**DOKUMEN LAPORAN**

**GPS MONITORING SISTEM**

LAPORAN

*LSKK-SW-LP-01-GPS*

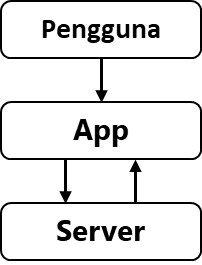


**PT. LANGGENG SEJAHTERA KREASI KOMPUTASI**

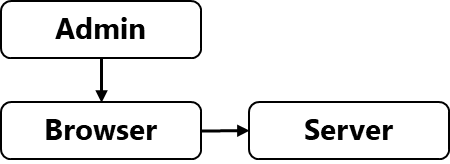
**©2019**

***UPDATE* DESAIN PERANGKAT LUNAK 2019 08 21**

1. **Desain Sistem**

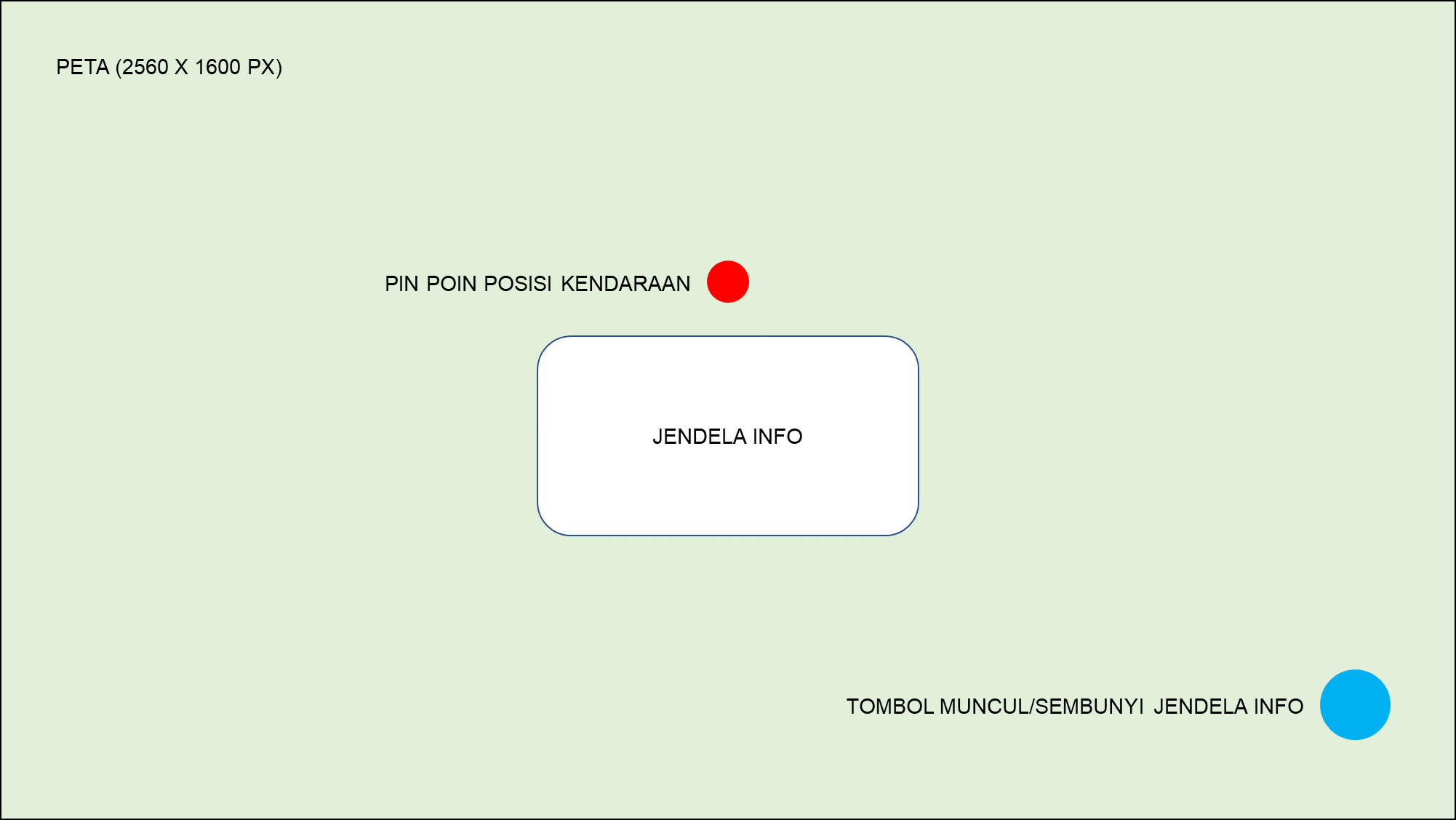
****

Gambar *Update* desain sistem modular diagram pengguna 2019 08 21



Gambar *Update* desain sistem modular diagram admin 2019 08 21

1. **Desain Antarmuka**

****

Gambar 3 *Update* desain perangkat lunak 2019 08 21

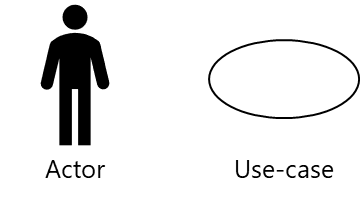
Desain perangkat lunak memiliki empat elemen utama yaitu; Peta, Pin Poin Posisi Kendaraan, Jendela Info, dan Tombol Muncul/Sembunyi dengan keterangan yang dapat dilihat pada Tabel 1.

Tabel 1 Keterangan desain perangkat lunak 2019 08 21

|  |  |
| --- | --- |
| **Elemen** | **Keterangan** |
| Peta (2560 x 1600 px) | Menampilkan peta dengan ukuran resolusi *full res* 2560 x 1600 piksel |
| Pin Poin Posisi Kendaraan | Titik posisi kendaraan yang terpasang GPS Module |
| Jendela Info | Informasi Latitude, Longitude, dan Kecepatan kendaraan |
| Tombol Muncul/Sembunyi Jendela Info | Memunculkan atau menyembunyikan Jendela Info |

