

## Subject : Edge and Fog Computing

### Experiment No. 7

**Title:** Edge-based data analytics algorithms

**Aim :** Develop and evaluate edge-based data analytics algorithms in an edge simulator.

**Theory:** Edge analytics is an approach to data collection and analysis in which an automated analytical computation is performed on data at a sensor, network switch or other device instead of waiting for the data to be sent back to a centralized data store. Edge analytics has gained attention as the internet of things (IoT) model of connected devices has become more prevalent. In many organizations, streaming data from manufacturing systems, industrial equipment, pipelines and other remote devices connected to the IoT creates a massive glut of operational data, which can be difficult -- and expensive -- to manage. By running the data through an analytics algorithm as it's created -- at the edge of a corporate network -- companies can set parameters on what information is or isn't worth sending to a cloud or on-premises data store for later use.

Analyzing data as it's generated can also decrease latency in the decision-making process on connected devices. For example, if the sensor from a manufacturing system detects the likely failure of a specific part, business rules built into the analytics algorithm interpreting the data at the network edge can automatically shut down the machine and send an alert to plant managers so that the part can be replaced. This can save time compared with transmitting the data to a central location for processing and analysis, potentially letting organizations reduce or avoid unplanned equipment downtime. Real-time analysis of data facilitates real-time decision-making. An example of this kind of immediate turnaround includes rerouting workloads to machines in a factory when issues in a particular machine, detected by IoT edge devices, predict that the performance of that machine will degrade. Edge computing consists of clusters of servers that located close to the IoT sensing devices for timely response to service requests while conserving bandwidth consumption rate and latency delay. On the other hand, IoT sensing devices can offload their sensed data to the edge servers when the load exceed their capabilities. The proximity between edge and the IoT devices, provides an opportunity to control the latency delay between the IoT devices and the traditional cloud. In addition, the sensed data collected from IoT devices is stored and immediately processed by the edge servers, with only a fraction of the data being sent to a cloud data center for long-term processing.



## Subject : Edge and Fog Computing Experiment No. 7

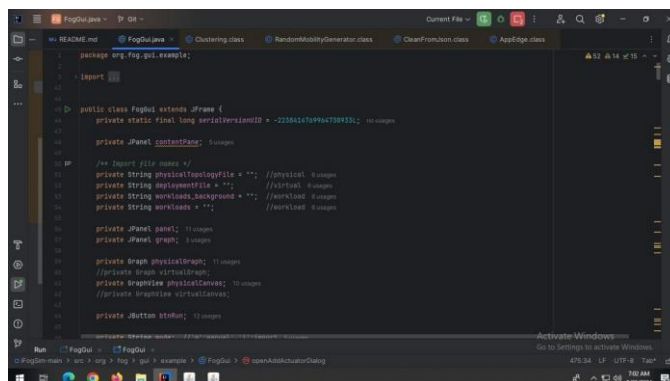


How is edge analytics used?

One of the most common use cases for edge analytics is monitoring edge devices. This is particularly true for IoT devices. A data analytics platform might be deployed for monitoring a large collection of devices and ensuring the devices are functioning normally. If a problem does occur, an edge analytics platform might be able to take corrective action automatically. If automatic remediation isn't possible, then the platform could instead provide the IT staff with actionable insights to help them fix the problem.

### Procedure:

#### Step 1: Open ifogsimulator



#### Step 2 : AFTER GETTING OUTPUT WINDOW (FOG TOPOLOGY CREATOR)



**AC Patil**  
College of Engineering

Jawahar Education Society's  
**A.C. Patil College of Engineering, Kharghar**  
**Department of CSE(IoT-CS BC)**

