

**PEMROSESAN PARALEL  
DOKUMENTASI UAS**



**DISUSUN OLEH :**

**Kelompok 6**

Jonathan Jeremia V.V.S.	( 09011181823007 )
Tri Shena Orivia Pasin	( 09011181823029 )
Thesa Jiornmia	( 09011281823031 )
M Al Insyirah Satria H	( 09011381823083 )
Endy Afri Wildanu	( 09011281924076 )

**JURUSAN SISTEM KOMPUTER  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS SRIWIJAYA  
2020**

## **LAPORAN PROJECT AKHIR**

### **“MEMBUAT APLIKASI PERHITUNGAN LUAS DAN KELILING DI CONTROLLER DENGAN MENGAMBIL DATA PADA NODE – NODE VIRTUAL YANG TELAH DIHUBUNGKAN”**

Hal – hal yang di persiapkan :

1. Membuat Program di windows dengan menggunakan bahasa pemrograman python

- 3 program tersebut adalah ;
  - 1. Main\_apps
  - 2. Manualcall, dan
  - 3. Allcall

Penjelasan mengenai program yang dibuat terlampir dibawah.

2. Membuat 4 Virtual Mesin berOS Centos 7 dengan mode CLI

Virtual Mesin yang kami gunakan adalah Aplikasi Oracle VM VirtualBox versi 6.1  
Kami membuat Virtual mesin untuk menghasilkan beberapa node dengan IP  
address yang berbeda, dan juga mengubungkannya melalui jaringan adapter bridged

3. Menginstal Modul Paramiko

Dalam pembuatan aplikasi, kami menggunakan Modul paramiko yang telah di  
telah disediakan di bahasa pemrograman python untuk menghubungkan controller  
(windows) dengan beberapa virtual mesin yang lainnya agar dapat mengambil data  
yang ada di dalam virtual mesin tersebut.

4. Membuat Program di Virtual Mesin menggunakan bahasa pemrograman python

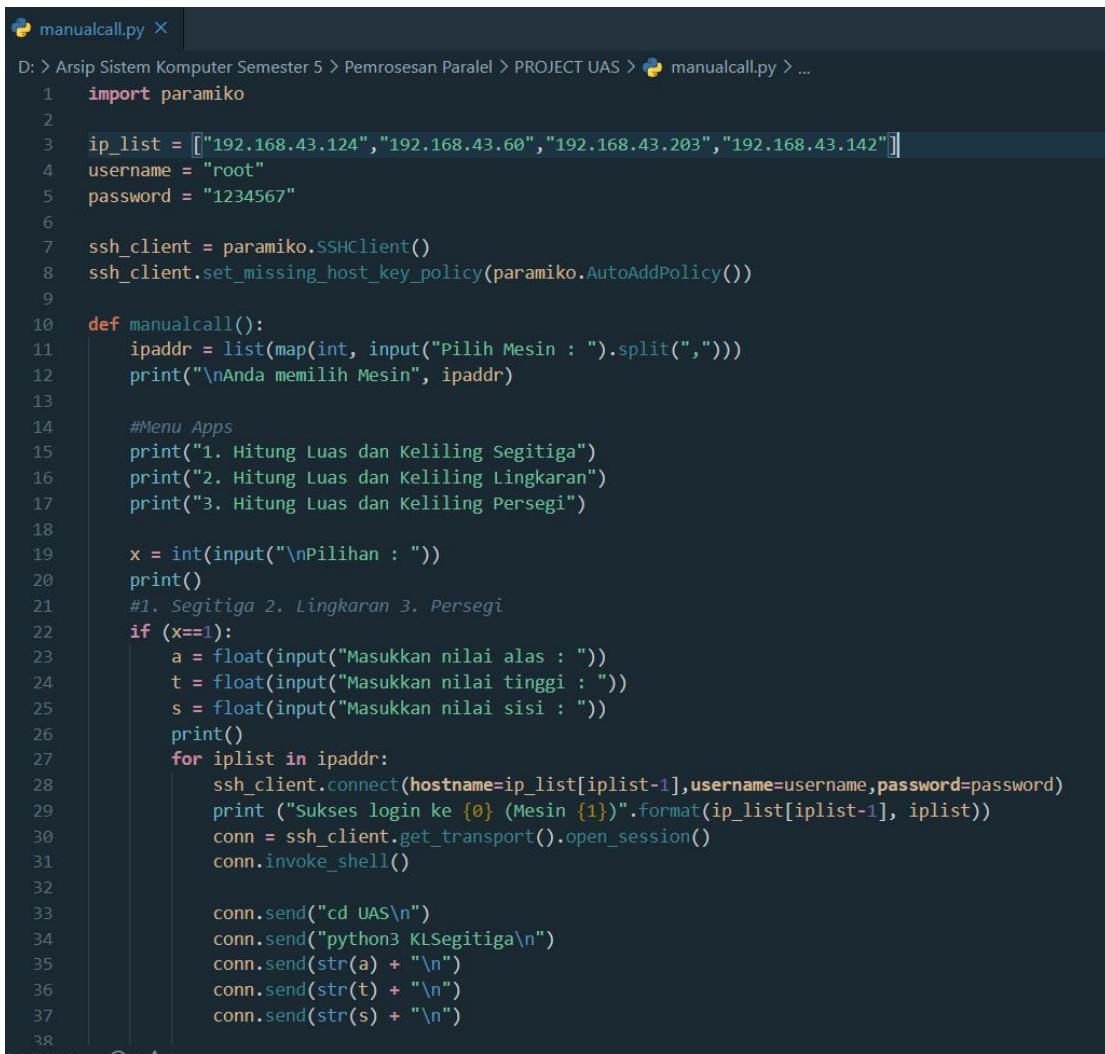
Ada 3 program yang akan dibuat ;