***UPDATE* UML PERANGKAT LUNAK 2019 08 21**

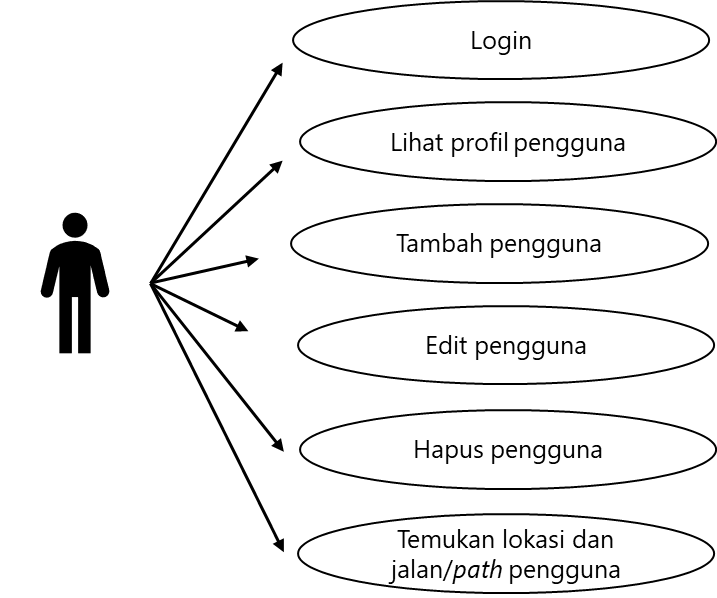
1. ***Use-Case* Diagram**



Gambar *Update* desain simbol *actor* dan *use-case* 2019 08 21

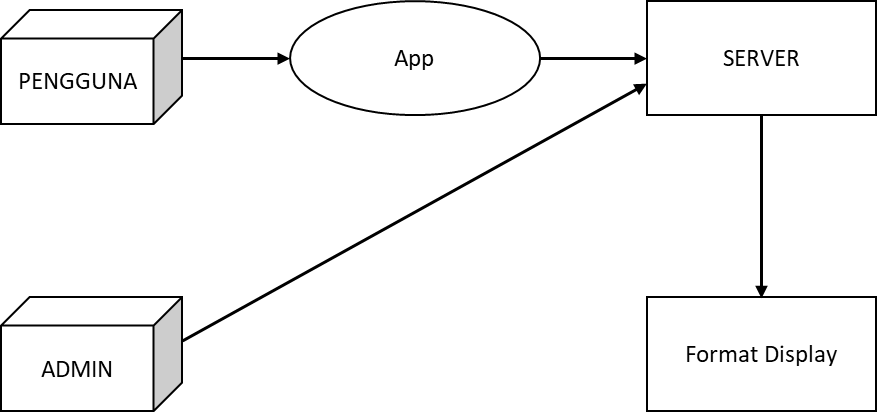


Gambar *Update use-case* diagram pengguna 2019 08 21

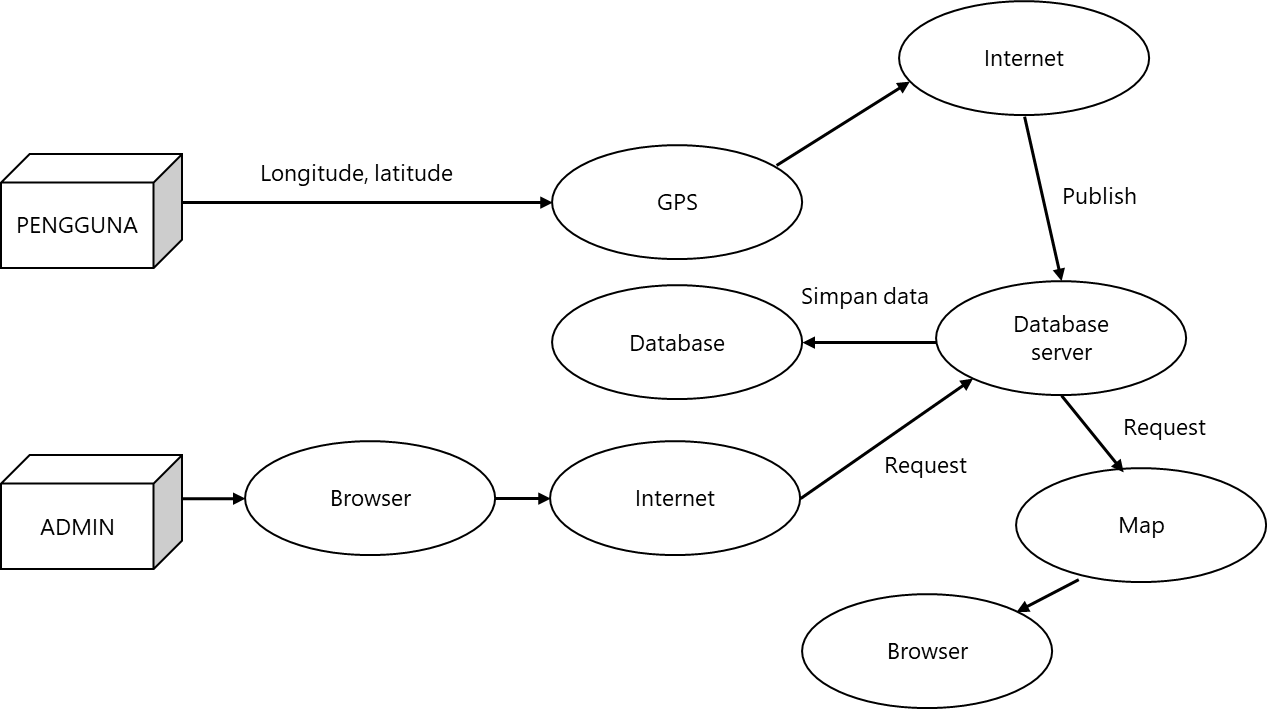


Gambar *Update use-case* diagram admin 2019 08 21

1. ***Data Flow* Diagram**

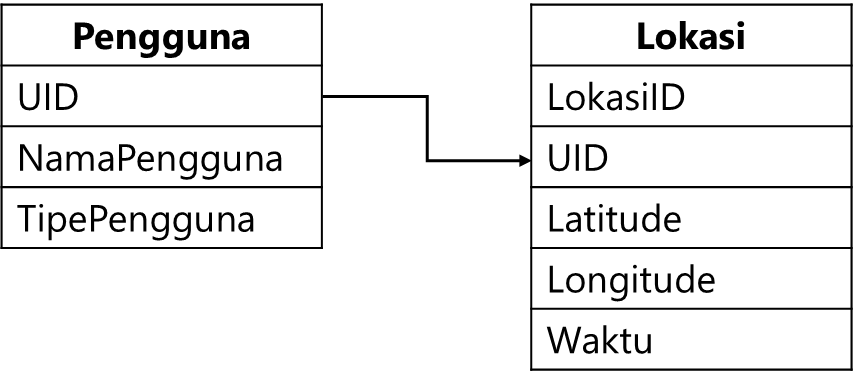


