

PRACTICAL 4

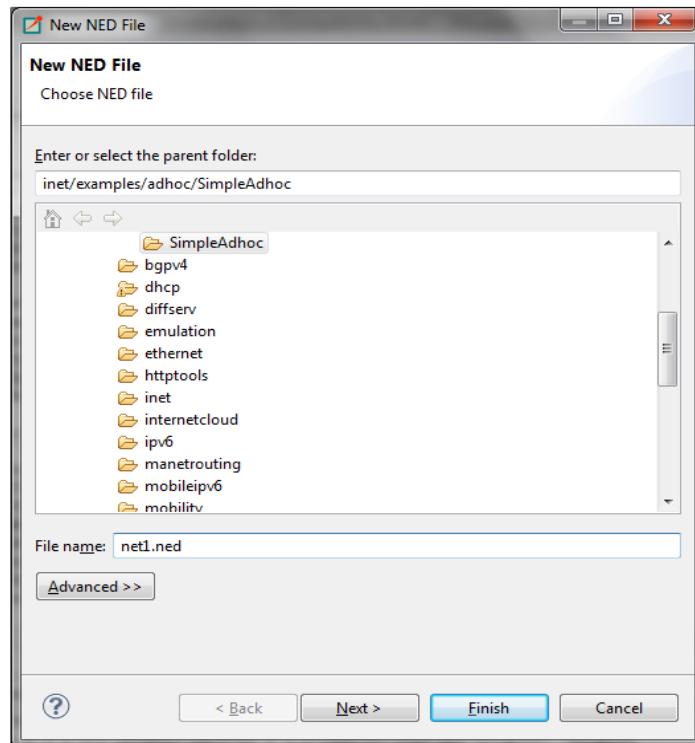
Aim: Create simple Adhoc network.

Description:

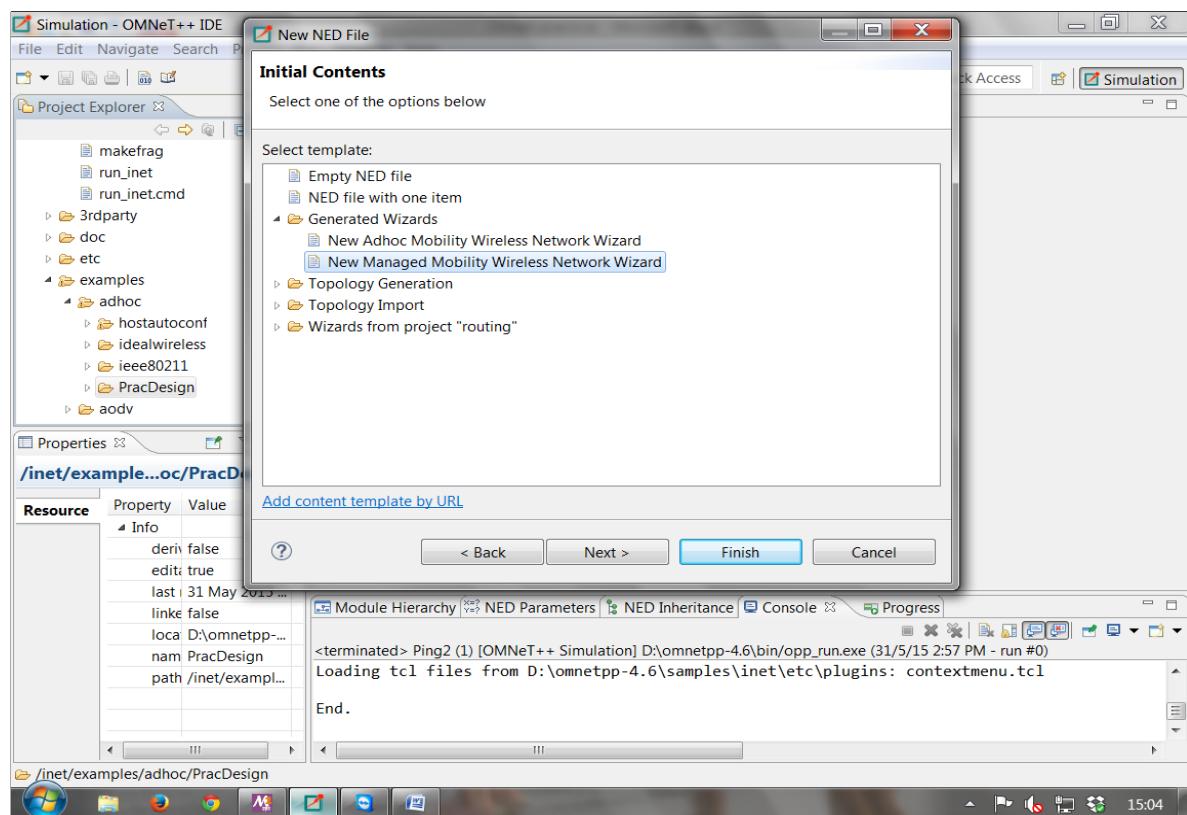
- Simulator used : Omnet++
- Simulator can be downloaded from below link:
<https://omnetpp.org/omnetpp> (recommended version is omnet++ 4.2.2).
- After installing Omnet++, we need to install inet framework which is specially designed for wireless simulation. You can download inet framework from below link.
<https://inet.omnetpp.org/Download.html>
- After downloading there are certain steps to be followed to include this framework in omnet++ as follows:
 - Download the INET sources.
 - Unpack it into the directory of your choice: (tar xvfz inet-<version>.tgz)(recommended is tar or .tgz)
 - Recommeded version is inet 2.1
 - Start the Omnet++ IDE, and import the project via *File -> Import -> Existing Projects to the Workspace*. A project named inet should appear.
 - Build with *Project -> Build*, or hit ctrl+b
 - Now you should be able to launch example simulations.

Steps for practical:

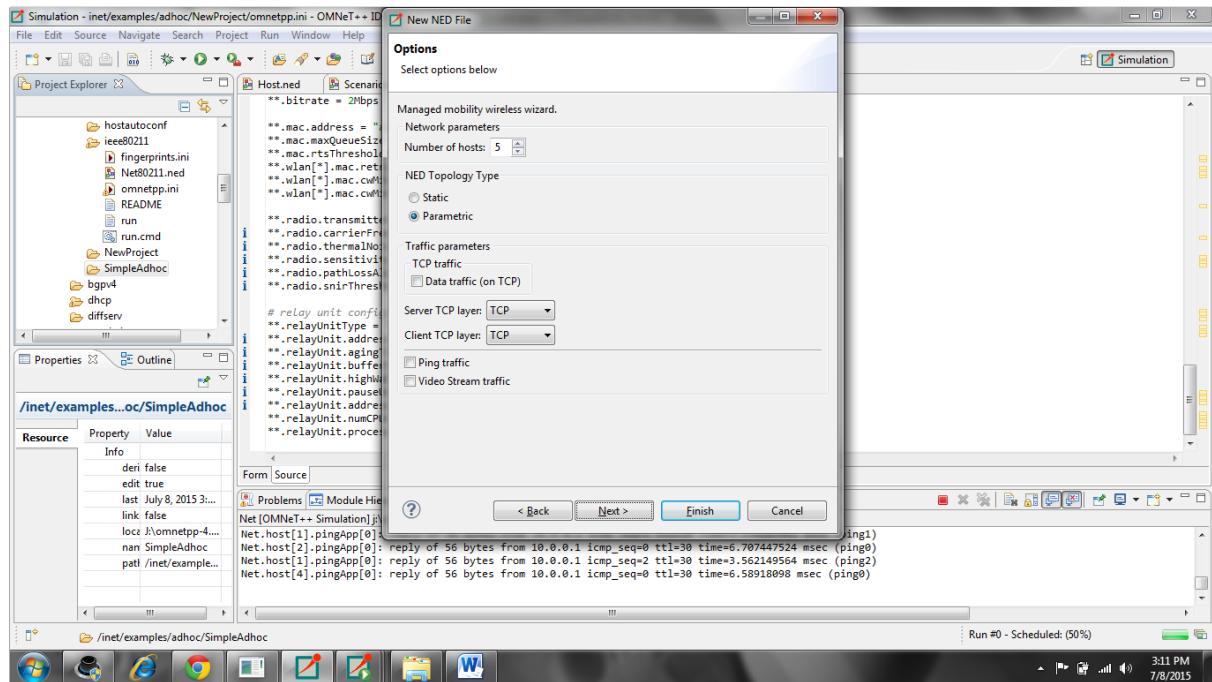
- Then open inet/examples/
- Right click on adhoc -create new folder as SimpleAdhoc.
- Right click on your newly created folder and select NED file. Give name as Net1.



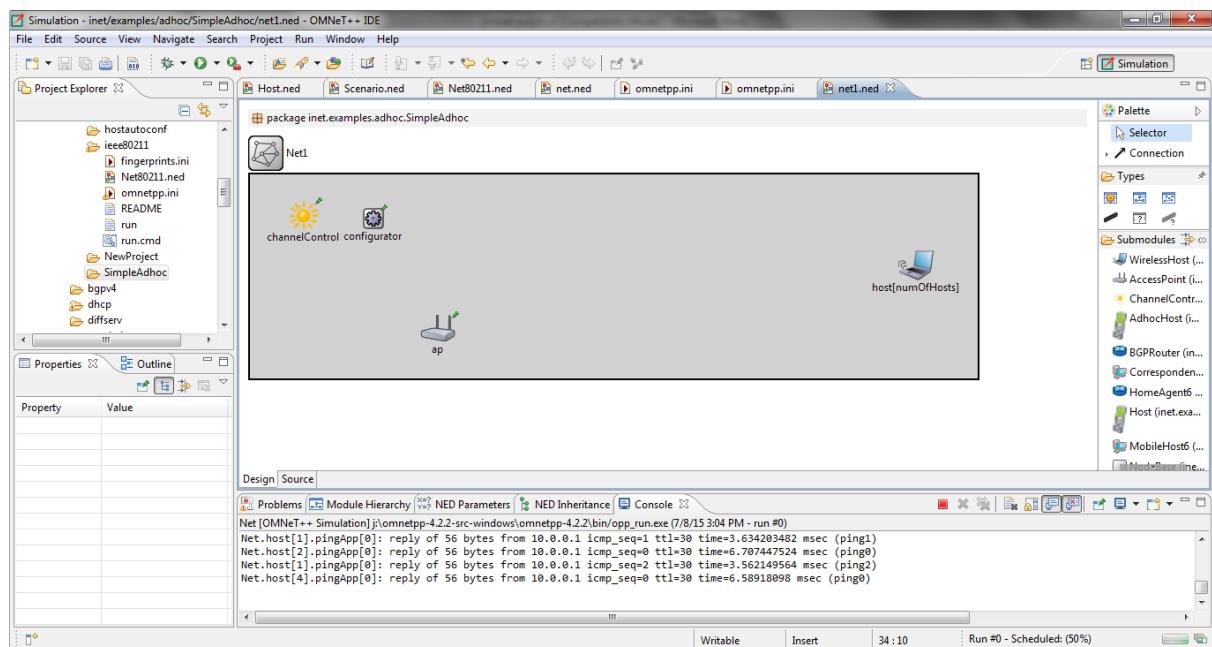
Click on new manages mobility wireless network wizard.



then configure as follows



then click on finish.



Below is the code that will be available in source part of net1.ned once configured.

```
package inet.examples.adhoc.SimpleAdhoc;
```

```
// numofHosts: 5
```

```
import inet.networklayer.autorouting.ipv4.Ipv4NetworkConfigurator;
import inet.nodes.inet.WirelessHost;
import inet.nodes.wireless.AccessPoint;
import inet.world.radio.ChannelControl;
```

```
network Net
```

```
{
parameters:
  int numOfHosts;

submodules:
  host[numOfHosts]: WirelessHost
  {
    @display("r=,,#707070");
  }

  ap: AccessPoint
  {
    @display("p=213,174;r=,,#707070");
  }

  channelControl: ChannelControl
  {
    numChannels = 2;
    @display("p=61,46");
  }

  configurator: IPv4NetworkConfigurator
  {
    @display("p=140,50");
  }
}
```

On design part you will find components appearing according to the code as the above snapshot.

Same as do this in omnetpp.ini file :

Source code for omnetpp.ini:

```
[General]
network = Net1

*.numOfHosts = 5

#debug-on-errors = true
tkenv-plugin-path = ../../etc/plugins

**.constraintAreaMinX = 0m
**.constraintAreaMinY = 0m
**.constraintAreaMinZ = 0m
**.constraintAreaMaxX = 600m
**.constraintAreaMaxY = 400m
**.constraintAreaMaxZ = 0m
**.debug = true
**.coreDebug = false
**.host**.channelNumber = 0

# channel physical parameters
*.channelControl.carrierFrequency = 2.4GHz
*.channelControl.pMax = 2.0mW
*.channelControl.sat = -110dBm
*.channelControl.alpha = 2

# mobility
```

```

**.host*.mobilityType = "MassMobility"
**.host*.mobility.initFromDisplayString = false
**.host*.mobility.changeInterval = truncnormal(2s, 0.5s)
**.host*.mobility.changeAngleBy = normal(0deg, 30deg)
**.host*.mobility.speed = truncnormal(20mps, 8mps)
**.host*.mobility.updateInterval = 100ms

# ping app (host[0] pinged by others)
*.host[0].numPingApps = 0
*.host[*].numPingApps = 2
*.host[*].pingApp[*].destAddr = "host[0]"
**.pingApp[0].startTime = uniform(1s,5s)
**.pingApp[1].startTime = 5s+uniform(1s,5s)
**.pingApp[*].printPing = true

# nic settings
**.wlan[*].bitrate = 2Mbps

**.wlan[*].mgmt.frameCapacity = 10

**.wlan[*].mac.address = "auto"
**.wlan[*].mac.maxQueueSize = 14
**.wlan[*].mac.rtsThresholdBytes = 3000B
**.wlan[*].mac.retryLimit = 7
**.wlan[*].mac.cwMinData = 7

**.wlan[*].radio.transmitterPower = 2mW
**.wlan[*].radio.thermalNoise = -110dBm
**.wlan[*].radio.sensitivity = -85dBm
**.wlan[*].radio.pathLossAlpha = 2
**.wlan[*].radio.snrThreshold = 4dB

```

[Config Ping1]
description = "host1 pinging host0"

[Config Ping2] #__interactive__
description = "n hosts"
leave numHosts undefined here

```

**.mobility.constraintAreaMinZ = 0m
**.mobility.constraintAreaMaxZ = 0m
**.mobility.constraintAreaMinX = 0m
**.mobility.constraintAreaMinY = 0m
**.mobility.constraintAreaMaxX = 600m
**.mobility.constraintAreaMaxY = 400m
**.debug = false
**.coreDebug = false

