

## LAMPIRAN

### Lampiran 1

#### Script view register

```
<div class="login-area">
  <div class="container">
    <div class="login-box ptb--100">
      <form method='post' action=/register'>
        @csrf
        @include('error_layout.error')
        <div class="login-form-head">
          <h4>Sign up</h4>
          <p>Hello there, Sign up and Join with Us</p>
        </div>
        <div class="login-form-body">
          <div class="form-gp">
            <label for="exampleInputName1">Full Name</label>
            <input type="text" name="username" id="exampleInputName1">
            <i class="ti-user"></i>
          </div>
          <div class="form-gp">
            <label for="exampleInputEmail1">Email address</label>
            <input type="email" name="email" id="exampleInputEmail1">
            <i class="ti-email"></i>
          </div>
          <div class="form-gp">
            <label for="exampleInputPassword1">Password</label>
            <input
              type="password"
              name="password"
              id="exampleInputPassword1">
            <i class="ti-lock"></i>
          </div>
          <div class="form-gp">
            <label for="exampleInputPassword2">Confirm Password</label>
            <input
              type="password"
              name="password_confirmation"
              id="exampleInputPassword2">
            <i class="ti-lock"></i>
          </div>
          <div class="submit-btn-area">
            <button id="form_submit" type="submit">Submit <i class="ti-arrow-
              right"></i></button>
          <div class="login-other row mt-4">
            </div>
          </div>
          <div class="form-footer text-center mt-5">
            <p class="text-muted">Don't have an account? <a href="/">Sign
              in</a></p>
          </div>
        </div>
      </form>
    </div>
  </div>
</div>
@endsection
```

## Kode program proses dan view dashboard

```

public function index()
{
    //Jika yang login adalah Role Admin
    $user_id = "";
    if(session('role_id') == 1 or session('role_id') == 2)
    {
        $user_id = 'admin';
    }else{
        $user_id = session('user_id');
    }
    $user = User::find(session('user_id'));
    $laporan_user = $user->laporans;
    $status_laporan_belum = (new Laporan)->getTotalLaporanByStatus(0, $user_id);
    $status_laporan_telah = (new Laporan)->getTotalLaporanByStatus(1, $user_id);
    $status_laporan_selesai = (new Laporan)->getTotalLaporanByStatus(2, $user_id);
    return view('index', [
        'laporans' => $laporan_user,
        'status_belum' => count($status_laporan_belum),
        'status_telah' => count($status_laporan_telah),
        'status_sudah' => count($status_laporan_selesai)
    ]);
}
@extends('base_layout.app')
@section('title', 'DASHBOARD')
@section('index_content')
    <h2>{{ $status_sudah }}</h2>
    </div>
    <canvas id="seolinechart1" height="50"></canvas>
    </div>
    </div>
    </div>
    <div class="col-md-6 mt-md-5 mb-3">
        <div class="card">
            <div class="seo-fact sbg2">
                <div class="p-4 d-flex justify-content-between align-items-center">
                    <div class="seofct-icon"><i class="ti-share"></i> Masih Di
                </div>
                <h2>{{ $status_telah }}</h2>
            </div>
        </div>
    </div>
    </div>
    @if (auth()->user()->role_id == 2 || auth()->user()->role_id == 1)
    <div class="col-lg-12">
        <div class="card">
            <div class="card-body pb-0">
                <h4 class="header-title">MAP PERSEBARAN KERUSAKAN</h4>
                <div id="map_persebaran" style="height: 800px;"></div>
            </div>
        </div>
    </div>
    @endif

    <div class="col-xl-4 col-lg-4 mt-5">
        <div class="card">

```

“lanjutan”

```
<div class="card-body">
  <h4 class="header-title">Garis Waktu</h4>
  <div class="timeline-area">
    @foreach ($laporans as $laporan)
      <div class="timeline-task">
        @if($laporan->status == 0)
          <div class="icon bg4">
            <i class="fa fa-exclamation-triangle"></i>
          </div>
        @elseif($laporan->status == 1)
          <div class="icon bg1">
            </div>
        </div>
        deskripsilaporan}}</p>
      </div>
    @endforeach
  </div>
</div>
</div>
</div>
</div>
</div>
@endif

</div>
</div>
@endsection
@push('scripts')
<script>
  var legend = L.control({position: 'bottomright'});
  var warna_total = [ '#ff8000', '#adf542', '#22e0bd', '#fc0367'];
  var labels = ['Jalan Primer', 'Jalan Sekunder', 'Jalan Nasional', 'Jalan Residential'];
  var map = L.map('map_persebaran')
    .setView([-6.966667, 110.416664], 12);

  var
    basemap =
    L.tileLayer('https://api.tiles.mapbox.com/v4/{id}/{z}/{x}/{y}.png?access_token=pk.eyJ1IjoibWFwYm94IiwiYSI6ImNpejY4NXVycTA2emYycXBndHRqcmZ3N3gifQ.rJcFIG214AriSLbB6B5aw', {
      attribution: '&copy;',
      href="http://www.openstreetmap.org/copyright">OpenStreetMap</a> &copy; <a
      href="http://cartodb.com/attributions">CartoDB</a>',
      subdomains: 'abcd',
      id: 'mapbox.light',
      maxZoom: 19
    }).addTo(map);

  legend.onAdd = function (map) {
    var div = L.DomUtil.create('div', 'info-legend')
    for (var i = 0; i < warna_total.length; i++) {
      div.innerHTML += '<i style="background:' + warna_total[i] + "'></i> ' + labels[i] +
    <br>';
    }
    return div;
  };

  legend.addTo(map);

  var google_roadmap = new L.Google('ROADMAP');
  var google_hybrid = new L.Google('HYBRID');
```

