

**LAPORAN TUGAS MATA KULIAH
GEOGRAPHIC INFORMATION SYSTEM**

Penerapan Mapnik Python Dalam Pembuatan Peta Area Pulau Nusa Penida

DWI SETIAWAN

04315048

Teknik Informatika



**FAKULTAS ILMU KOMPUTER
UNIVERSITAS NAROTAMA SURABAYA
2018**

Penerapan Mapnik Python Dalam Pembuatan Peta Area Pulau Nusa Penida

Pendahuluan

Nusa Penida adalah sebuah gugusan kepulauan kecil yang terletak disebelah tenggara Bali yang dipisahkan oleh Selat Badung dan berbatasan langsung dengan Samudra Hindia atau yang bisa dikatakan juga sebagai salah satu pulau terluar Indonesia. Nusa Penida terdiri dari 3 pulau utama yang memiliki penghuni yakni Pulau Nusa Penida, Pulau Lembongan, dan Pulau Ceningan dan tidak hanya itu masih banyak juga kepulauan yang memiliki luas lebih kecil dari pulau utama yang terletak di sebelah selatan Pulau Nusa Penida.

GIS atau *Geographic Information System* merupakan mata kuliah wajib yang memiliki 2 sks yang memvisualisasikan atau mempresentasikan dunia nyata berbasis komputer sebagaimana sama halnya dengan visualisasi dunia nyata kedalam sebuah lembaran kertas. Kelebihan dari GIS sendiri salah satunya yaitu lebih fleksibel daripada visualisasi yang menggunakan lembaran kertas. Objek-objek yang akan direpresentasikan yaitu berupa peta atau map, contohnya seperti sungai, area daratan, area lautan, jalan raya, dan lain-lain.

Berdasarkan latar belakang yang telah dipaparkan pada paragraf diatas, maka pada proyek ujian praktikum ini, penulis mengangkat judul **“Penerapan Mapnik Python Dalam Pembuatan Peta Area Pulau Nusa Penida”** sebagai proyek ujian praktikum matakuliah GIS. Pada proyek ini akan dibangun sebuah Sistem Informasi Geografis mengenai area Pulau Nusa Penida.

