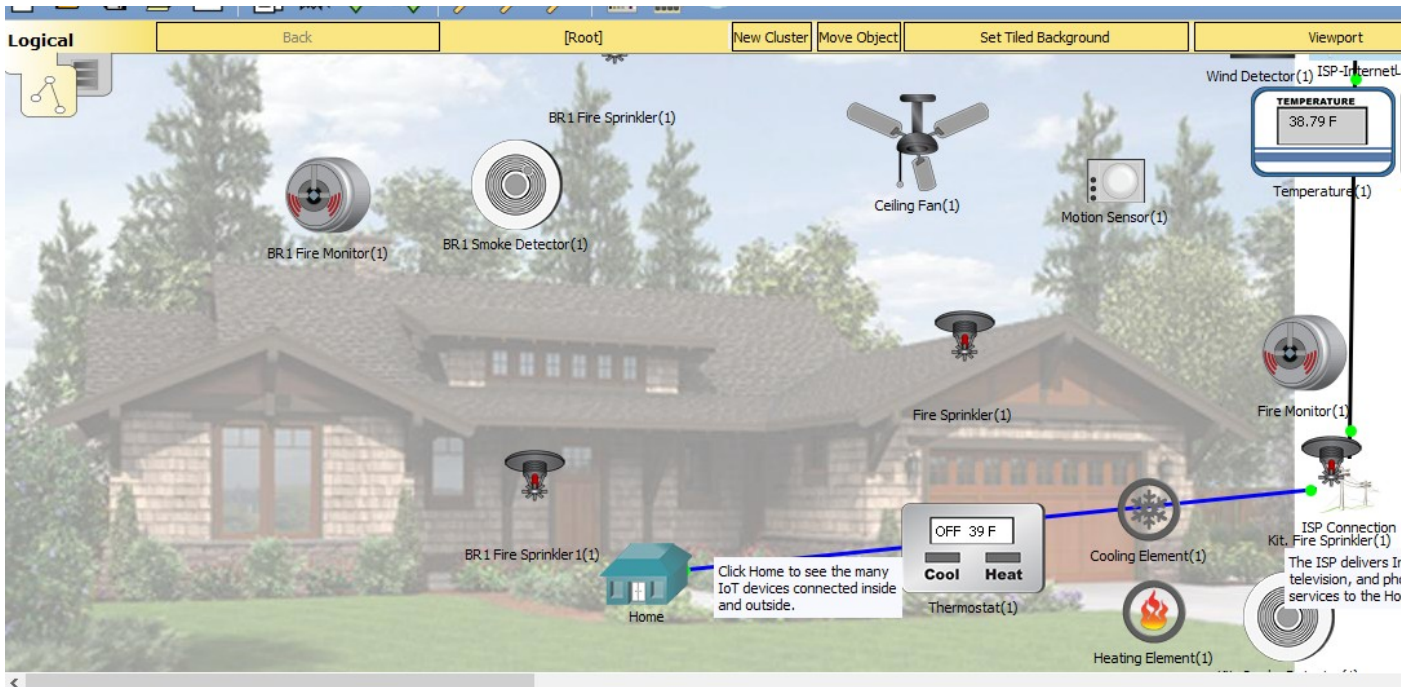


IoT-PT

IOT con Packet Tracer Cisco



Indice Generale

1 Dispositivi IoT in Packet Tracer.....	3
Registration Server.....	3
Home gateway.....	5
Microcontrollori.....	6
SBC - Single Board Controller (Raspberry PI).....	6
MCU - Microcontroller Unit (Arduino).....	7
2 Sitografia.....	8



IOT-PT

Questo testo è pubblicato sotto licenza **Creative Commons - Attribuzione - Non commerciale - Condividi allo stesso modo 3.0 Unported** .

Le informazioni contenute in questa opera possono essere diffuse e riutilizzate in base alle condizioni poste dalla licenza GNU General Public License, come pubblicato dalla Free Software Foundation. - Per le condizioni consulta: <http://creativecommons.org/licenses/by-nc-sa/3.0/deed.it>



Le utilizzazioni consentite dalla legge sul diritto d'autore e gli altri diritti non sono in alcun modo limitati da quanto sopra.

Il documento è scaricabile da , sezione download, per fini esclusivamente didattici e non commerciali. Il documento è scaricabile liberamente da:

<http://moodle.isisfacchinetti.it/mod/data/view.php?id=21&rid=517>

E' possibile richiedere la dispensa in formato cartaceo, all'Ufficio Stampa dell'ISIS Facchinetti

Segnalazioni di errori, critiche e consigli sono molto graditi e possono essere inoltrati a paolo.macchi@libero.it, oppure lasciando un commento al momento del download se siete registrati.

