3.2.2.5 - Install and Configure IoT Devices

Part 1. Setting up the Home Network

Step 1: Set up the wired network

Connect FastEthernet on Home Switch to Home Gateway using a coper straight--through cable.

Connect FastEthernet0 on PC -> FastEthernet on HomeSwitch

Set IPConfig of PC to DHCP

Step 2: Set up the wireless network

Home Gateway -> Config -> Internet -> select DHCP

Home Gateway -> Wireless -> change SSID to MyHome Gateway, auth to WPA2-PSK, use CiscoIoT as passphrase

Laptop -> Desktop -> PC Wireless -> Connect -> select MyHomeGateway -> connect

Step 3: Connect the IoT devices to the network

Connect Coffee Pot to FastEthernet using copper straight-through cable

Coffee Pot -> Config -> FastEthernet0 -> select DHCP

Lamp/Ceiling Fan -> Config -> Wireless0 -> connect to MyHomeGateway

Part 2. Interacting with IoT devices

Step 1: Acces the IoT devices locally

Click Alt + Lamp a few times, how many settings does the lamp have: 3

Alt + Lamp Switch: 2

Alt + Ceiling Fan: 3, Switch: 2

Alt + Coffee Pot: 2, Swtich: 2

Step 2: Configure the IoT devices for remote access

Lamp/Ceiling Fan/Coffee Pot -> Config -> IoE Server heading -> Remote server:

Server address: www.register.pka

Username: admin

Password: admin

-> connect

Step 3: Access the IoT devices remotely

3.2.2.5 - Install and Configure IoT Devices (cont)

Laptop/Smartphone -> Desktop -> Webbrowser -> www.register.pka -> login with admin:admin

Devices can be controlle

Coffee Pot ins Internet hängen

Netcat starten: `nc -l 1234`

Javascript Code beim Coffee Pot unter Advanced -> Programming einfügen:

```
var state = 0;
var client = new RealTCPClient();
var coffeestatus = ["Off","On"];
function setup() {
// 17 Zeilen ...
setState(state);
client.connect("ip", 1234);
}
// 10 Zeilen ...
function setState(newState) {
// 10 Zeilen; client.send als letzte Zeile einfügen:
client.send(getName()+": "+coffeestatus[state]+"\n");
}
```

Netcat bekommt nun (nach Neustart und Cofee Pot Stop/Start/Altclick) Nachrichten

Mozilla Webthings Gateway

Lubuntu in VM Winstallieren, Firefox in der VM starten, Mozilla Webgateway installieren und starten, im Browser auf Configseite gehen.

Codeexample

```
Dependency installieren: pip install webthing
from __future__ import division
from webthing import (Action, Event, Property, Single Thing, Thing, Value,
                                                -
          Web Thi ngS erver)
import logging
import time
```

Codeexample (cont)

```
> import uuid
# Ein Event dass Überhitzung indiziert
class OverheatedEvent(Event):
    def __init__(self, thing, data):
        Event.__init__(self, thing, 'overheated', data=data)
# eine Action, welche die Lampe "faden" lässt
class FadeAction(Action):
    def __init__(self, thing, input_):
        Action.__init__(self, uuid.uuid4().hex, thing, 'fade', input_=input_)
    def perform_action(self):
        time.sleep(self.input['duration'] / 1000)
        self.thing.set_property('brightness', self.input['brightness'])
        self.thing.add_event(OverheatedEvent(self.thing, 102))
def make_thing():
    # lamp thing erstellen, kann ein/aus geschalten werden, hat verschiedene properties
    thing = Thing(
        'urn:dev:ops:my-lamp-1234',
        'My Lamp',
        ['OnOffSwitch', 'Light'],
        'A web connected lamp'
    )
    thing.add_property(
        Property(thing,
                 'on',
                 Value(True),
                 metadata={
                     '@type': 'OnOffProperty',
                     'title': 'On/Off',
                     'type': 'boolean',
                     'description': 'Whether the lamp is turned on',
                 }))
    thing.add_property(
```


Sponsored by ApolloPad.com
Everyone has a novel in them. Finish Yours!
https://apollopad.com

Codeexample (cont)

```
>         Property(thing,
                'brightness',
                Value(50),
                metadata={
                    '@type': 'BrightnessProperty',
                    'title': 'Brightness',
                    'type': 'integer',
                    'description': 'The level of light
from 0-100',
                    'minimum': 0,
                    'maximum': 100,
                    'unit': 'percent',
                }))
    thing.add_available_action(
        'fade',
        {
            'title': 'Fade',
            'description': 'Fade the lamp to a
given level',
            'input': {
                'type': 'object',
                'required': [
                    'brightness',
                    'duration',
                ],
                'properties': {
                    'brightness': {
                        'type': 'integer',
                        'minimum': 0,
                        'maximum': 100,
                        'unit': 'percent',
                    },
                    'duration': {
                        'type': 'integer',
                        'minimum': 1,
                        'unit': 'milliseconds',
                    },
                },
            },
```

Codeexample (cont)

```
>         },
        FadeAction)
    thing.add_available_event(
        'overheated',
        {
            'description':
            'The lamp has exceeded its safe
operating temperature',
            'type': 'number',
            'unit': 'degree celsius',
        })
    return thing
def run_server():
    # thing erstellen
    thing = make_thing()
    # webserver starten
    # If adding more than one thing, use
MultipleThings() with a name.
    # In the single thing case, the thing's
name will be broadcast.
    server = WebThingServer(SingleThing(-
thing), port=8888)
    try:
        logging.info('starting the server')
        server.start()
    except KeyboardInterrupt:
        logging.info('stopping the server')
        server.stop()
        logging.info('done')
if __name__ == '__main__':
    logging.basicConfig(
        level=10,
        format="%(asctime)s %(filename)s:%(-
lineno)s %(levelname)s %(message)s"
    )
    # server starten
    run_server()
```


Sponsored by **ApolloPad.com**
Everyone has a novel in them. Finish Yours!
https://apollopad.com