```

```
**.channelNumber = 0
```

```

# channel physical parameters
*.channelControl.carrierFrequency = 2.4GHz
*.channelControl.pMax = 20.0mW
*.channelControl.sat = -110dBm
*.channelControl.alpha = 2

```

mobility

```
**.host[*].mobilityType = "MassMobility"
```

```

**.host[*].mobility.changeInterval = truncnormal(2s, 0.5s)
**.host[*].mobility.changeAngleBy = normal(0deg, 30deg)
**.host[*].mobility.speed = truncnormal(20mps, 8mps)
**.host[*].mobility.updateInterval = 100ms

# nic settings
**.bitrate = 2Mbps

**.mac.address = "auto"
**.mac.maxQueueSize = 14
**.mac.rtsThresholdBytes = 3000B
**.wlan[*].mac.retryLimit = 7
**.wlan[*].mac.cwMinData = 7
**.wlan[*].mac.cwMinMulticast = 31

**.radio.transmitterPower = 20.0mW
**.radio.carrierFrequency = 2.4GHz
**.radio.thermalNoise = -110dBm
**.radio.sensitivity = -85dBm
**.radio.pathLossAlpha = 2
**.radio.snrThreshold = 4dB

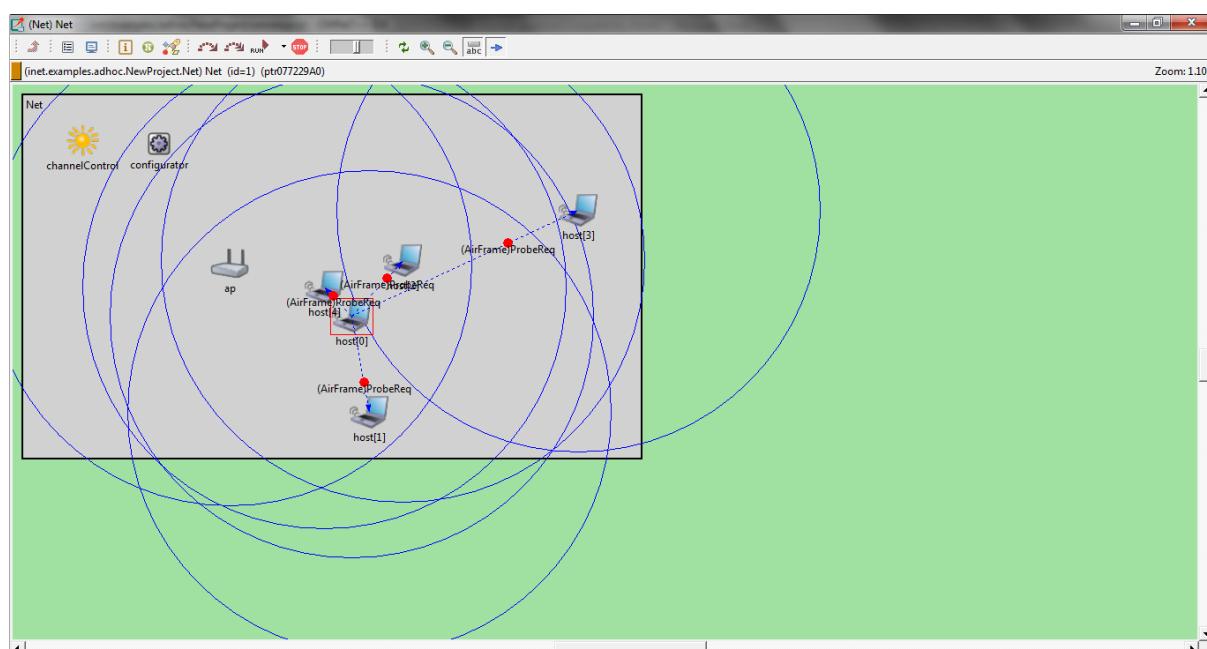
# relay unit configuration
**.relayUnitType = "MACRelayUnitNP"
**.relayUnit.addressTableSize = 100
**.relayUnit.agingTime = 120s
**.relayUnit.bufferSize = 1MiB
**.relayUnit.highWatermark = 512KiB
**.relayUnit.pauseUnits = 300 # pause for 300*512 bit (19200 byte) time
**.relayUnit.addressTableFile = ""
**.relayUnit.numCPUs = 2
**.relayUnit.processingTime = 2us

```

EXECUTION:

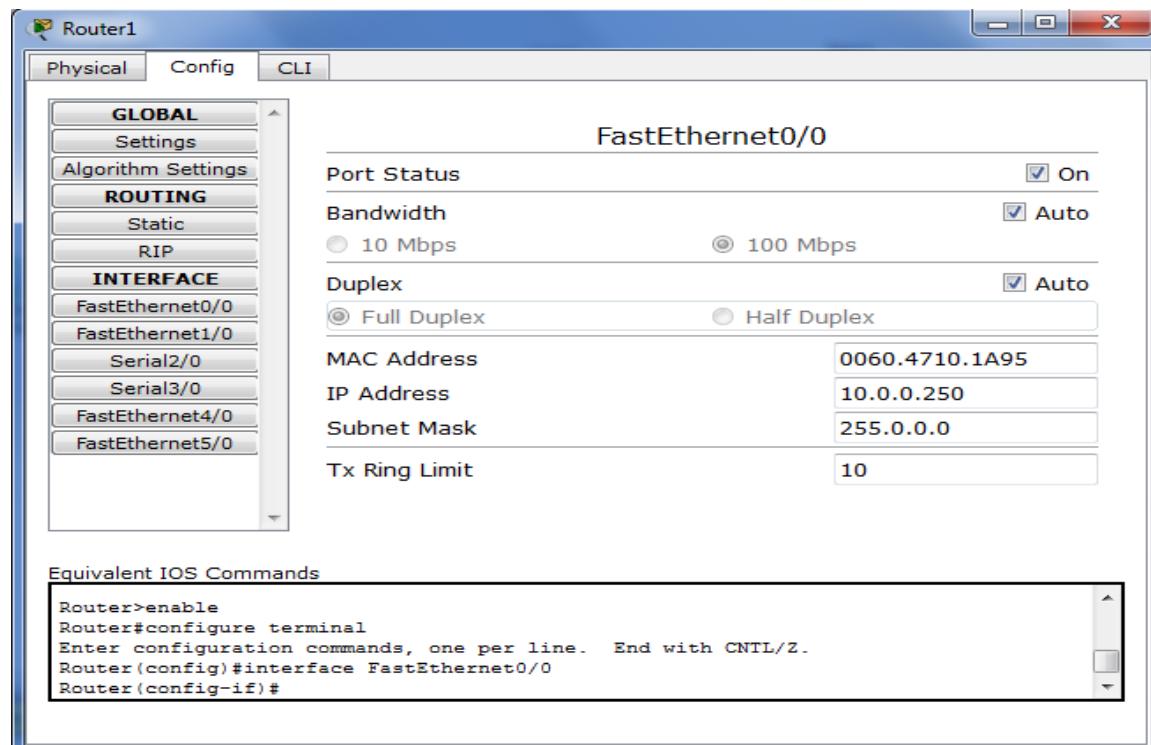
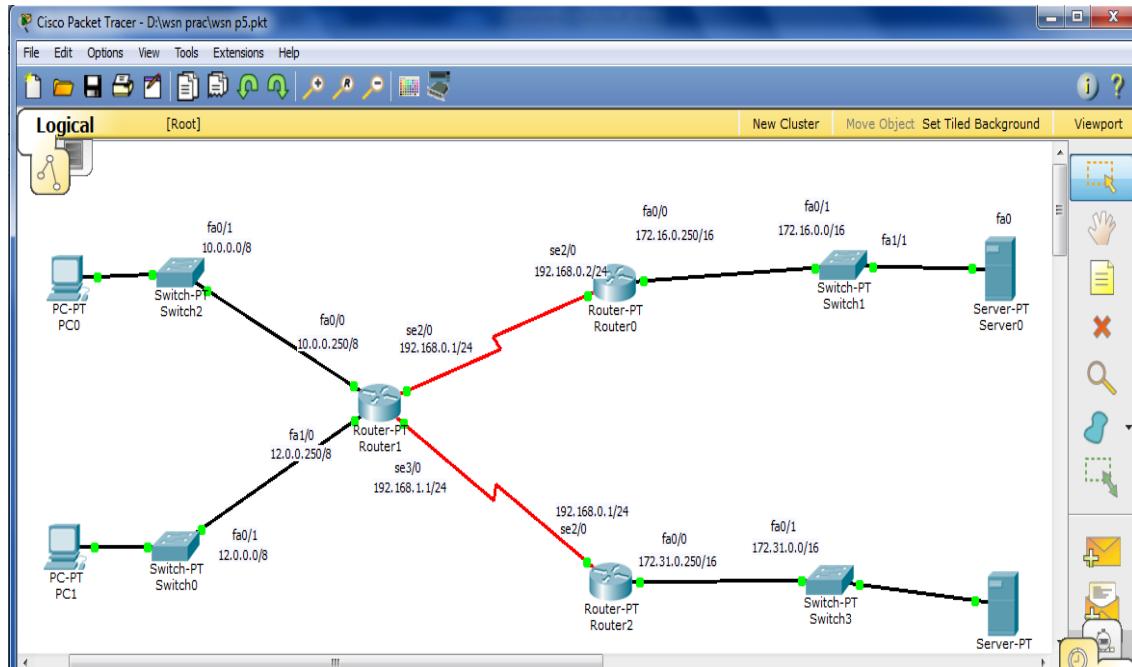
Now try to execute by right click on ned file Run as-1-Omnet++ simulation.

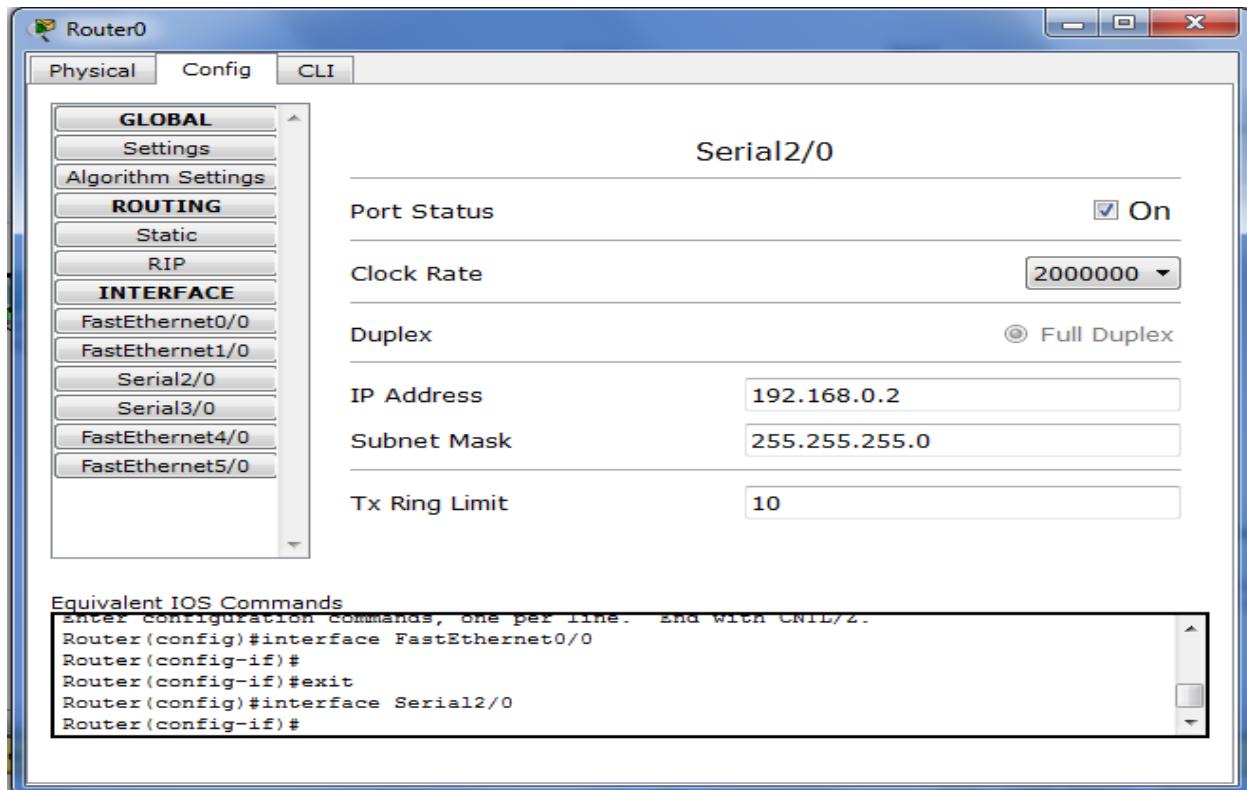
OUTPUT:



PRACTICAL 5

Aim :- Understanding , Reading and Analyzing Routing Table of a network.





The screenshot shows the IOS Command Line Interface for Router1. The terminal window displays the following output:

```

*LINEPROTO-5-UPDOWN: Line protocol on Interface Serial3/0, changed state to up

Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface FastEthernet0/0
Router(config-if)#exit
Router(config)#exit
Router#
*SYS-5-CONFIG_I: Configured from console by console

Router#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
      * - candidate default, U - per-user static route, o - ODR
      P - periodic downloaded static route

Gateway of last resort is not set

C    10.0.0.0/8 is directly connected, FastEthernet0/0
C    12.0.0.0/8 is directly connected, FastEthernet1/0
C    192.168.0.0/24 is directly connected, Serial2/0
C    192.168.1.0/24 is directly connected, Serial3/0
Router#

```

PRACTICAL 6

Aim : Create a basic MANET implementation simulation for Packet animation and Packet Trace.

