



## **Master Course**

### **Advanced Internet of Things (IoT): Systems, Networks, Edge Intelligence and Secure Connected Applications**

#### **Curriculum**

Duration:40 Hours

#### **Lesson 1: Fundamentals of Computer Networks**

- Definition of computer networks
- Types of networks
- Packet switching
- Segmentation
- Packet structure
- Header, payload, control information
- Routers, switches, modems, ISP, servers

#### **Learning Outcomes**

- Explain how data travels through a network
- Differentiate network types by scale and ownership
- Analyze packet-based communication

## **Hands-On**

- Packet journey simulation
- Smart temperature sensor network simulation

## **Lesson 2: Addressing and Network Layer Foundations**

- IP addressing
- Public vs private IP
- NAT
- IPv4 vs IPv6
- MAC addressing
- Device identity in local networks

## **Learning Outcomes**

- Interpret IP and MAC relationships
- Explain NAT operation
- Distinguish public and private addressing

## **Hands-On**

- What is My IP?
- Public vs private IP detection

## **Lesson 3: OSI Model and Protocol Stack Engineering**

- OSI 7-layer model
- Application layer
- Presentation layer
- Session layer
- Transport layer
- Network layer
- Data link layer
- Physical layer

### **Learning Outcomes**

- Map protocols to OSI layers
- Explain inter-layer dependencies
- Analyze protocol interactions

### **Lesson 4: Transport Protocols and Reliable Communication**

- TCP
- UDP
- Connection-oriented communication
- Connectionless communication
- Reliability vs latency trade-offs

### **Learning Outcomes**

- Compare TCP and UDP
- Select protocols for real-time systems
- Analyze protocol suitability for IoT

## **Hands-On**

- TCP vs UDP communication analysis

## **Lesson 5: Socket Programming and Client-Server Communication**

- Socket fundamentals
- IP + Port addressing
- Client-server architecture
- Python socket library
- Connection establishment
- send / recv patterns

## **Learning Outcomes**

- Build client-server systems
- Implement two-node communication
- Design local distributed applications

## **Hands-On**

- Pizza Shop client-server
- Dice Roller
- Bakery Chat

## **Lesson 6: Interactive Network Applications**

- Echo servers
- Request-response systems
- Interactive server logic

## **Hands-On**

- Python Echo Bot
- Number Guessing Server

## **Lesson 7: Web and Internet Protocols**

- HTTP fundamentals
- Client-server web exchange
- APIs
- Data over internet
- Request/response lifecycle

## **Learning Outcomes**

- Explain web communication flow
- Retrieve remote data using APIs
- Integrate internet services into IoT systems

## **Lesson 8: Email Protocols and Automated Messaging Systems**

- SMTP
- Secure email delivery

- TLS concepts
- App password authentication

## **Hands-On**

- Homework reminder bot
- IoT alert email system
- Password reset system
- OTP verification system

## **Lesson 9: REST APIs and Flask Servers**

- Flask server
- Endpoints
- JSON responses
- GET, POST, PUT, DELETE

## **Learning Outcomes**

- Design REST APIs
- Build web-accessible services

## **Hands-on**

- To-do REST server
- Calculator API

## **Milestone Project**

Smart School Safety System

## **Lesson 10: Real-Time Communication with WebSockets**

- Persistent bidirectional communication
- WebSockets vs HTTP
- Real-time dashboards
- Live sensor streaming

### **Learning Outcomes**

- Build live communication channels
- Stream data continuously

### **Milestone Project**

Smart Notification Infrastructure

## **Lesson 11: WebSockets and Real-Time Communication**

- Persistent communication
- Bidirectional channels
- Async communication

### **Learning Outcomes**

- Design real-time communication systems

### **Hands-on**

- Terminal group chat

## **Lesson 12: IoT Communication Flow**

- Sensing
- Communication
- Processing
- Action
- Visualization

### **Learning Outcomes**

- Explain full IoT data lifecycle

### **Hands-on**

- Sensor-to-dashboard simulation

## **Lesson 13: IoT System Architecture**

- End-to-end architecture
- Sensor → controller → broker → client
- Functional decomposition
- Architecture diagrams