Gambar *Update* level-0 data *flow* 2019 08 21

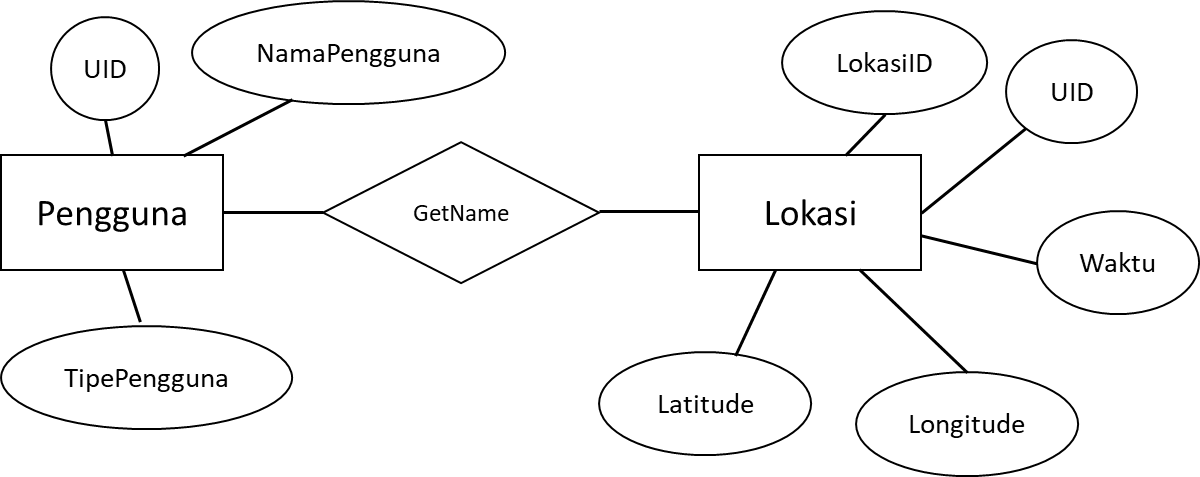


Gambar *Update* level-1 data *flow* 2019 08 21

1. **Skema Diagram**

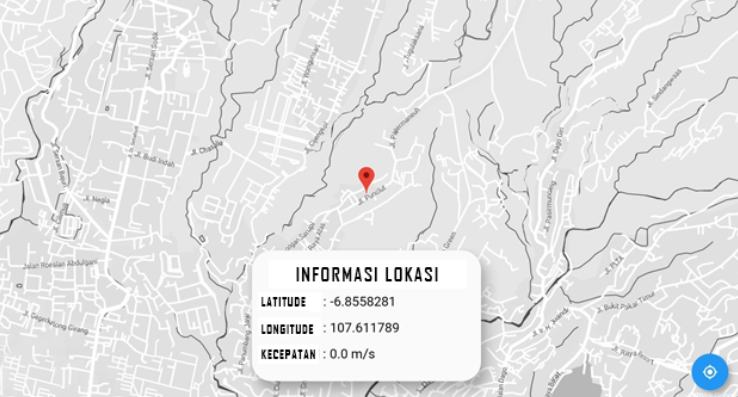


Gambar *Update* skema diagram 2019 08 21



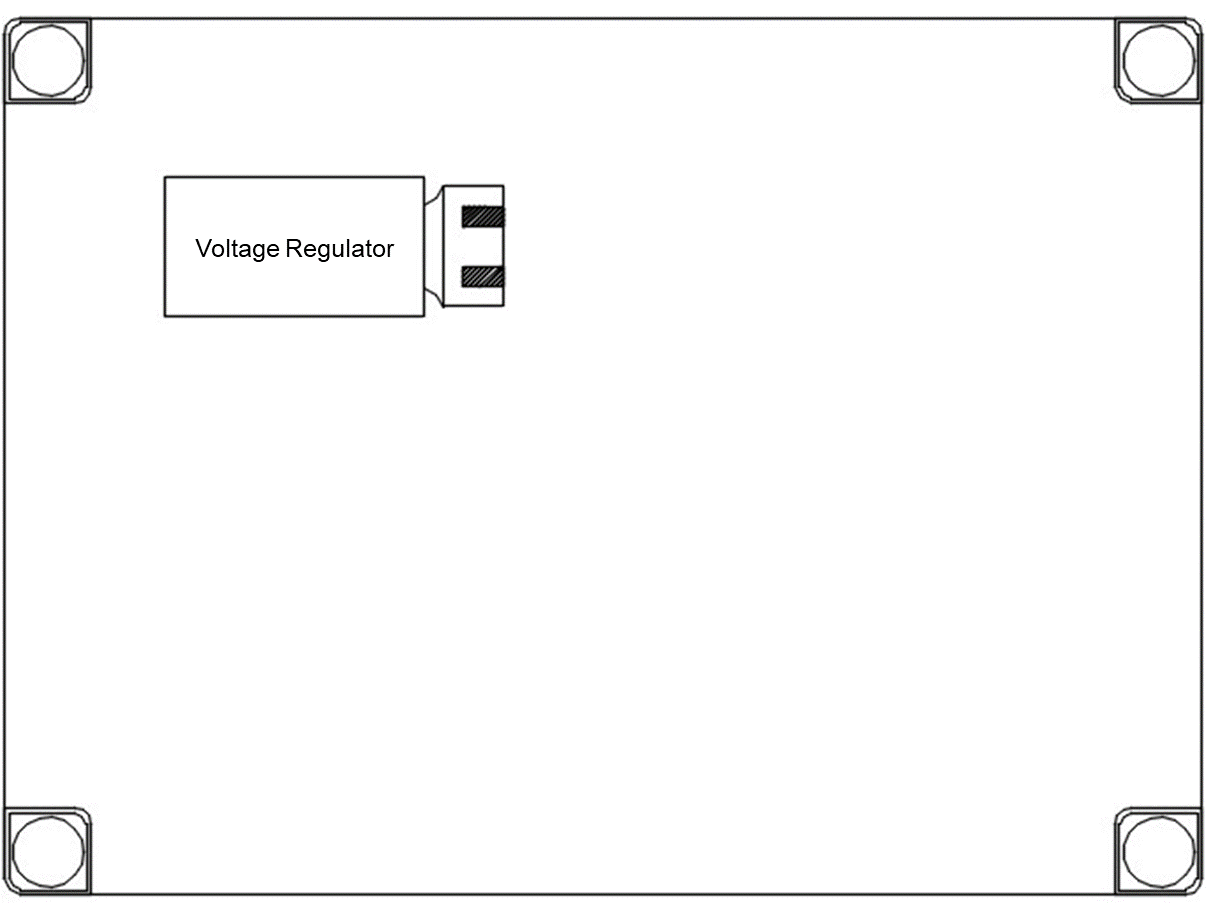
Gambar *Update* ER diagram 2019 08 21

***UPDATE* TAMPILAN PERANGKAT LUNAK 2019 08 21**

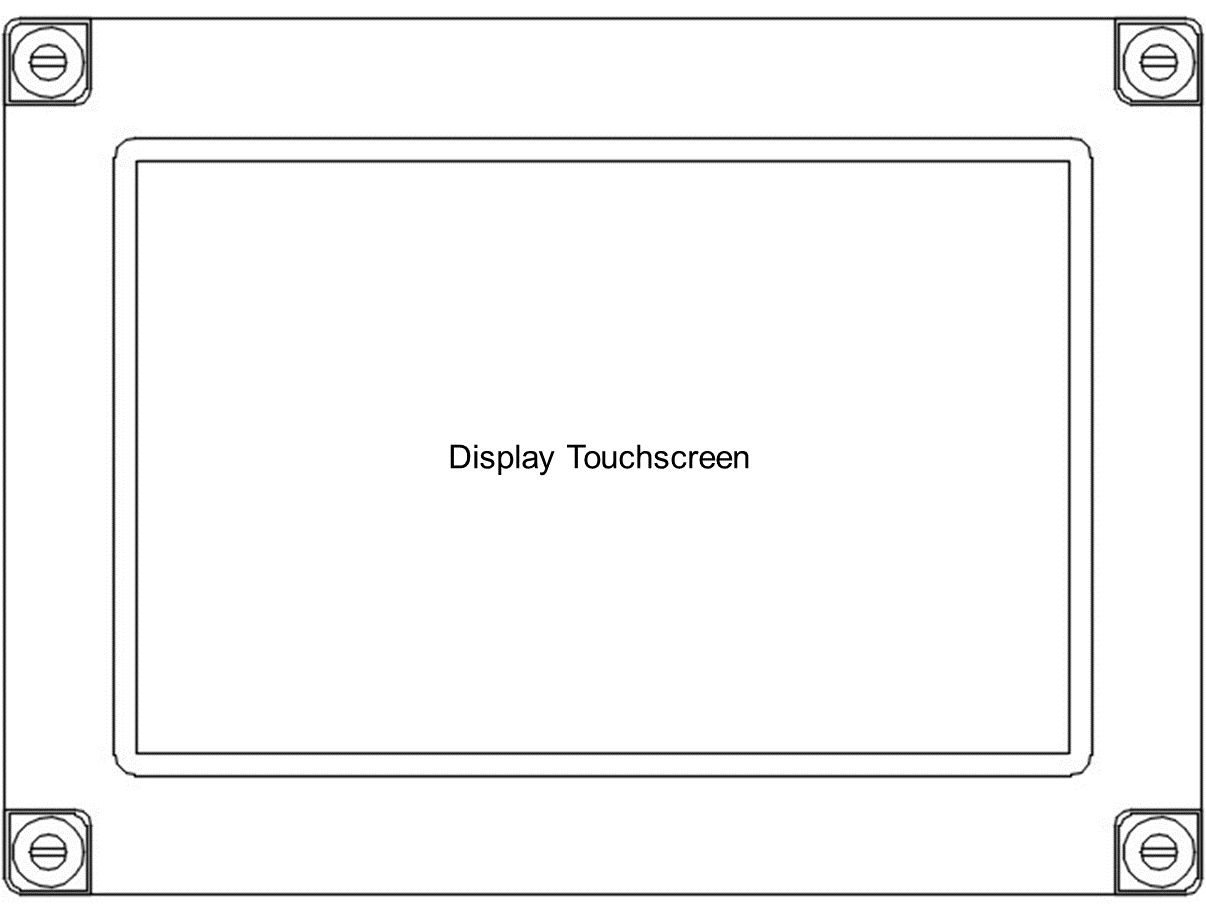


Gambar *Update* tampilan perangkat lunak 2019 08 21

***UPDATE* TAMPILAN PERANGKAT LUNAK 2019 08 21**



Gambar *Update* layout perangkat keras *Voltage Converter* 2019 08 21



Gambar *Update* layout perangkat keras *Display Touchscreen* 2019 08 21

***UPDATE SOURCE CODE* 2019 08 21**

Tabel 2 Baris kode import paket library

|  |
| --- |
| // Import paket library material servis dan peta  import 'package:flutter/material.dart';  import 'package:flutter/services.dart';  import 'package:google\_maps\_flutter/google\_maps\_flutter.dart';  import 'package:location/location.dart'; |

Tabel Baris kode mulai app

|  |
| --- |
| // Mulai app  void main() => runApp(MyApp());  class MyApp extends StatelessWidget {  @override  Widget build(BuildContext context) {  return MaterialApp(  home: MapView(),  );  }  }  class MapView extends StatefulWidget {  @override  \_MapViewState createState() => \_MapViewState();  } |

