Flood Monitoring and Early warning System

Set-up Connection in Wokwi (Assuming you're using an ESP8266):

Sign in to Wokwi (if not already).

Create a new project.

Add an ESP8266 module and any required sensor components to your project.

Connect the sensor to the appropriate pins on the ESP8266.

Configure Wi-Fi settings on the ESP8266 for internet connectivity.

Write and upload the Python script to the ESP8266 using Wokwi's online IDE.

Simulate the project to test its functionality.

Sample Python Code (for sending sensor data to the Early Warning Platform):

Here's a simplified example of Python code that reads data from a water level sensor and sends it to a hypothetical Early Warning Platform. Note that the actual code will depend on the sensor type and communication protocol used.

PYTHON CODE

```
import machine
import urequests
# Configure your Wi-Fi network
wifi_ssid = "YourSSID"
wifi_password = "YourPassword"

# Configure the Early Warning Platform URL
platform_url = "http://your-early-warning-platform.com/api"

# Initialize the sensor (replace with actual sensor initialization)
sensor = machine.Sensor()

# Connect to Wi-Fi
wifi = machine.WiFi(wifi_ssid, wifi_password)

while True:
    # Read water level from the sensor (replace with actual sensor reading)
    water_level = sensor.read()

    # Send data to the Early Warning Platform
    data = {"water_level": water_level}
    response = urequests.post(platform_url, json=data)

    # Check for a successful response
    if response.status_code == 200:
        print("Data sent successfully")
#Delay for a period before taking the next reading
    machine.sleep(60000) # Sleep for 1 minute (adjust as needed)
```

Simulation:
Sample Output:

The sample output would depend on the IoT device's debugging or serial output. You may see messages like "Data sent successfully" when the data is successfully transmitted to the Early Warning Platform.

The components required for building an IoT flood monitoring and early warning system are:

IoT Sensors (e.g., water level sensors)

IoT Device (e.g., Raspberry Pi, Arduino, ESP8266/ESP32)

Internet Connectivity

Early Warning Platform

Python Script

Power Supply