Trabalho final

Data final de entrega do trabalho: **20 de junho 2020**
Para além do trabalho que cada grupo realizará autonomamente, haverá aulas específicas para apoio à realização do trabalho.
Componentes a entregar:
1) Ficheiro ZIP com as componentes desenvolvidas, incluindo ficheiros README com informações sobre configurações, pressupostos de execução, testes etc.
2) Documento em formato PDF com descrição da solução: Diagramas de arquitetura e de interação entre as partes envolvidas; protocolos de aplicação e formatos de dados envolvidos nas interações, bem como os aspectos relevantes da implementação, nomeadamente o processo de garantia e controlo da elasticidade de processamento e eventuais pontos de falha.

Objetivos: Saber planear e realizar um sistema para submissão e execução de tarefas, com requisitos de elasticidade, utilizando de forma integrada serviços da Google Cloud Platform de armazenamento e de comunicação, nomeadamente, Storage, Firestore e Pub/Sub, e o serviço de computação, Compute engine.

Descrição: Desenvolva um sistema, designado `CNText`, para detetar texto em imagens e traduzir esse texto para diferentes linguagens. O sistema deve ter elasticidade, aumentando ou diminuindo a sua capacidade de processamento de imagens. As funcionalidades do sistema estão disponíveis para aplicações cliente através de uma interface gRPC com as seguintes operações:

- Iniciar sessão indicando o nome do utilizador e retornando um identificador de sessão que será usado nas restantes operações;
- Terminar sessão indicando o identificador de sessão;
- Submissão de imagem para deteção e tradução de texto. Esta operação recebe um ficheiro e o identificador da língua de tradução (“pt”, “en”, “es”, “it”, ...), retornando um descritor de pedido único do serviço para usar na operação de interrogação. Note que ficheiros de pequena dimensão podem ser transferidos num único pedido. No entanto, valoriza-se o suporte do envio de ficheiros de qualquer dimensão.
- Obter, para um determinado descritor de submissão, o texto detetado e, eventualmente, a língua detetada no texto, com a respetiva tradução para uma outra língua solicitada.

O sistema *CNText* usa os serviços Cloud Storage, Firestore, Pub/Sub e Compute Engine, de acordo com a Figura 1:

- O Cloud Storage armazena as imagens a processar;
- O Firestore guarda os textos detetados nas imagens e as traduções;

- O Pub/Sub é usado para troca desacoplada de mensagens entre os componentes;
- O Compute Engine através de máquinas virtuais ou grupos de instâncias que executam os programas (*workers*) de deteção de texto (OCR) e de tradução (Translation).

A visão geral do sistema, e as diferentes interações entre os seus componentes, é apresentada na Figura 1.

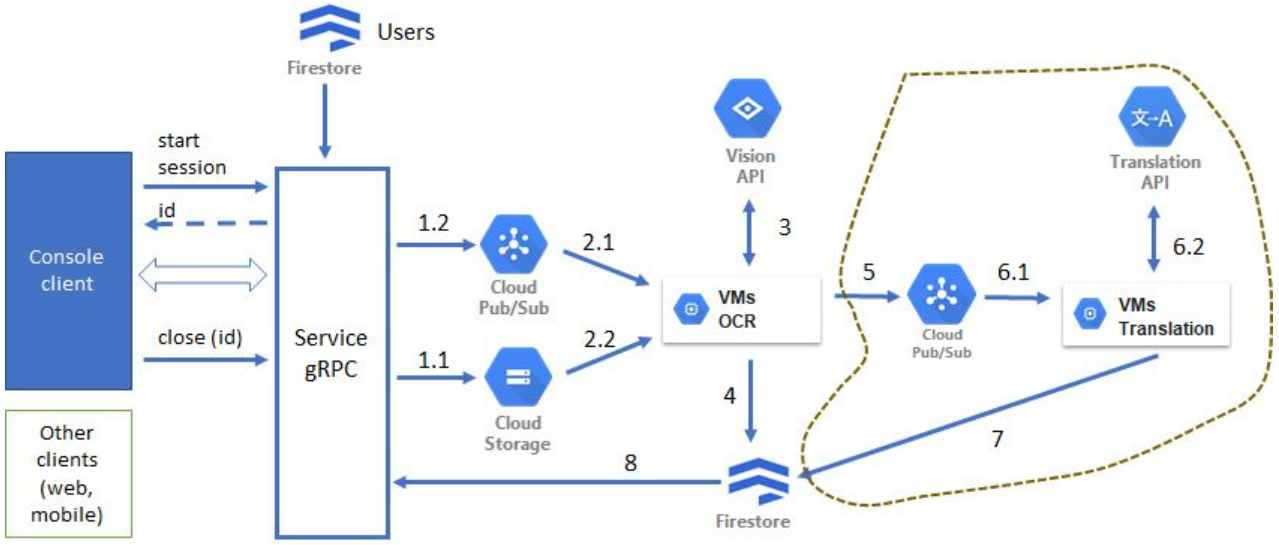

Figura 1: Componentes do *CNText* e respetivas interações

Qualidade de serviço e elasticidade:

O sistema *CNText* oferece dois níveis de qualidade de serviço: *free* e *premium*. Os utilizadores estão associados a um dos níveis de serviço. Essa associação é estabelecida previamente na coleção *users* da base de dados Firestore. Em cada momento, o serviço gRPC sabe quantos utilizadores *free* e *premium* têm sessão iniciada.