### **Learning Outcomes**

- Read and design IoT architectures
- Decompose systems into functional layers

### **Hands-on**

- Draw architecture for smart home

- Draw architecture for smart health monitor

## **Lesson 14: MQTT Fundamentals**

- Publish/Subscribe
- Broker
- Topics
- Lightweight messaging

### **Learning Outcomes**

- Explain broker-based communication
- Design topic structures

### **Hands-on**

- First MQTT publish

## **Lesson 15: MQTT with Embedded Devices**

- Wi-Fi setup
- Broker connection
- Device publishing

### **Learning Outcomes**

- Connect embedded boards to cloud brokers

### **Hands-on**

- Board-to-cloud message system

## **Lesson 16: Bidirectional MQTT Systems**

- Subscribe callbacks
- Cloud-to-device messages
- Device responses

### **Learning Outcomes**

- Implement two-way device communication

### **Hands-on**

- MQTTX communication
- Cloud LED control

### **Milestone Project**

Smart Street Light System

## **Lesson 17: MQTT Architecture Design**

- Topic design
- Publisher/subscriber mapping
- Broker routing

### **Learning Outcomes**

- Architect multi-device MQTT systems

## **Lesson 18: Advanced Interactive MQTT Projects**

- Event-driven IoT interaction

## **Hands-on**

- Musical IoT board
- Dice cloud system

## **Lesson 19: IoT Games and Interaction Design**

- Human-device interaction
- Sensor play systems
- Feedback loops

## **Learning Outcomes**

- Design interactive IoT experiences

## **Hands-on**

- Reaction game
- Touch game
- Interactive challenge systems

## **Lesson 20: Game System Architecture for IoT**

- Input architecture
- State logic
- Timing systems

## **Learning Outcomes**

- Build architecture for physical interaction systems

## **Milestone Project**

Interactive IoT Game Platform

### **Lesson 21: REST API with Physical Devices**

- Board → Flask communication
- Device POST
- Browser GET

### **Learning Outcomes**

- Integrate embedded devices with REST systems

### **Hands-on**

- Mood Lamp Controller

### **Lesson 22: REST Architecture for IoT Systems**

- Device-server-client flow
- State storage

### **Learning Outcomes**

- Design REST-based IoT architecture

### **Lesson 23: Real-Time Architecture for Physical Systems**

- Event propagation
- Push pipelines

### **Lesson 24: Driver Functions and Modular Embedded Design**

- Modular hardware abstraction

- Reusable functions

## **Lesson 25: State Machines for Embedded Systems**

- States
- Events
- Transitions

## **Lesson 26: Non-Blocking Embedded Systems**

- Event loops
- Timing without delays
- Optimization

## **Milestone Project**

Smart Embedded Controller

## **Lesson 27: Parallel Processing and Concurrent Tasks**

- Threads
- Queues
- Concurrent tasks

## **Lesson 28: Edge AI and TinyML**

- TinyML
- Embedded inference

- Local classification

## **Lesson 29: Predictive Analytics for IoT**

- Time-series behavior
- Prediction logic
- Sensor trend forecasting

## **Lesson 30: IoT Security**

- Authentication
- Secure communication
- Device protection

## **Capstone Project**

- Smart City Platform
  - Secure Health Monitoring
-

## **Lesson 31: Parallel Processing and Embedded Concurrency**

- Processes
- Threads
- Queues
- Concurrent task management

### **Learning Outcomes**

- Handle multiple simultaneous sensor tasks
- Separate acquisition and communication threads

## **Lesson 32: Driver Functions and Modular Embedded Programming**

- Function abstraction
- Sensor drivers
- Modular code
- Hardware abstraction layers

### **Hands-On**

- Sensor driver library
  - Smart environment monitor
-