“lanjutan”

```
var google_satelit = new L.Google('SATELLITE');
var osm = L.tileLayer('http://{s}.tile.osm.org/{z}/{x}/{y}.png', {});
var Esri_WorldImagery =
L.tileLayer('http://server.arcgisonline.com/ArcGIS/rest/services/World_Imagery/MapServer/tile/{z}/{y}/{x}');
var mapbox =
L.tileLayer('https://api.tiles.mapbox.com/v4/{id}/{z}/{x}/{y}.png?access_token=pk.eyJ1IjoibWFwYm94IiwiYSI6Im9pZW56aWZjZjNmRjNzk3ZmE2MTcwOTEwMGY0MzU3YjUzOWFmNWZhIn0.Y8bhBaUMqFiPrDRW9hieoQ', {
    maxZoom: 18,
    id: 'mapbox.streets'
});

getDataLaporan();
residentialHighway();
administration_Kecamatan_Semarang();
primaryHighway();
secondaryHighway();
trunkHighway();

function secondaryHighway(){
    var counties = $.ajax({
        url:
        "https://gist.githubusercontent.com/lintangtimur/8a12ca78fb8ee68b1b4e07e8115adf4/raw/52398a75eda3129f4448fd58932ff8f7f8433e56/secondary_highway_semarang.geojson",
        dataType: "json",
        success: console.log("Secondary Highway Loaded"),
        error: function(xhr) {
            alert(xhr.statusText)
        }
    })
    function onEachFeature(feature, layer) {
        var popupContent = "<p>Name:" + feature.properties.name + "<br>Highway: " +
            feature.properties.highway +
            "<br>Surface: " + feature.properties.surface + "</p>";

        if (feature.properties && feature.properties.popupContent) {
            popupContent += feature.properties.popupContent;
        }

        layer.bindPopup(popupContent);
    }
    function polystyle(feature) {
        return {
            weight: 2.4,
            color: "#adf542",
            opacity: 1
        };
    }

    $.when(counties).done(function() {
        // Add requested external GeoJSON to map
        var kyCounties = L.geoJSON(counties.responseJSON, {
```

“lanjutan”

```
onEachFeature: onEachFeature,
    style: polystyle
  }).addTo(map);
});
}
function primaryHighway(){
  var counties = $.ajax({
    url:
"https://gist.githubusercontent.com/lintangtimur/341f83925c9144e7c01b5c2e8a42ec8d/ra
w/610a35f2b8166c20f0b58c4c3ce23d7f92cefd8a/primary_semarang.geojson",
    dataType: "json",
    success: console.log("ADM Semarang Loaded"),
    error: function(xhr) {
      alert(xhr.statusText)
    }
  })
  function onEachFeature(feature, layer) {
    var popupContent = "<p>Name:"+feature.properties.name + "<br>Highway: "+
      feature.properties.highway +
      "<br>Surface:
"+feature.properties.surface+"</p>";

    if (feature.properties && feature.properties.popupContent) {
      popupContent += feature.properties.popupContent;
    }

    layer.bindPopup(popupContent);
  }
  function polystyle(feature) {
    return {
      weight: 2.4,
      color: "#ff8000",
      opacity: 1
    };
  }

  $.when(counties).done(function() {

    // Add requested external GeoJSON to map
    var kyCounties = L.geoJSON(counties.responseJSON, {
      onEachFeature: onEachFeature,
      style: polystyle
    }).addTo(map);
  });
}
function trunkHighway(){
  var counties = $.ajax({
    url:
"https://gist.githubusercontent.com/lintangtimur/2e0561d8f77e84d25a41233ebd73cb86/ra
w/c1074c339b86034f58f1fb15a922c766cd9625d5/trunk_highway_semarang.geojson",
    dataType: "json",
    success: console.log("Trunk Highway Semarang Loaded"),
    error: function(xhr) {
      alert(xhr.statusText)
    }
  })
}
```

