## Coincent 3-Year Program in Embedded systems and the Internet of Things
## Partnered by Apsis Solution

### Year 1: Live Industrial Training – Build Your Foundation

Gain hands-on industry exposure from day one with 2.5 months of live training in a professional environment. Learn the latest tools and technologies through skill-focused sessions, guided by expert mentors from the industry

### Embedded systems and the Internet of Things Curriculum
### Module 1: Fundamentals of IoT

**Learning Objectives:**

- Understand what IoT is and why it matters
- Identify real-world applications
- Learn the structure of IoT architecture

**Topics Covered:**

- What is IoT? Definitions & Evolution
- Applications: Smart Homes, Healthcare, Agriculture, Industry 4.0
- **IoT Architecture:**
    - Perception layer (sensors, actuators)
    - Network layer (communication)
    - Application layer (interface, cloud)
- Embedded systems: basics, components
- Microcontroller vs Microprocessor

**Tools:** Slides, videos, real-world case studies

## Module 2: Arduino and TinkerCAD Simulation

**Learning Objectives:**

- Learn to simulate circuits and code using TinkerCAD
- Understand Arduino UNO board basics

**Topics Covered:**

- Arduino UNO Pinout, onboard components
- TinkerCAD setup and usage
- Connecting LED and resistor
- Blinking LED program (`digitalWrite`, `delay`)

**Hands-On:** Blinking LED simulation

**Assignment:** LED blink with varied delays and multiple LEDs

## Module 3: Digital I/O and Serial Communication

**Learning Objectives:**

- Interface with basic I/O components
- Use Serial Monitor to send and receive data

**Topics Covered:**

- Digital Inputs: Pushbuttons
- Digital Outputs: LEDs, Buzzer
- Serial communication with PC
- `Serial.begin()`, `Serial.print()`, `Serial.read()`

**Project Idea:** LED ON/OFF using serial commands

## Module 4: Analog Input and PWM

**Learning Objectives:**

- Work with analog sensors
- Learn PWM control for brightness and speed

**Topics Covered:**

- Analog inputs: Potentiometer, LDR
- PWM control: Fading LED
- `analogRead()`, `analogWrite()`
- LED forward voltage and safety

**Mini Project:** RGB LED color control using potentiometer

## Module 5: Sensors and Data Acquisition

**Learning Objectives:**

- Interface environmental and motion sensors
- Capture real-time sensor data

**Topics Covered:**

- PIR sensor (Motion detection)
- DHT11 (Temp & humidity)
- Ultrasonic sensor (Distance measurement)
- IR sensor, Gas sensor, Soil moisture sensor

**Hands-On Projects:**

- Automatic light using PIR
- Water level indicator using Ultrasonic

## Module 6: Actuators and Motor Control

**Learning Objectives:**

- Understand and control actuators
- Use motor driver circuits

**Topics Covered:**

- DC Motors – forward, reverse
- L293D motor driver
- Servo Motor – angle control
- Motor commands via serial or buttons

**Mini Project:** Line follower robot (Simulated)

## Module 7: Robotics Integration

**Learning Objectives:**

- Build basic robotic systems
- Combine sensors with actuators

**Topics Covered:**

- Robot design principles
- Remote-controlled robot using pushbuttons
- Obstacle avoidance logic using ultrasonic
- Speed control using PWM

**Projects:**

- Obstacle Avoidance Robot
- Smart Vacuum Cleaner Prototype

## Module 8: Introduction to IoT Platforms & Protocols

**Learning Objectives:**

- Understand IoT system components
- Learn about boards and protocols

**Topics Covered:**

- ESP8266 vs ESP32 vs Arduino UNO
- Wi-Fi, Bluetooth, LoRa
- HTTP vs MQTT protocol
- Cloud computing for IoT

**Case Study:** Smart irrigation system architecture

## Module 9: IoT Cloud – ThingSpeak

**Learning Objectives:**

- Send sensor data to the cloud
- Visualize real-time data online

**Topics Covered:**

- Creating ThingSpeak channel
- HTTP API keys
- Sending DHT11 data to cloud
- Reading from ThingSpeak using Arduino

**Project:** Remote temperature logging system

## Module 10: Home Automation Projects

**Learning Objectives:**

- Build a smart home interface
- Use sensors and relays to control appliances

**Topics Covered:**

- Relay module wiring and usage
- IR and Bluetooth control
- Cloud-triggered relay control

Adafruit IO dashboards

**Project:** IoT-based smart home controller

## Module 11: App Development with MIT App Inventor

**Learning Objectives:**

- Create mobile apps to control IoT devices
- Use Bluetooth and Wi-Fi communication

**Topics Covered:**

- Introduction to MIT App Inventor
- UI design for control panels
- App-to-Arduino communication (via Bluetooth/Wi-Fi)
- Text-to-Speech, Voice control