Steps for practical:

1. Then open inet/examples/
2. Right click on manetrouting -create new folder as MobileNet.
3. Right click on your newly created folder and select NED file. Give name as Net1.
select new adhoc mobility wireless network wizard

```

package inet.examples.manetrouting.mymanet;

// numOfHosts: 10
// parametric: true
// static: false

import inet.networklayer.autorouting.ipv4.I Pv4NetworkConfigurator;
import inet.nodes.inet.AdhocHost;
import inet.world.radio.ChannelControl;

network Net1
{
    parameters:
        int numHosts;
    submodules:
        host[numHosts]: AdhocHost
        {
            parameters:
                @display("r=,,#707070");
        }

        channelControl: ChannelControl
        {
            parameters:
                @display("p=60,50");
        }

        configurator: IPv4NetworkConfigurator
        {
            @display("p=140,50");
        }
}

```

a file omnetpp.ini will be created with the following code

```

[General]
network = Net1
#record-eventlog = true
#eventlog-message-detail-pattern = *:(not declaredOn(cMessage) and not declaredOn(cNamedObject)
and not declaredOn(cObject))

*.numHosts = 10

num-rngs = 3
**.mobility.rng-0 = 1
**.wlan[*].mac.rng-0 = 2

```

```
#debug-on-errors = true

tkenv-plugin-path = ../../etc/plugins

**.channelNumber = 0

# channel physical parameters
*.channelControl.carrierFrequency = 2.4GHz
*.channelControl.pMax = 2.0mW
*.channelControl.sat = -110dBm
*.channelControl.alpha = 2
*.channelControl.numChannels = 1

# mobility
**.host[*].mobilityType = "MassMobility"
**.mobility.constraintAreaMinZ = 0m
**.mobility.constraintAreaMaxZ = 0m
**.mobility.constraintAreaMinX = 0m
**.mobility.constraintAreaMinY = 0m
**.mobility.constraintAreaMaxX = 600m
**.mobility.constraintAreaMaxY = 400m
**.mobility.changeInterval = truncnormal(2s, 0.5s)
**.mobility.changeAngleBy = normal(0deg, 30deg)
**.mobility.speed = truncnormal(20mps, 8mps)
**.mobility.updateInterval = 100ms

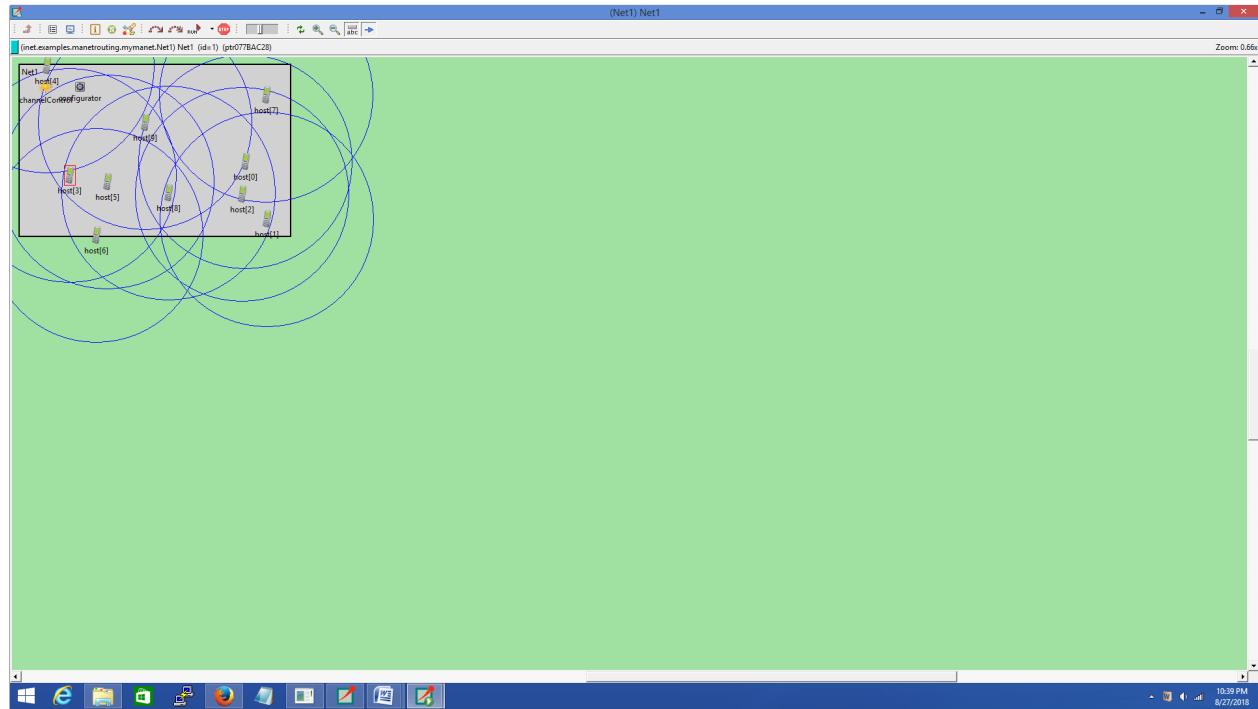
# ping app (host[0] pinged by others)
*.host[0].pingApp[0].destAddr =
**.host[*].numPingApps = 1
*.host[*].pingApp[0].destAddr = "host[0]"
*.host[*].pingApp[0].startTime = uniform(1s,5s)
*.host[*].pingApp[0].printPing = true

# nic settings
**.wlan[*].bitrate = 2Mbps

**.wlan[*].mgmt.frameCapacity = 10
**.wlan[*].mac.address = "auto"
**.wlan[*].mac.maxQueueSize = 14
**.wlan[*].mac.rtsThresholdBytes = 3000B
**.wlan[*].mac.retryLimit = 7
**.wlan[*].mac.cwMinData = 7
**.wlan[*].mac.cwMinMulticast = 31

**.wlan[*].radio.transmitterPower = 2mW
**.wlan[*].radio.thermalNoise = -110dBm
**.wlan[*].radio.sensitivity = -85dBm
**.wlan[*].radio.pathLossAlpha = 2
**.wlan[*].radio.snrThreshold = 4dB
```

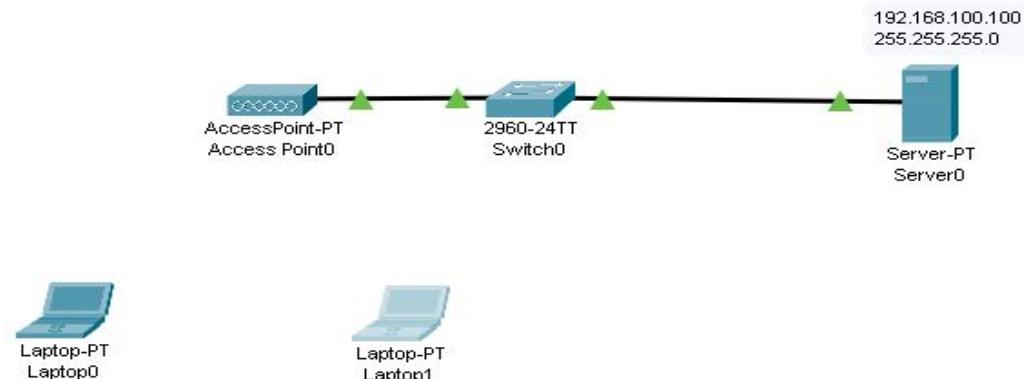
Right click on ned file and run it as omnetpp simulation



PRACTICAL 7

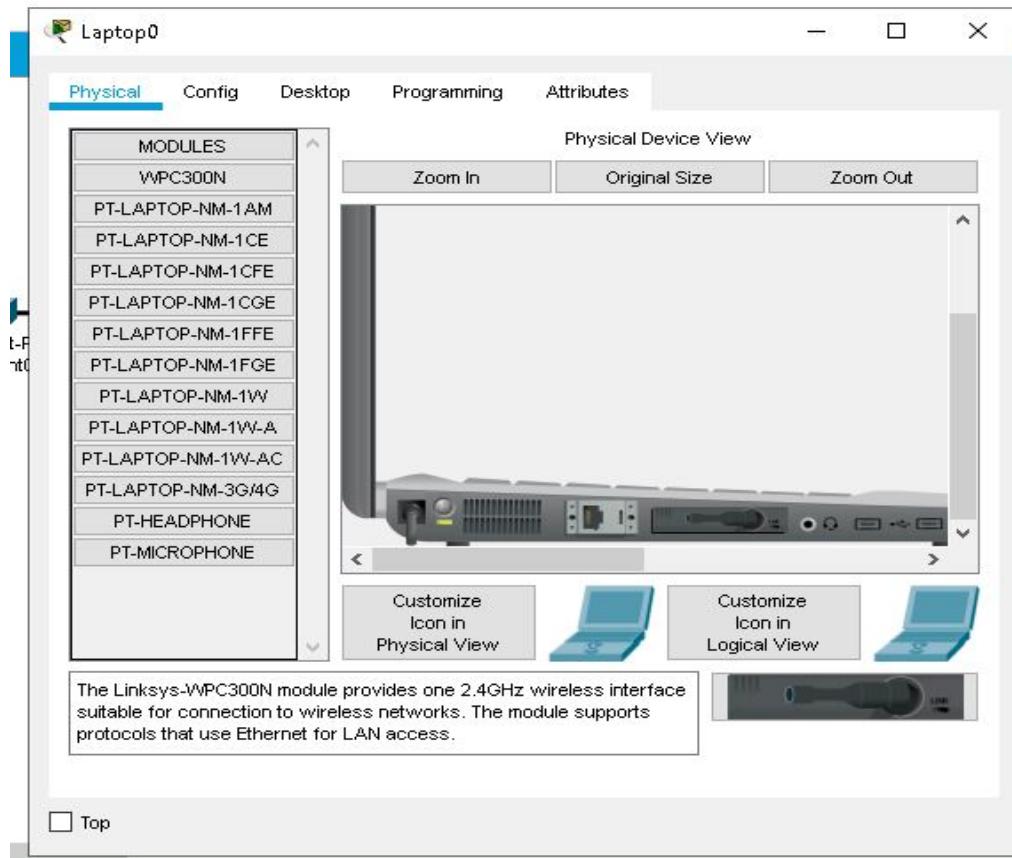
Aim : Implementation of wireless sensor network simulation.

- 1) Create the following topology.

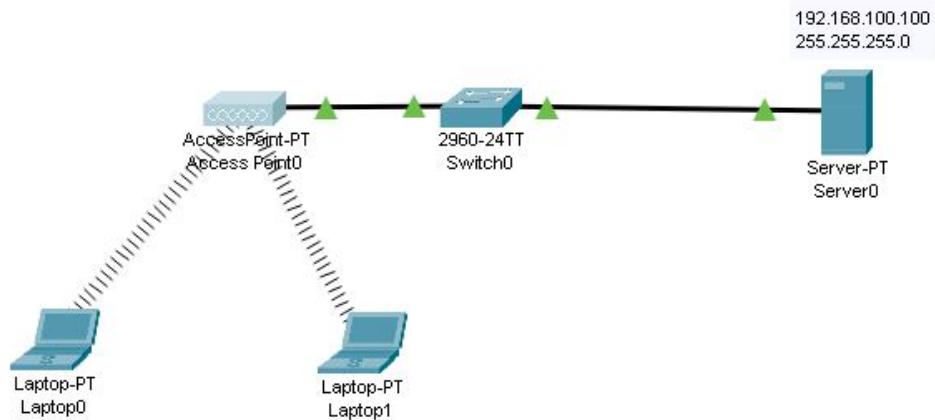


Assign the IP address and mask of server as 192.168.100.100 and 255.255.255.0 respectively.

- 2) To implement wireless connectivity, change the module of laptop devices as shown in the figure. Repeat for both the laptop.

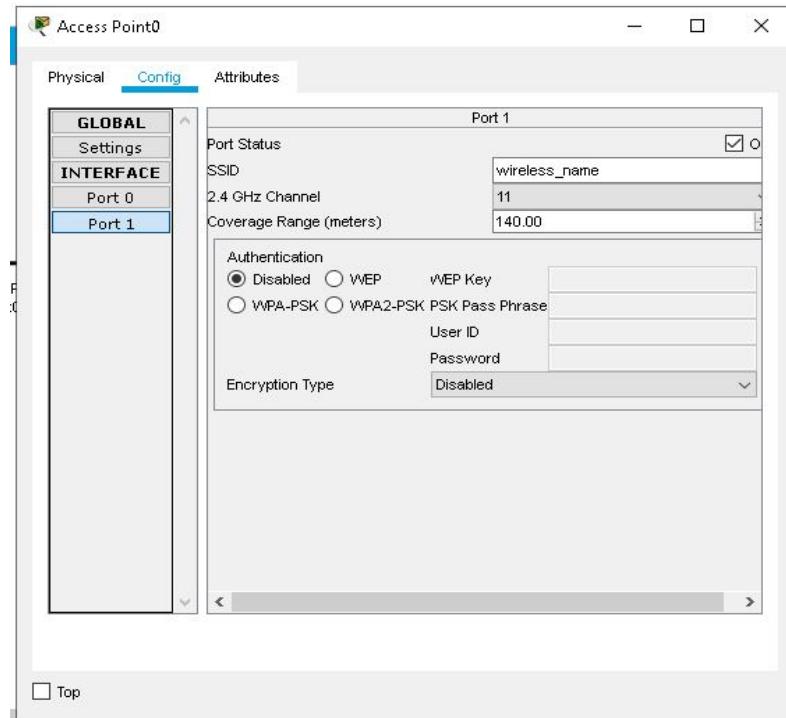


Once the changes are made and laptop is turned on, wireless implementation is displayed.