## Subject : Edge and Fog Computing Experiment No. 7

ADD FOG DEVICE with required specification

NAME: DATA CENTER 1

LEVEL: 0

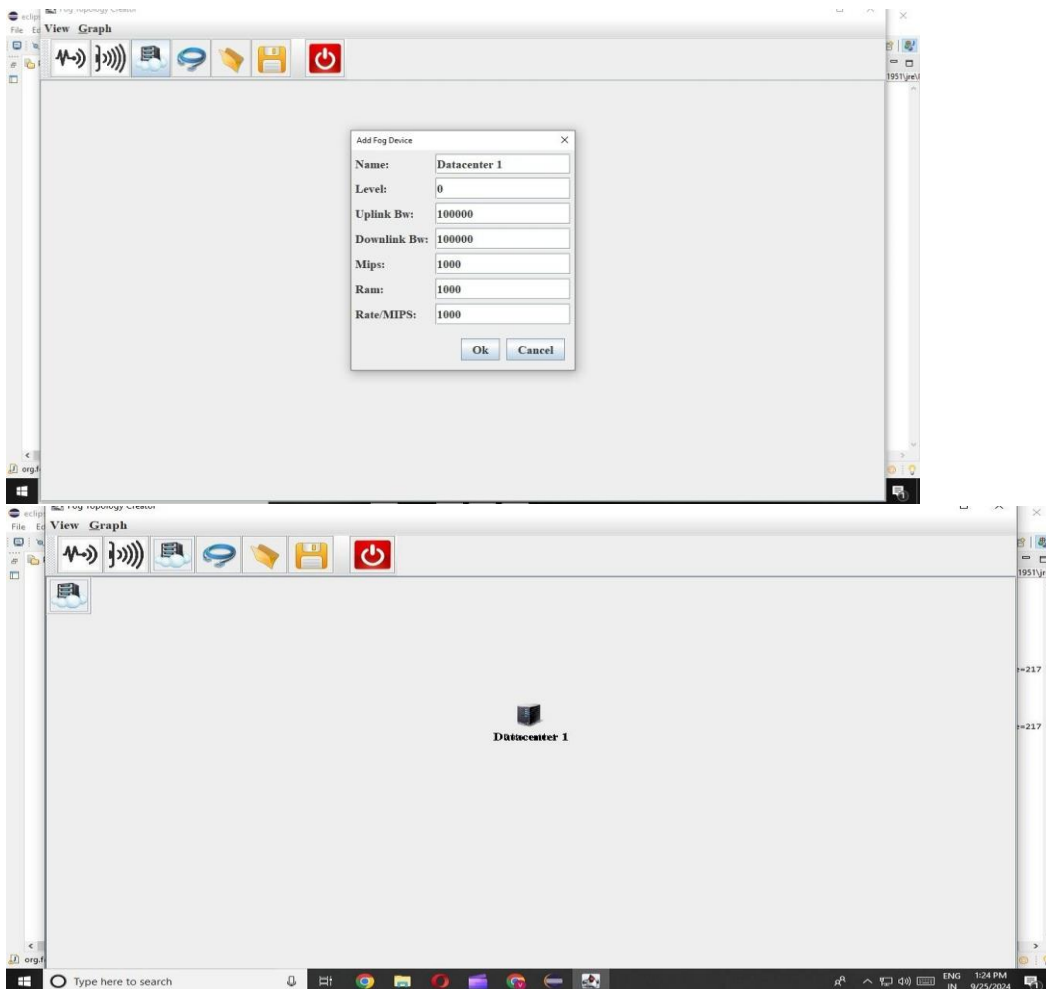
UPLINK Bw: 100000

Download Bw:10000

Mips:1000

Ram:1000

Rate/MIPS:1000



**Step3)Add fog device**

**NAME: gate way 0**



**AC Patil**  
College of Engineering

Jawahar Education Society's  
**A.C. Patil College of Engineering, Kharghar**  
**Department of CSE(IoT-CS BC)**