“lanjutan”

```
function onEachFeature(feature, layer) {
    var popupContent = "<p>Name:" + feature.properties.name + "<br>Highway: " +
        feature.properties.highway +
        "<br>Surface: " + feature.properties.surface + "</p>";

    if (feature.properties && feature.properties.popupContent) {
        popupContent += feature.properties.popupContent;
    }

    layer.bindPopup(popupContent);
}

function polystyle(feature) {
    return {
        weight: 2.4,
        color: "#22e0bd",
        opacity: 1
    };
}

$.when(counties).done(function() {
    // Add requested external GeoJSON to map
    var kyCounties = L.geoJSON(counties.responseJSON, {
        onEachFeature: onEachFeature,
        style: polystyle
    }).addTo(map);
});

function residentialHighway(){
    var counties = $.ajax({
        url:
        "https://gist.githubusercontent.com/lintangtimur/e9252c628fc98ec05773f427646dfff5/raw/42fae4015ead11819c5ad85fa278027bc375b509/residential_semarang_geojson",
        dataType: "json",
        success: console.log("Residential Semarang Loaded"),
        error: function(xhr) {
            alert(xhr.statusText)
        }
    })
}

function onEachFeature(feature, layer) {
    var popupContent = "<p>Name:" + feature.properties.name + "<br>Highway: " +
        feature.properties.highway +
        "<br>Surface: " + feature.properties.surface + "</p>";

    if (feature.properties && feature.properties.popupContent) {
        popupContent += feature.properties.popupContent;
    }

    layer.bindPopup(popupContent);
}

function polystyle(feature) {
    return {
        weight: 2.4,
        color: "#fc0367",
        opacity: 1
    };
}
```

“lanjutan”

```
};
}

$.when(counties).done(function() {

    // Add requested external GeoJSON to map
    var kyCounties = L.geoJSON(counties.responseJSON, {
        onEachFeature: onEachFeature,
        style: polystyle
    }).addTo(map);
});
}
function administrationSemarang(){
    var counties = $.ajax({
        url:
        "https://gist.githubusercontent.com/lintangtimur/2b32a7278641c7fb13ff13fea1f0bd/raw/
        bc5f5e59daf0cf1357e9fd76bc162c0822a2eb24/adm_semarang.geojson",
        dataType: "json",
        success: console.log("ADM Semarang Loaded"),
        error: function(xhr) {
            alert(xhr.statusText)
        }
    })
    function onEachFeature(feature, layer) {
        var popupContent = "<p>Name:" + feature.properties.name + "<br>Highway: " +
            feature.properties.highway +
            "<br>Surface: " + feature.properties.surface + "</p>";

        if (feature.properties && feature.properties.popupContent) {
            popupContent += feature.properties.popupContent;
        }

        layer.bindPopup(popupContent);
    }
    function polystyle(feature) {
        return {
            weight: 2.4,
            opacity: 1
        };
    }
}

$.when(counties).done(function() {

    // Add requested external GeoJSON to map
    var kyCounties = L.geoJSON(counties.responseJSON).addTo(map);
});
}

function administration_Kecamatan_Semarang(){
    var counties = $.ajax({
        url:
        "https://gist.githubusercontent.com/lintangtimur/9137f00eb8ebfdc4c9ebd30afcce3982/raw/
        93496517d3316aa2ff993b9f94bd51d6cb1caec2/kecamatan_semarang_tanpamarker.ge
        ojson",
        dataType: "json",
```

“lanjutan”

```
success: console.log("ADM Kecamatan Semarang Loaded"),
  error: function(xhr) {
    alert(xhr.statusText)
  }
})
function onEachFeature(feature, layer) {
  var popupContent = "<p>Kecamatan:"+feature.properties.name+"</p>";

  if (feature.properties && feature.properties.popupContent) {
    popupContent += feature.properties.popupContent;
  }

  layer.bindPopup(popupContent);
}
function polystyle(feature) {
  return {
    weight: 2.4,
    opacity: 1
  };
}

$.when(counties).done(function() {
  // Add requested external GeoJSON to map
  var kyCounties = L.geoJSON(counties.responseJSON, {
    onEachFeature: onEachFeature,
  }).addTo(map);
});

function getDataLaporan(){
  var counties = $.ajax({
    url: "{{route('laporanjson')}}",
    dataType: "json",
    success: console.log("Get loaded point laporans"),
    error: function(xhr) {
      alert(xhr.statusText)
    }
  })
  function onEachFeature(feature, layer) {

    var popupContent =
    src="+feature.properties.imgLaporan[0]+"/><p>Kecamatan:
    "+feature.properties.kecamatan+"</p>";

    if (feature.properties && feature.properties.popupContent) {
      popupContent += feature.properties.popupContent;
    }

    layer.bindPopup(popupContent);
  }
  function polystyle(feature) {
    return {
      weight: 5.4,
      color: "#f242f5",

