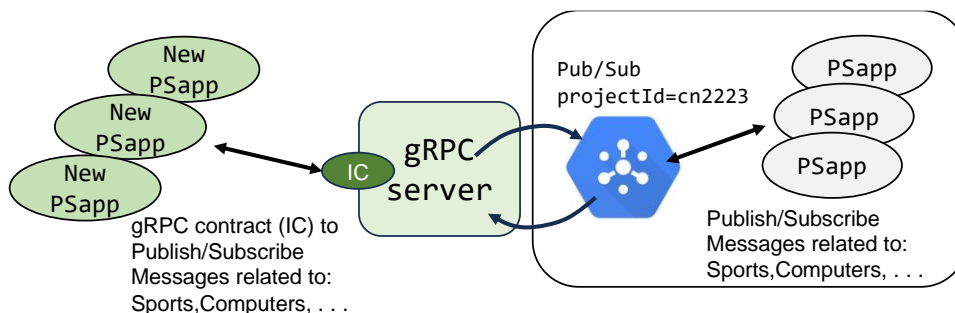


EXAME DE ÉPOCA NORMAL (2ª PARTE, COM CONSULTA, DURAÇÃO: 75 min., 9 valores)

- ♦ **Na questão 9, onde tem de apresentar código Java, não precisa de indicar o *pom.xml*, as declarações de *import* ou detalhes que não sejam essenciais (por exemplo *static main*). Utilize um misto de código Java e comentários para indicar eventuais pressupostos;**
- ♦ **A resposta a cada uma das questões (9, 10 e 11) tem de ser entregue em folhas separadas.**

9) Considere o cenário apresentado na figura, em que existe uma aplicação (PSapp) que utiliza o serviço GCP Pub/Sub para suportar um sistema de troca de mensagens entre múltiplos utilizadores usando o padrão *publish/subscribe*, em que existem múltiplos tópicos relacionados com temas genéricos do tipo Sports, Computers, etc. A aplicação *PSapp*, de acordo com escolha dos utilizadores, permite listar tópicos existentes, publicar novas mensagens num tópico e subscrever a receção de todas as novas mensagens que venham a ser publicadas nos tópicos subscritos.

Face ao problema de gerir as múltiplas contas de serviço com permissão de acesso ao serviço Pub/Sub do GCP (uma para cada utilizador) foi decidido criar um servidor gRPC, para alojar numa VM do serviço GCP com uma única conta de serviço, que sirva de intermediário para uma nova aplicação *New PSapp*. Assim, esta nova aplicação cliente não tem a noção da existência do serviço Pub/Sub do GCP.



Note que não tem de apresentar qualquer alteração à aplicação *PSapp*.

- a) **[1,5 val]** Proponha o contrato *protobuf*, a implementar pelo servidor *gRPC* e a ser usado pelo cliente (*New PSapp*) por forma a ser possível:
 - (1) Listar os tópicos existentes;
 - (2) Publicar uma mensagem (*userID, text*) num determinado tópico;
 - (3) Registar o interesse para receber todas as mensagens publicadas num tópico.
- b) **[2,0 val]** Escreva o código Java essencial da classe que implementa o servidor onde deve constar a assinatura dos métodos que implementam o contrato. Em particular, apresente o código referente à operação (3) usada pelos clientes para receberem todas as mensagens publicadas num tópico.
- c) **[1,5 val]** Escreva o código Java essencial da aplicação cliente (*New PSapp*) que subscreve a receção de mensagens do tópico “Sports” e de seguida publica uma mensagem nesse mesmo tópico.

(Questões 10 e 11 no verso)

♦ **As respostas às questões 10 e 11 têm de ser entregues em folhas separadas**

10) [2,0 val] [Não precisa apresentar código]

Considere um cenário em que se pretende desenvolver um sistema Cliente/Servidor, com distribuição de carga (*load balancing*) por múltiplos servidores. Um cliente pode fazer milhares de pedidos e podem existir milhares de clientes.

Considerando uma seleção entre os vários ambientes estudados (Java RMI, gRPC e serviços da Google Cloud Platform (GCP)), apresente e descreva uma proposta de diagrama de arquitetura.

11) [2,0 val] [Não precisa apresentar código]

Na empresa CN23 existem dezenas de funcionários que tratam em simultâneo as mensagens sobre reclamações dos clientes (cada mensagem é unicamente tratada por um funcionário diferente). O funcionário analisa a mensagem e de acordo com a sua interpretação reenvia a mesma para ser tratada nos departamentos de Vendas, Faturação ou Suporte, onde posteriormente outros funcionários tratam e dão solução às reclamações através de contacto direto com o cliente (fora do âmbito do sistema, por exemplo, via telefone). Para efeitos de registo e possíveis futuras auditorias, todas as mensagens tratadas, com indicação da solução do problema, terão de ser guardadas num único ficheiro de *Logging*.

Utilizando o serviço Pub/Sub do GCP apresente e descreva um diagrama de arquitetura do sistema para a empresa CN23, indicando aplicações, tópicos, subscrições, etc.