- 1. klling.py : merupakan kodingan menghitung luas dan keliling lingkaran
- 2. klsegi.py : merupakan kodingan menghitung luas dan keliling segitiga
- 3. Klpers.py : merupakan kodingan menghitung luas dan keliling persegi

Kodingan ini di simpan di dalam masing – masing virtual mesin yang telah dibuat.

## Kodingan untuk di Windows

1. Pada file kodingan manualcall.py, digunakan untuk memanggil VM Linux yang sudah dibuat dengan cara memilih Mesin yang ingin digunakan terlebih dahulu untuk menghitung luas dan keliling. Kodingan ini akan berjalan ketika main\_apps.py memilih pilihan ‘1’. Setelah itu, akan muncul pilihan lagi untuk memilih menghitung luas dan keliling dari ketiga pilihan yang ada dan akan menampilkan hasil dari mesin yang kita pilih.

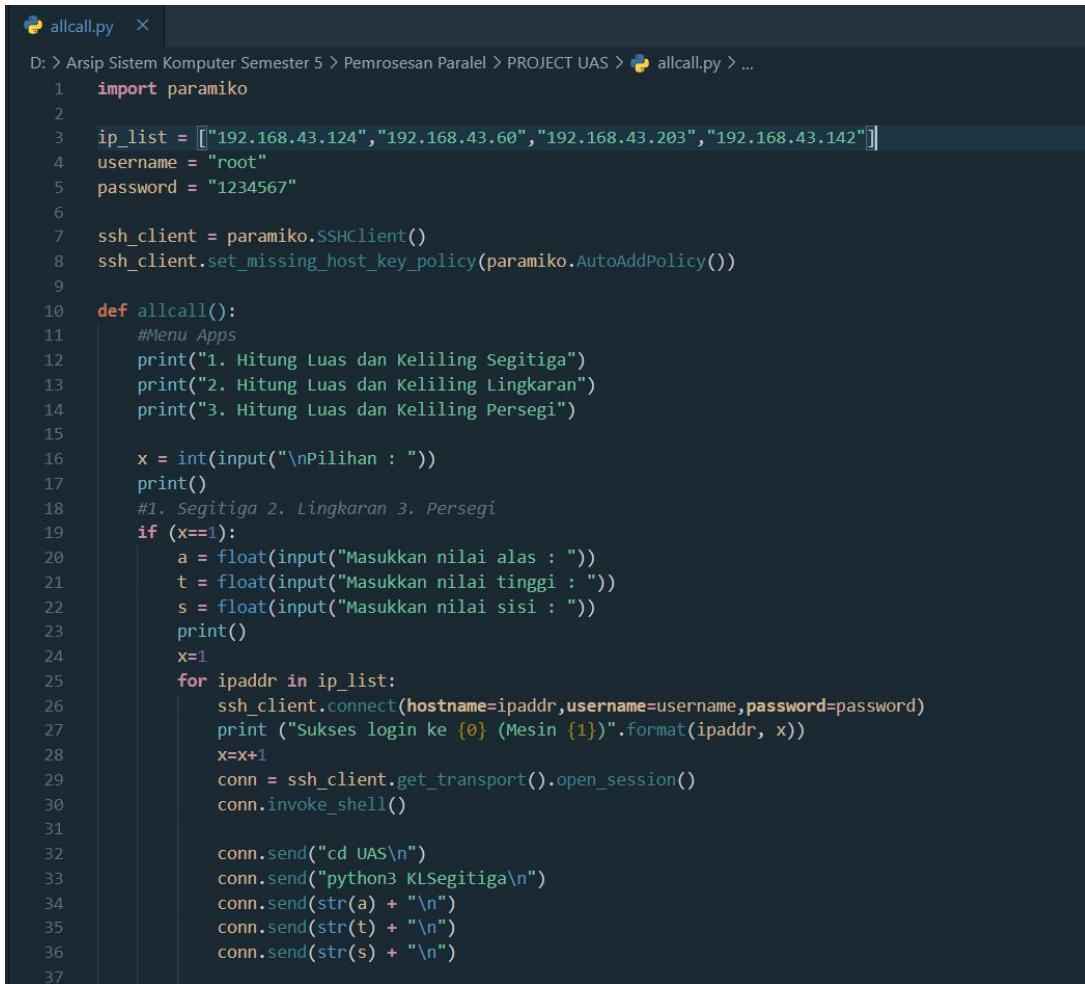


The screenshot shows a code editor window with the file 'manualcall.py' open. The code is written in Python and uses the paramiko library to connect to multiple hosts via SSH. It prompts the user to choose a machine from a list of four IP addresses (192.168.43.124, 192.168.43.60, 192.168.43.203, 192.168.43.142). After connecting, it asks the user to select an application (1. Segitiga, 2. Lingkaran, 3. Persegi) and provides options to calculate area or circumference. The code then sends commands to the selected host to run a specific Python script (KLSegitiga.py) which performs the calculations based on user input for base, height, or side lengths.

```
manualcall.py X
D: > Arsip Sistem Komputer Semester 5 > Pemrosesan Paralel > PROJECT UAS > manualcall.py > ...
1 import paramiko
2
3 ip_list = ["192.168.43.124", "192.168.43.60", "192.168.43.203", "192.168.43.142"]
4 username = "root"
5 password = "1234567"
6
7 ssh_client = paramiko.SSHClient()
8 ssh_client.set_missing_host_key_policy(paramiko.AutoAddPolicy())
9
10 def manualcall():
11     ipaddr = list(map(int, input("Pilih Mesin : ").split(",")))
12     print("\nAnda memilih Mesin", ipaddr)
13
14     #Menu Apps
15     print("1. Hitung Luas dan Keliling Segitiga")
16     print("2. Hitung Luas dan Keliling Lingkaran")
17     print("3. Hitung Luas dan Keliling Persegi")
18