```



“lanjutan”

```
opacity: 1
  };
}
$.when(counties).done(function() {

  // Add requested external GeoJSON to map
  var kyCounties = L.geoJSON(counties.responseJSON, {
    style: function (feature) {
      return feature.properties && feature.properties.style;
    },
    onEachFeature: onEachFeature,
    style: polystyle,
    pointToLayer: function (feature, latlng) {

      return L.circleMarker(latlng, {
        radius: 8,
        fillColor: "#ff7800",
        color: "#000",
        weight: 1,
        opacity: 1,
        fillOpacity: 0.8
      });
    }
  }).addTo(map);
});

function InitLayerOverlay(){
  var primary = $.ajax({
    url:
    "https://gist.githubusercontent.com/lintangtimur/341f83925c9144e7c01b5c2e8a42ec8d/ra
w/610a35f2b8166c20f0b58c4c3ce23d7f92cefd8a/primary_semarang.geojson",
    dataType: "json",
    error: function(xhr) {
      alert(xhr.statusText)
    }
  })

  var secondary = $.ajax({
    url:
    "https://gist.githubusercontent.com/lintangtimur/8a12ca78fb8ee68b1b4e07e8115adf4/ra
w/52398a75eda3129f4448fd58932ff8f7f8433e56/secondary_highway_semarang.geojson
",
    dataType: "json",
    error: function(xhr) {
      alert(xhr.statusText)
    }
  })

  var trunk = $.ajax({
    url:
    "https://gist.githubusercontent.com/lintangtimur/2e0561d8f77e84d25a41233ebd73cb86/ra
w/c1074c339b86034f58f1fb15a922c766cd9625d5/trunk_highway_semarang.geojson",
    dataType: "json",
    error: function(xhr) {
```

“lanjutan”

```
alert(xhr.statusText)
  }
})

$.when(primary, secondary, trunk).done(function(a,b,c) {

  // Add requested external GeoJSON to map
  var hasilOverlay = L.geoJSON(counties.responseJSON, {
    onEachFeature: onEachFeature,
    style: polystyle
  }).addTo(map);
  var hasilOverlay = L.geoJSON(counties.responseJSON, {
    onEachFeature: onEachFeature,
    style: polystyle
  }).addTo(map);
});
}</script>
@endpush
```

Lampiran 3

Kode program proses laporan detail

```
@push('scripts')
<script>
  getGeolocationPermission();
  $('#btnFindMyPos').on('click', function(){
  })

  $(".btn-tambah").click(function(){
    var html = $(".clone").html();
    $(".increment").after(html);
  });
  $("body").on("click", ".btn-danger",function(){
    $(this).parents(".controlgroup").remove();
  });

  $('button').on('click', function(){
    getGeolocationPermission();
  })

  function getGeolocationPermission()
  {
    navigator.geolocation.getCurrentPosition(
      // Success callback
      function(position) {
        console.log(position)

        */
        var lat = position.coords.latitude
        var lng = position.coords.longitude

        $('#latitude').val(lat)
```

```

        $('#longitude').val(lng)

$.get('https://nominatim.openstreetmap.org/reverse?format=jsonv2&lat='+lat+'&lon='+lng,
function(data){
    $('#tipe_jalan').val(data.type)
    $('#nama_jalan').val(data.address.road)
    })
},

// Optional error callback
function(error){
    console.error(error);
}
);
}

$('#kategori_id').on('change', function(){

    var kategori_id = $(this).val();
    if(kategori_id == 1){
        //Jika Jembatan

$('.field-kerusakan-jembatan').toggleClass('hidden-kerusakan');
        $('.field-kerusakan-jalan').addClass('hidden-kerusakan')
    }else if(kategori_id == 2){
        $('.field-kerusakan-jalan').toggleClass('hidden-kerusakan');
        $('.field-kerusakan-jembatan').addClass('hidden-kerusakan')
    }else{
        $('.field-kerusakan-jembatan').addClass('hidden-kerusakan')
        $('.field-kerusakan-jalan').addClass('hidden-kerusakan')
    }
})

</script>
@endpush

/**
 * menampilkan tampilan detail laporan dari si user
 *
 * @param Request $req
 * @param string $laporan_id laporan_id dari dashboard timeline
 * @return void
 */

public function laporan_detail(Request $req, $laporan_id)
{
    $laporan = Laporan::find($laporan_id);
    $imagelaporan = json_decode($laporan->imagelaporan);

    // WARNING: Cek di table validasi, apakah laporan yang perlu divalidasi?
    $validate = (new Laporan)::find($laporan_id)->validate->where('status_validated', '=',
0);

    $permintaan = "";