(in copertina -)

-Per la generazione di qr code: <https://www.qrstuff.com/>

Convenzioni usate nel testo:



rappresenta una curiosità, un approfondimento

NOTA rappresenta una nota



rappresenta una esercitazione o esempio

link di riferimento Rappresenta un codice o dei risultati o una segnalazione storica

1 Dispositivi IoT in Packet Tracer

La gestione dinamica di sensori ambientali (temperatura, gas, pressione, luce, ...) in Packet Tracer ha permesso di rendere la simulazione dei dispositivi IoT più realistica. Ad esempio se una vecchia auto emette gas di scarico inquinanti, un rilevatore di fumo può essere utilizzato per innescare un allarme; una finestra si aprirà quando la temperatura dell'ambiente diventa eccessiva.

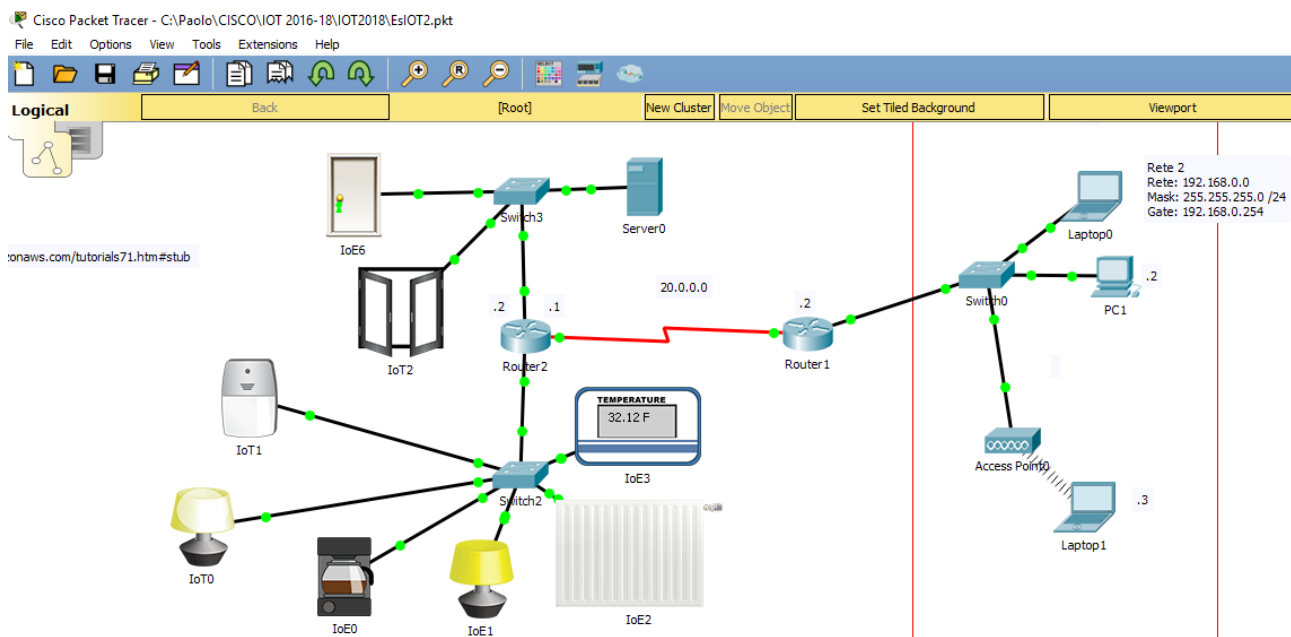
Gli "smart thing" sono oggetti fisici che possono connettersi a **Registration Server** (configurato con il servizio *IoE*) o **Home Gateway** tramite un'interfaccia di rete. Sono suddivisi in 4 sottocategorie:

- Home
- Smart City
- Industrial
- Power Grid.

