

## Laboratório 3

### Objetivo:

- Ganhar prática e consolidar os conceitos fundamentais do serviço Cloud Storage
- Utilização da API do Cloud Storage
- Consolidar conceitos sobre gRPC

1) Usando a interface *web*, e seguindo os passos apresentados nos slides, crie um *bucket* regional na Europa e outro na Ásia.

2) Nesta alínea use a interface *web* e a ferramenta *gsutil* já existente na *cloud shell*.

a) Aceda à Google Cloud Shell.



b) Na *Cloud Shell* faça *download* da seguinte imagem do site *nasa.org*:

```
wget https://images-assets.nasa.gov/image/iss067e022083/iss067e022083~orig.jpg
```

c) Renomeie o ficheiro para *space-station.jpg*.

```
mv iss067e022083~orig.jpg space-station.jpg
```

d) Faça *upload* do ficheiro para o *bucket* da Ásia usando a ferramenta *gsutil* (<https://cloud.google.com/storage/docs/gsutil/commands/cp>).

```
gsutil cp space-station.jpg gs://...
```

3) Estão disponíveis para instalação local em computadores pessoais as ferramentas de linha de comandos do Google Cloud SDK (*gcloud* e *gsutil*). Instale e configure o *Google Cloud SDK* no seu computador, seguindo as instruções indicadas em: <https://cloud.google.com/sdk/docs/install-sdk>. Um dos primeiros passos de configuração é dar consentimento para acesso aos recursos existentes na sua conta Google GCP e no projeto que já criou anteriormente. Este passo é feito através do protocolo de autorização OAuth 2.0. A autorização de acesso aos recursos é feita no *browser*, após se autenticar na conta Google usada na criação do projeto.

4) Execute o comando *perfdiag* do *gsutil* no seu computador, usando como teste o *bucket* na Europa e na Ásia. (<https://cloud.google.com/storage/docs/gsutil/commands/perfdiag>):

```
gsutil perfdiag <bucket>
```

Observe e analise os resultados do *throughput* de escrita e leitura. A opção *-o <file>* guarda os resultados no ficheiro indicado.

5) Considere o projeto *maven CNStorageBaseAlunos*, disponível em anexo ao enunciado, que possibilita algumas operações básicas de acesso ao serviço *Storage*: listagem de *buckets* de um projeto; criação e remoção de *buckets* e *upload/download* de *blobs*, etc.

a) Crie uma *service account* com permissões (*role*) de *Storage Admin* e execute o programa configurando a variável de ambiente para o ficheiro com a chave privada da *service account*:

```
GOOGLE_APPLICATION_CREDENTIALS=<pathname do ficheiro json com chave privada>
```

b) Acrescente outras operações para consolidar conceitos, especialmente a operação para colocar um *blob* com permissões de acesso público (consulte os slides).

