ISEL
INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA

Relatório de Projeto

CNV2024TF

Grupo 11 | CN | 25-05-24

Orientadores:

José Simão

Fernanda Passos

João Mota – 49508
Ricardo Rovisco – 49487
Tiago Neves – 48292

Índice

Introdução

Neste relatório vamos demonstrar o desenvolvimento do sistema *CNV2024TF* que nos foi proposto na unidade curricular de Computação na Nuvem.

O sistema consiste numa aplicação de deteção de características em ficheiros de imagem (JPG, PNG, etc) traduzindo as mesmas de inglês para português bem como efetuar a submissão e o download de imagens.

Com o desenvolvimento desta aplicação pretende-se saber planear e realizar sistemas de submissão e execução de tarefas de computação na nuvem usando serviços da Google Cloud Platform para armazenamento, comunicação e computação com Cloud Storage, Firestore, Pub/Sub, Compute Engine e Cloud Functions.

Abstract

In this report, we will demonstrate the development of the CNV2024TF system proposed in the Cloud Computing curricular unit.

The system consists of an application that detects features in image files (JPG, PNG, etc.) translating them from English to Portuguese as well as submitting and downloading images.

With the development of this application, the aim is to know how to plan and carry out systems for submitting and executing computing tasks in the cloud using Google Cloud Platform services for storage, communication and computing with Cloud Storage, Firestore, Pub/Sub, Compute Engine and Cloud Functions.

Formulação do Problema

Problemas – Cliente

O componente Cliente é responsável por interagir com os utilizadores ou aplicações cliente, fornecendo uma interface intuitiva para submissão de imagens, obtenção de resultados e informações sobre as imagens processadas. Ele utiliza a interface gRPC para comunicar com o servidor.

Lookup

A função Lookup desempenha um papel fundamental na disponibilidade e no load balancing do sistema. Ele é responsável por obter os endereços IP das instâncias do servidor gRPC, permitindo que o cliente se conecte a uma instância disponível. Essa funcionalidade é implementada como uma Cloud Function, facilitando a atualização dos endereços IP em tempo real.

Operações base

No modulo de cliente, as operações base são submitFile, getImageLabels, getNamesFromDateAndLabel e downloadFile. Todas estas operações irão ser explicadas de forma mais detalhada ao longo do relatório.

Problemas – Servidor

O componente Servidor é a parte central do sistema, que recebe as solicitações do cliente através da interface gRPC e coordena o processamento dos ficheiros. Ele interage com os serviços da Google Cloud Platform, como Cloud Storage, Firestore, Pub/Sub, Compute Engine, Cloud Functions, Vision API e Translation API, para armazenar os ficheiros, identificar caracteristicas, obter labels e armazenar informações relevantes. O servidor também retorna os resultados ao cliente.

A divisão do projeto em partes bem definidas facilita o desenvolvimento, a manutenção e a compreensão do projeto. Cada componente possui responsabilidades específicas possibilita uma solução robusta e escalável na nuvem.

Cloud Storage

O serviço Cloud Storage armazena as imagens a processar, enviadas pelo cliente. Estas imagens são Blobs que são guardados num Bucket específico.

Firestore

O serviço Firestore guarda a informação relevante sobre processamente de uma imagem, nomeadamente o identificador do pedido, data do processamento, as características detetadas na mesma e a sua respetiva tradução.

Pub/Sub

O serviço Pub/Sub é usado para a troca desacoplada de mensagens entre os servidores gRPC e a aplicação de processamento de imagens, LabelsApp.

Problemas – LabelsApp

A aplicação Labels tem como objetivo processar as imagens fornecidas e armazenar as mesmas, para tal recorre a duas APIs, Vision API e Translation API, para as funcionalidades de processamento e recorre à Firestore para armazenamento.

Vision API

A Google Vision API é usada para detetar caracteristicas nas imagens. O problema principal relacionado a essa API é realizar a deteção precisa das caracteristicas nas imagens fornecidas. Garantir a deteção correta e precisa das caracteristicas é essencial para o funcionamento adequado da aplicação.