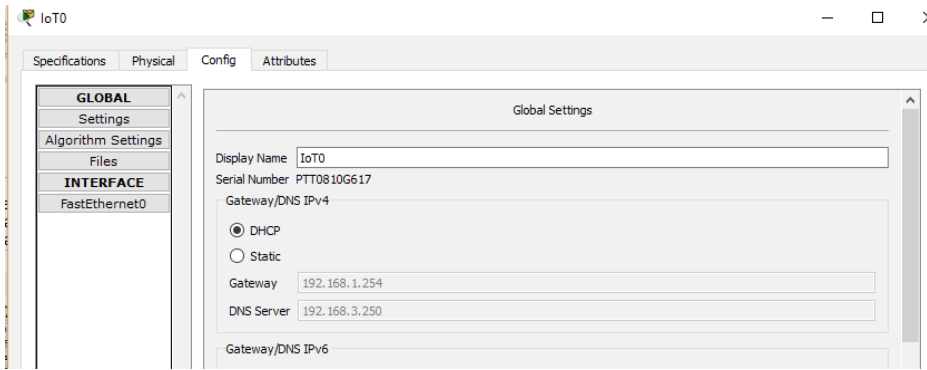
I "components" sono oggetti fisici che si connettono al **microcontrollore (MCU-PT)** o ai **computer a scheda singola (SBC-PT)**. In genere non dispongono di un'interfaccia di rete. Si tratta di dispositivi semplici che comunicano solo attraverso gli ingressi e uscite analogiche o digitali. Si distinguono in:

- **Attuatori:** questi componenti intervengono sull'ambiente esterno
- **Sensori:** questi componenti percepiscono le caratteristiche fisiche dell'ambiente tramite fotocellule, sensore di temperatura, potenziometri, pulsanti...

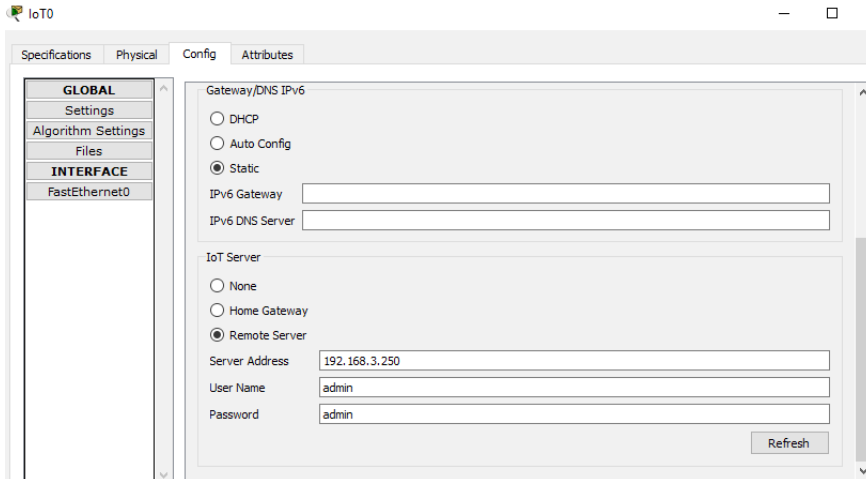
Regstration Server



1. inserire l'elemento IoT (ad esempio la lampada o altro oggetto)
2. impostare l'indirizzo IP



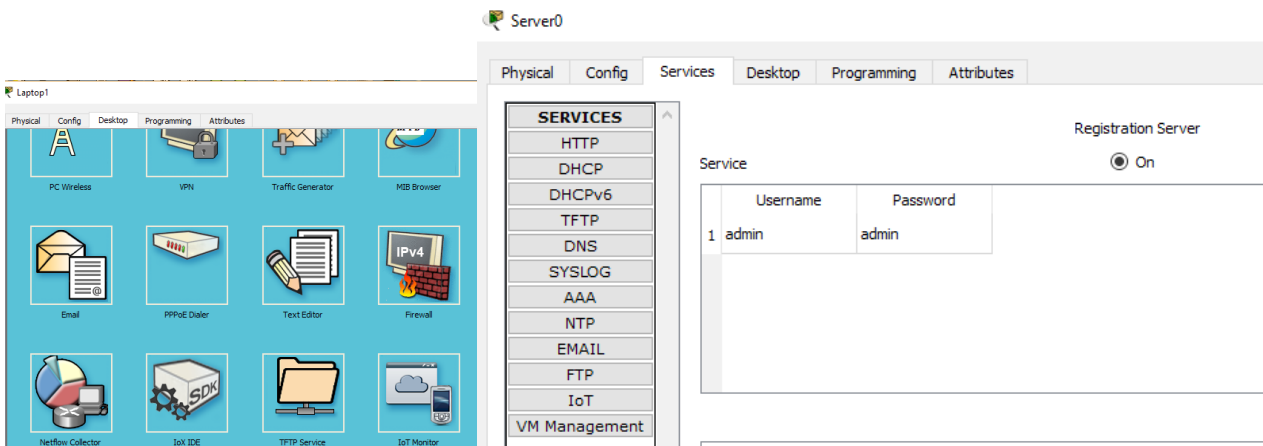
3. impostare l'indirizzo (o il nome) del Server remoto con login & password



