

Tugas UTS Sistem Komputasi Terdistribusi

Nama : Muhammad Misbahul Munir

NIM : 145150300111027

Kelas : A

Soal membuat Rest webservice dengan socket UDP

Source code

Server

```
from flask import Flask, abort
import json
import socket, sys
app = Flask(__name__)
PORT = 7555
MAX = 65535

temp= []
@app.route('/node', methods=['GET'])
def semua():
    return json.dumps(temp)

@app.route('/node/<int:node_id>', methods=['GET'])
def satu(node_id):
    node = None
    for n in temp :
        if n["id"] == node_id :
            node = n
    if node :
        return json.dumps(node)
    else :
        abort(404)

if __name__=='__main__':
    app.run(debug=True, port=7555)
    s = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
    s.bind(('', PORT))
    while True:
        datax, address = s.recvfrom(MAX)
        temp.append(datax)
        print 'The client says', datax
```

No	Source code	Penjelasan
1	<code>from flask import Flask, abort</code>	-Import data dari library flask
2	<code>import json</code>	-Import data dari library json
3	<code>import socket, sys</code>	-Import data dari library socket, sys
4	<code>app = Flask(__name__)</code>	-Inisialisasi object flask framework
5	<code>PORT = 7555</code>	-Deklarasi PORT 7555
6	<code>MAX = 65535</code>	-Deklarasi maksimal data yang dibawa UDP = 65535
7	<code>temp= []</code>	-Membuat variabel temp dengan tipe array
8	<code>@app.route('/node',</code>	-Membuat rute dengan method GET
9	<code>methods=['GET'])</code>	-Deklarasi layanan web service API '/node' dengan method 'GET' untuk melayani permintaan semua data sensor
10	<code>def semua():</code>	-Deklarasi fungsi semua dengan isi marshaling
11	<code>return json.dumps(temp)</code>	
12	<code>@app.route('/node/<int:node_id>', methods=['GET'])</code>	-Deklarasi layanan web service API '/node/<int:node_id>' dengan method 'GET' untuk melayani permintaan satu data sensor
13	<code>def satu(node_id):</code>	-Deklarasi fungsi satu dengan isi untuk mencari nilai id node
14	<code>node = None</code>	
15	<code>for n in temp :</code>	
16	<code>if n["id"] == node_id :</code>	
17	<code>node = n</code>	
18	<code>if node :</code>	
19	<code>return</code>	
20	<code>json.dumps(node)</code>	-Marshaling node
21	<code>else :</code>	
22	<code>abort(404)</code>	
23	<code>if __name__=='__main__':</code>	-Jika nama sama dengan main maka akan membuka koneksi di port 7555
24	<code>app.run(debug=True,</code>	
25	<code>port=7555)</code>	
26	<code>s =</code>	-Membuat variabel dengan nama s yang menampung fungsi kembalian socket dengan nilai kembalian AF_INET yang menunjukkan bahwa menggunakan ipv4, SOCK_DGRAM yang menunjukkan bahwa menggunakan protocol udp.
27	<code>socket.socket(socket.AF_INET</code>	
28	<code>, socket.SOCK_DGRAM)</code>	
29	<code>s.bind(('', PORT))</code>	- Membuat fungsi s.bind dengan nilai kembalian port yang merupakan alamat untuk berkomunikasi dengan client, PORT yang merupakan port yang digunakan untuk komunikasi dengan client
30	<code>while True:</code>	- perulangan terus menerus
31	<code>datax, address =</code>	
32	<code>s.recvfrom(MAX)</code>	-Menerima data dari node
33	<code>temp.append(datax)</code>	-menambahkan nilai datax pada penampung temp

29	<code>print 'The client says', datax</code>	-menampilkan data yang diterima oleh server
----	---	---