3) Change the SSID of server as follows:

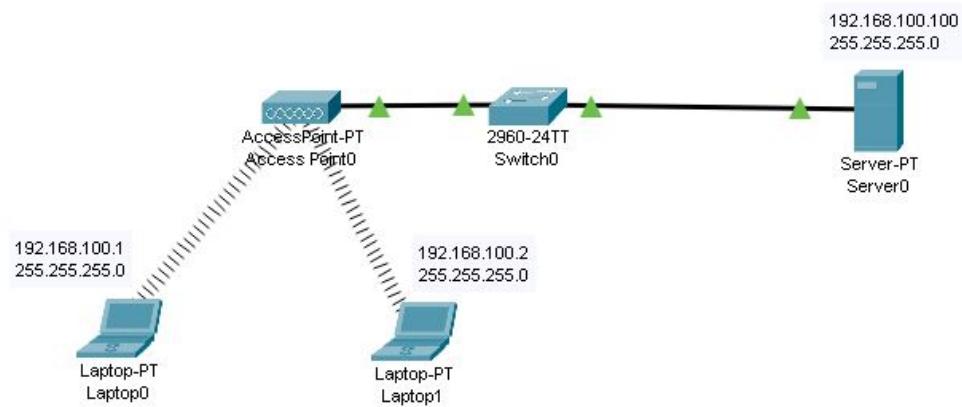
SSID : wireless_name 2.4 GHz channel : 11



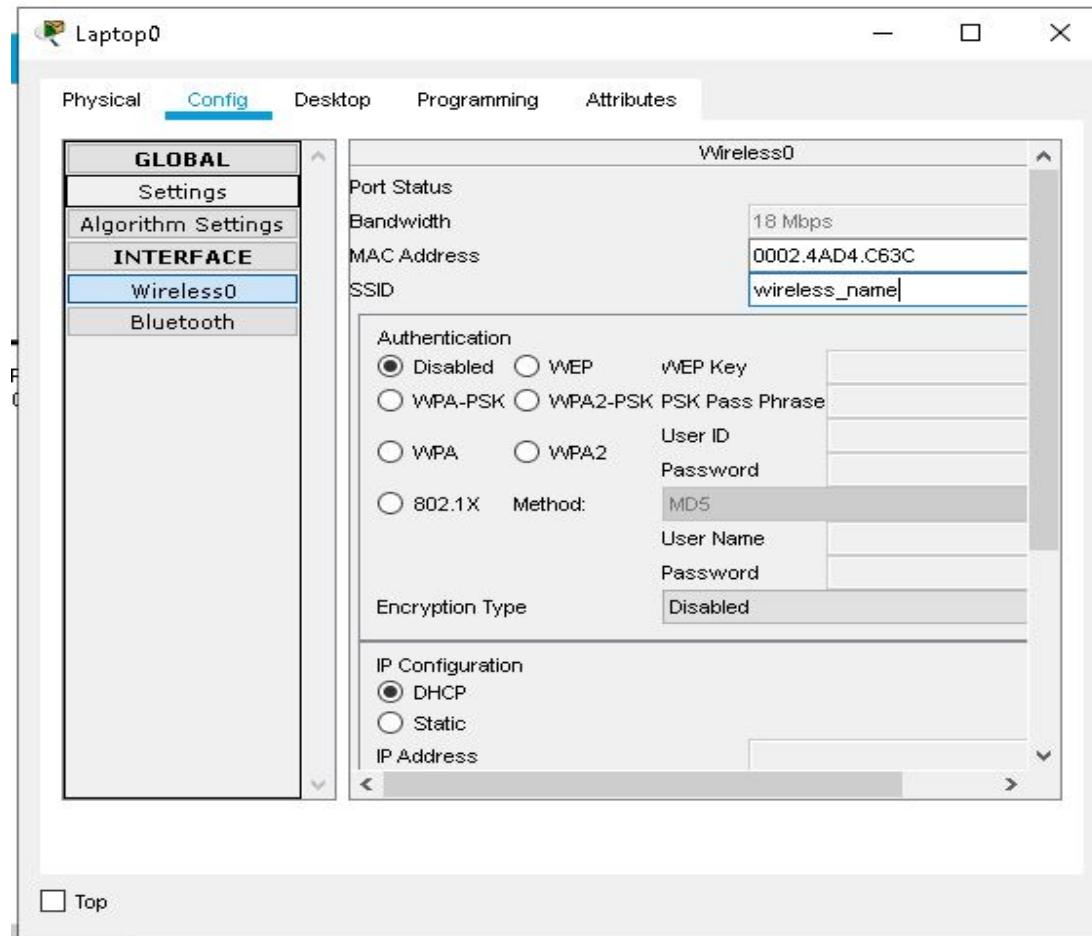
4) Set the IP address of laptop devices as

Laptop0 : 192.168.100.1

Laptop1 : 192.168.100.2



- 5) In the config, change the SSID of laptop device to wireless_name as shown below.
Do the same for another laptop device.



- 6) From the laptop device command prompt, ping the server through the following command:

```
Ping 192.168.100.100
```

Laptop0

Physical Config Desktop Programming Attributes

Command Prompt

```
Link-local IPv6 Address..... ::  
IP Address..... 0.0.0.0  
Subnet Mask..... 0.0.0.0  
Default Gateway..... 0.0.0.0  
  
Wireless0 Connection:  
Link-local IPv6 Address..... FE80::202:4AFF:FE04:C63C  
IP Address..... 192.168.100.1  
Subnet Mask..... 255.255.255.0  
Default Gateway..... 0.0.0.0  
  
C:\>ping 192.168.100.100  
  
Pinging 192.168.100.100 with 32 bytes of data:  
  
Reply from 192.168.100.100: bytes=32 time=35ms TTL=128  
Reply from 192.168.100.100: bytes=32 time=26ms TTL=128  
Reply from 192.168.100.100: bytes=32 time=25ms TTL=128  
Reply from 192.168.100.100: bytes=32 time=25ms TTL=128  
  
Ping statistics for 192.168.100.100:  
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),  
    Approximate round trip times in milli-seconds:  
        Minimum = 25ms, Maximum = 35ms, Average = 27ms  
  
C:\>
```

Top

7) From laptop1 ping laptop0.

Laptop1

Physical Config Desktop Programming Attributes

Command Prompt

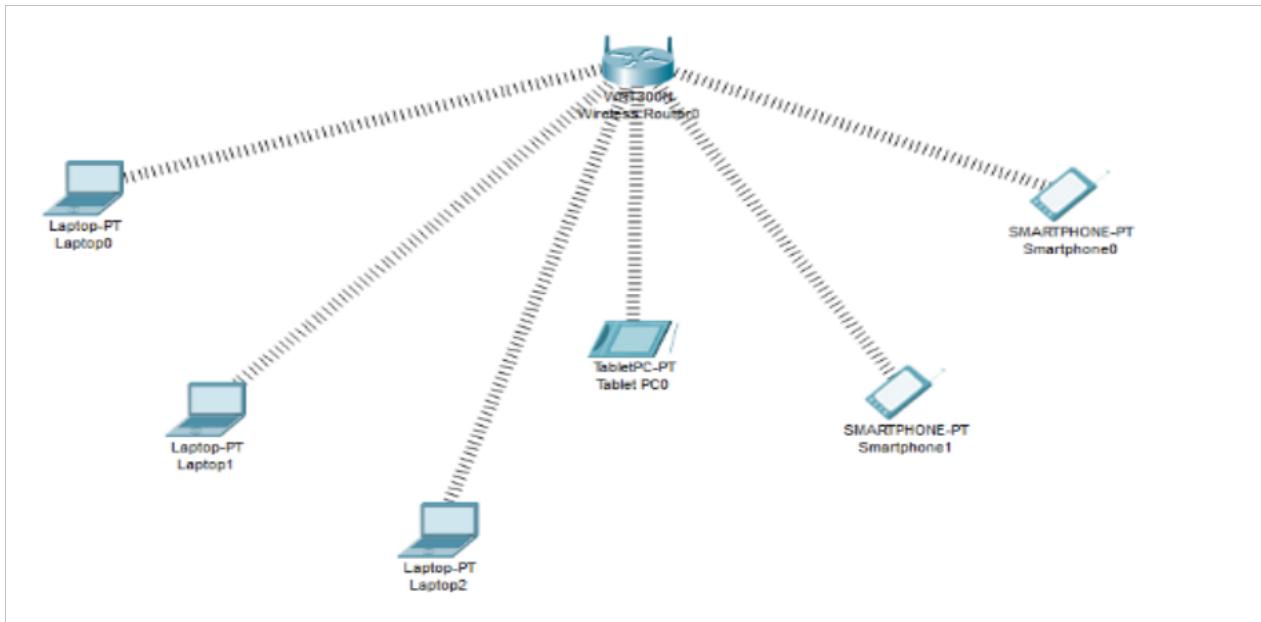
```
Packet Tracer PC Command Line 1.0  
C:\>  
  
ping 192.168.100.1  
  
Pinging 192.168.100.1 with 32 bytes of data:  
  
Reply from 192.168.100.1: bytes=32 time=26ms TTL=128  
Reply from 192.168.100.1: bytes=32 time=17ms TTL=128  
Reply from 192.168.100.1: bytes=32 time=15ms TTL=128  
Reply from 192.168.100.1: bytes=32 time=14ms TTL=128  
  
Ping statistics for 192.168.100.1:  
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),  
    Approximate round trip times in milli-seconds:  
        Minimum = 14ms, Maximum = 26ms, Average = 18ms  
  
C:\>
```

Top

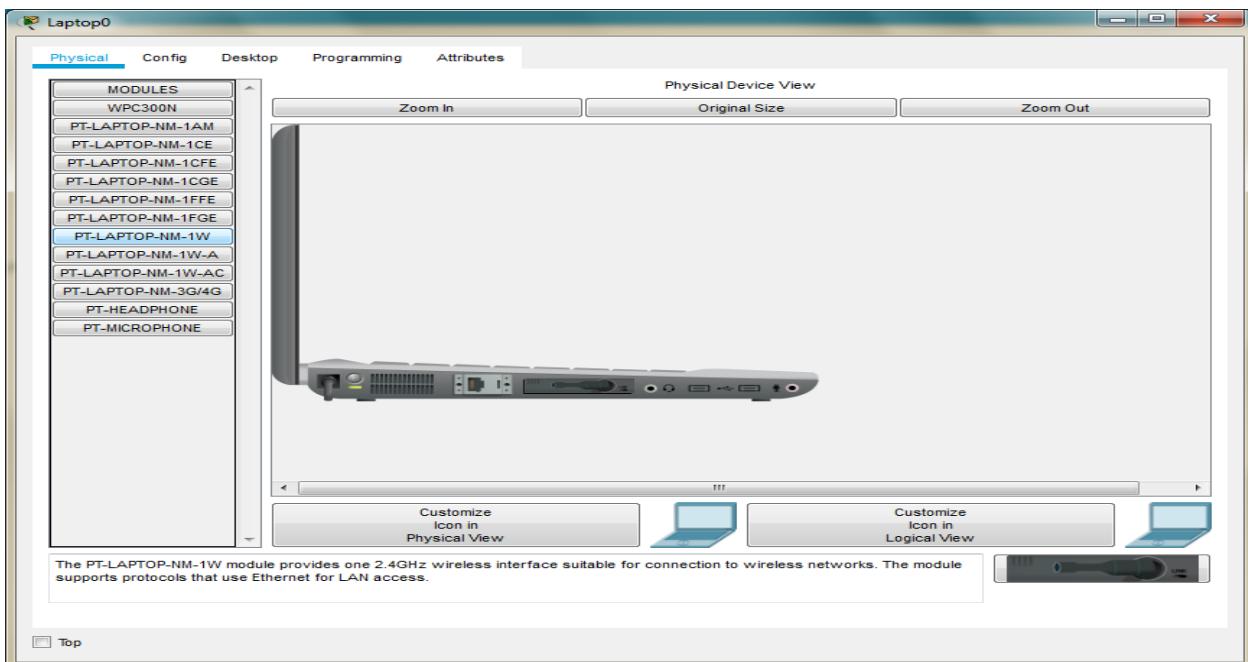
PRACTICAL 8

Aim: Create MAC protocol simulation implementation for wireless sensor Network.

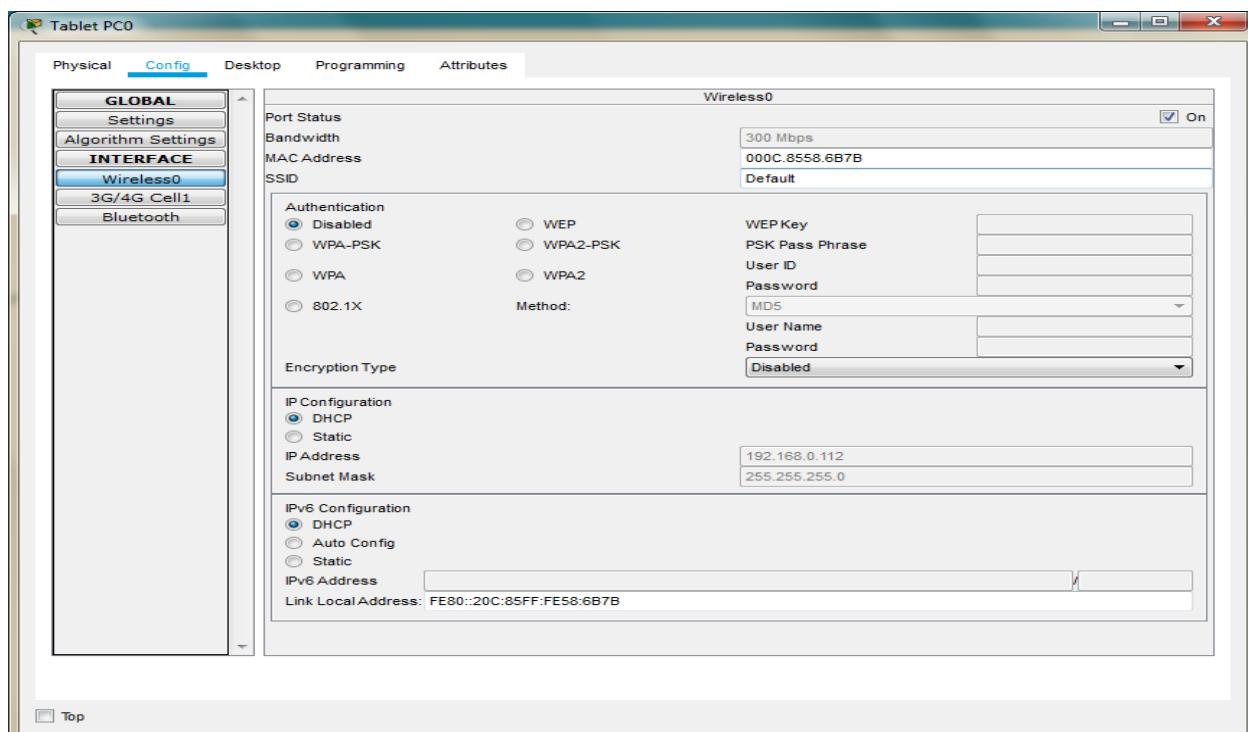
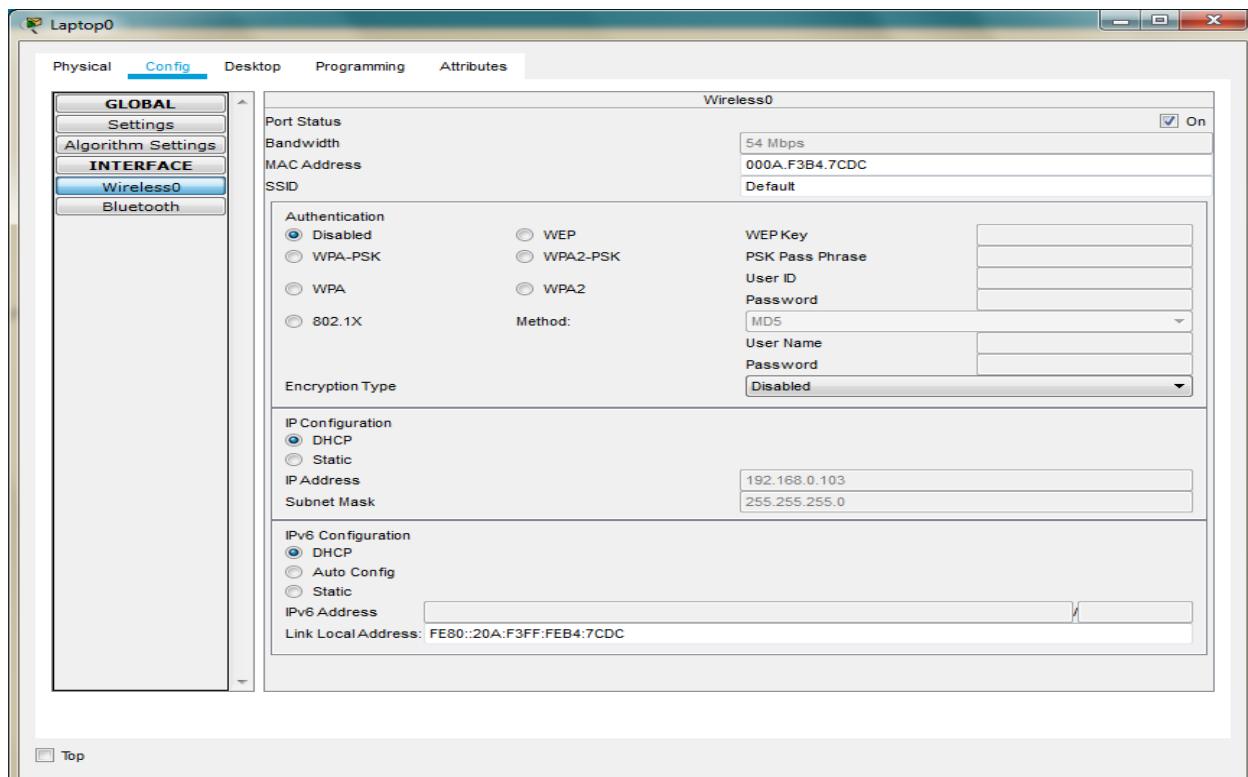
Consider the following topology