19     x = int(input("\nPilihan : "))
20     print()
21     #1. Segitiga 2. Lingkaran 3. Persegi
22     if (x==1):
23         a = float(input("Masukkan nilai alas : "))
24         t = float(input("Masukkan nilai tinggi : "))
25         s = float(input("Masukkan nilai sisi : "))
26         print()
27         for iplist in ipaddr:
28             ssh_client.connect(hostname=ip_list[iplist-1],username=username,password=password)
29             print ("Sukses login ke {} (Mesin {})".format(ip_list[iplist-1], iplist))
30             conn = ssh_client.get_transport().open_session()
31             conn.invoke_shell()
32
33             conn.send("cd UAS\n")
34             conn.send("python3 KLSegitiga\n")
35             conn.send(str(a) + "\n")
36             conn.send(str(t) + "\n")
37             conn.send(str(s) + "\n")
```

```
39     output = conn.recv(65535).decode('ascii')
40     print(output)
41
42 elif (x==2):
43     r = float(input("Masukkan nilai jari-jari : "))
44     print()
45     for iplist in ipaddr:
46         ssh_client.connect(hostname=ip_list[iplist-1],username=username,password=password)
47         print ("Sukses login ke {0} (Mesin {1})".format(ip_list[iplist-1], iplist))
48         conn = ssh_client.get_transport().open_session()
49         conn.invoke_shell()
50
51         conn.send("cd UAS\n")
52         conn.send("python3 KLLingkaran\n")
53         conn.send(str(r) + "\n")
54
55         output = conn.recv(65535).decode('ascii')
56         print(output)
57
58 elif (x==3):
59     s = float(input("Masukkan nilai sisi : "))
60     print()
61     for iplist in ipaddr:
62         ssh_client.connect(hostname=ip_list[iplist-1],username=username,password=password)
63         print ("Sukses login ke {0} (Mesin {1})".format(ip_list[iplist-1], iplist))
64         conn = ssh_client.get_transport().open_session()
65         conn.invoke_shell()
66
67         conn.send("cd UAS\n")
68         conn.send("python3 KLPersegi\n")
69         conn.send(str(s) + "\n")
70
71         output = conn.recv(65535).decode('ascii')
72         print(output)
73
74 else:
75     print("Tidak tersedia. Silahkan ulangi kembali.")
```