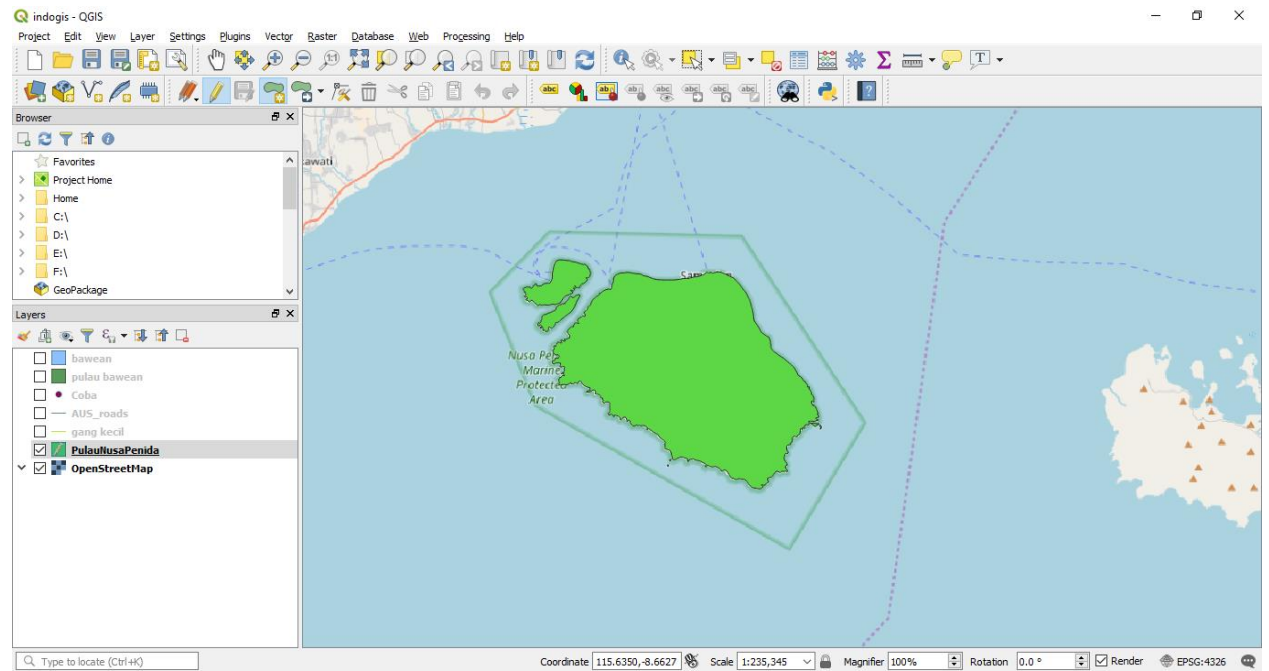
Bahan dan Metode

Dalam menyelesaikan praktikum ini penulis menggunakan beberapa program beserta tools pendukungnya yang semuanya dijalankan pada *Operation System Windows 7 64 bit*. Dalam *coding* pemrograman penulis menggunakan bahasa pemrograman *Python* versi 2.7.14 yang sudah *include* dengan *PIP (Python Instalaler Package)* versi 10.0.1 yang berfungsi untuk pemasangan *tools* pendukung pada *python*, *Mapnik* versi 2.2.0 sebagai *framework python* pendukung pemrograman geografis, *QGis Desktop* 3.0.0 yang digunakan untuk pembuatan *Shapefile* area Pulau Nusa Penida dengan mengatur *Coordinate Reference System* pada *WGS 84* yang memiliki *Authority Id* *EPSG:4326*, *Sublime Text 3* sebagai *Text Editor* nya, *Google Chrome* untuk *view map* area Pulau Nusa Penida, *PostgreSQL database*, *PostGIS* sebagai wadah pendukung dalam pemrosesan data geometri database *PostgreSQL*, *XAMPP*, serta *Command Prompt* untuk melakukan *rendering*.

Metode pelaksanaannya dilakukan pada tanggal 19 dan 20 April 2018 bertempat di Kampus Universitas Narotama Surabaya dengan agenda dihari pertama yaitu, *meng-create vector polygon* area Pulau Nusa Penida yang kemudian disimpan dalam format layer *shapefile (.shp)* lengkap beserta atributnya menggunakan *QGIS Desktop* versi 3.0.0. dan agenda di hari kedua yaitu *import data shapefile* Pulau Nusa Penida kedalam database *PostgreSQL* menggunakan *PostGIS*, kemudian dilanjutkan *coding* pemrograman-nya dan setelah itu dilanjutkan pada proses *render map* menggunakan *flask* dan menampilkannya ke *web server*.

Hasil Dan Pembahasan

Pembuatan *Shapefile* area Pulau Nusa Penida menggunakan *QGis Desktop* 3.0.0, adapun tampilannya sebagai berikut :