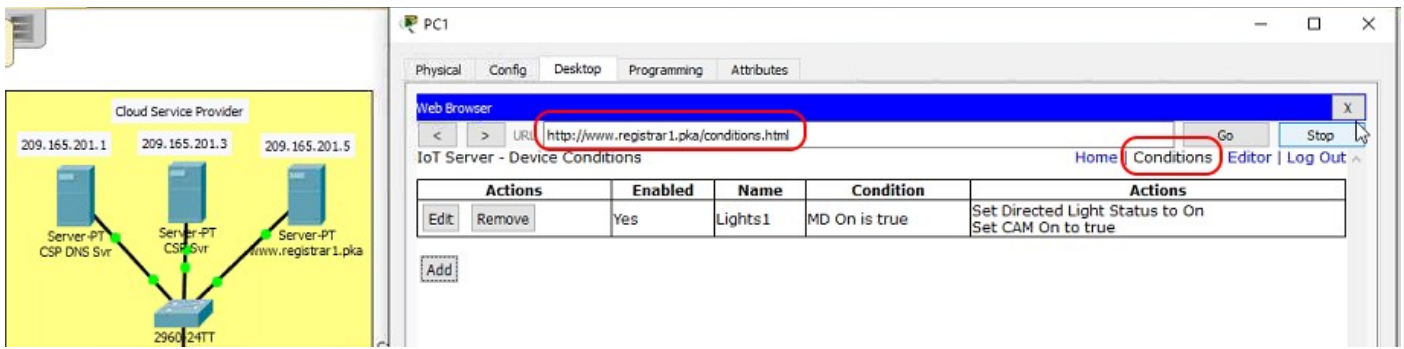
4. Da Desktop del PC remoto (es. laptop1) scegliere IoTmonitor



- Su Server è possibile cambiare login e password (di default admin,admin)



E' possibile "programmare" i dispositivi servendosi di eventi-azioni (Conditions/Actions) andando sul server remoto e impostando le Conditions:



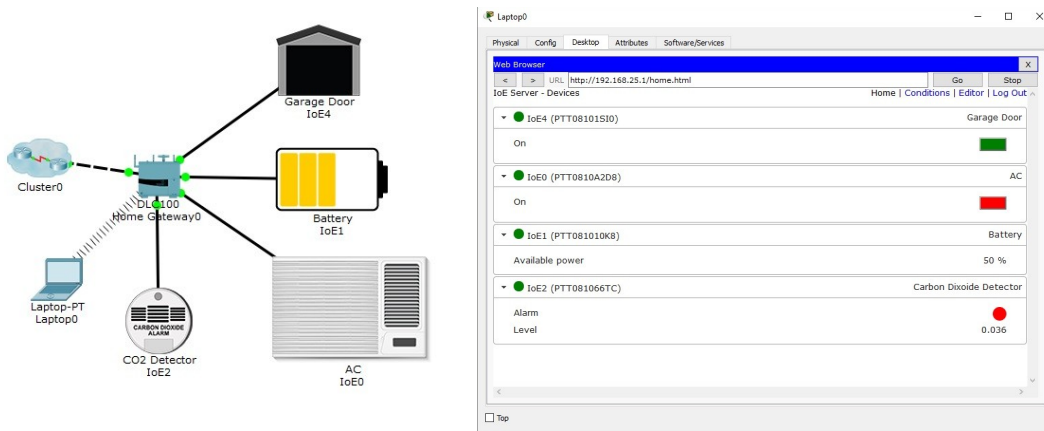
Home gateway

Home Gateway fornisce 4 porte Ethernet e un punto di accesso wireless configurato con lo ssid "HomeGateway" sul canale 6. È possibile configurare WEP / WPA-PSK / WPA2 enterprise per proteggere le connessioni wireless.

Gli oggetti IoE possono essere gestiti da remoto tramite un'interfaccia web ospitata da Home Gateway, ma è accessibile anche tramite l'indirizzo IP di Internet .