## Subject : Edge and Fog Computing Experiment No. 7

**LEVEL: 1**

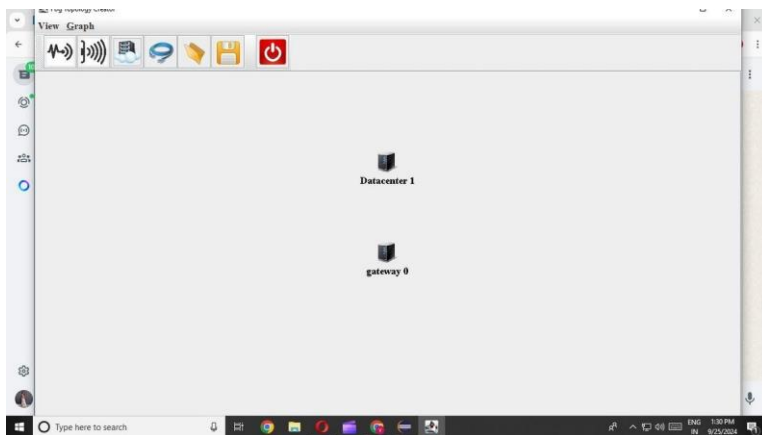
**UPLINK Bw: 1000**

**Download Bw:1000**

**Mips:100**

**Ram:1000**

**Rate/MIPS:10**

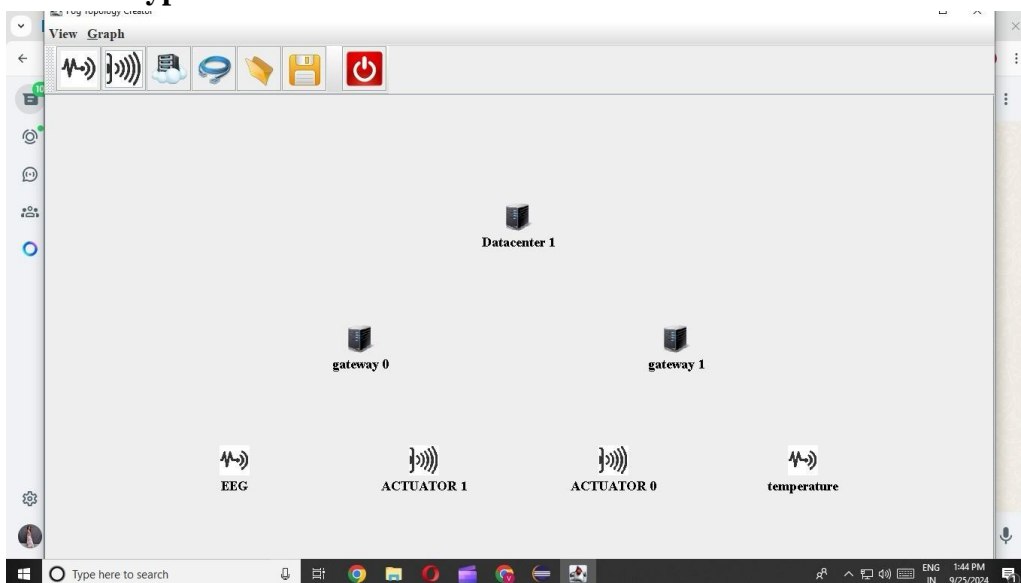


Repeat step 3

**Step 5) add actuator**

**Name: actuator0**

**Actuator type: actuator0**



**Step8) add link**



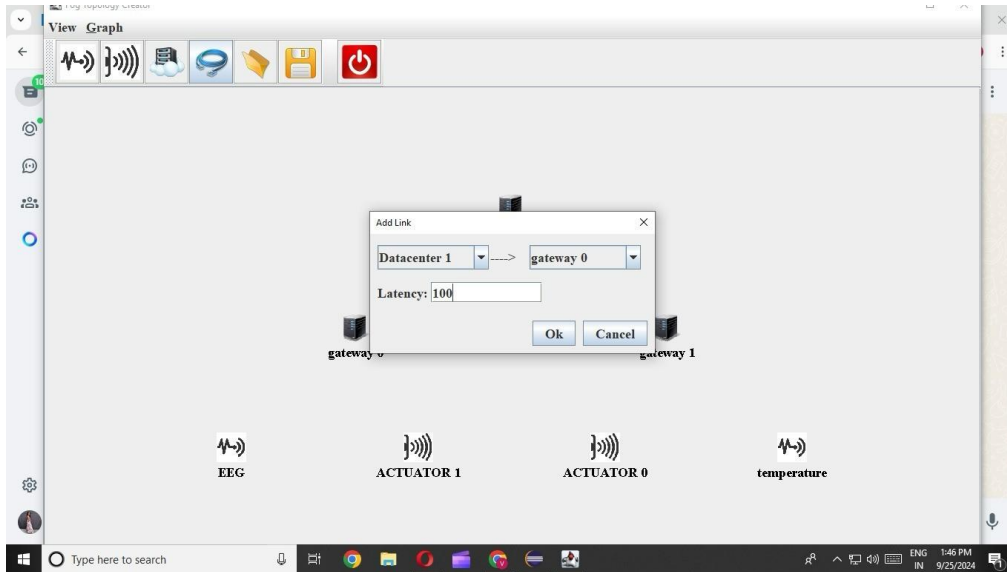
**AC Patil**  
College of Engineering

Jawahar Education Society's  
**A.C. Patil College of Engineering, Kharghar**  
**Department of CSE(IoT-CS BC)**