Translation API

A Google Translation API é utilizada para realizar a tradução das labels que foram geradas.

Problemas – Logging

A aplicação Logging tem como objetivo armazenar na Firestore os registos dos pedidos de submissão de imagens.

Problemas – Contratos

O componente Contratos define o contrato da interface gRPC, estabelecendo as operações disponíveis e os parâmetros necessários para cada operação. Isso promove uma comunicação consistente entre o cliente e o servidor, garantindo que ambos estejam em conformidade com o contrato estabelecido.

Diagramas de Arquitetura

Organização do Projeto

O sistema desenvolvido está dividido em 5 componentes distintos:

- Cliente;
- Servidor;
- Contratos;
- *LabelsApp*;
- *Logging*;

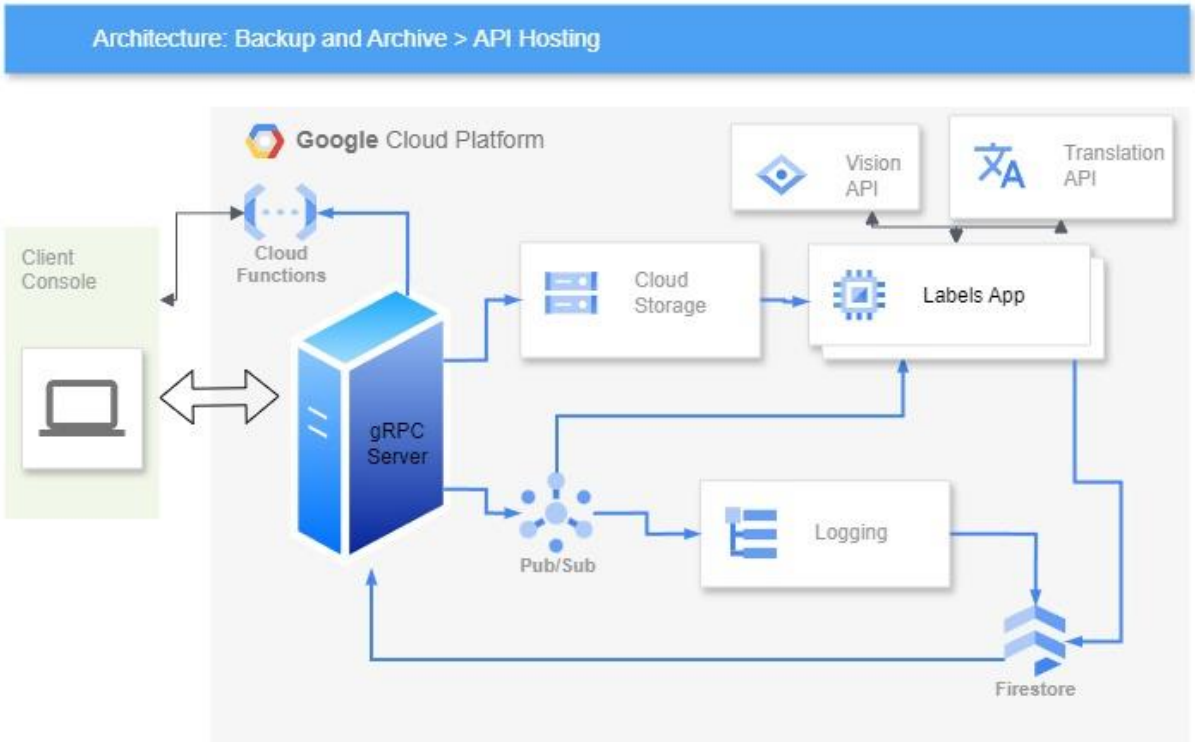

Figura 1 Arquitetura do Sistema CNV2024TF

Componentes do Sistema

Cliente

O componente Cliente é o responsável pela interação do utilizador com a aplicação. Este utiliza tecnologia gRPC (Google Remote Procedure Call) para que o utilizador possa interagir a partir de um serviço remoto.

