik *upload* maka akan keluar tampilan seperti ini. Lalu pilih *file* kode program yang ingin di-*upload*, bisa juga di lakukan dengan men-*drag file* ke dalam *web hosting*. Untuk lebih mempermudah bisa menggunakan *file .zip* untuk di *upload*. Tunggu proses *upload* hingga selesai



Gambar 3. 46 Menu *Upload File*

- Setelah itu, buat *database* baru di *server hosting* dengan cara memilih menu *MySQL database* yang ada di menu utama *cpanel* dan beri nama

database yang ingin dibuat klik *create*

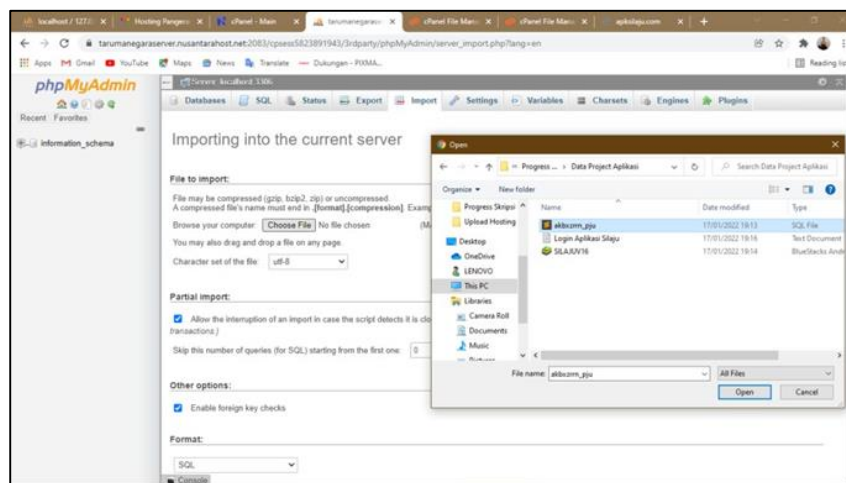


Gambar 3. 46 *Create New Database*

5. Kemudian meng-*upload database* lokal ke *server hosting* dengan cara kembali ke menu utama *cpanel* pilih *phpMyAdmin* klik *import* dan pilih *file* yang memiliki ekstensi *.sql* yang telah di-*export* dari *database* lokal.



Gambar 3. 47 Menu Php Myadmin



Gambar 3. 47 *Import Basis Data Localhost Ke Hosting*

6. Setelah kode program dan *database* berhasil di-*upload* ke *server hosting*, aplikasi silaju dapat diakses secara *online* oleh *user*

3.7.9 Proses Pengujian Kelayakan Aplikasi (Uji Usability)

Pengujian kelayakan aplikasi dilakukan dengan menggunakan metode kuesioner yang diambil melalui *google form* guna mengetahui tingkat

kelayakan dari aplikasi yang telah dibuat. Terdapat 10 pertanyaan dalam kuesioner yang akan diisi oleh 30 responden yang diambil secara acak setelah menggunakan aplikasi. Setiap responden akan menjawab setiap pertanyaan dengan 5 alternatif jawaban yang bisa dipilih, dimana masing-masing jawaban diberi nilai yang mengacu padaskala *likert* dengan rincian sebagai berikut:

- SB (Sangat Baik) = mendapat nilai 5
- B (Baik) = mendapat nilai 4
- C (Cukup) = mendapat nilai 3
- K (Kurang) = mendapat nilai 2
- SK (Sangat Kurang) = mendapat nilai 1

Tabel 3. 10 Tabel Kusioner

NO	PERTANYAAN	JUMLAH RESPONDEN				
		SB	B	C	K	SK
1	Bagaimana tampilan antar muka dari aplikasi eroadnet?	8	12	3	0	0
2	Bagaimana fungsional dari tombol-tombol dan navigasi yang ada pada aplikasi eroadnet	9	8	3	0	0
3	Bagaimana kecepatan dalam mengakses menu dan informasi yang terdapat pada aplikasi eroadnet?	9	8	3	0	0
4	Bagaimana kepuasan dalam penyajian menu dan informasi yang tersedia pada aplikasi eroadnet?	10	7	3	0	0
5	Bagaimana manfaat dari adanya aplikasi eroadnet?	6	11	3	0	0
6	Tanggapan pengguna secara keseluruhan terhadap aplikasi eroadnet?	7	11	2	0	0
7	Bagaimana kepuasan dalam					

	penyajian menu dan informasi yang tersedia pada aplikasi silaju?	10	7	3	0	0
8	Bagaimana kelengkapan fitur-fitur yang ada pada aplikasi eroadnet?	10	7	3	0	0

Setelah diperoleh jawaban responden dari setiap pertanyaan yang diajukan, maka dapat dilakukan perhitungan persentase kelayakan aplikasi menggunakan rumus sebagai berikut:

$$\text{Persentase Kelayakan (\%)} = \frac{\Sigma \text{skor yang diobservasi}}{\text{skor ideal}} \times 100\%$$

Dimana:

$$\text{Skor Ideal} = \Sigma \text{Jumlah Responden} \times \text{Nilai Tertinggi Skala Likert}$$

Dengan total responden sebanyak 20 orang maka akan didapat jumlah skor ideal 100 poin untuk seluruh item pertanyaan (20 responden x 5 poin = 100 poin, jika semua responden menjawab dengan SB). Berdasarkan skor yang telah ditetapkan dapat dihitung persentase kelayakan sebagai berikut:

Untuk mendapatkan nilai total persentase dari semua pertanyaan di atas menggunakan perhitungan rata-rata dengan rumus sebagai berikut Total persentase kelayakan (%):

$$= \frac{\Sigma \text{ presentase nilai pertanyaan}}{\Sigma \text{ pertanyaan}}$$

$$= \frac{82\% + 86\% + 86\% + 87\% + 89\% + 92\% + 81\% + 84\%}{8}$$

$$= 75,875$$