```

“lanjutan”

```

$status_validate = false;

$status = [
    0 => 'Belum di proses',
    1 => 'Telah di proses',
    2 => 'Sudah selesai'
];

if (count($validate) > 0) { //----- ada data yang perlu divalidasi
    $permintaan = $status[
        $validate->first()->permintaan
    ];

    $status_validate = true;
}

$data = DB::table('laporans')
->join('users', 'users.user_id', '=', 'laporans.user_id')
->join('kategoris', 'kategoris.kategori_id', '=', 'laporans.kategori_id')
->join('kategori_kerusakans', 'kategori_kerusakans.kode_kerusakan', '=',
'laporans.kode_kerusakan')
->select(
    'laporans.laporan_id',
    'laporans.imagelaporan',
    'users.username',
    'laporans.user_id',
    'laporans.judullaporan',
    'laporans.deskripsilaporan',
    'kategoris.nama_kategori',
    'kategori_kerusakans.kerusakan_kategori',
    'laporans.kecamatan',
    'laporans.nama_jalan',
    'laporans.tipe_jalan',

    'laporans.status',
    'laporans.created_at'
)
->where('laporan_id', '=', $laporan_id)
->first();

// dd(json_decode($laporan->imagelaporan));
return view('user.laporan_detail', [
    'laporan' => $laporan, 'data'=> $data, 'status' => $status[$data->status],
    'status_validate' => $status_validate, 'permintaan' => $permintaan
]);
}

```

“lanjutan”

Lampiran 4

kode program view laporan detail

```

@extends('base_layout.app')

@section('title', 'Laporan Detail | '. $laporan->judullaporan)
@section('laporan_detail')
<div class="container">
    <div class="col-lg-6 col-md-12">

```

“lanjutan”

```
('message' )}
    </div>
  </div>
  @endif
  @include('error_layout.error')
  <h4 class="card-title">Informasi Laporan</h4>

  <div class="table-responsive">
    <table class="table table-
      <button class="btn btn
    </div>
  </div>
  </div>
  </div>
  </div>
  @endsection
  @push('scripts')
  <script>

var map = L.map('maplaporan').setView([{{$laporan->latitude}}, {{$laporan->longitude}},
20);
L.tileLayer('https://{s}.tile.openstreetmap.org/{z}/{x}/{y}.png', {
  attribution: '&copy;
  href="https://www.openstreetmap.org/copyright">OpenStreetMap</a> contributors'
}).addTo(map);

var circle = L.circle([{{$laporan->latitude}}, {{$laporan->longitude}}], {
  color: 'red',
  fillColor: '#f03',
  fillOpacity: 0.5,
  radius: 20
}).addTo(map);
circle.bindPopup("{{ $laporan->nama_jalan }}");
$('#status').on("change", function(){
  var status = $(this).val();
  $('#hidden_status').val(status);
  var laporan_id = $('#laporan_id').val()

  if(status == 1 || status == 2){
    //Jika telah di proses
    $('#fieldUbahStatus').toggleClass('hidden-kerusakan');
    $('#fieldUbahStatus').removeClass('hidden-kerusakan');
  }else{
    $('#fieldUbahStatus').addClass('hidden-kerusakan')
  }
})

$(".btn-tambah").click(function(){
  var html = $(".clone").html();
  $(".increment").after(html);
});
$("body").on("click", ".btn-danger", function(){
  $(this).parents(".controlgroup").remove();
});
</script>
@endpush
```

## kode program getDataLaporanJSON

```

public function getDataLaporan()
{
    $laporan = Laporan::all();
    $kecamatanans = DB::table('kecamatanans')->get();
    $laporan_template = [
        'type' => 'FeatureCollection',
        'features'=> []
    ];

    foreach ($laporan as $key => $value) {
        $imageLaporan = [];
        // iterate image laporan
        foreach (json_decode($value->imagelaporan) as $k => $v) {
            $imageLaporan[] = Storage::url($v->path);
        }

        // Check setiap kecamatan
        $asliKecamatan = "";
        foreach ($kecamatanans as $key => $kecamatan) {
            $marker = [$value->longitude, $value->latitude];
            if (inside_polygon($marker, json_decode($kecamatan->boundary))) {
                $asliKecamatan = $kecamatan->name;
            }
        }

        $laporan_template['features'][] = [
            'geometry'=> [
                'type' => 'Point',
                'coordinates'=> [
                    (float)$value->longitude, (float)$value->latitude
                ],
            ],
            'type' => 'Feature',
            'properties'=> [
                'popupContent'=> $value->judullaporan,
                'imgLaporan' => $imageLaporan,
                'kecamatan' => $asliKecamatan
            ],
            'id'=> $value->laporan_id
        ];
    };
    return response()->json($laporan_template);
}

```

kode program iinside\_polygon, untuk mengecek sebuah (lng,ltid) didalam sebuah polygon.

```

/**
 * check apakah titik ada didalam polygon
 *