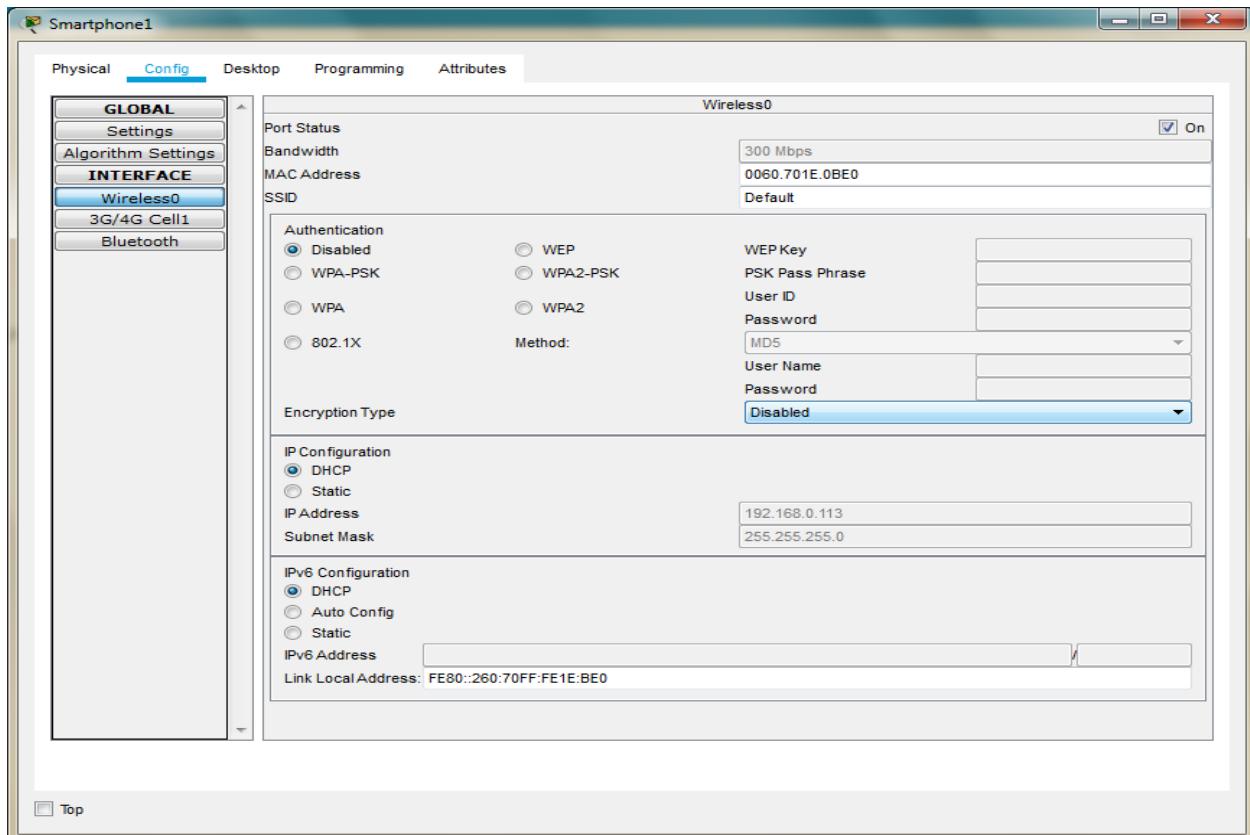


Adding the wireless interface to each Laptops



Copy the MAC address of each component as follows

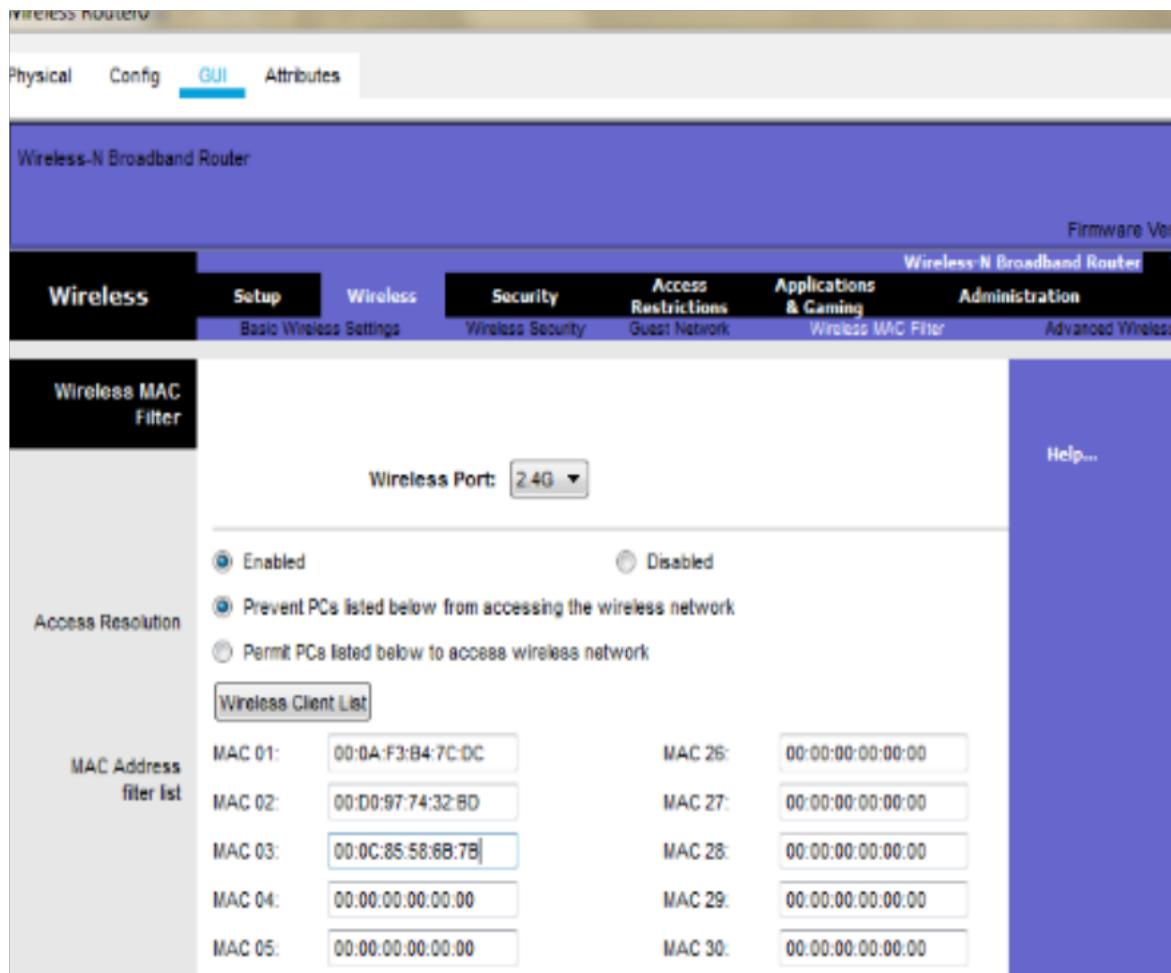




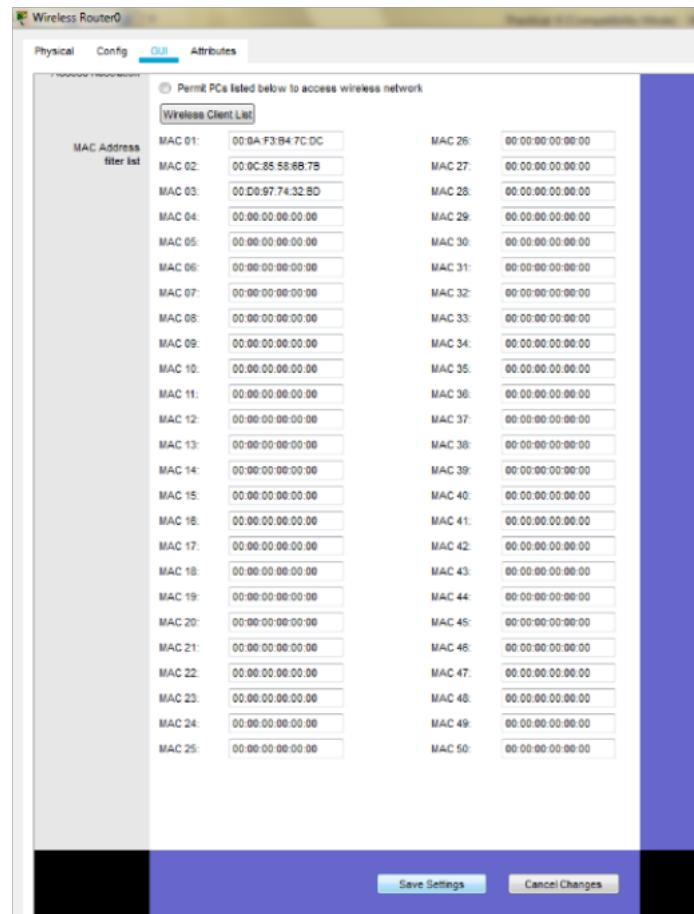
We note the following MAC addresses and convert them to the following form

Component	MAC Address	Converted MAC address
Laptop0	000A.F3B4.7CDC	00:0A:F3:B4:7C:DC
Laptop1	0001.4269.6539	00:01:42:69:65:39
Laptop2	0060.5CB8.B919	00:60:5C:B8:B9:19
TabletPC	000C.8558.6B7B	00:0C:85:58:6B:7B
SmartPhone0	00D0.9774.32BD	00:D0:97:74:32:BD
SmartPhone1	0060.701E.0BE0	00:60:70:1E:0B:E0

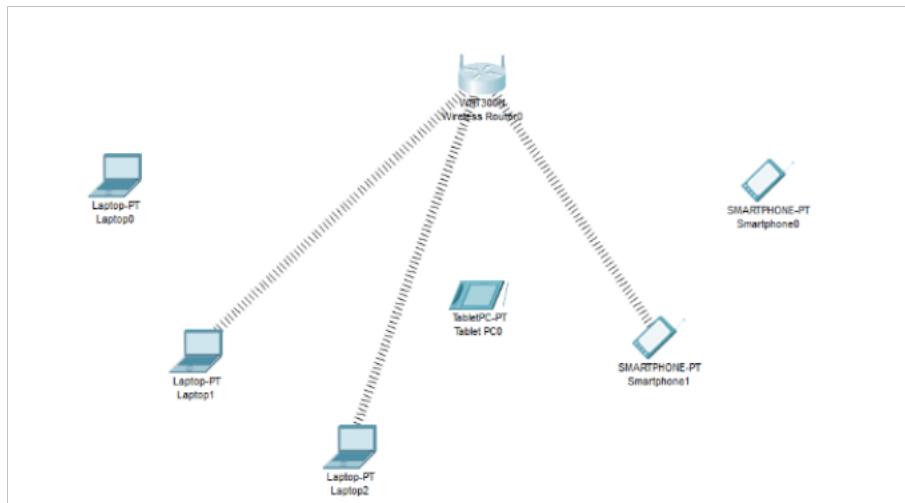
Now we add few addresses in the wireless MAC filter of the Wireless Router and then use the given options for either allow or deny the Wireless access



