Escola Superior de Tecnologia e Gestão

**Supermercado/Comércio Inteligente**

****

Gonçalo Dias Pestana

José Francisco Sousa Fernandes

Trabalho de Projeto da unidade curricular de Tecnologias de Internet

Leiria, junho de 2022

# Lista de Figuras

Elemento a figurar, **quando aplicável**.

[Figura 1 - Arquitetura IoT genérica (altere para a sua) 2](#_Toc102681365)

[Figura 2 - Exemplo de um diagrama para representação de um evento 3](#_Toc102681366)

# Lista de tabelas

Elemento a figurar, **quando aplicável**.

**Não foi encontrada nenhuma entrada do índice de ilustrações.**

# Lista de siglas e acrónimos

Elemento a figurar, **quando aplicável**.

|  |  |
| --- | --- |
| API  CPT  ESTG  IOT | Interface de Programação de Aplicações  Cisco Packet Tracer  Escola Superior de Tecnologia e Gestão  Internet of Things |
| IPLeiria  LCD  MCU  SBC | Instituto Politécnico de Leiria  Dysplay de cristal líquido  Microcontrolador  Single-board computer |
| TI | Tecnologias de Internet |
| UC | Unidade Curricular |
|  |  |
|  |  |
|  |  |
|  |  |

Cuidados na elaboração da lista de siglas e acrónimos:

* Ordenação alfabética;
* Apenas as que sejam relevantes para a leitura do texto.

Adicionar mais entradas à tabela, caso seja necessário (a tabela não tem contornos, mas está no texto).

Índice

Trata-se de um elemento **obrigatório**. Nota: **o índice nunca figura do índice.**

[Lista de Figuras iv](#_Toc102681374)

[Lista de tabelas v](#_Toc102681375)

[Lista de siglas e acrónimos vi](#_Toc102681376)

[1. Introdução 1](#_Toc102681377)

[2. Arquitetura 2](#_Toc102681378)

[3. Implementação 3](#_Toc102681379)

[4. Cenário de Teste 4](#_Toc102681380)

[5. Resultados obtidos 5](#_Toc102681381)

[6. Conclusão 6](#_Toc102681382)

[7. Bibliografia 7](#_Toc102681383)

[8. Anexos 8](#_Toc102681384)

# Introdução

No âmbito da disciplina de Tecnologias de Internet, foi-nos proposto a realização de um projeto subordinado ao tema Supermercado/Comércio Inteligente.

O objetivo deste trabalho é que os estudantes apliquem os conhecimentos adquiridos nas aulas teóricas e pratico-laboratoriais da unidade curricular (UC) de Tecnologias de Internet (TI), de modo a definir e desenvolver um protótipo de uma solução de Internet of Things (IoT).

O protótipo a desenvolver deverá utilizar um cenário real ou virtual, através da utilização do software Cisco Packet Tracer (CPT), com a utilização de sensores, atuadores e controladores com capacidade de comunicação, de modo a interagir em tempo real com um servidor web, disponibilizando assim a informação *Anytime* & Anywhere.

Neste projeto serão implementadas as seguintes funcionalidades:

* Capacidade de atuação;
* Capacidade de sensing;
* Capacidade de transmissão e disponibilização da informação em tempo real;
* Disponibilização da informação Anytime & Anywhere;
* Capacidade de definição e envio de eventos por software e hardware;

Na estrutura deste projeto podemos identificar duas partes distintas, mas interligadas.

Uma dessas partes é a simulação de um ambiente virtual através do software Cisco Packet Tracer, onde são colocados diversos sensores e atuadores, com o principal objetivo de simular uma superfície comercial inteligente. A outra parte consiste na criação de um Website onde podemos visualizar as informações enviadas pelos censores, como por exemplo, os dados recolhidos por estes e o seu estado funcionamento.

# Arquitetura

Uma imagem com texto, mapa, interior

Descrição gerada automaticamente

Figura 1-Arquitetura IOT

# Implementação

Quando o sensor de movimento deteta alguém a entrar no supermercado (Cenário do Packet Tracer), este irá despoletar uma ação na dashboard através da API.

Quando a porta for aberta (Cenário do Packet Tracer), este irá despoletar uma ação na dashboard através da API.

Quando um incendio for detetado (Cenário do Packet Tracer), irá despoletar uma ação tanto do buzzer como do sprinkler(“Extintor”), alterando valores na dashboard através da API

Quando for clicado o botão de “Tirar Foto” na Dashboard, irá ser enviada informação à API que é necessária para tirar uma foto, e a mesma irá inforar o python para realizar a ação.