Gambar 1. Tampilan area Pulau Nusa Penida

Coding dikerjakan menggunakan *Sublime Text 3* sebagai *text* editor nya dan *python* versi 2.7.14 sebagai bahasa pemrogramannya. Adapun tampilan *coding*-nya seperti berikut :

```

1  from __future__ import division
2  from collections import namedtuple
3  import math
4
5  from flask import Flask, Blueprint, render_template
6
7  import mapnik
8
9  tileserver = Blueprint('tiles', __name__)
10 web_ui = Blueprint('web_ui', __name__, template_folder='templates')
11
12 POSTGIS_TABLE = dict(
13     host='localhost',
14     port=5432,
15     user='postgres',
16     password='12345',
17     dbname='kelasgis',
18     table='(select ST_MPolyFromText(ST_AStext(geom)) as geom, nama from NusaPenida) as NusaPenida')
19
20 LAYER_NAME = 'pantal'
21
22 WGS84 = 'proj4=longlat +ellps=WGS84 +towgs84=0,0,0,0,0,0,0 +no_defs' # 4326
23 # UTM32N = 'proj4=utm +zone=32 +datum=WGS84 +units=m +no_defs' # 32632
24
25 # Google Mercator - EPSG:900913
26 # GOOGLE_MERC = ('proj4=merc +a=6378137 +b=6378137 +lat_ts=0.0 +lon_0=0.0 '
27 #               ' +srs=EPSG:900913 +units=m +nadgrids=@null +no_defs')
28 DATA_PROJECTION = mapnik.Projection(WGS84)
29
30 TILE_WIDTH = 256 # Square tiles only!
31
32
33 minmax = lambda val, lower, upper: min(max(val, lower), upper)
34
35 tile_coords = namedtuple('TileCoords', 'x,y')
36 geo_coords = namedtuple('Coords', 'lat,lon')
37
38
39 def deg2num(lat_deg, lon_deg, zoom):
40     """Convert coordinates to tile number"""
41
42     lat_rad = math.radians(lat_deg)
43     n = 2.0 ** zoom
44     xtile = int((lon_deg + 180.0) / 360.0 * n)
45     ytile = int(
46         (1.0 - math.log(math.tan(lat_rad) + (1 / math.cos(lat_rad)))
47          / math.pi) / 2.0 * n)
48     return tile_coords(x=xtile, y=ytile)
49
50
51 def num2deg(xtile, ytile, zoom):
52     """Convert tile number to coordinates (of the upper corner)"""
53
54     n = 2.0 ** zoom
55     lon_deg = xtile * n * 360.0 / 180.0
56     lat_rad = math.atan(math.sinh(math.pi * (1 - 2 * ytile / n)))
57     lat_deg = math.degrees(lat_rad)
58     return mapnik.Coord(y=lat_deg, x=lon_deg)
59
60
61 class TiledMapRenderer(object):
62     def __init__(self, mapobj):
63         self.m = mapobj
64
65     def render_tile(self, z, x, y):
66         """
67         :param z: Zoom level
68         :param x: Tile horizontal position
69         :param y: Tile vertical position
70         """
71
72         topleft = num2deg(x, y, z)
73         bottomright = num2deg(x + 1, y + 1, z)
74
75         # Bounding box for the tile
76         bbox = mapnik.Box2d(topleft, bottomright)
77
78         bbox = DATA_PROJECTION.forward(bbox)
79
80         print("Bounding box: ", bbox)
81