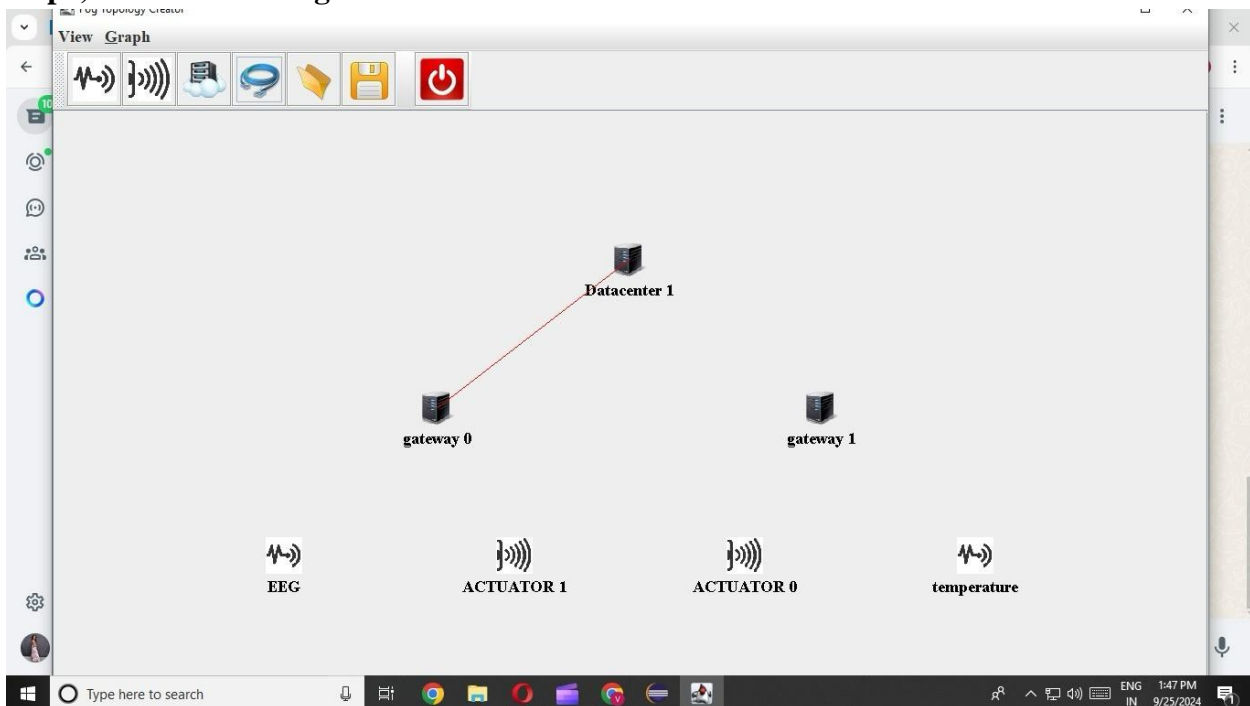
## Subject : Edge and Fog Computing Experiment No. 7

**Datacenter 1 -> gateway 0**

**Latency: 100**



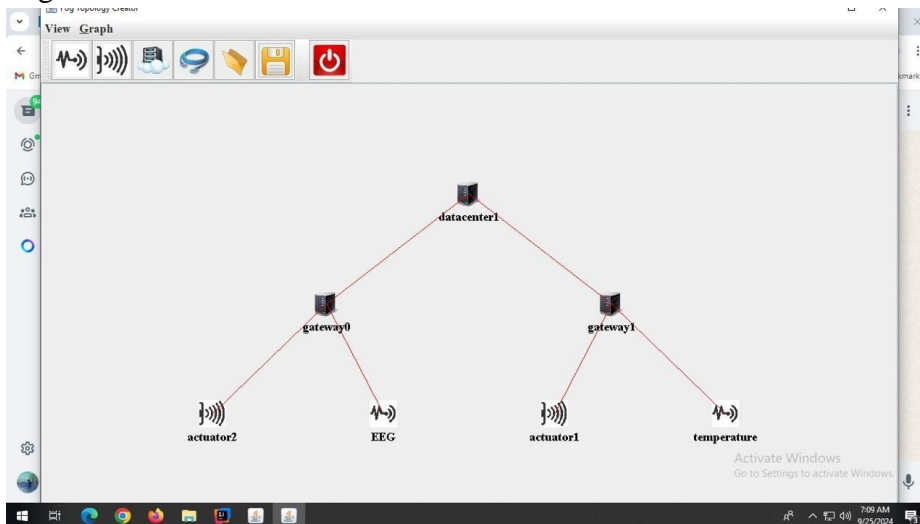
**Step9) Continue linking**





## Subject : Edge and Fog Computing Experiment No. 7

### Fog Architecture



### Result:

```
Run  FogGui x  FogGui x
=====
=====
=====
COORD MAP{Actuator []=Coordinates [abscissa=252, ordinate=486], FogDevice [mips=1000 ram=1000 upBw=1000 downBw=2000]=Coordinates [abscissa=840, ordinate=3.
Start Node : actuator2
Start Node : gateway1
Target Node : actuator1
Target Node : temperature
Start Node : datacenter1
Target Node : gateway0
Target Node : gateway1
Start Node : gateway0
Target Node : actuator2
Target Node : EEG
Start Node : EEG
Start Node : actuator1
Start Node : temperature
```

### Conclusion :

In this experiment, we successfully developed and evaluated edge-based data analytics algorithms using the iFogSimulator tool. By configuring and linking fog devices such as data centers, gateways, and actuators, we demonstrated the implementation of an edge computing architecture. This architecture allowed us to process data locally, minimizing latency and reducing bandwidth consumption, which is crucial in real-time decision-making and IoT environments. The use of edge analytics enabled us to take immediate corrective actions, showcasing the importance of edge computing in enhancing the performance and reliability of IoT systems.