2. Pada file kodingan allcall.py digunakan untuk memanggil seluruh VM Linux tanpa memilih mesin yang ingin digunakan terlebih dahulu. Kodingan ini akan berjalan ketika main\_apps.py memilih pilihan ‘2’. Setelah itu, akan muncul pilihan lagi untuk memilih menghitung luas dan keliling dari ketiga pilihan yang ada dan akan menampilkan hasilnya dari seluruh mesin yang ada.



```

allcall.py  X
D: > Arsip Sistem Komputer Semester 5 > Pemrosesan Paralel > PROJECT UAS > allcall.py > ...
1 import paramiko
2
3 ip_list = ["192.168.43.124", "192.168.43.60", "192.168.43.203", "192.168.43.142"]
4 username = "root"
5 password = "1234567"
6
7 ssh_client = paramiko.SSHClient()
8 ssh_client.set_missing_host_key_policy(paramiko.AutoAddPolicy())
9
10 def allcall():
11     #Menu Apps
12     print("1. Hitung Luas dan Keliling Segitiga")
13     print("2. Hitung Luas dan Keliling Lingkaran")
14     print("3. Hitung Luas dan Keliling Persegi")
15
16     x = int(input("\nPilihan : "))
17     print()
18     #1. Segitiga 2. Lingkaran 3. Persegi
19     if (x==1):
20         a = float(input("Masukkan nilai alas : "))
21         t = float(input("Masukkan nilai tinggi : "))
22         s = float(input("Masukkan nilai sisi : "))
23         print()
24         x=1
25         for ipaddr in ip_list:
26             ssh_client.connect(hostname=ipaddr,username=username,password=password)
27             print ("Sukses login ke {0} (Mesin {1})".format(ipaddr, x))
28             x=x+1
29             conn = ssh_client.get_transport().open_session()
30             conn.invoke_shell()
31
32             conn.send("cd UAS\n")
33             conn.send("python3 KLSegitiga\n")
34             conn.send(str(a) + "\n")
35             conn.send(str(t) + "\n")
36             conn.send(str(s) + "\n")
37

```

```
37     output = conn.recv(65535).decode('ascii')
38     print(output)
39
40
41 elif (x==2):
42     r = float(input("Masukkan nilai jari-jari : "))
43     print()
44     x+=1
45     for ipaddr in ip_list:
46         ssh_client.connect(hostname=ipaddr,username=username,password=password)
47         print ("Sukses login ke {0} (Mesin {1})".format(ipaddr, x))
48         x=x+1
49         conn = ssh_client.get_transport().open_session()
50         conn.invoke_shell()
51
52         conn.send("cd UAS\n")
53         conn.send("python3 KLLingkaran\n")
54         conn.send(str(r) + "\n")
55
56         output = conn.recv(65535).decode('ascii')
57         print(output)
58
59 elif (x==3):
60     s = float(input("Masukkan nilai sisi : "))
61     print()
62     x+=1
63     for ipaddr in ip_list:
64         ssh_client.connect(hostname=ipaddr,username=username,password=password)
65         print ("Sukses login ke {0} (Mesin {1})".format(ipaddr, x))
66         x=x+1
67         conn = ssh_client.get_transport().open_session()
68         conn.invoke_shell()
69
70         conn.send("cd UAS\n")
71         conn.send("python3 KLPersegi\n")
72         conn.send(str(s) + "\n")
73
74         output = conn.recv(65535).decode('ascii')
75         print(output)
76
77 else:
78     print("Tidak tersedia. Silahkan ulangi kembali.")
```

3. Pada file kodingan main\_apps.py merupakan perintah untuk menjalankan kodingan yang ada pada file manualcall.py dan allcall.py. pilihan ‘1’ untuk manualcall.py dan pilihan ‘2’ untuk allcall.py. Selain kedua pilihan tersebut maka akan muncul “ Tidak tersedia. Silahkan ulangi kembali. “. Pada main\_apps.py ini juga terdapat pilihan apakah akan keluar atau tidak dari main\_apps.py. Ketik ‘y’ lalu tekan enter, maka akan keluar dari program. Sedangkan ketik ‘n’ jika tetap ingin melakukan perhitungan.