Quando forem clicados os botões de “Fechar/Abrir Portão”, irá ser enviada informação à API que é necessária para abrir/fechar o portão, e a mesma irá informar o python para realizar a ação.

Quando for clicado o rocket switch (Cenário do Packet Tracer), irá acender a lampada.

# Cenário de Teste

Para a realização deste projeto foram utilizados os seguintes softwares:

* Cisco Packet Tracer;
* Visual Studio Code;
* Uniform Server;

O Cisco Packet Tracer foi o software utilizado com o principal objetivo de simular a implementação e funcionamento de diversos dispositivos, sendo estes sensores, atuadores e dispositivos de rede. Este software permitiu-nos a implementação de vários subsistemas, tais como: sistema de iluminação, sistema de recolha de variáveis de ambiente (temperatura, humidade, luminosidade), porta automática, portão de garagem automático e, por fim, um sistema detetor de incêndio.

## Sistema de iluminação

O sistema de iluminação desenvolvido é composto por três elementos: um interruptor, um MCU e uma lâmpada. Estando dois destes elementos ligados a portas digitais do MCU, sendo o interruptor um dispositivo de input e a lâmpada um dispositivo de output. O MCU foi programado para que quando o interruptor fosse ligado, a lâmpada seria ligada também, estando sempre a enviar pedidos com o método POST para a API. Estes pedidos contêm o estado da lâmpada, ou seja, se esta se encontra ligada ou desligada e são registados numa base de dados, cujos conteúdos são mostrados na Dashboard.

## Sistema de recolha de variáveis de ambiente

O sistema de recolha de variáveis de ambiente é composto por quatro elementos: sendo três destes sensores (Humidade, Temperatura e luminosidade), e outro um MCU. Estes três sensores estão conectados a portas analógicas do MCU. Como no sistema anterior, o MCU foi programado para receber o valor dos sensores e enviá-los para a base de dados, fazendo para isso um pedido POST para a API.

## Sistema de porta automática

O sistema de porta automática é composto por um sensor de movimento e três atuadores, sendo estes: uma porta, uma camara e um LCD, todos conectados a um SBC. Quando o sensor de movimento deteta algo, os três atuadores realizam uma determinada ação. Neste sistema apenas dois dos seus elementos estão presentes na Dashboard, sendo estes a porta e o sensor de movimento. O envio da informação é realizado como nos sistemas anteriores.

## Sistema Portão de garagem automático

Este sistema é apenas composto por um portão de garagem, que se encontra conectado a um MCU. A diferença entre este sistema e os restantes, é que este utiliza o método GET para enviar pedidos para a API, abrindo ou fechando caso o valor seja respetivamente 1 ou 0.

## Sistema detetor de incêndio

O sistema detetor de incêndio foi provavelmente o mais complexo de se realizar, tendo sido necessário para a sua criação: um sensor, três atuadores e um MCU, para os conectar. Para conseguir simular o funcionamento deste sistema foi necessária a programação de um objeto que agisse como fogo, para ser detetado pelo sensor de incêndio. Quando detetado pelo sensor de incêndio, todos os atuadores reagem consoante a sua função. Por fim, todos os elementos pertencentes a este sistema com exceção de um LCD estão presentes na Dashboard.

# Resultados obtidos

No cenário do Cisco Packet Tracer, os MCU’S estão a enviar as informações de diversos sensores/atuadores para a API, a cada 5 segundos. Por outro lado, o script em Python está, a cada 5 segundos, a pedir o estado da webcam, e quando é pedido pela API para ligar a mesma, ativa a webcam do PC e tira uma foto. O SBC está, da mesma forma, a enviar informações da porta, do sensor de movimento e da webcam, a cada 5 segundos, para a API.

# Conclusão

Resumidamente, este projeto foi desenvolvido em prol de uma solução o mais eficaz possível, dentro dos possíveis, de forma a que tudo o que esteja implementado dê o menor trabalho possível ao utilizador. Existem pontos a melhorar, como a implementação de uma framework de Python/PHP, e também algumas bibliotecas que, eventualmente, poderiam dar jeito. Ambos os realizadores do projeto têm opinião que este projeto veio dar bastantes bases que, efetivamente, serão bastante uteis no futuro, e que o essencial dos objetivos enunciados foram cumpridos.

# Bibliografia

Inserir aqui a bibliografia.

Bibliografia – quando se coloca toda a bibliografia consultada;

# Anexos

Anexos, caso seja necessário incluir blocos de código maiores que sejam referenciados nos capítulos anteriores.