Inicialmente este componente estabelece uma conexão com o serviço remoto e então configura o stub gRPC e fornece um menu com as diferentes funcionalidades da aplicação para que o utilizador possa interagir com o serviço.

A funcionalidades disponibilizadas ao utilizador incluem:

1. Submissão de Imagens – Nesta operação, o utilizador fornece o nome da imagem que pretende submeter cujo conteúdo será posteriormente processado numa stream de blocos e guardado como um blob no serviço Cloud Storage devolvendo o identificador do pedido.
2. Download de imagens – O utilizador pode efetuar o download de uma imagem anteriormente submetida fornecendo o identificador do blob armazenado no serviço Cloud Storage bem como o caminho para onde quer efetuar o download da imagem.
3. Obter as características de uma imagem – A aplicação permite ao utilizador obter uma lista com as características de uma imagem anteriormente submetida com o identificador fornecido pelo utilizador.
4. Obter as imagens que foram submetidas entre duas datas com uma característica - O utilizador pode efetuar um pedido à aplicação para que lhe seja fornecido uma lista com os identificadores de imagens anteriormente submetidas entre duas datas e com uma determinada característica (estes parâmetros são fornecidos à aplicação pelo utilizador).
5. Aumento ou diminuição das instâncias do servidor ou da LabelsApp – O utilizador pode solicitar um aumento ou uma diminuição das instâncias tando do Servidor como da LabelsApp. Para tal, o mesmo tem de fornecer o número de instâncias que deseja ter tendo anteriormente indicado qual das operações deseja efetuar.
6. Obter os IPs das diferentes instâncias do servidor – Esta operação tem como objetivo fornecer ao utilizador uma lista dos IPs de cada instância ativa do Servidor.
7. Sair da aplicação

Servidor

No componente Servidor é onde vai ocorrer o processamento dos diferentes pedidos efetuados pelo utilizador no componente Cliente.

Este componente efetua o redirecionamento para os devidos módulos da aplicação. Se a intenção do utilizador for submeter uma imagem, o servidor, utilizando Pub/Sub, efetua a publicação para as diversas subscrições nomeadamente a LabelsApp, onde a mesma será armazenada, e a LoggingApp onde será armazenado o pedido efetuado à aplicação. Por outro lado, se o utilizador efetuar um pedido relacionado a alguma das outras operações da aplicação para obter qualquer tipo de informação sobre imagens submetidas, os mesmos serão feitos utilizando a Cloud Storage.

Para que fosse possível devolver a resposta do Servidor para o Cliente onde será apresentada ao utilizador, são enviadas pelo cliente *Streams* dos diferentes modelos de dados referidos nos respetivos contratos. Na operação de submissão de imagem foi necessário criar uma classe *SubmitFileStream* para que o identificador da imagem submetida fosse corretamente apresentado ao utilizador.

Contratos

Para descrever os serviços e o formato de dados utilizado nos mesmos, entre o Cliente e o Servidor, foi necessário definir dois contratos de protocolo para que estes consigam comunicar de uma forma eficiente.

Contrato de Serviço

Neste contrato estão definidos os serviços relacionados com imagens seja de submissão, download ou obtenção de características.