```

“lanjutan”

```
* @param array $test_point marker/point
* @param array $points array of polygon
* @return boolean
*/
function inside_polygon($test_point, $points)
{
    $p0 = end($points);

    $ctr = 0;
    foreach ($points as $p1) {
        if ($test_point[1] == $p0[1]) {
            $test_point[1] += 0.0000000001;
        } //epsilon
        if ($p0[1] != $p1[1]) {
            $sinterp = ($test_point[1] - $p0[1]) / ($p1[1] - $p0[1]);
            if ($sinterp >= 0 && $sinterp < 1) {
                $slong = $sinterp * $p1[0] + (1 - $sinterp) * $p0[0];
                if ($slong > $test_point[0]) {
                    // if so, count it:
                    $ctr++;
                }
            }
        }
    }
    $p0 = $p1;
}

return ($ctr & 1);
}
```

kode program tampilan view data report

```
@extends('base_layout.app')
@section('title','Data Report')
@section('datareport_content')
<div class="container">
    <div class="col-lg-6 col-ml-12">
        <div class="card">
            <div class="card-body">
                @if (session('message'))
                <div class="alert-items">
                    <div class="alert alert-primary" role="alert">
                        {{ session('message') }}
                    </div>
                </div>
                @endif
                @include('error_layout.error')
            </div>
        </div>
    </div>
</div>
</div>
</div>
```

Lampiran 7

“lanjutan”

```

<div class="col-12 mt-5">
  <div class="row">
    <div class="col-12 mt-5">
      <div class="card">
        <div class="card-body">
          <h4 class="header-title">Report Master</h4>
          <div class="data-tables">
            <table id="dataTable_datareport" class="text-center">
              <thead class="bg-light text-capitalize">
                <tr>
                  <th>USERNAME</th>
                  <th>Judul Laporan</th>
                  <th>Deskripsi Laporan</th>
                  <th>Kategori</th>
                  <th>Kategori Kerusakan</th>
                  <th>Kecamatan</th>
                  <th>Nama Jalan</th>
                  <th>Tipe Jalan</th>
                  <th>Status</th>
                  <th>Dibuat Tgl</th>
                  <th>Action</th>
                </tr>
              </thead>
              <tbody>
                <tr>
                </tbody>
              </tbody>
            </table>
          </div>
        </div>
      </div>
    </div>
  </div>
</div>
</div>

```



kode program datatable untuk data report

```

@push('scripts')
<script>
$( "#dataTable_datareport" ).DataTable({
  processing: true,
  serverSide: true,
  ajax: '{{route("admin.datareport.json")}}',
  drawCallback: function(settings){
    $(' .adminhapususer' ).click(function(){
      var user_id = $(this).data('user');

      //Pop up konfirmasi untuk delete user
      Swal.fire({
        title: `Apakah anda yakin ingin menghapus ${user_id}?`,
        text: "Anda tidak dapat mengembalikan data ini lagi!",
        type: 'warning',
        showCancelButton: true,
        confirmButtonColor: '#3085d6',

```

Lampiran 8



“lanjutan”

```
cancelButtonColor: '#d33',
confirmButtonText: 'Ya, hapus!'
}).then((result) => {
  if (result.value) {
    $.ajax({
      method: 'post',
      url: '{{route("userdelete")}}',
      data: {
        user_id: user_id,
        _token: $('meta[name="csrf-token"]').attr('content')
      }
    })
  }
})
});
```

```
},
columns: [
  { data: 'username', name: 'username' },
  { data: 'judullaporan', name: 'judullaporan' },
  { data: 'deskripsilaporan', name: 'deskripsilaporan' },
  { data: 'nama_kategori', name: 'nama_kategori' },
  { data: 'kerusakan_kategori', name: 'kerusakan_kategori' },
  { data: 'kecamatan', name: 'kecamatan' },
  { data: 'nama_jalan', name: 'nama_jalan' },
  { data: 'tipe_jalan', name: 'tipe_jalan' },
  { data: 'status', name: 'status' },
  { data: 'created_at', name: 'created_at' },
  { data: 'action', name: 'action' }
]
});
</script>
@endpush
```

kode program admin.datareport.json

```
/**
 * json data
 *
 * @return void
 */
public function reportallJson()
{
    $data = DB::table('laporans')
        ->join('users', 'users.user_id', '=', 'laporans.user_id')
        ->join('kategoris', 'kategoris.kategori_id', '=', 'laporans.kategori_id')
        ->join('kategori_kerusakans', 'kategori_kerusakans.kode_kerusakan', '=',
'laporans.kode_kerusakan')
        ->select(
            'laporans.laporan_id',
            'users.username',
            'laporans.user_id',
            'laporans.judullaporan',
            'laporans.deskripsilaporan',
```

Lampiran 9

“lanjutan”

```
'kategoris.nama_kategori',
  'laporans.kecamatan',
  'laporans.nama_jalan',
  'kategori_kerusakans.kerusakan_kategori',
  'laporans.tipe_jalan',
  'laporans.status',
  'laporans.created_at'
)
->get();

return Datatables::of($data)
->editColumn('status', function ($data) {
  $status = [
    0 => 'Belum di proses',
    1 => 'Telah di proses',
    2 => 'Sudah selesai'
  ];

  return $status[$data->status];
})
->addColumn('action', function ($data) {
  return "<a class=btn btn-primary href='\" . route('laporandetail', $data-
>laporan_id) . \">Lihat & Ubah status</a>";
})
->rawColumns(['action'])
->make(true);
}
```

