Hands-On IoT Architecture and Smart Systems using ESP32

Day	Session	Title	Contents	Tasks
DAY 1	FN 9.00 am to 10.30 am	**Introduction, Evolution and Future of IoT**	Overview of IoT: Definition, architecture and applications. Exploring the transformation of IoT from basic Machine-to-Machine (M2M) communication to today's interconnected smart systems. Latest trends in IoT, including the role of edge computing, AI integration, and how emerging technologies are shaping the future of IoT across industries.	PPT Presentation and Group discussion on how emerging technologies will influence the future of IoT.
	BREAK		Introduction to microcontrollers and their role in IoT.	
	10.45 am to 12.15 pm	**Introduction to IoT Board (Microcontroller, sensors and actuator)**	Digital I/O, Serial Communication, Sensors and actuators. Detailed explanation of working of L298N driver.	**Hands-on**: Dancing LEDs and buzzer with Toggle switch using ESP32 and controlling motors (dc, stepper, servo) using L298N motor driver.

	AN 1.00 pm to 2.30 pm	**Analog I/O and PWM** **LCD Display and Data Presentation**	Use analog pins to read data from devices such as potentiometers, allowing variable input values. Simulate analog output using PWM (Pulse Width Modulation) for controlling LED brightness or motor speed. Connect and control a 16x2 character LCD using the LiquidCrystal library. Display text and sensor data.	**Hands-on:** Adjusting the brightness of an LED or the speed of a motor using a potentiometer. Read temperature data from a DHT11 sensor, displaying it on the LCD.
	BREAK			
	2.45 pm to 4.15pm	**Timers, Delays and interrupts.**	Implementing delays in your program using delay() and how to keep track of time without halting the program using millis(). Using interrupts to handle events like button presses in real-time, independent of the main code execution.	**Hands-on:** Building a system where pressing a button (using interrupts) controls an LED, ensuring it operates independently of other tasks in

Hands-On IoT Architecture and Smart Systems using ESP32

DAY 2	FN 9.00 am to 10.30 am	**Real-Time Operating System (RTOS) on ESP32**	Introduction to FreeRTOS, the built-in Real-Time Operating System for ESP32, enabling real-time task management. Create and manage multiple tasks, control task scheduling, and set task priorities for better resource management.	**Hands-on:** Implement a multi-tasking example where an LED blinks while reading sensor data simultaneously, using FreeRTOS for task scheduling.
	BREAK 10.45 am to 12.15 pm	**Introduction to I2C and SPI Protocols**	I2C (Inter-Integrated Circuit) and SPI (Serial Peripheral Interface) protocols for communication with sensors and displays. Connect and control I2C devices like OLED displays or Real-Time Clocks (RTC), and display data on them.	**Hands-on:** Display sensor data on an I2C OLED display or interface with an SPI device like an SD card module.

	AN 1.00 pm to 2.30 pm **BREAK** 2.45 pm to 4.15pm	**Advanced Networking with ESP32 (Wi-Fi Modes: Station, SoftAP, and Dual Mode)** **Bluetooth and BLE**	Different modes of Wi-Fi communication: Station mode (client), Soft Access Point (AP), and Dual Mode (acting as both AP and client). Bluetooth Classic and Bluetooth Low Energy (BLE), focusing on the low-power advantages of BLE for IoT applications and architecture of BLE.	**Hands-on:** Build a dual-mode Wi-Fi system where the ESP32 acts as both a client and a server, allowing communication between devices. **Hands-on:** Implement a BLE server on the ESP32 that broadcasts sensor data to connected devices.

Hands-On IoT Architecture and Smart Systems using ESP32

DAY 3	FN 9.00 am to 10.30 am **BREAK** 10.45 am to 12.15 pm	**Web Server Development: WebSocket Server** **Serving HTML/CSS/JS from ESP32**	Set up a WebSocket server on the ESP32 to allow real-time, bidirectional communication between clients and the server Serve web pages with HTML, CSS, and JavaScript directly from the ESP32, allowing users to interact with the device through a web interface.	**Hands-on:** Build an interactive web dashboard to monitor sensor data and control the ESP32's outputs in real time.
	AN 1.00 pm to 2.30 pm **BREAK** 2.45 pm to 4.15pm	**IoT integration and cloud communication: MQTT Protocol in Depth** **Connecting ESP32 to Cloud Services (AWS IoT/ Blynk)**	Explore the MQTT (Message Queuing Telemetry Transport) protocol, widely used for IoT communication due to its lightweight, publish/ subscribe architecture. Connect the ESP32 to popular cloud platforms such as AWS IoT/ Blynk to send and receive data.	**Hands-on:** Set up an MQTT connection to a cloud service, sending sensor data from the ESP32 and receiving control commands for home automation.