- 6) Considere a existência de um serviço de troca de mensagens (*Forum*) cujo envio e receção de mensagens está organizado em tópicos. O contrato do serviço é apresentado em seguida e é fornecido em anexo no ficheiro `forum.proto`:

```
syntax = "proto3";
import "google/protobuf/empty.proto";
option java_multiple_files = true;
option java_package = "forum";
package forum;

service Forum {
    // subscribe a topic
    rpc topicSubscribe(SubscribeUnSubscribe) returns (stream ForumMessage);
    // unsubscribe a topic
    rpc topicUnSubscribe(SubscribeUnSubscribe) returns (google.protobuf.Empty);
    // get all topics in server
    rpc getAllTopics(google.protobuf.Empty) returns (ExistingTopics);
    // send a message to a topic
    rpc publishMessage(ForumMessage) returns (google.protobuf.Empty);
}

message SubscribeUnSubscribe {
    string usrName = 1;
    string topicName = 2;
}

message ExistingTopics {
    // list of topic names
    repeated string topicName = 1;
}

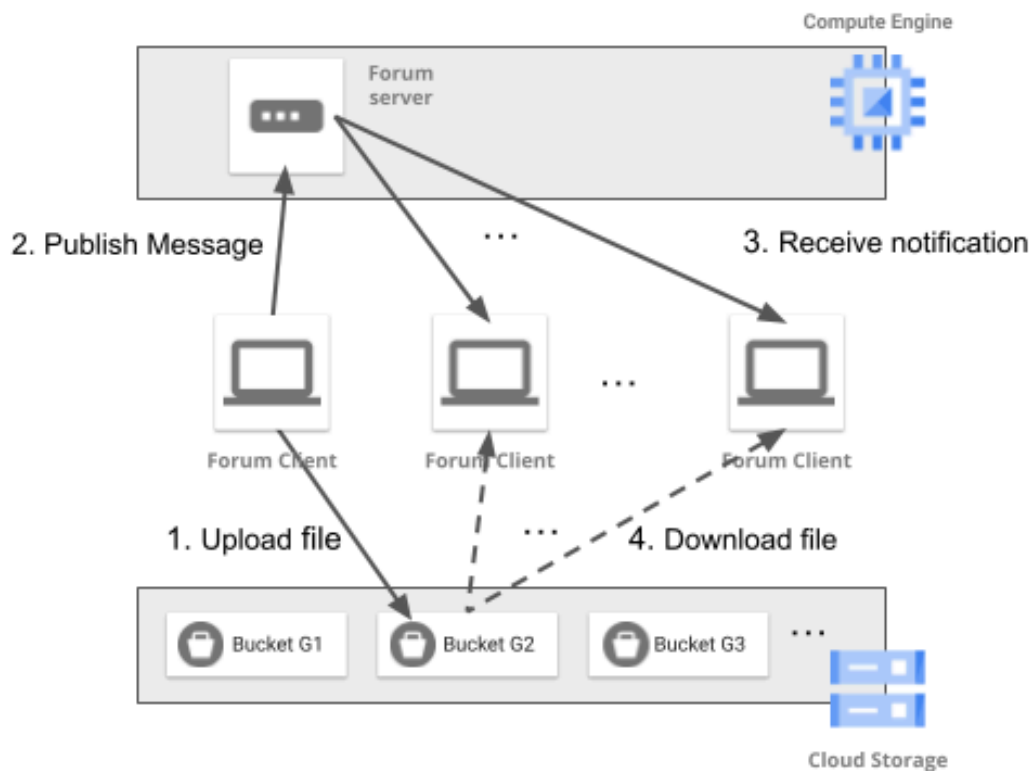
message ForumMessage {
    string fromUser = 1;
    string topicName = 2;
    string txtMsg = 3;
}
```

Pretende-se realizar um **cliente Java do serviço *Forum*** (já em execução numa VM GCP) que possibilite a partilha de ficheiros (*blobs*) alojados no serviço *Google Cloud Storage*.

As mensagens enviadas para o *Forum* podem representar apenas texto, ou um texto com o nome de um *bucket* e de um *blob* com acesso público, onde a aplicação cliente escreveu previamente o ficheiro que pretende partilhar. As mensagens que o cliente envia para um tópico têm o formato `<texto>;<bucketName>;<blobName>` onde:

- i) `<texto>` é obrigatório e representa a mensagem a enviar para o tópico
- ii) `;<bucketName>;<blobName>` é opcional e representa o nome do *bucket* onde foi escrito o *blob*

A aplicação cliente quando recebe uma mensagem de um tópico que subscreveu, analisa-a segundo este formato e, caso esteja presente o nome de um *bucket* e de um *blob*, faz *download* do mesmo para um ficheiro local, tal como apresentado na figura seguinte:



Tenha em conta que a aplicação cliente:

- Deve usar o serviço *Forum* disponível no endereço **35.209.47.109** porto **8000**;
- Para teste em máquina local é fornecido um JAR (*forum-server-Lab03.jar*) com um servidor já operacional que implementa o contrato e que pode executar na sua máquina local, em qualquer porto passado como argumento. O servidor fornecido escreve num ficheiro de *log* os eventos mais relevantes.