

Trabalho de aprofundamento 2

Francisco Xavier Santos Petronilho, Rui Miguel Silva Oliveira

UNIVERSIDADE DE AVEIRO



Trabalho de aprofundamento 2

DEPARTAMENTO DE ELETRÓNICA,
TELECOMUNICAÇÕES E INFORMÁTICA

UNIVERSIDADE DE AVEIRO

Francisco Xavier Santos Petronilho, Rui Miguel Silva Oliveira
(89241) franciscoxaviersp@ua.pt, (89216) ruimigueloliveira@ua.pt

Abril 2018

0.1 Resumo

Python é sem dúvida uma linguagem extremamente versátil e com a elaboração deste trabalho podemos entender mais profundamente as suas vantagens e a capacidade de resolução de problemas do nosso quotidiano.

Inicialmente foi nos proposto criar um cliente com a capacidade de aceder remotamente a uma sonda de temperatura, humidade e vento, sendo que esta possui a capacidade de enviar os dados através de uma rede sem fios. A aquisição é feita de forma automática e constante, emitindo um valor novo a cada 10 segundos.

O objetivo do trabalho é registar os dados num ficheiro Comma Separated Values (CSV) e apresentar algumas indicações sobre a necessidade de se levar certas peças de roupa ou acessórios consoante o clima.

A comunicação com a sonda é feita através de um socket Transmission Control Protocol (TCP), na porta 8080 do servidor xcoa.av.it.pt, enviando-se comandos de texto e recebendo-se objectos JavaScript Object Notation (JSON).

Conteúdo

0.1	Resumo	1
	Conteúdo	2
	Lista de Figuras	3
1	Introdução	4
2	Desenvolvimento	5
2.1	Comunicação com a Sonda	5
2.2	Tipos de Documentos	6
2.2.1	CSV	6
2.2.2	JSON	6
2.3	Mecanismos de Decisão	7
2.4	Encriptação	8
3	Conclusão	9
	Bibliografia	10

Lista de Figuras

2.1	Sequência de primitivas utilizadas num Socket TCP	5
2.2	Mecanismos de Decisão	7

Capítulo 1

Introdução

Qualquer projeto para que seja mais simples de compreender e também metódico na sua resolução do seu todo, terá de ser elaborado de forma faseada. Deste modo, para a resolução do tema que nos foi proposto, inicialmente tivemos que compreender em geral como se processa comunicação com a sonda, estudando deste modo o que seria um “socket Transmission Control Protocol (TCP)”.

A partir podemos tratar da informação que recebíamos do servidor para ser utilizada mais tarde nos métodos de decisão.

Por fim foi introduzida a possibilidade de troca de informação com o servidor de forma encriptada.

Capítulo 2

Desenvolvimento

2.1 Comunicação com a Sonda

O termo “comunicação”, neste caso com uma sonda, requer além demais conhecimento no que toca a conceitos de comunicação. Assim, para além das aulas onde aprendemos o essencial para conseguir-mos perceber no geral como tudo isto funcionava, tivemos de interiorizar o modelo “Cliente-Servidor”. Mais concretamente foi necessário perceber o que seria um socket e todo o tipo de programação associada ao mesmo. Depois desta ambientação este novo tipo de “programação” foi necessário conhecer a sequência de primitivas utilizadas num Socket TCP.

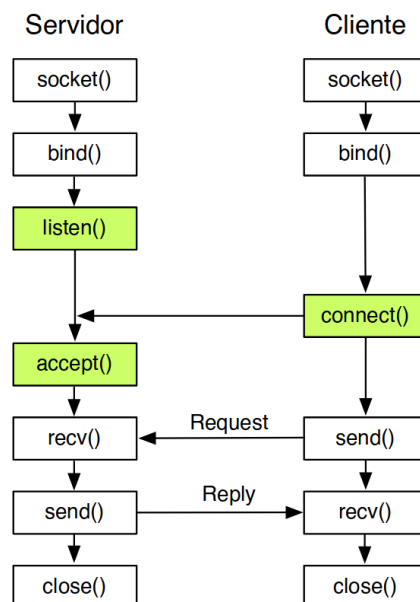


Figura 2.1: Sequência de primitivas utilizadas num Socket TCP

2.2 Tipos de Documentos

Relembramos que a comunicação com a sonda é feita através de um socket Transmission Control Protocol (TCP), na porta 8080 do servidor xcoa.av.it.pt, enviando-se comandos de texto e recebendo-se objectos JavaScript Object Notation (JSON) e que o objetivo do trabalho é registar dados num ficheiro CSV.