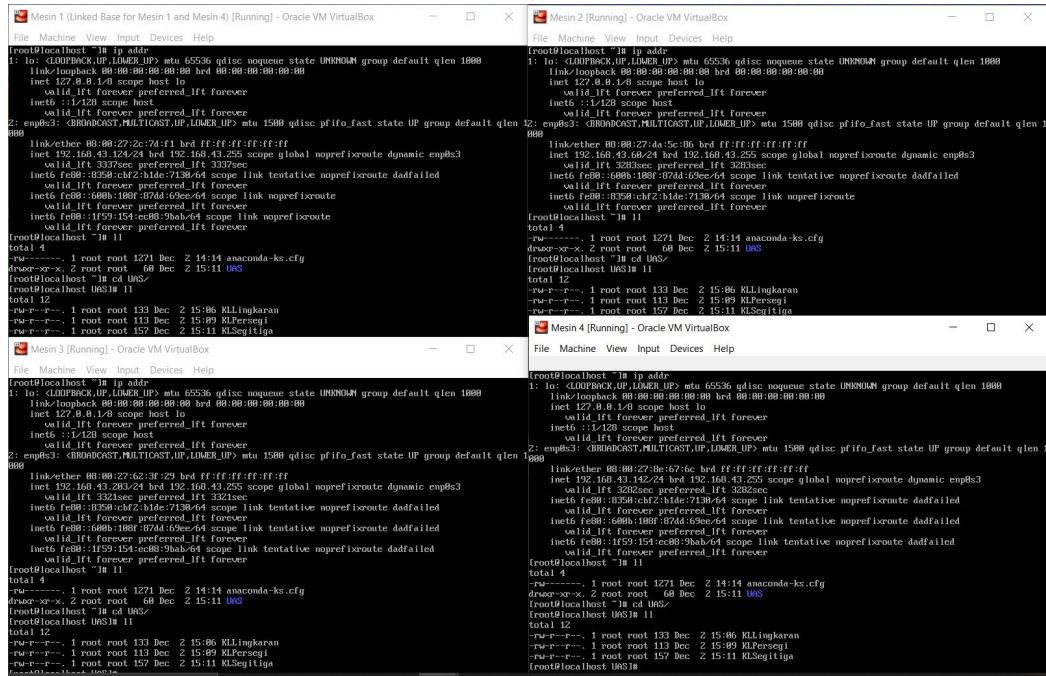
```

main_apps.py X
D: > Arsip Sistem Komputer Semester 5 > Pemrosesan Paralel > PROJECT UAS > main_apps.py > ...
1 import paramiko
2 import os
3 from manualcall import manualcall
4 from allcall import allcall
5
6 ip_list = ["192.168.43.124", "192.168.43.60", "192.168.43.203", "192.168.43.142"]
7
8 os.system("title Main Menu")
9
10 def confirm():
11     print("Exit? (y/n)", end=" ")
12     y = str(input())
13     if (y=="y" or y=="Y" or y=="yes"):
14         print()
15     elif (y=="n" or y=="N" or y=="no"):
16         menu()
17
18 def menu():
19     os.system('cls')
20
21     #Menu
22     print("MENU APPS (Menghitung Luas & Keliling Segitiga, Lingkaran, dan Persegi)")
23
24     #Print List ip dan mesin
25     x=0;
26     for ipaddr in ip_list:
27         x=x+1
28         print("Mesin", x,":", ipaddr)
29
30     print("\n1. Pilih Manual 2. Ambil Semua")
31     x = int(input("Pilihan : "))
32     if (x==1):
33         manualcall()
34     elif (x==2):
35         allcall()
36     else:
37         print("Tidak tersedia. Silahkan ulangi kembali.")
38
39     confirm()
40
41 menu()

```

## **Proses Menjalankan Program**

1. Bukalah sebanyak empat VM Linux, dan periksa masing-masing IP di setiap mesin menggunakan perintah ip addr. Setiap mesin harus memiliki IP yang berbeda. Pada keempat mesin ini, Mesin 1 memiliki ip 192.168.43.124, Mesin 2 memiliki ip 192.168.43.60, Mesin 3 memiliki ip 192.168.43.203, dan Mesin 4 memiliki ip 192.168.43.142.