Os níveis de serviço influenciam a forma como os pedidos de tradução de cada utilizador são processados:

- Para tratar os pedidos do nível *free* existe apenas 1 VM de OCR e 1 VM de tradução, isto é, só é iniciado o processamento de um pedido após a conclusão do anterior;
- Para tratar os pedidos do nível *premium* existe um número variável de VMs (instance groups) para OCR e tradução, isto é, múltiplos pedidos em simultâneos. Valoriza-se soluções em que cada VM possa também processar vários pedidos em simultâneo. É responsabilidade do serviço gRPC aumentar ou diminuir o número de VMs, usando uma métrica que entenda adequada.

Fluxo de operações:

O fluxo de operações do sistema é o seguinte, tendo em conta os números apresentados na Figura 1:

- Os utilizadores iniciam sessão, através do cliente gRPC, indicando o seu identificador de utilizador (note que os utilizadores têm de estar previamente criados na coleção `users` do Firestore, com indicação do *username* e nível de serviço). O serviço retorna um identificador de sessão, que é guardado pelo cliente gRPC e usado nos pedidos seguintes. Por simplificação, assume-se que não existe autenticação de utilizadores;

- Após a submissão de uma imagem, a mesma é guardada no Cloud Storage (1.1) e é retornado ao cliente gRPC um identificador único para posteriormente ser possível pedir o texto traduzido;

- Os pedidos de OCR chegam às VMs através do serviço Pub/Sub (1.2). Os pedidos *free* são enviados para o tópico `free-ocr`, ao qual está associada uma subscrição que é consumida por uma única VM. Os pedidos premium são enviados para o tópico `premium-ocr` ao qual está associada uma subscrição que é consumida por várias VMs (*work-queue pattern*).

- Cada VM (ou *worker*) de OCR recebe o nome da imagem a processar (2.1) e obtém o seu conteúdo do Cloud Storage (2.2), interagindo depois com a API de Visão (3). O texto detetado é guardado no Firestore (4).

- Após o processamento a imagem é apagada do *Storage e é* enviada uma notificação com o resultado do OCR para um tópico Pub/Sub (5), seguindo a mesma lógica para os pedidos *free* e *premium*, isto é, tópicos diferentes (`free-translate` e `premium-translate`) no que diz respeito à comunicação, e VMs diferentes para o processamento.

- Cada *worker* de tradução recebe o texto a traduzir (6.1) e interage com a API de tradução (6.2). O texto traduzido é guardado no Firestore juntamente com o respetivo texto original (7);

- A qualquer momento, as aplicações cliente do serviço podem pedir o texto traduzido, com base no descritor do pedido. O serviço gRPC consulta a base de dados Firestore para obter o texto traduzido (8).

- Todos os pedidos têm de validar o identificador de sessão passado como parâmetro, recusando os pedidos com identificador inválido.

Aspetos de implementação:

Tenha em conta que:

- Para especificar uma operação num contrato gRPC que recebe o conteúdo de um ficheiro pode usar o tipo **`bytes`** da linguagem `protobuf`, que na API Java da Google corresponde ao tipo **`ByteString`**.
  (https://developers.google.com/protocol-buffers/docs/proto3#scalar)
- A API de visão faz OCR sobre imagens, retornando a totalidade do texto detetado e a descrição das zonas dentro da imagem (não usadas no caso do trabalho) onde foram detetadas as várias partes do texto.
  (https://cloud.google.com/vision/docs/supported-files, https://cloud.google.com/vision/docs/languages)
- A API de tradução recebe, na forma de *strings*, um texto para traduzir, a língua desse texto, permitindo ser detetada automaticamente, e a língua do texto de destino. As línguas têm siglas pré-definidas, por exemplo, “pt”, “en”, “es” ou “it”.
  (https://github.com/googleapis/java-translate)

Critérios de avaliação do trabalho:

- Os pontos 5 a 7 identificados na Figura 1 (caixa tracejada) são desafios opcionais, isto é não existe tradução de textos. Caso não sejam realizados a nota máxima é 17 valores.
- O trabalho será avaliado segundo os seguintes critérios:
  - 35% - qualidade do relatório, que permita a um leitor entender claramente a arquitetura e as decisões de interação entre as partes, evitando apresentar código excepto se o mesmo ajudar a explicar detalhes relevantes. O relatório deve indicar os pressupostos assumidos, indicando eventuais comparações como outras decisões possíveis. Deve constar no relatório qual a(s) parte(s) onde cada elemento do grupo teve mais ou menos responsabilidade.
  - 50% - Operacionalidade, simplicidade e flexibilidade das soluções, nomeadamente na configuração e utilização da solução;
  - 15% - participação individual de cada elemento do grupo durante as aulas afetas à realização do trabalho, bem como na apresentação do trabalho à turma.

José Simão

Luís Assunção