Tabel Baris kode mendapatkan lokasi

|  |
| --- |
| // Dapatkan lokasi  class \_MapViewState extends State<MapView> {  final Map<String, Marker> \_markers = {};  final CameraPosition punclutPosition = CameraPosition(  target: LatLng(-6.1055922, 107.611744),  zoom: 14.0,  );  GoogleMapController mapController;  Location location = Location();  double lat = 0.0;  double lng = 0.0;  double spd = 0.0;  void onUserLocationChanged() {  location.changeSettings(  accuracy: LocationAccuracy.NAVIGATION,  );  location.onLocationChanged().listen((location) async {  setState(() {  \_markers.clear();  final marker = Marker(  markerId: MarkerId('${location.time.toString()}'),  position: LatLng(location.latitude, location.longitude),  infoWindow: InfoWindow(  title: location.accuracy.toString(),  snippet: '${location.latitude} & ${location.longitude}',  ),  );  \_markers[location.time.toString()] = marker;  mapController.animateCamera(  CameraUpdate.newCameraPosition(  CameraPosition(  target: LatLng(location.latitude, location.longitude),  zoom: 15.0,  ),  ),  );  lat = location.latitude;  lng = location.longitude;  spd = location.speed;  });  });  }  Future getUserLocation() async {  try {  location.changeSettings(  accuracy: LocationAccuracy.NAVIGATION,  interval: 500,  );  var currentLocation = await location.getLocation();  print(currentLocation.latitude);  print(currentLocation.longitude);  } on PlatformException catch (e) {  if (e.code == 'PERMISSION\_DENIED') {  print('Permission Denied');  }  }  } |

Tabel Baris kode menyesuaikan tema peta

|  |
| --- |
| // Sesuaikan tema peta  void changeMapMode() {  getJsonFile("assets/dark\_theme.json").then(setMapStyle);  } |

Tabel Baris kode mendapatkan tema JSON dari asset

|  |
| --- |
| // Dapatkan tema JSON dari Asset  Future<String> getJsonFile(String path) async {  return await rootBundle.loadString(path);  } |

Tabel Baris kode memasang tema peta

|  |
| --- |
| // Pasang tema peta  void setMapStyle(String mapStyle) {  print('map style : $mapStyle');  mapController.setMapStyle(mapStyle);  }  void \_onMapCreated(GoogleMapController controller) {  mapController = controller;  changeMapMode();  onUserLocationChanged();  }  @override  Widget build(BuildContext context) {  return SafeArea(  child: Scaffold(  body: Stack(  children: <Widget>[  googleMap(),  InformationWidget(lat: lat, lng: lng, spd: spd),  ],  ),  floatingActionButton: FloatingActionButton(  child: Icon(Icons.my\_location),  onPressed: () async {  await getUserLocation();  },  ),  ),  );  }  GoogleMap googleMap() {  return GoogleMap(  onMapCreated: \_onMapCreated,  initialCameraPosition: punclutPosition,  markers: \_markers.values.toSet(),  );  }  }  class InformationWidget extends StatelessWidget {  const InformationWidget({  Key key,  @required this.lat,  @required this.lng,  @required this.spd,  }) : super(key: key);  final double lat;  final double lng;  final double spd;  @override  Widget build(BuildContext context) {  return Align(  alignment: Alignment.bottomCenter,  child: Container(  margin: EdgeInsets.all(20.0),  width: 400,  height: 200,  decoration: BoxDecoration(  color: Colors.white70,  borderRadius: BorderRadius.all(  Radius.circular(30.0),  ),  ),  child: Material(  elevation: 14.0,  borderRadius: BorderRadius.all(Radius.circular(30.0)),  child: Container(  padding: EdgeInsets.all(15.0),  child: Column(  children: <Widget>[  Expanded(  child: Text(  'Detailed Information',  style: TextStyle(  color: Colors.black,  fontSize: 25.0,  fontWeight: FontWeight.w600),  ),  ),  SizedBox(  height: 10.0,  ),  Expanded(  child: TextInformation(  title: 'Latitude : ',  content: '$lat',  ),  ),  Expanded(  child: TextInformation(  title: 'Longitude : ',  content: '$lng',  ),  ),  Expanded(  child: TextInformation(  title: 'Speed : ',  content: '$spd m/s',  ),  ),  ],  ),  ),  ),  ),  );  }  } |

Tabel Baris kode jendela info

|  |
| --- |
| // Jendela info  class TextInformation extends StatelessWidget {  final String title;  final String content;  TextInformation({  this.title,  this.content,  });  @override  Widget build(BuildContext context) {  return Column(  children: <Widget>[  Expanded(  child: Row(  children: <Widget>[  Text(  '$title',  style: TextStyle(  color: Colors.black,  fontSize: 20.0,  fontWeight: FontWeight.w400),  ),  Text(  '$content',  style: TextStyle(  color: Colors.black,  fontSize: 20.0,  fontWeight: FontWeight.w400),  ),  ],  ),  ),  SizedBox(  height: 15.0,  ),  ],  );  }  } |