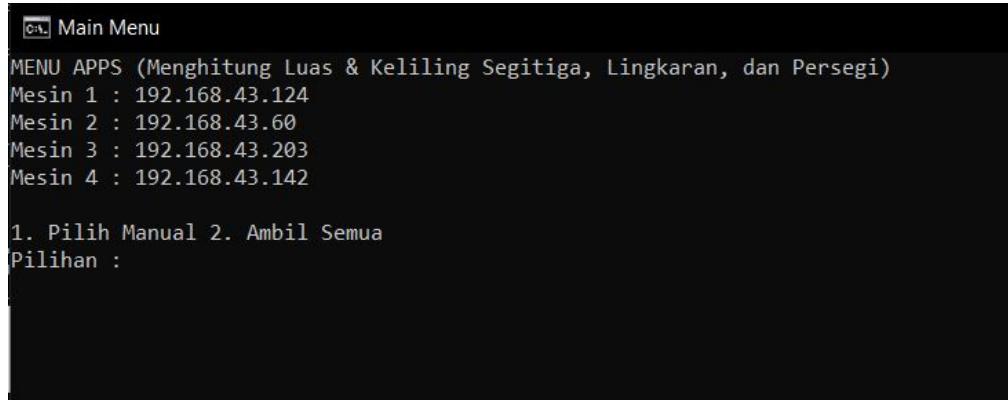


2. Jalankan kodingan main apps.py pada cmd

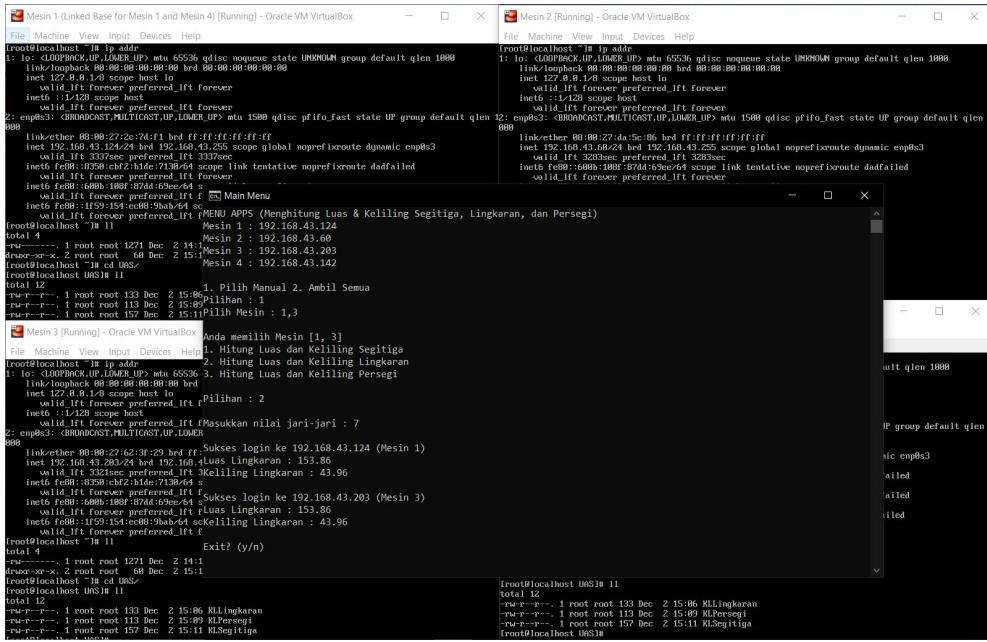
Data (D) > Arsip Sistem Komputer Semester 5 > Pemrosesan Paralel > PROJECT UAS				
Name	Date modified	Type	Size	
__pycache__	12/3/2020 11:04 AM	File folder		
alcall	12/3/2020 11:03 AM	Python Source File	3 KB	
main_apps	12/3/2020 11:03 AM	Python Source File	1 KB	
manualcall	12/3/2020 11:03 AM	Python Source File	3 KB	

```
C:\ Command Prompt  
Microsoft Windows [Version 10.0.19042.662]  
(c) 2020 Microsoft Corporation. All rights reserved.  
  
C:\Users\Harasa Ramdhany P>D:  
  
D:>cd "Arsip Sistem Komputer Semester 5"  
D:>Arsip Sistem Komputer Semester 5>cd "Pemrosesan Paralel"  
D:>Arsip Sistem Komputer Semester 5\Pemrosesan Paralel>cd "PROJECT UAS"  
D:>Arsip Sistem Komputer Semester 5\Pemrosesan Paralel\PROJECT UAS>python main_apps.py
```

3. Akan muncul tampilan seperti di bawah ini setelah main\_apps.py dijalankan. Pilih dari kedua pilihan tersebut. pilihan 1 akan menjalankan kodingan manualcall.py sedangkan pilihan 2 akan menjalankan allcall.py.

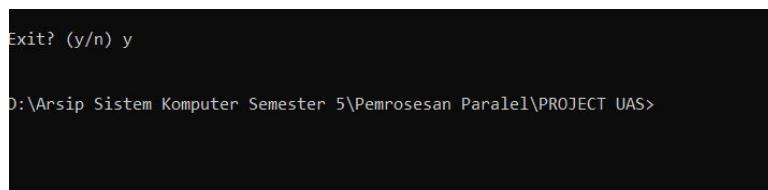


4. Dibawah ini merupakan tampilan ketika memilih “ Manual “, yang artinya memilih mesin yang ingin terlebih dahulu digunakan. Pada presentasi ini, kami memilih Mesin 1 dan Mesin 3 untuk melakukan perhitungan Luas dan Keliling Lingkaran.