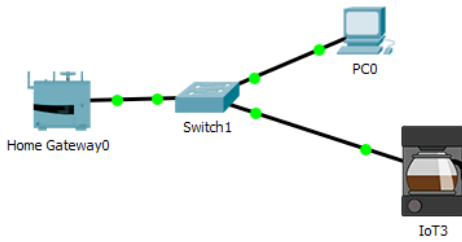
- Login Home Gateway (web): admin
- Password del gateway di casa: admin

L'immagine seguente mostra 4 oggetti IOE collegati ad un Home Gateway che è connesso a Internet tramite la porta Ethernet WAN.

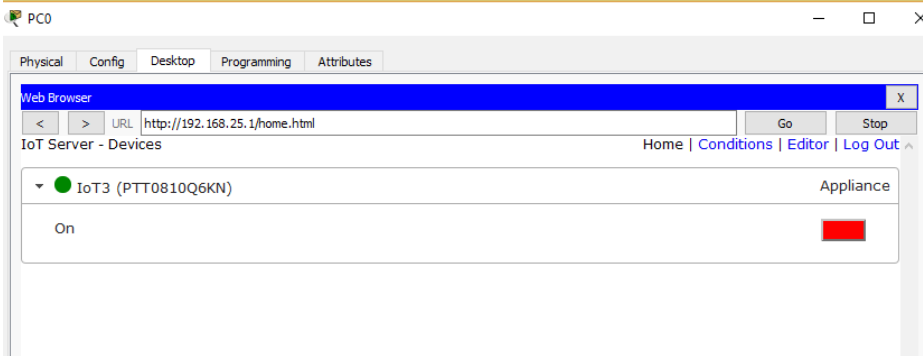


Esempio:

- 0.On Remote server choice IoT service and enable Registration Server (ON) and login and pass
- 0.1 Use device conditions on IoT Monitor

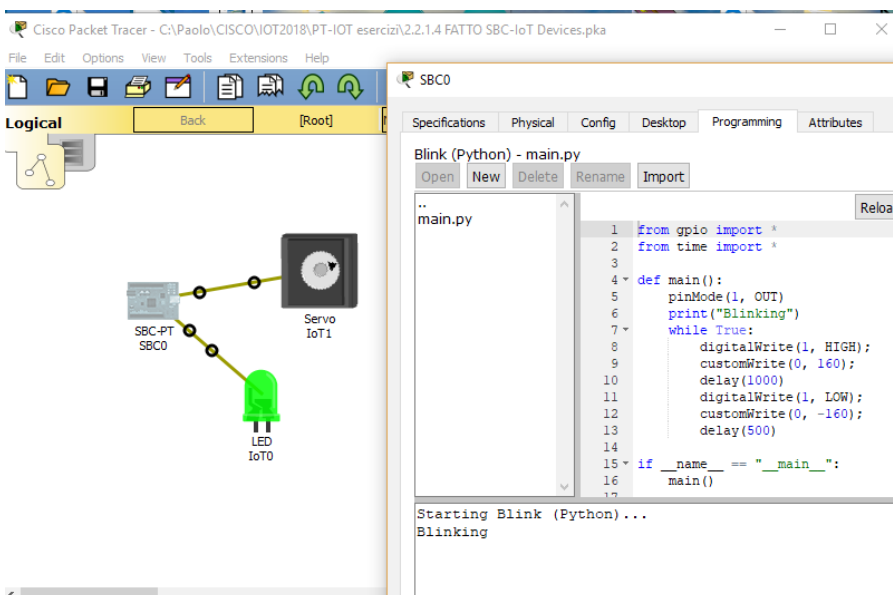


- Home Gateway
- PC0 DHCP
- Coffee DHCP
- on PC0 set IP address Home Gateway on Browser and admin, admin



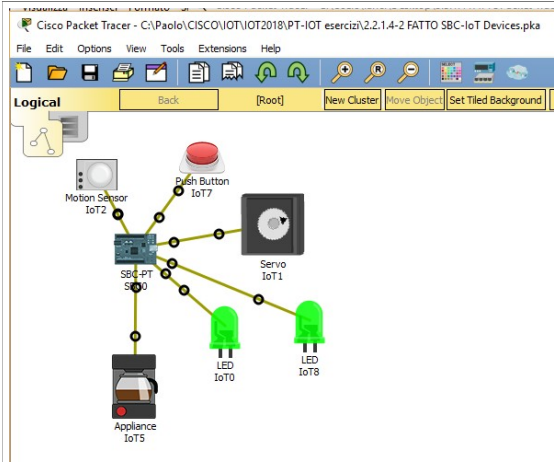
Microcontrollori

SBC - Single Board Controller (Raspberry PI)



Se aggiungo il bollitore (Appliance) su output D2, un pulsante in input (D3) che accende un altro LED su D4. E' possibile anche, tramite un sensore di movimento,

fare accendere la luce e far partire la macchina del caffè.
 Ad esempio quando al mattino mi alzo dal letto si accende la luce e parte il caffè!