Client

```
import httplib, urllib
import json

def semua():
    conn = httplib.HTTPConnection('127.0.0.1:7555')
    conn.request("GET", "/node")
    response = conn.getresponse()
    resp = json.loads(response.read())
    for id in resp :
        print id["id"], id["PH"], id["DO"]
    print resp

def satu(node_id):
    conn = httplib.HTTPConnection('127.0.0.1:7555')
    conn.request("GET", "/node/"+str(node_id))
    response = conn.getresponse()
    if response.status == 200 :
        resp = json.loads(response.read())
        print resp
    elif response.status == 404 :
        print "Node tidak ditemukan"
    else :
        print "Error"
```

No	Source code	Penjelasan
1	<code>import httplib, urllib</code>	-Import data dari library httlib
2	<code>import json</code>	-Import data dari library json
3	<code>def semua():</code>	-Deklarasi method semua untuk mendapatkan semua data sensor
4	<code> conn = httplib.HTTPConnection('127. 0.0.1:7555')</code>	-Membuat objek connection http ke server
5	<code> conn.request("GET", "/node")</code>	-Merequest API untuk mendapatkan seluruh data sensor
6	<code> response = conn.getresponse()</code>	-Mendapatkan respon dari server
7	<code> resp = json.loads(response.read())</code>	-Mengambil respon yang diterima

8	<pre> for id in resp : print id["id"], id["PH"], id["DO"] print resp </pre>	-perulangan untuk mendapatkan seluruh respon mencetak setiap data dalam perulangan
9	<pre> def satu(node_id): </pre>	-Deklarasi method satu untuk mendapatkan satu data sensor
10	<pre> conn = httplib.HTTPConnection('127. 0.0.1:7555') </pre>	-Membuat objek connection http ke server -Merequest API untuk mendapatkan seluruh data sensor
11	<pre> conn.request("GET", "/node/"+str(node_id)) </pre>	
12	<pre> response = conn.getresponse() </pre>	-Mendapatkan respon dari server
13	<pre> if response.status == 200 : </pre>	-Jika respon 200 maka file di load
14	<pre> resp = </pre>	
15	<pre> json.loads(response.read()) </pre>	
16	<pre> print resp </pre>	-Menampilkan respon
17	<pre> elif response.status == 404 : </pre>	-Jika respon 400
18	<pre> print "Node tidak ditemukan" </pre>	-Menampilkan Node tidak ditemukan
19	<pre> else : </pre>	-Selain itu menampilkan error
20	<pre> print "Error" </pre>	

Node

```

import socket, sys
import json
import time, datetime
from random import randint

SERVER_IP = '127.0.0.1'
PORT = 7555
MAX = 65535

def r(): # fungsi random nilai
    acak = randint(1, 100)
    return acak

incr = 0;
while True:
    t = time.time()
    waktu
    =
datetime.datetime.fromtimestamp(t).strftime('%d-%m-%Y %H:%M:%S')

```

```

incr = incr+ 1
data = {'id': incr,'PH': r() , 'DO': + r() , 'waktu': waktu}
datax = json.dumps(data)
s = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
s.sendto(datax,(SERVER_IP, PORT))
print "Data Sensor : ",datax,"\n"
time.sleep(3)
s.close()

```

No	Source code	Penjelasan
1	import socket, sys	-Mengimport library socket dan sys
2	import json	-Mengimport library json
3	import time, datetime from random import randint	-Mengimport library time, dan datetime -Mengimport library random agar dapat menggenerate random data 1 – 100
4	SERVER_IP = '127.0.0.1'	-Mengatur IP server
5	PORT = 7555	-Mengatur port yang digunakan server
6	MAX = 65535	-Mengatur maksimal datayang ditampung dalam pengiriman UDP
7	def r():	- fungsi random nilai
8	acak = randint(1, 100)	-Deklarasi variable acak dengan nilai awal 1 sampai 100
9	return acak	
10	incr = 0;	-Deklarasi variabel incr dengan nilai 0
11	while True:	-Perulangan while yang berjalan terus
12	t = time.time()	-Deklarasi variabel time
13	waktu = datetime.datetime.fromtime stamp(t).strftime('%d-%m-%Y %H :%M:%S')	-Merubah format waktu dengan urutan tanggal-bulan-tahun jam:menit:detik dan disimpan dalam variable waktu
14	incr = incr+ 1	-Increment + 1
15	data = {'id': incr,'PH': r() , 'DO': + r() , 'waktu': waktu}	-menyimpan data id,ph,do,waktu dalam variabel data
16	datax = json.dumps(data)	-Marshalling variabel datax
17	s = socket.socket(socket.AF_INET , socket.SOCK_DGRAM)	-Membuat variabel dengan nama serverSocket yang menampung fungsi kembalian socket dengan nilai kembalian AF_INET yang menunjukkan bahwa menggunakan ipv4, SOCK_DGRAM yang menunjukkan bahwa menggunakan protocol udp
18	s.sendto(datax,(SERVER_IP, PORT))	-Mengirim data ke client -Menampilkan data sensor