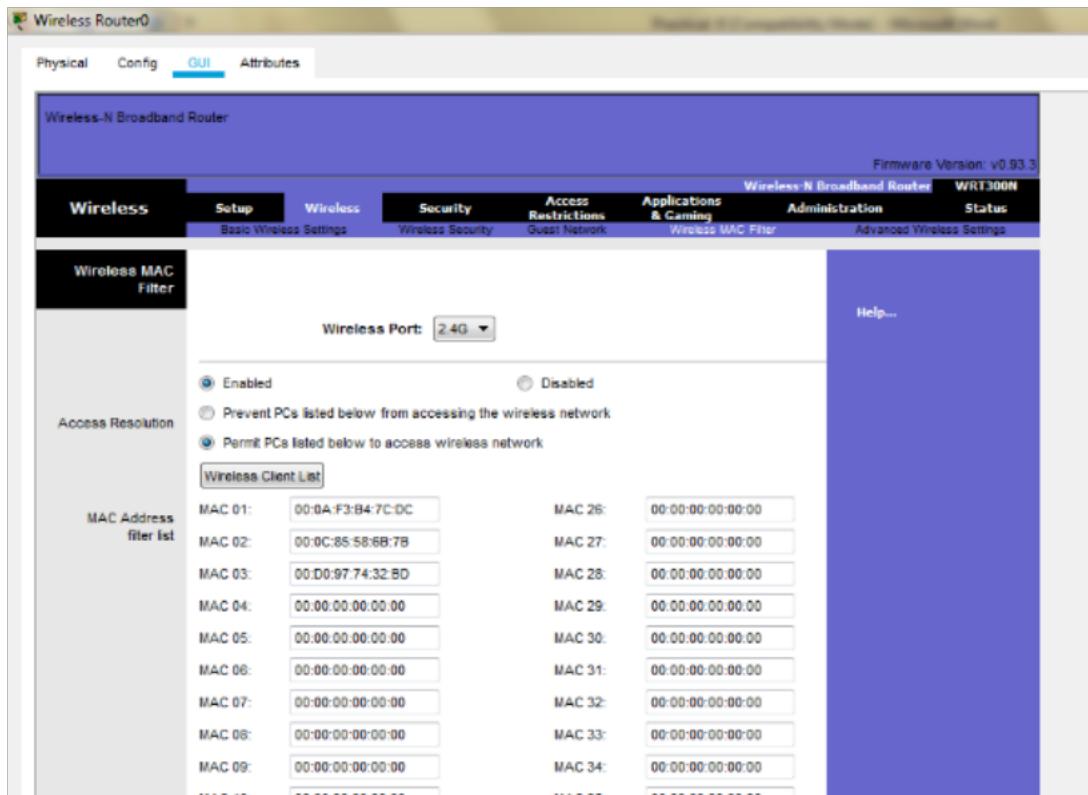
As seen in above screen shot we add the MAC address of Laptop0, TabletPC SmartPhone0 in the list so as to deny them accessing the Wireless network and then save the settings



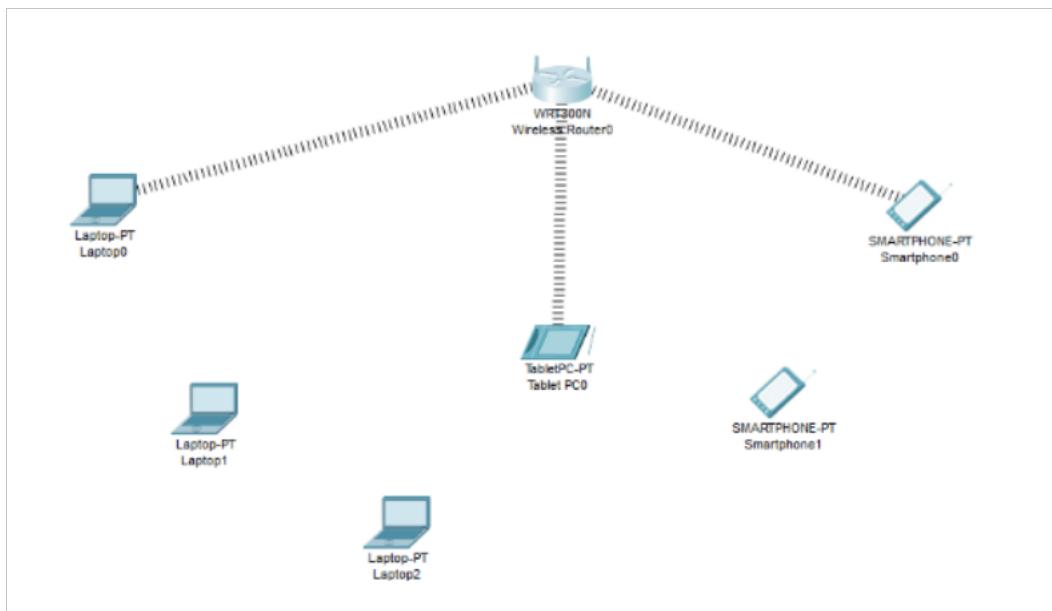
The result so obtained is as shown, the three devices denied any wireless connectivity



Similarly we can change the setting so that the above devices get wireless connectivity and the remaining devices do not get the wireless connectivity



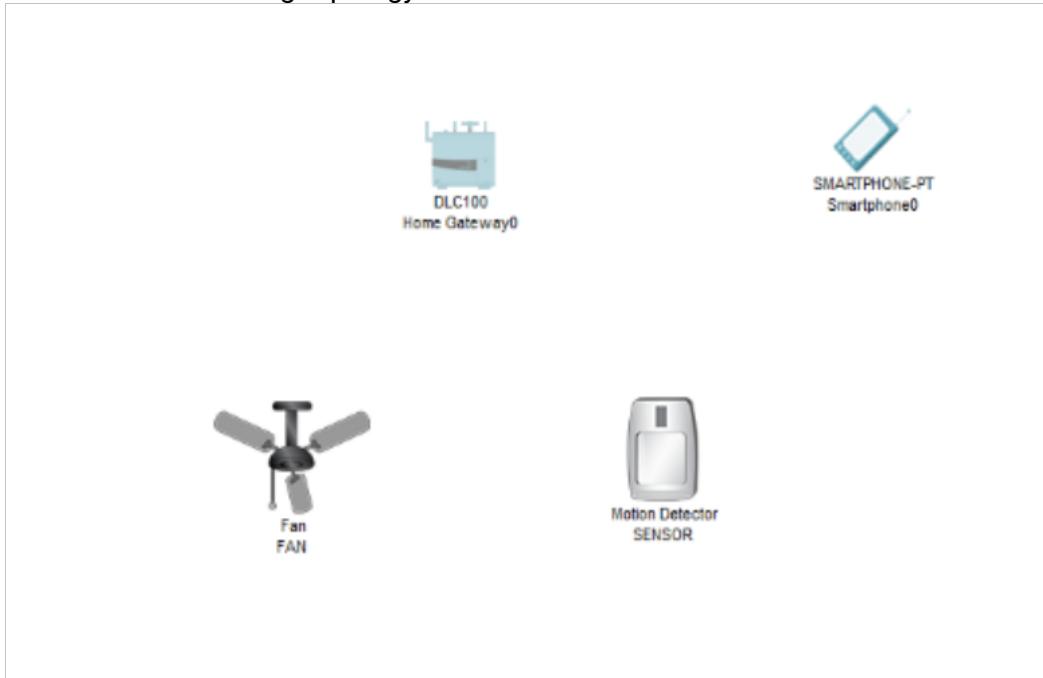
And save the setting and get the following