Lampiran 10

kode program chart statistik tampilan operator

```
@push('scripts')
<script>

init_stat_kecamatan();
init_stat_tipe_jalan();
init_stat_nama_jalan();
init_stat_kategori_kerusakan();
function init_stat_kecamatan(){
  $.ajax({
    url: '{{route("admin.statistic.json")}}',
    method: 'get',
    type: 'json'
  }).done(function(r){
    am4core.ready(function() {
      am4core.useTheme(am4themes_animated);
      var chart = am4core.create("chartdiv", am4charts.XYChart3D);
      chart.data = r
      let categoryAxis = chart.xAxes.push(new am4charts.CategoryAxis());

function init_stat_tipe_jalan(){
  $.ajax({
    url: '{{route("admin.statistic.tipejalanjson")}}',
    method: 'get',
    type: 'json'
```

“lanjutan”

```
}).done(function(r){
  am4core.ready(function() {
    am4core.useTheme(am4themes_animated);
    var chart = am4core.create("chartdiv_tipejalan", am4charts.PieChart3D);
    chart.hiddenState.properties.opacity = 0; // this creates initial fade-in

    chart.data = r
    chart.innerRadius = am4core.percent(40);
    chart.depth = 120;

    chart.legend = new am4charts.Legend();

    var series = chart.series.push(new am4charts.PieSeries3D());
    series.dataFields.value = "total";
    series.dataFields.depthValue = "total";
    series.dataFields.category = "tipe_jalan";
    series.slices.template.cornerRadius = 5;
    series.colors.step = 3;
    series.colors.list = [
      am4core.color("#845EC2"),
      am4core.color("#D65DB1"),
      am4core.color("#FF6F91"),
      am4core.color("#FF9671"),
      am4core.color("#FFC75F"),
      am4core.color("#F9F871"),
    ];
  });
}).fail(function(err){
  console.log("Error init stat tipe jalan")
  console.log(err)
})
}

function init_stat_nama_jalan(){
  $.ajax({
    url: '{{route("admin.statistic.namajalanjson")}}',
    method: 'get',
    type: 'json'
  }).done(function(r){
    am4core.ready(function() {
      let categoryAxis = chart.xAxes.push(new am4charts.CategoryAxis());
      categoryAxis.dataFields.category = "nama_jalan";
      categoryAxis.renderer.labels.template.rotation = 270;
      categoryAxis.renderer.labels.template.hideOversized = false;
      categoryAxis.renderer.minGridDistance = 20;
      categoryAxis.renderer.labels.template.horizontalCenter = "right";
      categoryAxis.renderer.labels.template.verticalCenter = "middle";
      categoryAxis.tooltip.label.rotation = 270;
      categoryAxis.tooltip.label.horizontalCenter = "right";
      categoryAxis.tooltip.label.verticalCenter = "middle";

      let valueAxis = chart.yAxes.push(new am4charts.ValueAxis());
      valueAxis.title.text = "Jumlah Laporan";
      valueAxis.title.fontWeight = "bold";
    });
  });
}
```

“lanjutan”

```
// Create series
var series = chart.series.push(new am4charts.ColumnSeries3D());
series.dataFields.valueY = "total";
series.dataFields.categoryX = "nama_jalan";
series.name = "Visits";
series.tooltipText = "{categoryX}: [bold]{valueY}[/]";
series.columns.template.fillOpacity = .8;

var columnTemplate = series.columns.template;
columnTemplate.strokeWidth = 2;
columnTemplate.strokeOpacity = 1;
columnTemplate.stroke = am4core.color("#FFFFFF");

columnTemplate.adapter.add("fill", function(fill, target) {
    return chart.colors.getIndex(target.dataItem.index);
})

columnTemplate.adapter.add("stroke", function(stroke, target) {
    return chart.colors.getIndex(target.dataItem.index);
})

chart.cursor = new am4charts.XYCursor();
chart.cursor.lineX.strokeOpacity = 0;
chart.cursor.lineY.strokeOpacity = 0;
});
}).fail(function(err){
    console.log("Error init stat nama jalan")
    console.log(err)
})
}

function init_stat_kategori_kerusakan()
{
    $.ajax({
        url: '{route("admin.statistic.kategorikerusakanjson")}',
        method: 'get',
        type: 'json'
    }).done(function(r){
        am4core.ready(function() {
            am4core.useTheme(am4themes_animated);
            var chart = am4core.create("chartdiv_kategori_kerusakan", am4charts.PieChart3D);
            series.colors.list = [
                am4core.color("#92e319"), //kecil
                am4core.color("#5250a"),
                am4core.color("#28130"),
                am4core.color("#28c71a"),
            ];

        });
    }).fail(function(err){
        console.log("Error init stat tipe jalan")
        console.log(err)
    })
}
</script>
```

**KUSIONER APLIKASI SISTEM INFORMASI  
GEOGRAFIS PENGADUAN KERUSAKAN JALAN DAN  
JEMBATAN KOTA SEMARANG**

**Section 1 - Pertanyaan Demografi**

1. Umur : \_\_\_\_\_ (A)  
 < 19 tahun (1)  19 tahun (2)  20 tahun (3)  
 21 tahun (4)  22 tahun (5)  23 tahun (6)  > 23 tahun (7)
  
2. Jenis Kelamin : \_\_\_\_\_ (G)  
 Laki - laki (1)  Perempuan (2)



## Section 2 – Pengujian Hipotesa

Pilih jawaban yang menurut anda sesuai dengan memberikan tanda (X).