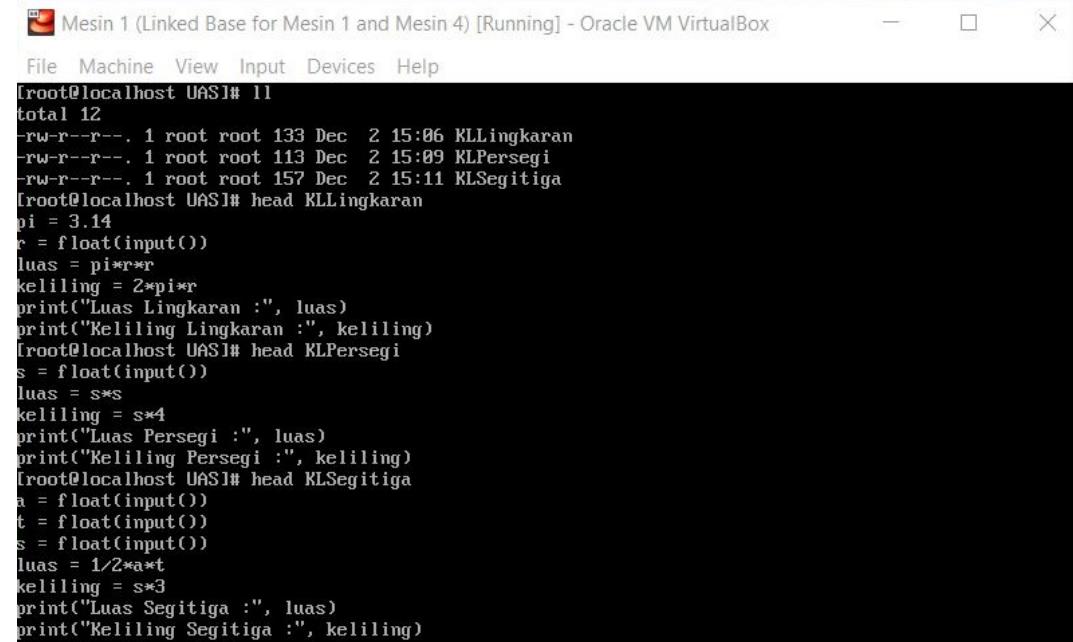


5. Kemudian, di bawah ini merupakan tampilan jika memilih “ Ambil Semua “ yang artinya menggunakan seluruh mesin. Pada presentasi ini kami mengambil seluruh mesin untuk melakukan perhitungan Luas dan Keliling Persegi.

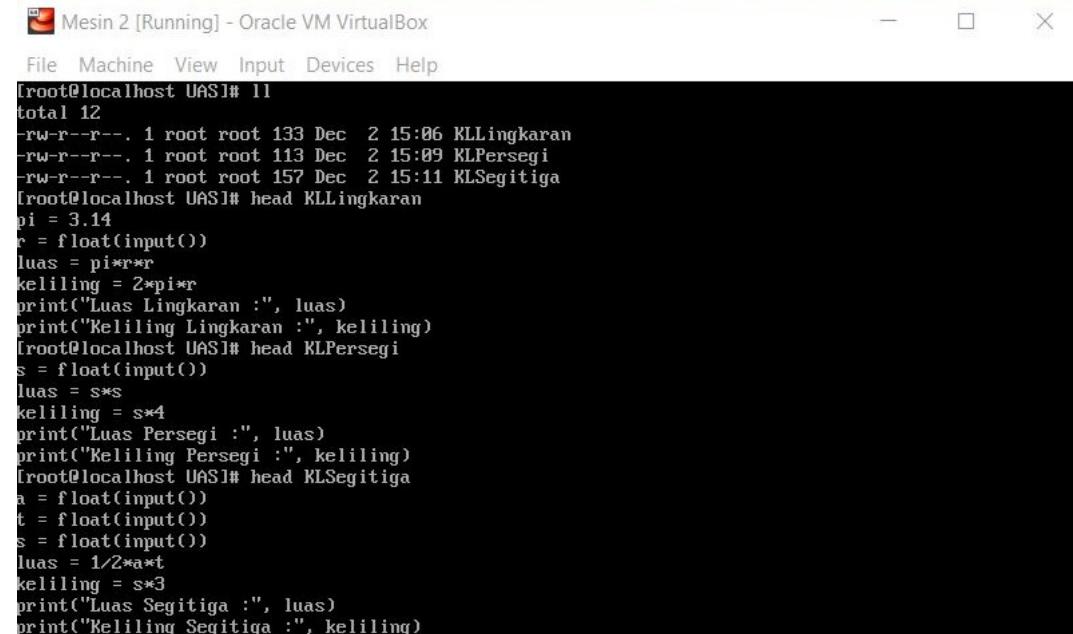
6. Dibawah ini merupakan pilihan jika ingin keluar atau tidak dari main\_apps.py, Ketik ‘y’ lalu tekan enter, maka akan keluar dari program. Sedangkan ketik ‘n’ jika tetap ingin melakukan perhitungan. Disini kami memilih ‘y’ yang artinya keluar dari program perhitungan.