Assim surgem dois termos novos que tocam à manipulação de documentos em dois formatos muito comuns, usados para a troca de informação entre aplicações e dispositivos através de sockets, CSV e JSON.

2.2.1 CSV

O formato Comma Separated Values CSV é muito comum para a troca de informação especialmente neste tipo de trabalhos onde se troca informações com uma sonda. Um ficheiro CSV é, como o próprio nome indica, é um ficheiro sem formatação em que os valores estão separados por vírgulas, delimitados por aspas e, em que, cada linha tem um registo diferente, ou seja, um artigo, cliente ou fornecedor por linha. Assim era possível receber os valores da sonda e desta forma serem impressos no terminal.

2.2.2 JSON

O formato JSON assenta basicamente em duas estruturas:

Uma colecção de pares: chave/valor em algumas linguagens de programação tal estrutura é entendida como um objecto, lista, matriz, etc;

Uma lista ordenada de valores nas linguagens de programação é caracterizado como um array, vector, lista, etc;

Este tipo de estruturas de dados são transversais a quase todo o tipo de linguagens de programação modernas, o que faz do JSON é excelente escolha no que se refere ao formato para intercâmbio de informação.

De facto é possível converter qualquer estrutura de dicionário ou lista para o formato JSON e vice-versa, o que é extremamente útil para transmitir informação sobre Sockets.

Deste modo este formato foi útil na elaboração do trabalho pois este é responsável por guardar os valores que interessavam.

2.3 Mecanismos de Decisão

Relembrando o objetivo principal do projeto, a utilidade pretendida seria a de enviar indicações sobre a possibilidade (ou necessidade) de se levar t-shirt, casaco, gorro ou outras peças de roupa consoante o clima naquele instante de tempo.

Deste modo ficou ao critério do grupo criar mecanismos de decisão conforme a humidade a temperatura e o vento. Resumindo sucintamente todas as possibilidades, ficaríamos com 15 mensagens a transmitir no terminal com todas as combinações possíveis das três variáveis anteriormente descritas. Estas seriam:

```
if mhum>80:
    if mtemp<10:
        if mwind>40:
            if mtemp<=0:
                print("Temperaturas negativas! Recomendado usar roupa quente e roupas impermeaveis devido a probabilidade de chuva e ventos fortes\n")
            elif mtemp<=10:
                print("Recomendado usar um casaco quente, e roupas impermeaveis devido a probabilidade de chuva e ventos fortes\n")

            elif mtemp<=20:
                print("Recomendado usar um casaco,e roupas impermeaveis devido a probabilidade de chuva e ventos fortes\n")
            elif mtemp<=30:
                print("Recomendado uma camada de roupa fina,e roupas impermeaveis devido a probabilidade de chuva e ventos fortes\n\n")
            elif mtemp>30:
                ("Recomendado usar roupa de verao(calcoes, t-shirt, chapau, etc),e roupas impermeaveis devido a probabilidade de chuva e ventos fortes\n\n")

        else:
            if mtemp<=0:
                print("Temperaturas negativas! Recomendado usar roupa quente e guarda-chuva\n")
            elif mtemp<=10:
                print("Recomendado usar um casaco quente, e guarda-chuva\n")

            elif mtemp<=20:
                print("Recomendado usar um casaco,e guarda-chuva\n")
            elif mtemp<=30:
                print("Recomendado uma camada de roupa fina,e guarda-chuva\n\n")
            elif mtemp>30:
                print("Recomendado usar roupa de verao(calcoes, t-shirt, chapau, etc),e guarda-chuva\n")

    else:
        if mtemp<=0:
            print("Temperaturas negativas! Recomendado usar roupa quente \n")
        elif mtemp<=10:
            print("Recomendado usar um casaco quente\n")

        elif mtemp<=20:
            print("Recomendado usar um casaco\n")
        elif mtemp<=30:
            print("Recomendado uma camada de roupa fina\n")
        else:
            print("Recomendado usar roupa de verao(calcoes, t-shirt, chapau, etc)\n")
```

Figura 2.2: Mecanismos de Decisão

2.4 Encriptação

Capítulo 3

Conclusão

Bibliografia

- [1] eLearning UA (Laboratórios de Informática)
<https://elearning.ua.pt/course/view.php?id=3470>
- [2] PplWare
<https://pplware.sapo.pt/tutoriais/json-javascript-objection-notation-o-sucessor->