Appliance

Usage: Connect Appliance to SBC or button and send a customWrite to turn on/off
Direct Control: ALT-click to interact

Local Control: Connect device to MCU/SBC/Thing. Use the "customWrite" API per Data Specifications

Remote Control: Connect device to Registration Server using Config Tab

Data Specifications: INPUT SLOT 0: 0 = Off 1 = On

Example:

Connect a button to Appliance, then send data: `customWrite(0, "1");` // turns on device

```

from gpio import *
from time import *
togglePushButtonValue = 0 # initialize Toggle Push Button sensor value global variable to 0

def main():
    pinMode(0, OUT)
    pinMode(1, OUT)
    pinMode(1, OUT)
    pinMode(5, IN)
    print("GO!")
    while True:
        togglePushButtonValue = digitalRead(3) # read Toggle Push Button
        sensor value
            digitalWrite(1, HIGH);
            customWrite(0, 160);
            customWrite(2, "1");
            if (togglePushButtonValue == HIGH): # evaluates to True if the Toggle
            Push Button sensor value is digital HIGH, otherwise false
                digitalWrite(4, HIGH) # turn on the LED4
                delay(1000)
            if(digitalRead(5)== HIGH):
                print("movimento")
                customWrite(2, "1");
                digitalWrite(4, HIGH)
                delay(1000)

            digitalWrite(1, LOW);
            digitalWrite(4, LOW) # turn off the LED4
            customWrite(0, -160);
            customWrite(2, "0");
            delay(500)

if __name__ == "__main__":
    main()
  
```

MCU - Microcontroller Unit (Arduino)

Network Description:

An MCU (microcontroller unit) is a board that has a USB port, six digital I/O ports, and four analog I/O ports. The digital I/O ports on the MCU allows a user to connect digital sensors and actuators. Meanwhile, the analog I/O ports allows a user to connect analog sensors and actuators.

In this demo, there is a single MCU connected to four sensors and four actuators. The four sensors include a digital switch, a digital toggle push button, an analog potentiometer, and an analog

MCU (Python) - main.py

```

main.py
1 from gpio import * # imports
2 from time import * # imports
3
4 switchValue = 0 # initialize
5 togglePushButtonValue = 0 # i
6 potentiometerValue = 0 # init
7 flexSensorValue = 0 # initial
8
9 def readFromSensors():
10     global switchValue # decl
11     global togglePushButtonVa
12     global potentiometerValue
13     global flexSensorValue #
14
15     switchValue = digitalRead
16     togglePushButtonValue = d
17     potentiometerValue = anal
18     flexSensorValue = analogR
19
20 def writeToActuators():
21     if (switchValue == HIGH):
22         customWrite(2, "2") #
23 <
  
```

Starting MCU (Python)...

2 Sitografia

CISCO - <http://static-pt-assets.s3.amazonaws.com/tutorials71.htm#stub>

14 Internet of Things

14-1 Actuators, Sensors and Smart Devices

- Explore the new IoT devices.

14-2 Smart Home

- How to create a Smart Home with the new IoT features.

14-3 Network Devices

- Explore devices and protocols.

14-4 Home Gateway and Registration Server

- Connect your IoT device to a server for remote control.

14-5 Physical Workspace

- Learn how to use containers, endpoints, and other physical workspace capabilities.

14-6 Web Server

- See what you can do with the HTTP server now.

14-7 Thing Editor and Device Manager

- Learn how to create your own custom IoT device.

14-8 Email Client and MCU

- Learn how to use the email client with the MCU.

14-9 Environment

- See how the environment affects the sensors.

14-10 Environment Key Frames

- *Control the environment variables to create a dynamic world.*

<http://www.packettracernetwork.com/internet-of-things/pt7-iot-devices-configuration.html> Packet Tracer 7.0 - IoT devices configuration

https://www.youtube.com/watch?v=pCoQE_gg-xk video con Blockly

<http://www.packettracernetwork.com/internet-of-things/pt7-iot-devices-configuration.html>