

Trabalho Prático

Trabalho realizado por:

João Pedro Brandão Moreira de Sousa, 8180175

Disciplina:

Administração de Sistemas Informáticos

Conteúdo

Introdução	3
Funcionamento do trabalho	4
Implementação e Escolhas	6
Módulos	6
Funções	Erro! Marcador não definido.
TinyDB	6
Kibana	Erro! Marcador não definido.
Referências	7

Introdução

O trabalho prático consiste na implementação de um sistema de monitorização utilizando a linguagem de programação Python no âmbito da Unidade Curricular de Administração de Sistemas Informáticos da Licenciatura de Engenharia Informática da Escola Superior de Tecnologia e Gestão. Assim sendo, este é composto por um agente que executa localmente em cada máquina, máquina virtual ou container, recolhendo informação do sistema em causa e enviando os resultados da recolha da informação para um servidor central.

Funcionamento do trabalho

Servidor

O servidor, uma vez ligado, é responsável por esperar e responder a pedidos dos agentes.

O servidor é constituído por duas funções que comunicam com o agente, “genKey” e “setData”. E outras duas para o tratamento e geração de gráficos no Kibana, “treatData” e “genKibana”.

A primeira função é responsável por receber e tratar as informações da máquina do agente, sendo assim, guarda o endereço de IP e o sistema operativo recebidos do agente e procura na base de dados se já existe alguma máquina com esses elementos, se já houver na base de dados uma key gerada para aquele conjunto de elementos o servidor apenas retorna a key como resposta, se ainda não houver, o servidor trata de gerar uma key e faz o registo na base de dados dessa key juntamente com o IP e Sistema Operativo, posteriormente envia como resposta a key ao agente.

A segunda função é responsável pelo tratamento das monitorizações do agente, assim sendo recebe as monitorizações, verifica se já existem dados daquela máquina no servidor e se já houver acrescenta a informação, caso ainda não exista ele adiciona.

Relativamente às funções para o kibana, a “genKibana” é utilizada apenas uma vez para cria um index para o kibana e a “treatData” é utilizada sempre que o servidor recebe dados para que estes sejam mandados para o kibana.

Agente

O cliente, uma vez ligado, manda pedidos e entre em ciclo para que de 5 em 5 minutos manda novamente mais pedidos.

Este é caracterizado por duas funções “askKey” e “sendData”.

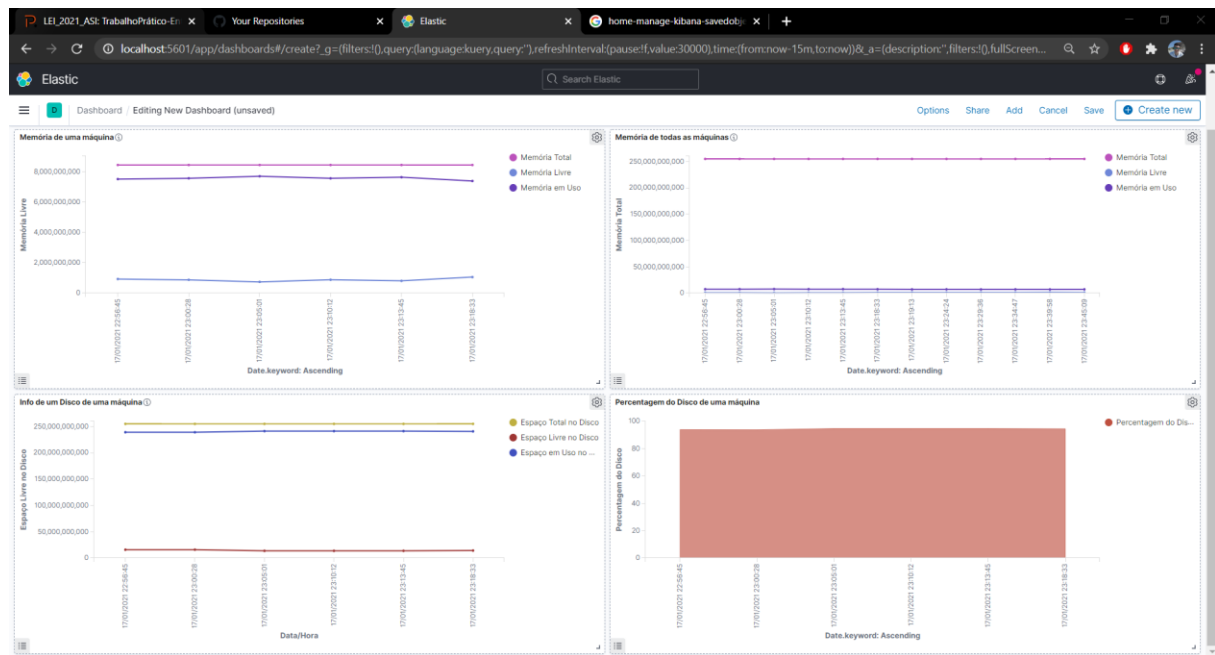
A primeira função responsável por enviar um pedido ao servidor para que este o registre na base de dados e lhe retorne uma key que ele guarda.

A segunda função é responsável por recolher os dados usando para isso funções que recolhem os dados da máquina e que estão presentes nos módulos.

Kibana

O Kibana pode ser acedido na porta 5601 do localhost e é responsável por apresentar os dados recolhidos.

As funções responsáveis pelo Kibana estão presentes no servidor porque são muito dependentes deste projeto e da forma como a base de dados está guardada.~



Implementação e Escolhas











Módulos

Foram usados módulos no lado do cliente para um melhor isolamento e reutilização do código. Dessa forma, foram criados 6 módulos que recolhem dados da máquina

TinyDB

Foi escolhida pela sua simplicidade de implementação e utilização.

Referências

-  **Regex - Make expressions** <https://regex101.com/>
-  **Commenting functions** <https://realpython.com/documenting-python-code/#commenting-code-via-type-hinting-python-35>
-  **Por no relatorio** <https://psutil.readthedocs.io/en/latest/>
-  **psutil · PyPI** <https://pypi.org/project/psutil/>
-  **Kibana** <https://kb.objectrocket.com/elasticsearch/how-to-bulk-index-elasticsearch-documents-from-a-json-file-using-python-753>
-  **Inodes** <https://stackoverflow.com/questions/20727179/calculate-percentage-inode-usage-using-python>
-  **Inodes** <https://www.geeksforgeeks.org/python-os-statvfs-method/>
-  **TinyDB** <https://tinydb.readthedocs.io/en/stable/usage.html>
-  **socket (portas...)** <https://docs.python.org/2/library/socket.html>
-  **Criar Index Pattern Kibana** <https://discuss.elastic.co/t/how-to-create-a-default-index-pattern-via-python-api/243085>