19	print "Data Sensor : ",datax,"\n"	
20	time.sleep(3)	-Delay 3 s
21	s.close()	-Menutup koneksi

Screenshot

Server

```
Terminal - ibah@ibah-Aspire-4739: /media/ibah/80E2680AE268072E/Kuliah/Semest - □ ×
File Edit View Terminal Tabs Help
ibah@ibah-Aspire-4739:/media/ibah/80E2680AE268072E/Kuliah/Semester 6/Sistem Komp
utasi Terdistribusi/webserver_node/Tugas UTSS$ python webserver_node.py
* Running on http://127.0.0.1:7755/ (Press CTRL+C to quit)
* Restarting with stat
* Debugger is active!
* Debugger PIN: 328-178-350
127.0.0.1 - - [12/Apr/2017 08:18:40] "GET /node/1 HTTP/1.1" 404 -
```

Client

```
Terminal - ibah@ibah-Aspire-4739: /media/ibah/80E2680AE268072E/Kuliah/Semest - □ ×
File Edit View Terminal Tabs Help
ibah@ibah-Aspire-4739:/media/ibah/80E2680AE268072E/Kuliah/Semester 6/Sistem Komp
utasi Terdistribusi/webserver_node/Tugas UTSS$ curl -i http://localhost:7755/node
/1
HTTP/1.0 404 NOT FOUND
Content-Type: text/html
Content-Length: 233
Server: Werkzeug/0.12.1 Python/2.7.12
Date: Wed, 12 Apr 2017 01:18:40 GMT

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//EN">
<title>404 Not Found</title>
<h1>Not Found</h1>
<p>The requested URL was not found on the server. If you entered the URL manual
ly please check your spelling and try again.</p>
ibah@ibah-Aspire-4739:/media/ibah/80E2680AE268072E/Kuliah/Semester 6/Sistem Komp
utasi Terdistribusi/webserver_node/Tugas UTSS$
```

Node

```
Terminal - ibah@ibah-Aspire-4739: /media/ibah/80E2680AE268072E/Kuliah/Semest - □ ×
File Edit View Terminal Tabs Help
ibah@ibah-Aspire-4739:/media/ibah/80E2680AE268072E/Kuliah/Semester 6/Sistem Komp
utasi Terdistribusi/webserver_node/Tugas UTSS$ python client.py
Data Sensor : {"DO": 72, "PH": 10, "waktu": "12-04-2017 08:14:23", "id": 1}
Data Sensor : {"DO": 94, "PH": 92, "waktu": "12-04-2017 08:14:26", "id": 2}
Data Sensor : {"DO": 35, "PH": 51, "waktu": "12-04-2017 08:14:29", "id": 3}
Data Sensor : {"DO": 48, "PH": 29, "waktu": "12-04-2017 08:14:32", "id": 4}
Data Sensor : {"DO": 19, "PH": 90, "waktu": "12-04-2017 08:14:35", "id": 5}
Data Sensor : {"DO": 49, "PH": 82, "waktu": "12-04-2017 08:14:38", "id": 6}
Data Sensor : {"DO": 32, "PH": 7, "waktu": "12-04-2017 08:14:41", "id": 7}
Data Sensor : {"DO": 83, "PH": 1, "waktu": "12-04-2017 08:14:44", "id": 8}
```