- Formato de dados:
    1. *TextMessage* – Utilizado pelos serviços de submissão/download de imagens e também pelo serviço de obtenção de características, para o envio e recebimento de mensagens/informações do Servidor.
    2. *InputFile* – Utilizado pelo serviço de submissão de imagens para enviar o conteúdo, o tipo de ficheiro e o nome da mesma para o Servidor.
    3. *FileLabels* – Utilizado pelo serviço de obtenção de características de uma imagem para receber a lista com as mesmas.
    4. *FileNames* – Usado pelo serviço de obtenção de imagens submetidas entre duas datas com uma determinada característica para receber a lista com os nomes das imagens.
    5. *DatesAndLabel* – Usado para enviar ao serviço de obtenção de imagens submetidas entre duas datas com uma determinada característica contendo as duas datas e a característica fornecida pelo utilizador.
    6. *DownloadedFile* – Usado pelo serviço de download de imagens para receber o conteúdo da imagem com o identificador fornecido pelo utilizador.
- Serviços:
    1. *submitFile* – Este serviço é responsável por tratar da operação de submissão de ficheiros. Recebe um ficheiro no formato *InputFile* e retorna uma mensagem no formato *TextMessage*.
    2. *getImageLabels* – Este serviço trata da operação de obtenção de características de uma determinada imagem. Para isso, recebe uma mensagem no formato TextMessage que representa o identificador de uma imagem e retorna a lista das características da mesma.
    3. *getNamesFromDateAndImages* – Neste serviço é tratada a operação de obtenção da lista das imagens anteriormente submetidas entre duas datas que tenham uma determinada característica. Para isso, esta operação recebe as datas e característica fornecidas pelo utilizador no formato *DatesAndLabel* e retorna uma lista que contém os identificadores de cada imagem com os parâmetros anteriormente referidos.
    4. *dowloadFile* – Este serviço é responsável pela operação de download de uma imagem. A operação recebe o identificador da imagem fornecido pelo utilizador no formato *TextMessage* e retorna o conteúdo da respetiva imagem no formato *DownloadFile*.

Contrato de Escalabilidade

Neste contrato estão definidos os serviços relacionados com imagens seja de submissão, download ou obtenção de características.

- Formato de dados:
    1. *ResizeRequest* – Este formato de dados é utilizado pelo serviço de redimensionamento do número de instãncias do Servidor ou *LabelsApp*. Este contém o identificador do projeto e do grupo de instâncias bem como a zona onde se localiza o projeto e o número de instâncias fornecido pelo utilizador.

- Serviços:
    1. *resizeInstance* – Este serviço é responsável por tratar da operação de aumento ou diminuição do número de instâncias do Servidor ou da *LabelsApp.* Para tal, o mesmo recebe o identificador do projeto, a zona do mesmo e o identificador do grupo de instâncias juntamente com o número fornecido pelo utilizador para aumentar ou diminuir as instâncias no formato *ResizeRequest*.

LabelsApp

Para um bom funcionamento da aplicação, quando uma imagem é submetida é preciso guardar a informação sobre a mesma. Este componente efetua o armazenamento das imagens bem como as suas caracteristicas no *Firestore.* Desta forma o componente em questão processa os labels usufruindo a *Vision API* redirecionando os mesmos para a *Translation API* para que a mesma efetue a tradução dos mesmos de inglês para português posteriormente efetuando o armazenamento no *Firestore*.

Logging

No sistema desenvolvido, é feito o registo dos pedidos de submissão de imagem efetuados pelo utilizador. Para isso o componente Logging é responsável por armazenar os mesmos no Firestore. Este armazena o identificador da imagem bem como a data em que a mesma foi submetida.

Conclusões

Com este trabalho podemos compreender de uma forma mais clara o funcionamento do modelo gRPC e das diferentes Google APIs utilizadas no mesmo.

Deparámo-nos com diversos problemas no desenvolvimento do sistema nomeadamente na configuração dos componentes, na conexão entre os mesmos e na utilização da Google Cloud. Contudo pensamos ter cumprido com sucesso o desenvolvimento do sistema pois conseguimos implementar todas as funcionalidades pretendidas para o mesmo.

Resumindo, conseguimos cumprir os objetivos do trabalho com sucesso tendo aprofundado os nossos conhecimentos sobre algumas das tecnologias usadas na área de Computação na Nuvem.

Distribuição de Trabalho

No que toca à distribuição de trabalho pelos elementos do grupo, desenvolvimento dos contratos e das funcionalidades base como a submissão de imagens, obtenção de características de imagens, obtenção dos identificadores das imagens submetidas entre duas datas e o download de imagens podemos afirmar que a distribuição foi feita de forma uniforme. Contudo, o desenvolvimento das últimas funcionalidades, como o aumento das instâncias do Servidor e da LabelsApp e do componente Logging apesar de todos os elementos terem trabalhado, o integrante do grupo João Mota foi quem teve uma maior participação no desenvolvimento dos mesmos.