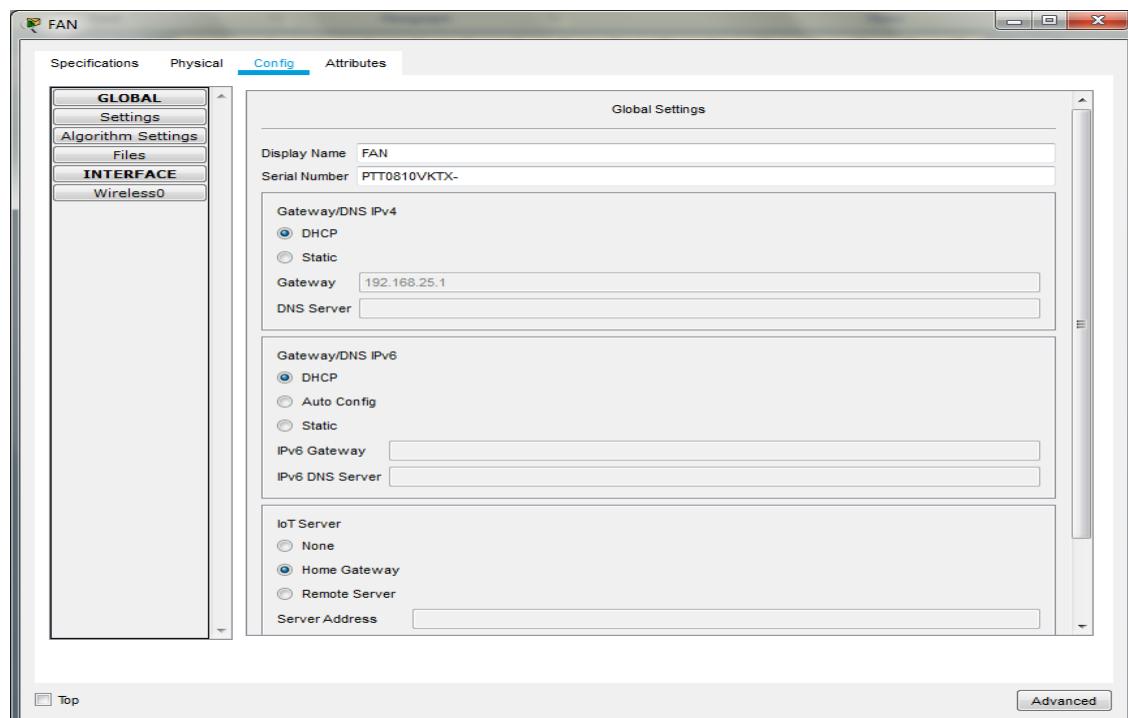
PRACTICAL 9

Aim: Simulate Mobile Adhoc Network with Directional Antenna

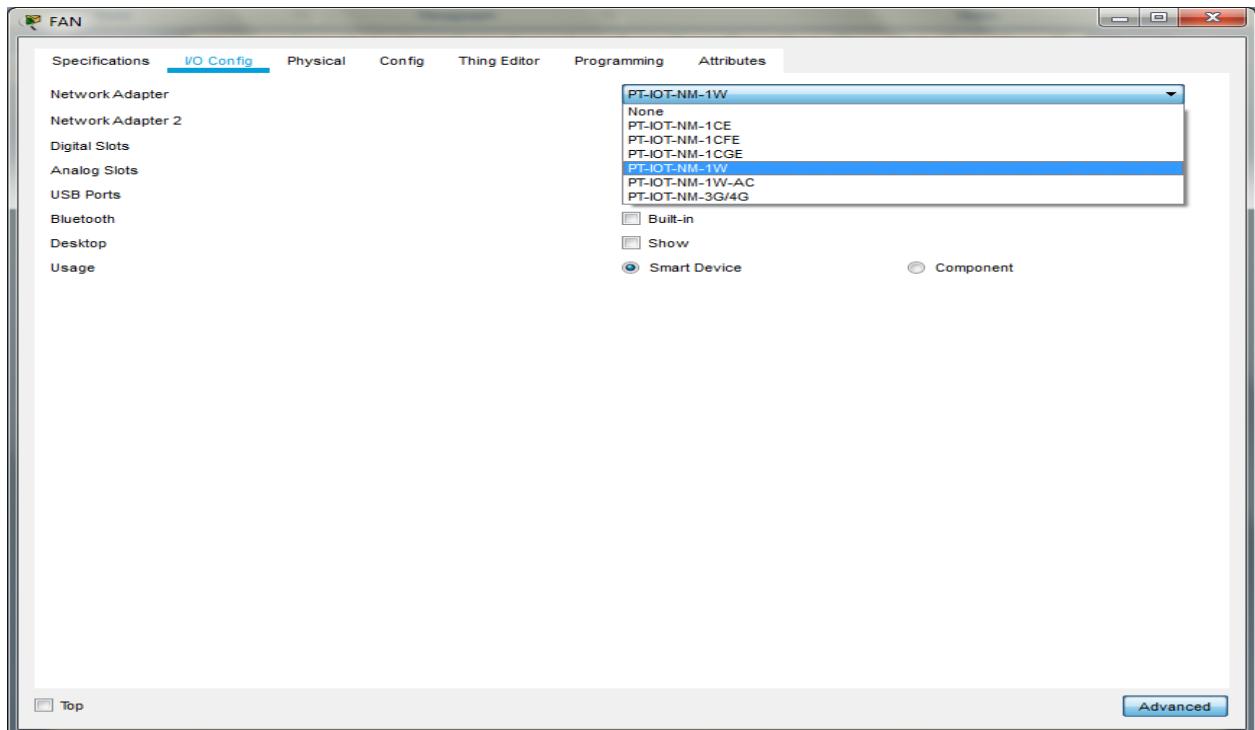
Consider the following topology



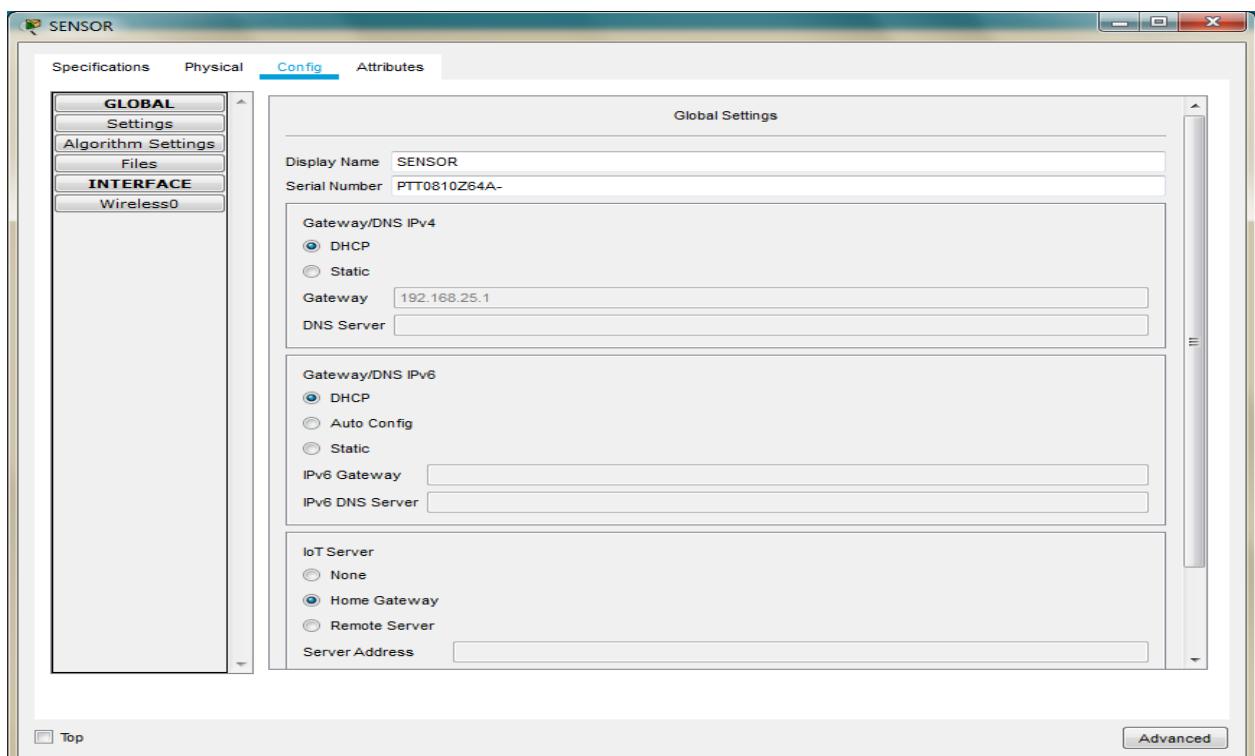
Click on the Fan and do the following



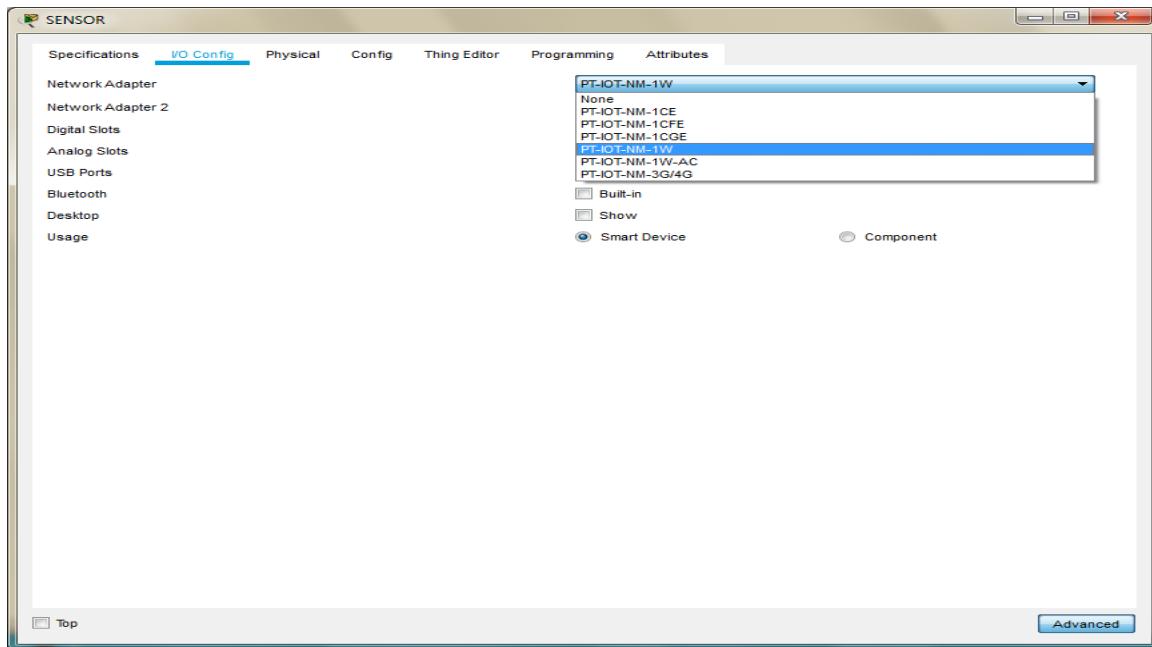
In the Advanced setting do the following for the Network adapter



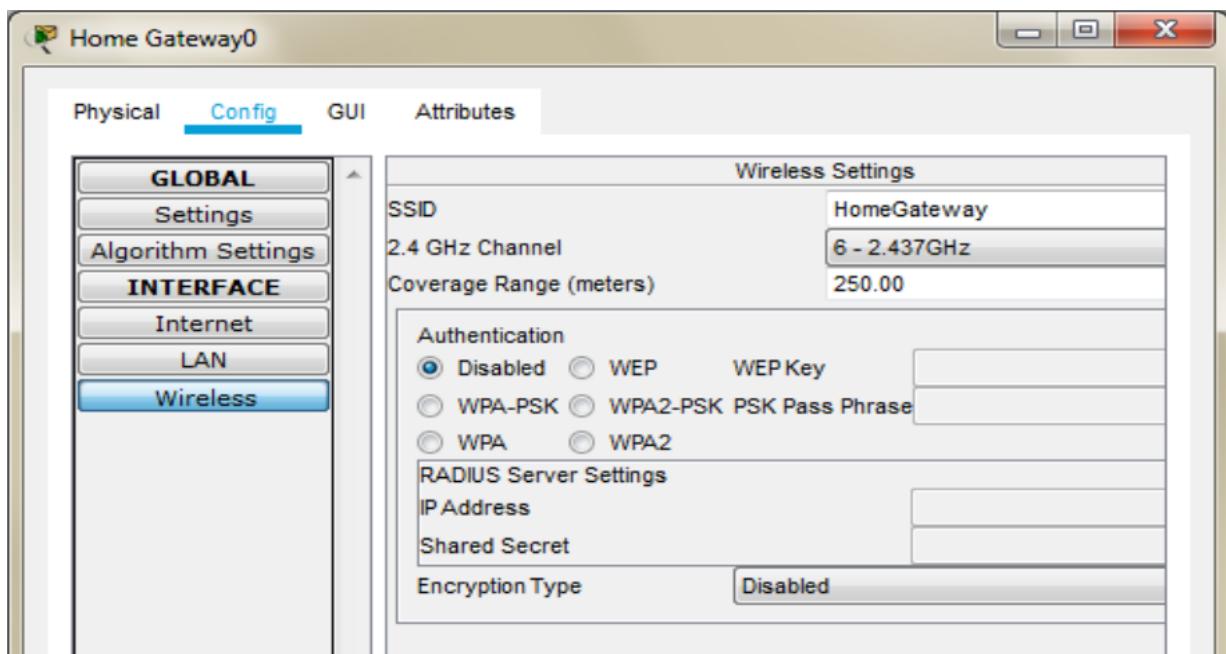
For the motion Detector sensor do the following



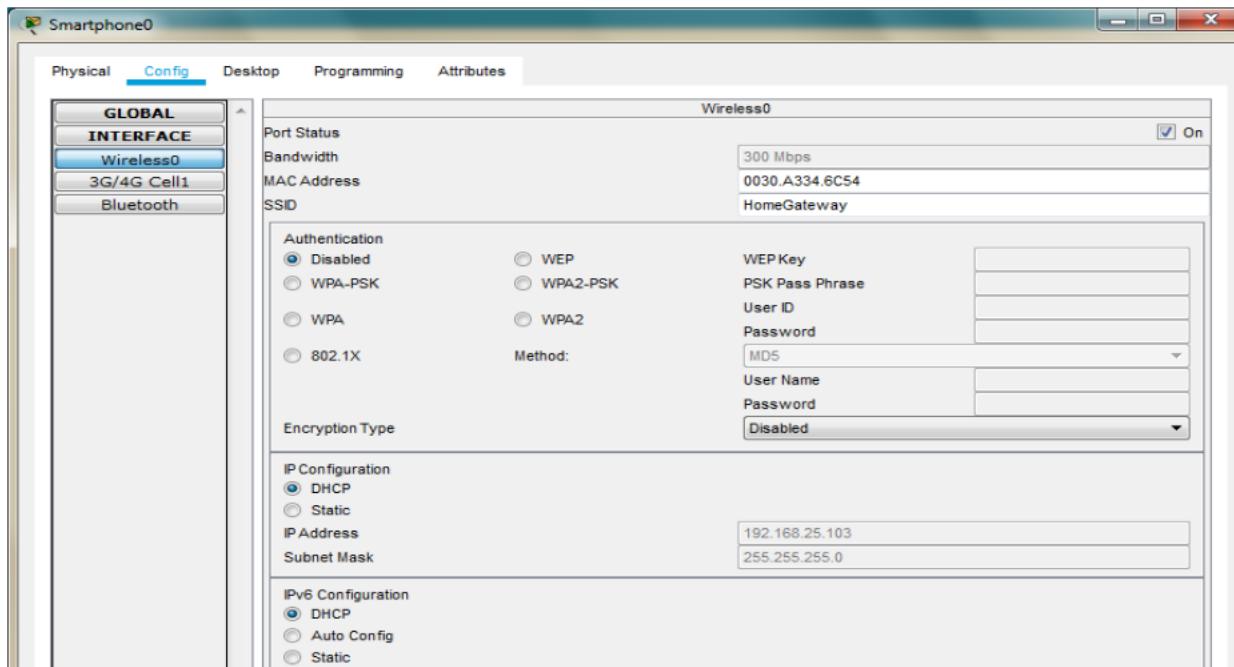
In the Advanced setting do the following for the Network adapter



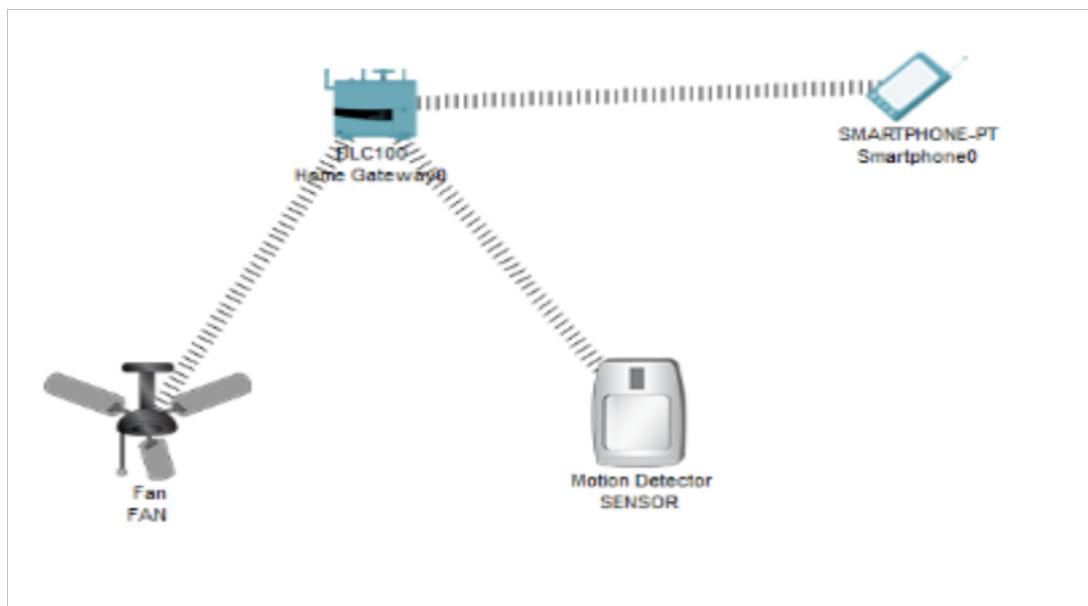
For the smartphone change the SSID to the SSID in the Home Gateway0



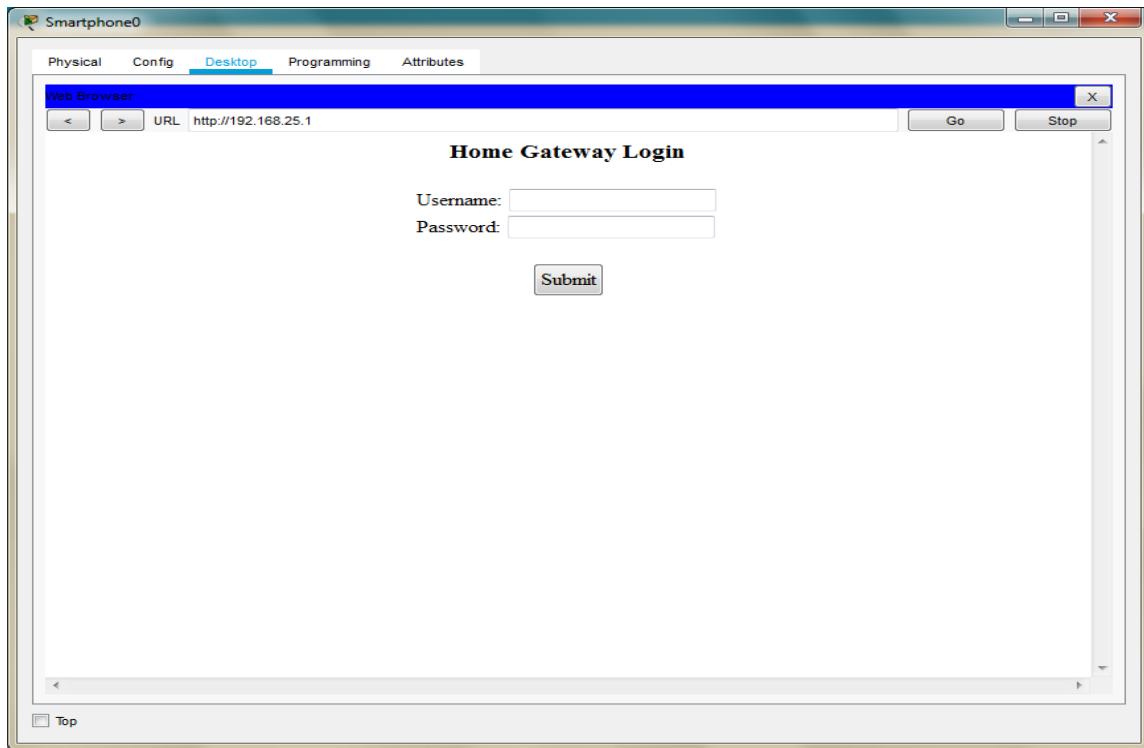
As seen above the SSID is HomeGateway, we use the same and set the SSID in the Smartphone



