

# *Facebook Stop Scraper*



MSc in Informatics and Computer Engineering  
Programming Paradigms  
EIC0065-2S

Diogo Miguel Sousa Barroso - 201105481 (ei11105@fe.up.pt)  
José Pedro Vieira de Carvalho Pinto - 201203811 (ei12164@fe.pt.up)  
Miguel Cruz Fernandes - 201105565 (ei12137@fe.up.pt)

Faculdade de Engenharia da Universidade do Porto  
Rua Roberto Frias, sn, 4200-465 Porto, Portugal

8 de Julho de 2016

## **Abstract**

Este trabalho consiste na recolha de dados sobre Operações STOP na página oficial de Facebook, o seu tratamento e armazenamento numa base de dados. Após esta fase, os dados estão prontos a serem mostrados em tabela ou gráfico numa plataforma web.

# Contents

<b>1</b>	<b>Introduction</b>	<b>4</b>
1.1	Objective . . . . .	4
1.2	Motivation . . . . .	4
<b>2</b>	<b>System Description</b>	<b>5</b>
2.1	Conceptual Description . . . . .	5
2.1.1	Functionalities . . . . .	5
2.1.2	Architecture . . . . .	5
2.1.3	Programming Languages . . . . .	5
2.2	Implementation Description . . . . .	6
2.2.1	Implementation Details . . . . .	6
2.2.2	Development Environment . . . . .	6
<b>3</b>	<b>Conclusion</b>	<b>7</b>
<b>4</b>	<b>Improvements</b>	<b>8</b>

# **1 Introduction**

## **1.1 Objective**

O objetivo do STOP Scrapper é o de alertar a comunidade para as Operações STOP existentes. Da mesma forma que a Polícia de Segurança Pública as divulga com uma intenção preventiva, este projeto pretende juntar todas aquelas que foram divulgadas e demonstrar, não só, em tempo real quais estão ativas, como também estatísticas acumuladas ao longo do tempo. Com isto, o utilizador pode retirar maior proveito dos dados fornecidos.

## **1.2 Motivation**

A ideia para o STOP Scrapper surgiu pela necessidade de um dos membros do grupo se lembrar de algumas das operações STOP da sua zona, não tendo algo pratico para assim o consultar. Assim, o primeiro objetivo foi o de construir uma aplicação para dispositivos móveis que alertasse o utilizador para operações nas suas proximidades. No entanto, após algumas dificuldades a nível de correspondência entre morada e coordenadas geográficas, decidiu-se construir um website que reúne estatísticas e mostra as operações a decorrer em tempo real. Este projeto tem interesse não academico e, potencialmente, de comercialização, e é interesse do grupo fazê-lo.

## 2 System Description

### 2.1 Conceptual Description

#### 2.1.1 Functionalities

Em termos de funcionalidades para este projecto, falando primeiro do ponto de vista do utilizador comum, temos uma interface web que disponibiliza uma pesquisa sobre as operações Stop que já ocorreram ou que irão ocorrer em todo o país. Esta pesquisa pode ser efetuada selecionando o distrito do qual queremos ver as operações Stop ou selecionando a opção “All” que apresentação todos os resultados disponíveis. A seleção da opção para a pesquisa é feita utilizando um menu dropdown e a cada pesquisa realizada está associada uma representação gráfica dos resultados na forma de uma tabela (com os resultados ordenados por data e hora) juntamente com dois gráficos circulares que apresentam a distribuição das operações Stop por mês e por hora, dando uma perspectiva dos dados que não é obtida quando apenas apresentados numa tabela.

Além da interface web construída, temos também disponível, uma Application Program Interface (API) que permite o acesso às todas funcionalidades presentes na interface web. No entanto, no servidor onde está alojada esta API temos também uma tarefa que se realiza a todo o minuto 0 de cada hora. Esta tarefa consiste na execução de um script de Python que trata da extração do conteúdo relativo as operações Stop utilizando a API publica do Facebook. Depois de extraído o conteúdo é armazenado numa base de dados MySQL onde é posteriormente acedida pela nossa API.

#### 2.1.2 Architecture

A aplicação num todo, encontra-se da seguinte maneira:

- A base de dados MySQL;
- O script de Python que recolhe a informação pública e a armazena na base de dados;
- O servidor NodeJS que contem a API pública e executa o script de python;
- A interface web que utiliza a API pública e representa os dados

#### 2.1.3 Programming Languages

- Python - ligação ao Facebook, tratamento de dados e ligação à base de dados;

- MySQL - base de dados;
- HTML, CSS - Front-end;
- Javascript - Usado em ambos no front-end e como linguagem do servidor.

## **2.2 Implementation Description**

### **2.2.1 Implementation Details**

Para a implementação do projecto, houveram vários aspectos que são de salientar:

- No desenvolvimento dos scripts de Python, estes foram usados exclusivamente no servidor desenvolvido. Durante o seu desenvolvimento foi utilizada a Graph API do Facebook, extraindo o conteúdo que necessitamos da página da PSP. Seguido disto foi implementado um parser que trata esta informação e, depois de feita a conexão à base de dados, trata do armazenamento da mesma.
- O servidor foi implementado, tendo como base NodeJS, utilizado a framework Hapi para gerir os pedidos HTTP que constituem a API. De modo, a correr ficheiros Python “.py”, foi utilizado o “PythonShell”. Como tínhamos como objectivo atualizar a base de dados com alguma frequência, utilizando o módulo “node-schedule” pudemos criar uma rotina de corre de hora em hora mantendo os registos de operações atualizados.
- A interface web foi desenvolvida utilizando HTML e CSS sendo que para obter os dados do servidor, utiliza-se Javascript e os pedidos Ajax que vêm com a biblioteca jQuery.

### **2.2.2 Development Environment**

O ambiente de desenvolvimento era bastante diversificado, dado o facto de cada membro do grupo estar a trabalhar numa componente diferente. Como editor de código, o software utilizado dependeu unicamente das preferencias de cada membro. No entanto para meios de testar o software utilizado toda a parte de backend não necessitou de qualquer interface sendo tudo verificado usando ferramentas de linha de comandos. Para a interface web, foi utilizado o Google Chrome.

### **3 Conclusion**

Com este projecto concluído verificamos que cumprimos todos os objectivos que nos propusemos no inicio do mesmo. Tivemos algumas dificuldades iniciais com a ligação a ser feita entre o servidor NodeJS e o Python, sendo que para a solução acabamos por recorrer a um módulo de NodeJS externo chamada PythonShell. Como grupo, tivemos o trabalho bem estruturado em 3 grandes parte que serviram como base de distribuição conseguindo desta forma com que todos os 3 membro trabalhassem igualmente no projecto.

## 4 Improvements

Como melhoramento seria interessante ter uma base de dados mais alargada, no sentido em que teríamos uma fonte maior do tipo de informação que pretendemos. No entanto, não é garantido que tais publicações sejam divulgadas, podendo o nosso site cair em desuso. Como tal, uma expansão focada na comunidade, em que cada pessoa pudesse também inserir locais de operações STOP na nossa base de dados, seria de elevado interesse.