Experiment No 3

AIM

To study Edge and Fog Computing simulators and understand their role in modeling, resource management, and performance evaluation of distributed IoT environments.

THEORY

1. Introduction to Computing Paradigms

- **Cloud Computing**: Centralized processing in remote data centers. While highly scalable, it suffers from **latency**, **bandwidth overhead**, and **mobility limitations** when dealing with IoT data.
- **Edge Computing**: Brings computation closer to the data source (IoT devices, gateways). This reduces **latency** and **network congestion**, making it suitable for real-time applications.
- **Fog Computing**: Extends Edge Computing by introducing an intermediate hierarchical layer between **Edge devices** and the **Cloud**. It enhances **resource distribution**, **scalability**, and **mobility support**.

---

2. Need for Simulators

- Deploying real-world Edge/Fog testbeds is **costly and complex**.
- **Simulators** such as iFogSim, CloudSim, and ENIGMA allow researchers to test and validate algorithms before actual deployment.
- Benefits:
    - **Cost-effective testing**
    - **Scalable experimentation**
    - **Performance evaluation under diverse scenarios**

---

3. Key Simulators

a) CloudSim

- Focuses mainly on **Cloud environments**.
- Provides simulation of resource allocation but **lacks Edge/Fog-specific features**.

b) iFogSim & iFogSim2

- Based on CloudSim, but tailored for **Fog and IoT systems**.
- Features:
  - Microservices and mobility support.
  - Suitable for small to medium-scale simulation.
  - Limitation: **Scalability issues** when simulating thousands of nodes.

c) ENIGMA

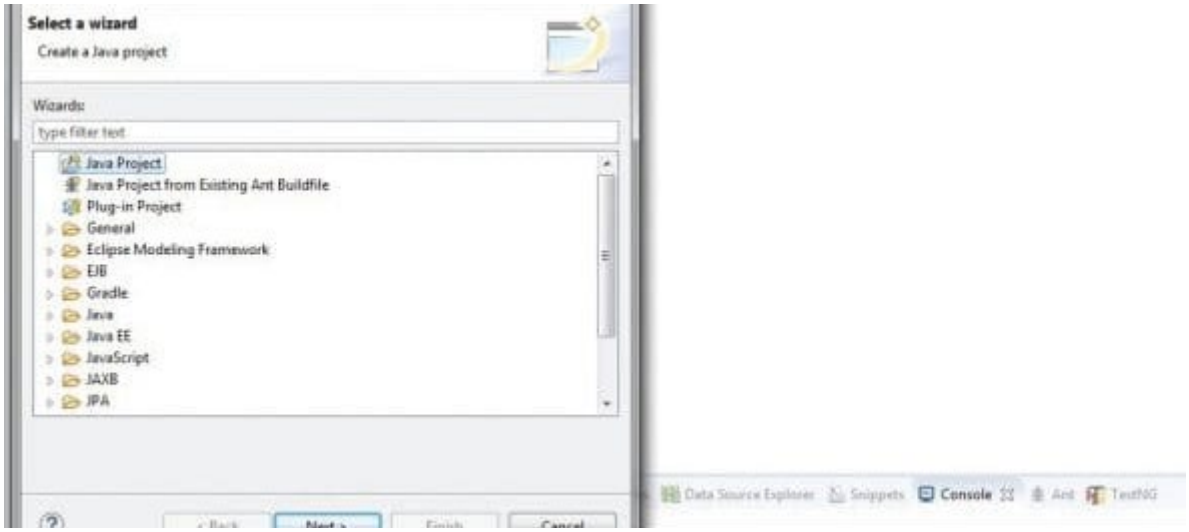
- Built on **SimGrid** framework.
- Designed for **large-scale, mobile, and dynamic Fog/Edge scenarios**.
- **Key Features**:
  - Mobility support and migration management
  - Real mobility dataset integration
  - Random mobility models
  - Microservice orchestration
  - Dynamic distributed clustering
  - Compatibility with CloudSim 5 and iFogSim tutorials