**Projects:**

- IoT Home Light App
- Speech Controlled Fan

## Module 12: MQTT Messaging Protocol

**Learning Objectives:**

- Learn lightweight IoT communication
- Use Mosquitto and Adafruit MQTT

**Topics Covered:**

- MQTT Broker, Publisher, Subscriber
- Installing Mosquitto locally
- Using Adafruit MQTT with Arduino
- Authentication with MQTT

**Project:** Sensor updates via MQTT on dashboard

## Module 13: Python for IoT

**Learning Objectives:**

- Learn Python basics and IoT scripting
- Control devices and log data with Python

**Topics Covered:**

- Python basics: variables, loops, conditions
- PySerial for Arduino communication
- Using MQTT and HTTP in Python
- DHT/Ultrasonic logging with Python

**Project:** Python dashboard for sensor data

## Module 14: Networking & Protocols

**Learning Objectives:**

- Learn how devices communicate on a network

**Topics Covered:**

- TCP/IP model and protocol stack
- UART, I2C, SPI communication
- Basics of sockets in Python
- Building a chatroom with sockets

## Year 2: Real-Time Projects – Apply What You've Learned

Transform your knowledge into real-world experience by working on 8 industry-level projects that build your technical and professional skills. Each project enhances your portfolio, strengthening your resume and showcasing your practical abilities. You'll also collaborate in teams, gaining valuable experience in communication, teamwork, and project management—just like in a real work environment.

**PROJECTS**

**Automatic Room Temperature Controller**
This project uses sensors like DHT11 or DHT22 to monitor room temperature and humidity in real-time. Based on sensor data, it controls fans or air conditioners using a microcontroller like Arduino or ESP32. The system maintains a desired temperature range automatically. It enhances energy efficiency and comfort in smart homes. Data can be displayed on an LCD or IoT dashboard. Tools used: **Arduino**, **Relay Module**, **DHT Sensor**, **ESP32**.

**Voice Control Home Automation**
This system enables controlling home appliances like lights and fans using voice commands via Google Assistant or Alexa. It integrates IoT platforms like Blynk or IFTTT with ESP32/NodeMCU and Wi-Fi. Voice input is processed through a smartphone or smart speaker. The system offers convenience, especially for the elderly or disabled. It supports remote and hands-free operation. Tools: **ESP32**, **Google Assistant**, **IFTTT**, **Relay Module**.

**Obstacle Avoidance Robot using Ultrasonic Sensor**
The robot detects obstacles using ultrasonic sensors and navigates by changing direction to avoid collisions. It uses an Arduino or Raspberry Pi to control motors based on sensor input. The system enhances autonomous mobility for robotics applications. It's ideal for environments where manual control is difficult. Widely used in research and automation. Tools: **Ultrasonic Sensor**, **Arduino UNO**, **Motor Driver**, **Wheels/Chassis**.

**IoT based Smart Factory System**
This project monitors factory parameters like temperature, vibration, or gas levels using sensors connected to ESP32. The data is transmitted to cloud platforms like ThingSpeak or Blynk for visualization and alerts. It improves production efficiency and safety through real-time monitoring. Automation can be triggered based on thresholds. Tools: **ESP32**, **Industrial Sensors**, **Blynk**, **ThingSpeak**, **MQTT**.

**IoT based Weather Data System using ESP32**
This system collects environmental data like temperature, humidity, and air pressure using sensors like DHT11 or BMP180. The ESP32 sends this data to cloud dashboards for live monitoring and logging. Users can access the weather data remotely via web or mobile apps. It's useful for agriculture, smart cities, and education. Tools: **ESP32**, **DHT11/BMP180**, **ThingSpeak**, **Blynk**.

**Smart Building using BLYNK**
This IoT solution automates and monitors building systems like lighting, security, and temperature using Blynk and ESP32. Data from sensors is sent to the Blynk app for real-time visualization and control. The system supports scheduling, alerts, and remote operation. It enhances energy efficiency, safety, and user convenience. Tools: **ESP32**, **Blynk App**, **Sensors (PIR, DHT)**, **Relay Module**.

**Year 3 – Placement & Internship Phase:**

In the 3rd year of Coincent's program, students are guaranteed an internship with partner companies, complete with a formal Internship Offer Letter and a Completion Certificate upon successful completion. This internship is a complimentary part of the 3-Year "Industrial Training + Internship" model, which also includes live classes, expert mentorship, and hands-on project work. This phase bridges academic learning with real-world application, providing students with valuable professional exposure before graduation.

Coincent also offers structured placement preparation to ensure students are job-ready. This includes portfolio building through 8 real-time projects, certifications aligned with Microsoft standards, and dedicated training for interviews. From mock interviews to resume reviews and HR/technical round prep, every element is designed to transition students from classroom learning to career success. By the 4th year, students are equipped not just with knowledge, but with experience, credentials, and confidence to enter the workforce.