All the devices are now connected to the Home Gateway



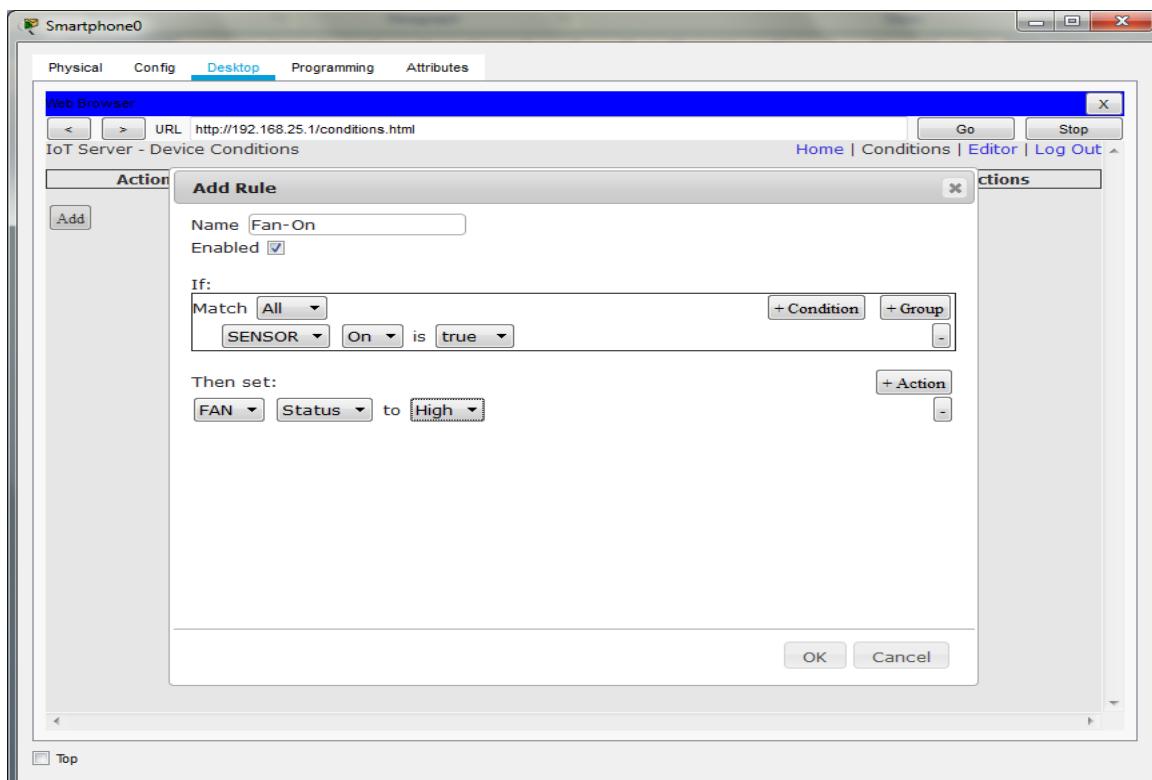
Now open the Web browser of the SmartPhone and type the IP address of the HomeGateway



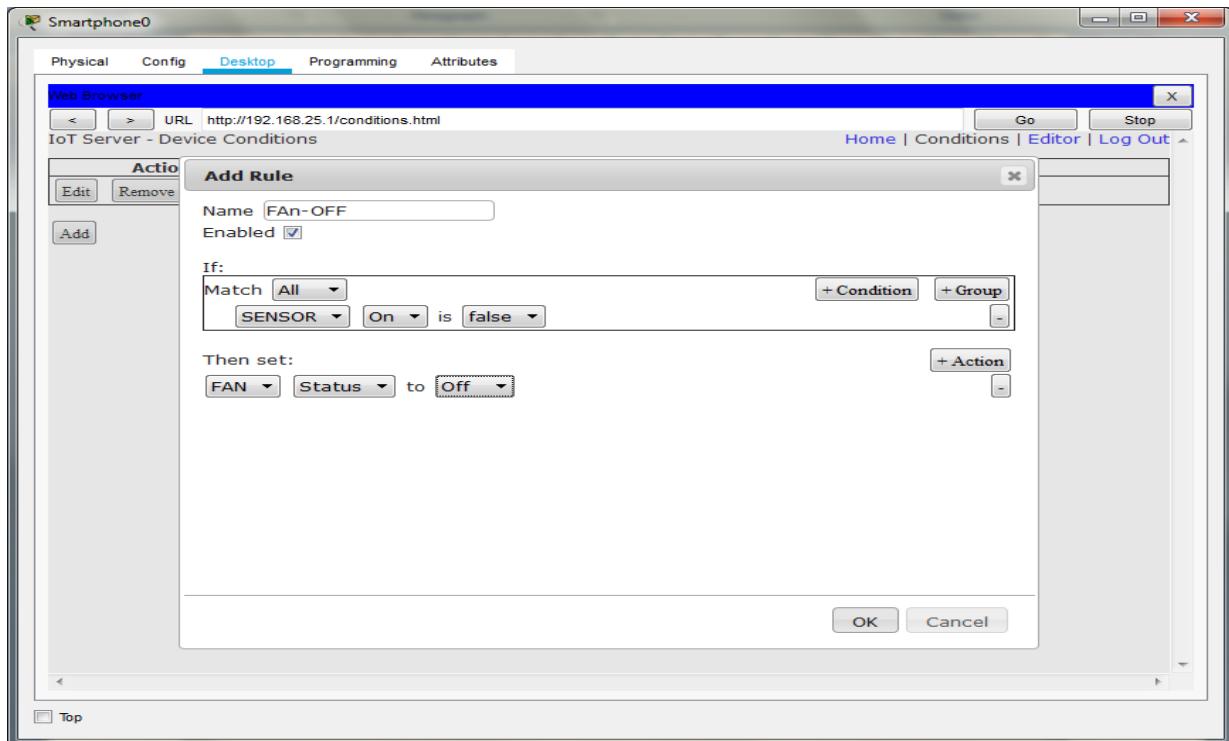
Username : admin

Password : admin

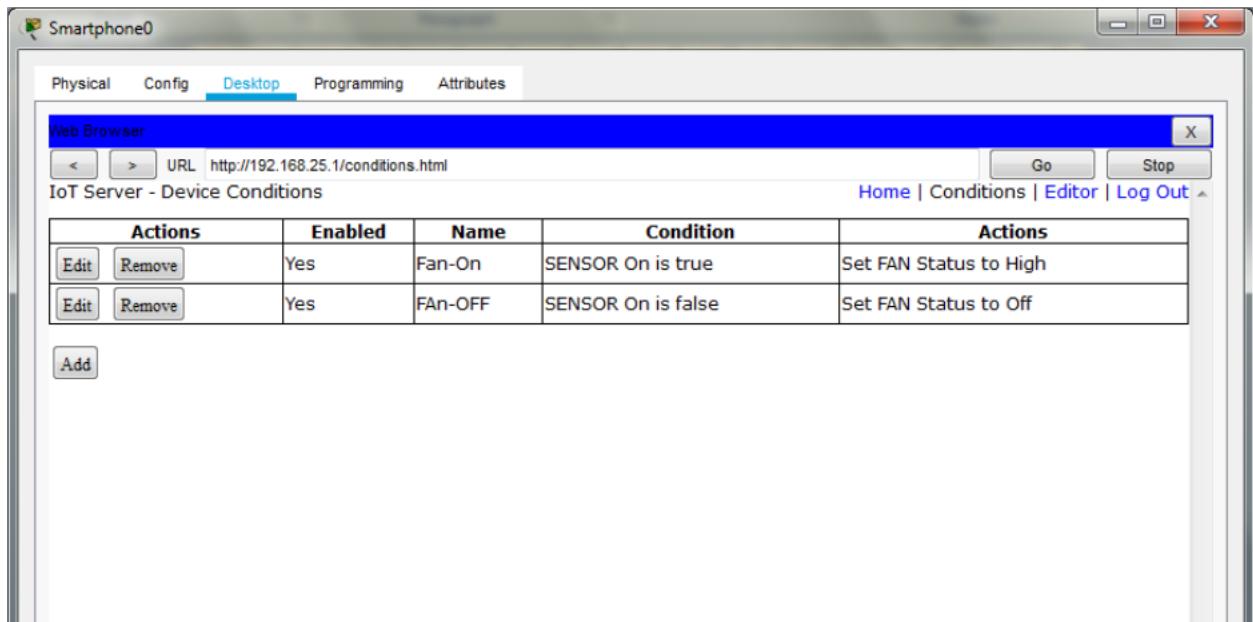
After logging click on conditions and do the following



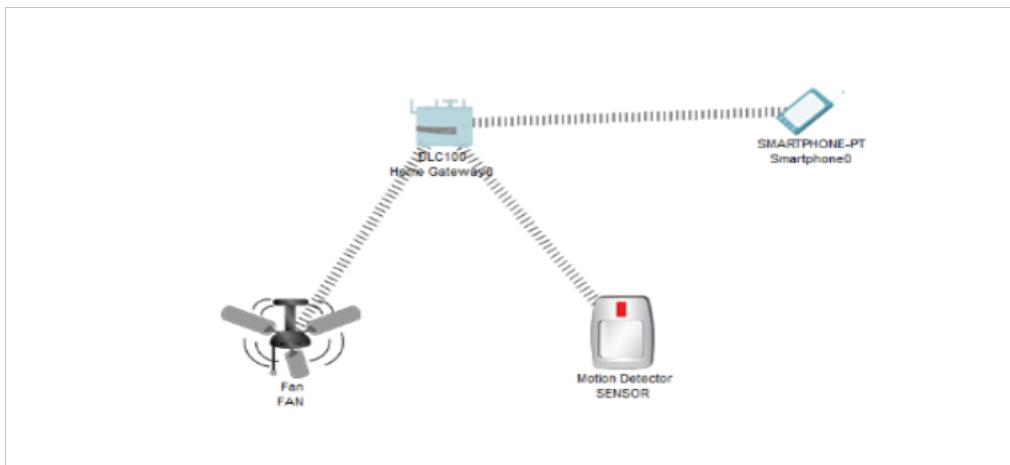
Add another condition as follows



Press the go button after adding the two conditions



In order to turn ON the fan Press the ALT key and left-click the mouse over the Sensor

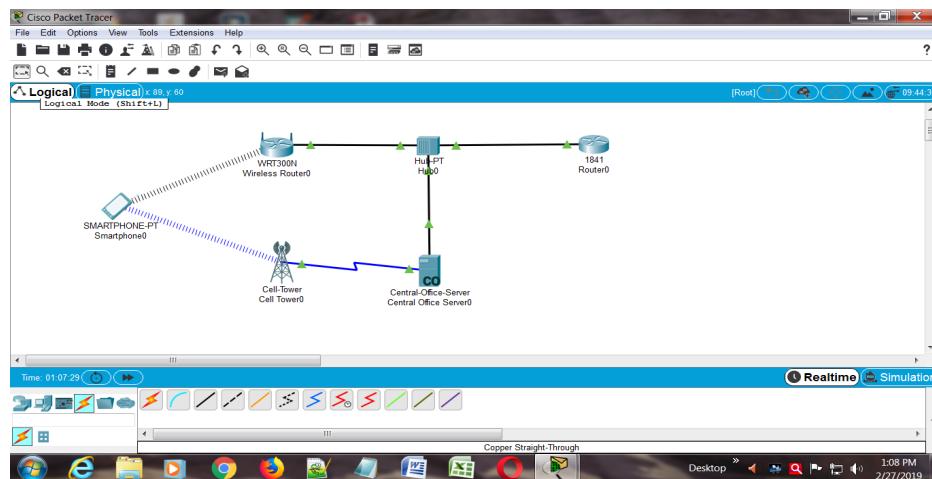


PRACTICAL 10

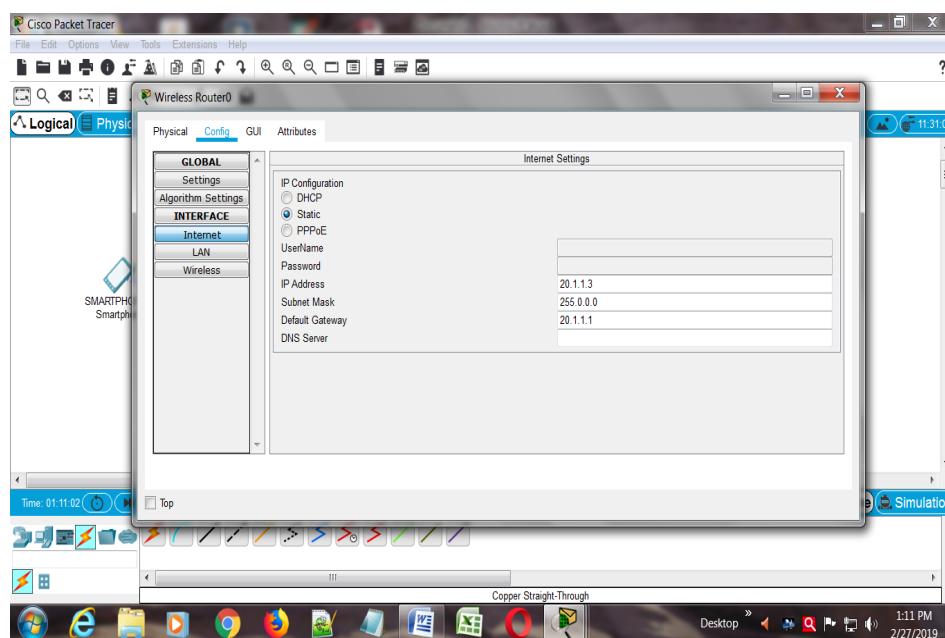
Aim :- Create a mobile netwrok using Cell Tower, Central Office Server, Web browser and Web Server. Simulate connection between them.

Steps :-

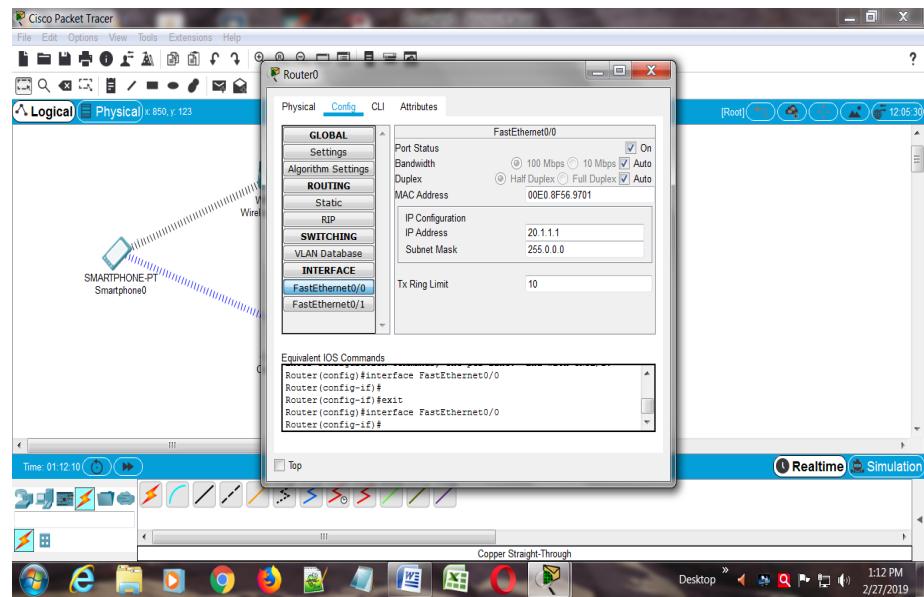
- 1) Create a network using smartphone, wireless router WRT300N, Hub-pt, 1841 Router, central-office-server, Cell-Tower.
- 2) Connect cell tower and central office server using coaxial cable.
- 3) Connect wireless router WRT300N, Hub-pt, 1841 Router, central-office-server using copper straight through wire.



- 4) Click on wireless router.in config tab select internet.in internet choose ip configuration as static and set ip address and default gateway.



5) Click on router 1841. In config tab select interface and give ip address.



6) Click on smartphone and ping router1841