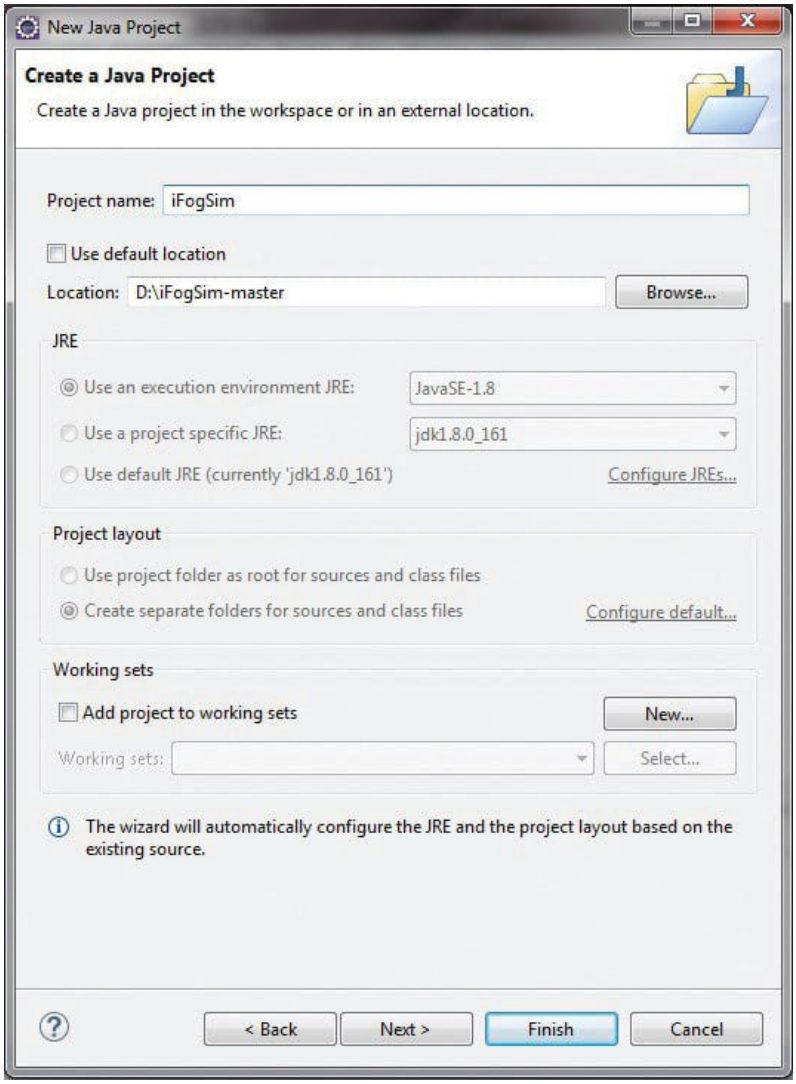
---

4. How to Run iFogSim2

Using Eclipse IDE

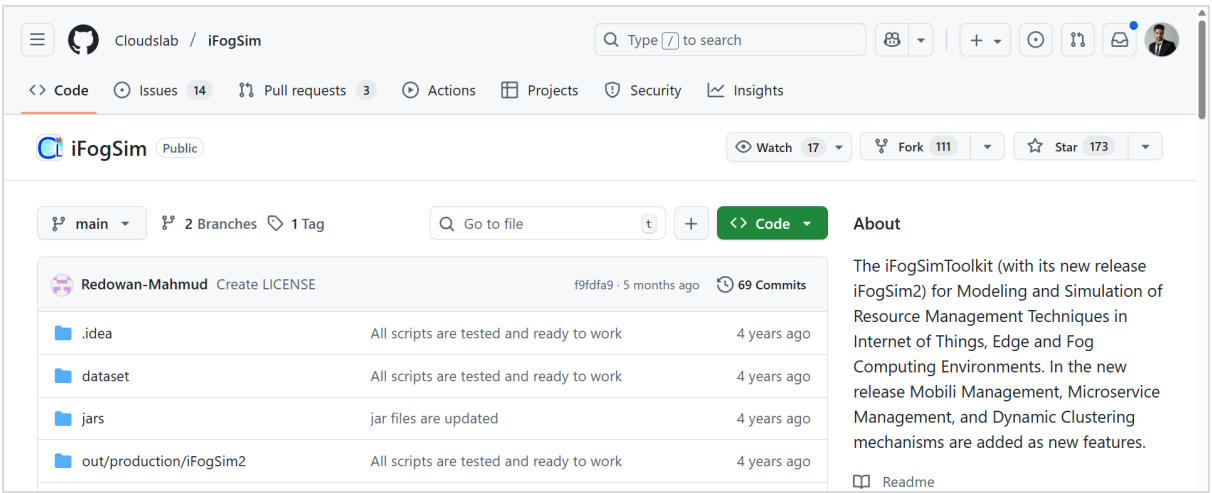
1. Create a **Java project** in Eclipse.





2. Initialize Git inside the project folder:

```
>> git init
>> git remote add origin https://github.com/Cloudslab/iFogSim
>> git pull origin main
```



3. Add required **JAR libraries** to the project.
4. Run example files such as:
    - TranslationServiceFog_Clustering.java
    - CrowdSensing_Microservices_RandomMobility_Clustering.java

Using IntelliJ IDEA

1. Clone the repository:

```
>> git clone https://github.com/Cloudslab/iFogSim
```

2. Select **"Project from Existing Sources"**.
3. Verify **Java version** and add external **JARs**.
4. Run example files (similar to Eclipse setup).

CONCLUSION

This experiment explored **Edge and Fog Computing simulators**, with emphasis on **ENIGMA**, a powerful tool for simulating large-scale IoT scenarios with mobility and dynamic clustering.

- **CloudSim**: Suited for centralized cloud-only systems.
- **iFogSim2**: Provides mobility and microservice support but struggles with large-scale scalability.
- **ENIGMA**: Excels in handling **dynamic, mobile, and large-scale environments**, making it more future-ready.

Future Scope: Integration of **AI/ML techniques** for predictive resource management and validation using **real-world datasets** can further improve simulation accuracy.