82         # self.m.resize(TILE_WIDTH, TILE_WIDTH)
83         self.m.zoom_to_box(bbox)
84
85         MIN_BUFFER = 256
86         self.m.buffer_size = max(self.m.buffer_size, MIN_BUFFER)
87
88         # Render image with default Agg renderer
89         im = mapnik.Image(TILE_WIDTH, TILE_WIDTH)
90         mapnik.render(self.m, im)
91         return im
92
93
94 @web_ui.route('/')
95 def index():
96     return render_template('leaflet_map.html')
97
98
99 @tileserver.route('/<layer>/<int:z>/<int:x>/<int:y>.png')
100 def render_tile(layer, z, x, y):
101     """
102     Render the tile using mapnik.
103     """
104
105     m = mapnik.Map(TILE_WIDTH, TILE_WIDTH)
106
107     s = mapnik.Style()
108     r = mapnik.Rule()
109
110     # line symbolizer = mapnik.LineSymbolizer()
111     line_symbolizer = mapnik.PolygonsSymbolizer()
112     line_symbolizer.fill = mapnik.Color('yellow')
113     line_symbolizer.fill_opacity = 0.7
114     # line symbolizer.allow_overlap = True
115     # line_symbolizer.transform = Scale(5,5)
116     r.symbols.append(line_symbolizer)
117
118     label = mapnik.TextSymbolizer(mapnik.Expression('[name]'), 'Dejavu Sans Book', 24, mapnik.Color('red'))
119
120     label.halo_radius = 1
121     label.avoid_edges = False
122     r.symbols.append(label)
123
124     s.rules.append(r)
125
126     m.append_style('My Style', s)
127
128     # Initialize layer from PostGIS table
129     ds = mapnik.PostGIS(POSTGIS_TABLE)
130     layer2 = mapnik.Layer(LAYER_NAME)
131     layer2.datasource = ds
132     layer2.styles.append('My Style')
133     m.layers.append(layer2)
134
135     m.zoom_all()
136
137     # ds = mapnik.Shapefile(file='C:\Users\ICS\Documents\kullah\gis\gis mapnik\pantal indonesia\IND_PNT_polyline.shp')
138     # layer = mapnik.Layer(LAYER_NAME)
139     # layer.datasource = ds
140     # layer.styles.append('My Style')
141     # m.layers.append(layer)
142     # m.zoom_all()
143
144     renderer = TiledMapRenderer(m)
145     im = renderer.render_tile(z, x, y)
146     # m.layers.append(layer)
147     # m.zoom_all()
148
149     renderer = TiledMapRenderer(m)
150     im = renderer.render_tile(z, x, y)
151
152     # im = mapnik.Image(TILE_WIDTH, TILE_WIDTH)
153     # mapnik.render(m, im, 15, 0, 0)
154     # im.save('tmp/b1a.png')
155     # mapnik.render_to_file(m, 'aaa.png', 'png')
156     return im.tostring('png'), 200, {'Content-type': 'image/png'}
157
158
159 def make_app():
160     app = Flask('tileserver')
161     app.register_blueprint(tileserver, url_prefix='/tiles')
162     app.register_blueprint(web_ui, url_prefix='')
163     return app
164
165
166 if __name__ == '__main__':
167     app = make_app()
168     app.run(debug=True, port=5000)