1 = Sangat Tidak Setuju

2 = Tidak Setuju

3 = Netral

4 = Setuju

5 = Sangat Setuju

No	Pernyataan	1	2	3	4	5
1	Aplikasi sistem informasi geografis pengaduan jalan dan jembatan Kota Semarang akan membantu saya memahami kerusakan lebih dalam (PE1)					
2	Dengan menggunakan aplikasi sistem informasi geografis pengaduan jalan dan jembatan Kota Semarang membantu saya melaporkan kerusakan dengan cepat. (PE2)					
3	Jika saya menggunakan aplikasi sistem informasi geografis pengaduan kerusakan jalan dan jembatan Kota Semarang, maka akan meningkatkan hasil yang maksimal. (PE3)					
4	Belajar menggunakan aplikasi sistem informasi geografis pengaduan jalan dan jembatan Kota Semarang mudah bagi saya . (EE1)					
5	Mudah bagi saya untuk menjadi terampil dalam menggunakan sistem informasi geografis pengaduan kerusakan jalan dan jembatan Kota Semarang. (EE2)					
6	Saya menemukan sistem aplikasi informasi geografis pengaduan kerusakan jalan dan jembatan Kota Semarang mudah digunakan. (EE3)					

7	Orang-orang yang penting bagi saya berpikir bahwa saya harus menggunakan aplikasi sistem informasi geografis pengaduan jalan dan jembatan Kota Semarang untuk membantu proses pelaporan kerusakan. (SI1)					
8	Orang yang memengaruhi perilaku saya berpikir bahwa saya harus menggunakan aplikasi sistem informasi geografis pengaduan jalan dan jembatan Kota Semarang untuk membantu proses pelaporan kerusakan. (SI2)					
9	Orang-orang yang pendapatnya saya hargai lebih suka menggunakan aplikasi sistem informasi geografis pengaduan jalan dan jembatan Kota Semarang untuk membantu proses pelaporan kerusakan.(SI3)					
10	Saya puas dengan kemampuan aplikasi sistem informasi geografis pengaduan kerusakan jalan dan jembatan Kota Semarang untuk membantu proses pengaduan. (PU1)					
11	Saya puas dengan pengalaman menggunakan aplikasi sistem informasi geografis pengaduan kerusakan jalan dan jembatan Kota Semarang untuk membantu proses pengaduan. (PU2)					
12	Saya memutuskan untuk tetap menggunakan sistem informasi geografis pengaduan kerusakan jalan dan jembatan Kota Semarang karena merasa puas. (PU3)					
13	Saya berencana menggunakan sistem informasi geografis pengaduan kerusakan jalan dan jembatan Kota Semarang untuk infrastruktur yang lebih baik. (BI1)					

14	Saya berniat akan menggunakan sistem informasi geografis pengaduan kerusakan jalan dan jembatan Kota Semarang untuk membantu kelayakan infrastruktur kedepannya. (BI2)					
15	Saya akan selalu menggunakan sistem informasi geografis pengaduan kerusakan jalan dan jembatan Kota Semarang untuk pelaporan infrasuktur jalan dan jembatan. (BI3)					





Submission author:  
15n10020 STEFANUS LINTANG TIMUR A.P.

Check ID:  
13643825

Check date:  
27.10.2019 12:38:07 GMT+0

Check type:  
Doc vs Internet + Library

Report date:  
28.10.2019 04:18:24 GMT+0

User ID:  
30361



File name: 15.n1.0020\_Stefanus Lintang Timur Aji Pamungkas.docx

File ID: 17883151 Page count: 36 Word count: 6059 Character count: 43592 File size: 2.55 MB

## 5.66% Matches

Highest match: 1.8% with source [http://digilib.uin-suka.ac.id/22261/2/12651075\\_BAB-I\\_IV-atau-V\\_DAFTAR-PUSTAKA.pdf](http://digilib.uin-suka.ac.id/22261/2/12651075_BAB-I_IV-atau-V_DAFTAR-PUSTAKA.pdf)

4.44% Internet Matches

12

Page 38

1.72% Library matches

26

Page 38

## 0.92% Quotes

Quotes

2

Page 39

No references found

## 0% Exclusions

No exclusions found

## Replacement

No replaced characters found