7. Dibawah ini merupakan kodingan Perhitungan Luas dan Keliling yang terdapat pada keempat VM Linux.



```
File Machine View Input Devices Help
[root@localhost UASJ]# ll
total 12
-rw-r--r--. 1 root root 133 Dec  2 15:06 KLLingkaran
-rw-r--r--. 1 root root 113 Dec  2 15:09 KLPersegi
-rw-r--r--. 1 root root 157 Dec  2 15:11 KLSegitiga
[root@localhost UASJ]# head KLLingkaran
pi = 3.14
r = float(input())
luas = pi*r*r
keliling = 2*pi*r
print("Luas Lingkaran :", luas)
print("Keliling Lingkaran :", keliling)
[root@localhost UASJ]# head KLPersegi
s = float(input())
luas = s*s
keliling = s*4
print("Luas Persegi :", luas)
print("Keliling Persegi :", keliling)
[root@localhost UASJ]# head KLSegitiga
a = float(input())
t = float(input())
s = float(input())
luas = 1/2*a*t
keliling = s*3
print("Luas Segitiga :", luas)
print("Keliling Segitiga :", keliling)
```



```
File Machine View Input Devices Help
[root@localhost UASJ]# ll
total 12
-rw-r--r--. 1 root root 133 Dec  2 15:06 KLLingkaran
-rw-r--r--. 1 root root 113 Dec  2 15:09 KLPersegi
-rw-r--r--. 1 root root 157 Dec  2 15:11 KLSegitiga
[root@localhost UASJ]# head KLLingkaran
pi = 3.14
r = float(input())
luas = pi*r*r
keliling = 2*pi*r
print("Luas Lingkaran :", luas)
print("Keliling Lingkaran :", keliling)
[root@localhost UASJ]# head KLPersegi
s = float(input())
luas = s*s
keliling = s*4
print("Luas Persegi :", luas)
print("Keliling Persegi :", keliling)
[root@localhost UASJ]# head KLSegitiga
a = float(input())
t = float(input())
s = float(input())
luas = 1/2*a*t
keliling = s*3
print("Luas Segitiga :", luas)
print("Keliling Segitiga :", keliling)
```

Mesin 3 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

```
[root@localhost UAS]# ll
total 12
-rw-r--r--. 1 root root 133 Dec  2 15:06 KLLingkaran
-rw-r--r--. 1 root root 113 Dec  2 15:09 KLPpersegi
-rw-r--r--. 1 root root 157 Dec  2 15:11 KLSegitiga
[root@localhost UAS]# head KLLingkaran
pi = 3.14
r = float(input())
luas = pi*r**2
keliling = 2*pi*r
print("Luas Lingkaran :", luas)
print("Keliling Lingkaran :", keliling)
[root@localhost UAS]# head KLPpersegi
s = float(input())
luas = s*s
keliling = s*4
print("Luas Persegi :", luas)
print("Keliling Persegi :", keliling)
[root@localhost UAS]# head KLSegitiga
a = float(input())
t = float(input())
s = float(input())
luas = 1/2*a*t
keliling = s*3
print("Luas Segitiga :", luas)
print("Keliling Segitiga :", keliling)
[root@localhost UAS]#
```

Mesin 4 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

```
[root@localhost UAS]# ll
total 12
-rw-r--r--. 1 root root 133 Dec  2 15:06 KLLingkaran
-rw-r--r--. 1 root root 113 Dec  2 15:09 KLPpersegi
-rw-r--r--. 1 root root 157 Dec  2 15:11 KLSegitiga
[root@localhost UAS]# head KLLingkaran
pi = 3.14
r = float(input())
luas = pi*r**2
keliling = 2*pi*r
print("Luas Lingkaran :", luas)
print("Keliling Lingkaran :", keliling)
[root@localhost UAS]# head KLPpersegi
s = float(input())
luas = s*s
keliling = s*4
print("Luas Persegi :", luas)
print("Keliling Persegi :", keliling)
[root@localhost UAS]# head KLSegitiga
a = float(input())
t = float(input())
s = float(input())
luas = 1/2*a*t
keliling = s*3
print("Luas Segitiga :", luas)
print("Keliling Segitiga :", keliling)
[root@localhost UAS]#
```

## Kesimpulan

Pada progress pembuatan project akhir ini, dimana kelompok kami berhasil membuat aplikasi perhitungan luas dan keliling di controller dengan mengambil data pada node – node virtual yang telah dihubungkan. Dengan membuat codingan untuk menghitung luas dan keliling yang diinginkan pada VM Linux, lalu menghubungkannya dengan codingan pada windows menggunakan modul paramiko yang terdapat pada codingan allcall.py dan manualcall.py. Lalu kami satukan allcall.py dan manualcall.py pada codingan main\_apps.py dengan cara memanggil fungsi manuallcall.py dan allcall.py pada codingan main\_apps.py. Lalu saat main\_apps.py di jalankan maka kita dihadapkan dengan dua pilihan, yang keduanya memiliki fungsi sama yaitu menghitung luas dan keliling yang kita inginkan dan mencantumkan ip list VM Linux yang melakukan proses perhitungan pada output codingan tersebut. Dan kita telah mendapatkan hasil luas dan keliling yang kita inginkan.

Project di Github : <https://github.com/hazekezia/Paramikoo>