```

Gambar 2. Tampilan Source code proyek

Import database dan Rendering yang dilakukan menggunakan command prompt. Adapun tampilannya sebagai berikut :

```
The filename, directory name, or volume label syntax is incorrect.

C:\WINDOWS\system32>cd C:\Program Files\PostgreSQL\9.6\bin

C:\Program Files\PostgreSQL\9.6\bin>shp2pgsql -d -s SRID "D:\Kuliah\6\GIS (Sistem Informasi Geografis)\ECP 1\NusaPenida\
PulauNusaPenida.shp" public.NusaPenida | psql -h localhost -U postgres -d kelasgis
Shapefile type: Polygon
Postgis type: MULTIPOLYGON[2]
Password for user postgres: 12345

SET
SET
      dropgeometrycolumn
-----
public.nusapenida.geom effectively removed.
(1 row)

DROP TABLE
BEGIN
CREATE TABLE
ALTER TABLE
      addgeometrycolumn
-----
public.nusapenida.geom SRID:0 TYPE:MULTIPOLYGON DIMS:2
(1 row)

INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
COMMIT
ANALYZE

C:\Program Files\PostgreSQL\9.6\bin>
```

Gambar 3. Tampilan proses *Import database*

```
Microsoft Windows [Version 10.0.16299.371]
(c) 2017 Microsoft Corporation. All rights reserved.

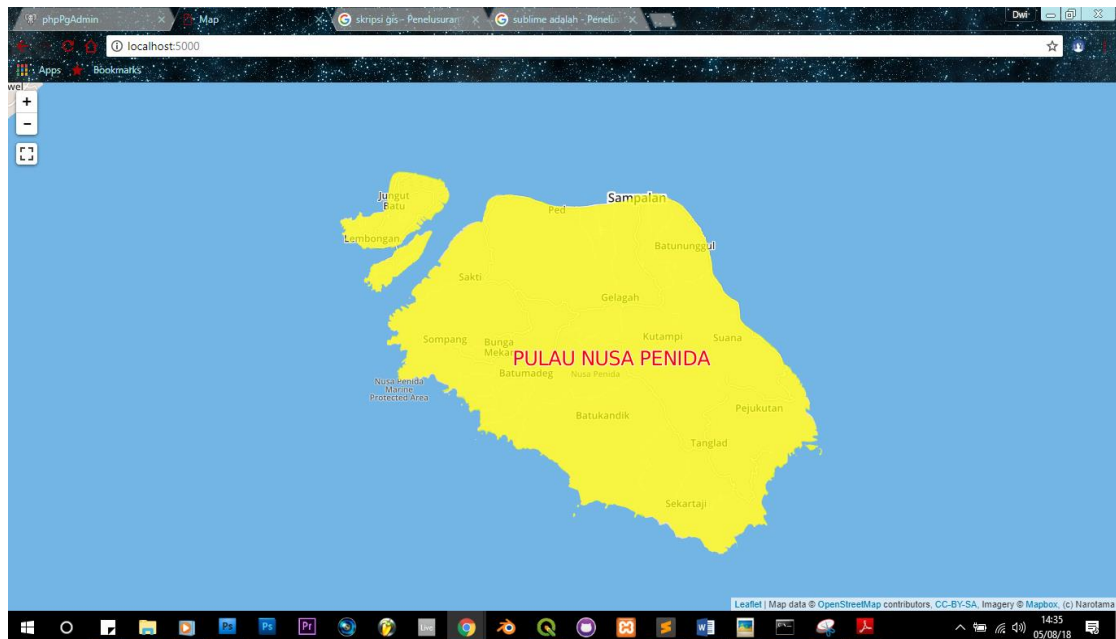
C:\WINDOWS\system32>cd d:
D:\

C:\WINDOWS\system32>d:
D:\>cd Kuliah\6\GIS (Sistem Informasi Geografis)\ECP 1\render-tiles-master

D:\Kuliah\6\GIS (Sistem Informasi Geografis)\ECP 1\render-tiles-master>python tileserver.py
* Restarting with stat
* Debugger is active!
* Debugger PIN: 213-188-223
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```

Gambar 4. Tampilan proses *Rendering* proyek

Hasil *import* database dan *rendering* di tes menggunakan Google Chrome melalui server *localhost:5000*.



Gambar 5. Tampilan map area Pulau Nusa Penida

Daftar Pustaka

- [1] I. . E. Swastikayana, “Sistem Informasi Geografis Web untuk pemetaan pariwisata kabupaten gianyar”, 2011.
- [2] Wikipedia.(2018, April.28) Nusa Penida [online] Available :
https://id.wikipedia.org/wiki/Nusa_Penida
- [3] Wikipedia.(2018, April.30) *Geographic Information System* [online] Available :
https://en.wikipedia.org/wiki/Geographic_information_system

Lampiran

LOGBOOK

Tools yang digunakan :

- OS WINDOWS 10
- Python versi 2.7.14
- Mapnik versi 2.2.0
- QGis versi 3.0.0

Kendala yang dihadapi :

- Instalasi packages libmapnik pada environment mapnik di linux, dan beberapa kendala lainnya yang lumayan ribet dan dengan performa laptop yang kurang maksimal dikarenakan memakai VirtualBox untuk menjalankan OS Linux.
 - o Solusi : beralih memakai OS Windows 10.
- Menggunakan OS Windows 10 sama halnya dengan yang menjadi kendala di linux, environment mapnik yang terkadang tidak cocok dengan versi python yang digunakan
 - o Solusi : Download mapnik yang cocok dengan versi python yang digunakan.
- Instalasi flask yang terlebih dahulu harus dilakukan instalasi pip pada python sempat terjadi eror dikarenakan versi python 2.7 masih belum include pip
 - o Solusi : Upgrade Python ke versi 2.7.14 yang include dengan pip, dan instalasi flask dapat berjalan aman
- Perbedaan coding Warna yang terjadi pada Windows dan Linux.
 - o Solusi : linux : “mapnik.Color(‘syntax warna’)
- Membuat kombinasi beberapa shapefile pada satu projek, proses render eror dikarenakan coding yang masih salah.
 - o Solusi : Copas coding tertentu agar menghasilkan algoritma yang sesuai dan dapat dilakukan proses render.