



Computação Distribuída

T2 – Point Of Sale

Semestre Inverno 2023/2024

Mestrado em Engenharia Informática e de Computadores

Trabalho Final

Estudantes: 47255 Gonçalo Silva A47255@alunos.isel.pt
47196 João Rocha A47280@alunos.isel.pt

Engenheiro: Luís Assunção

20 de março de 2023

Índice

1	Introdução Arquitetura Geral	4
2	Grpc Server.....	5
3	Rabbit	6
3.1	Point-Of-Sale (Producer) -> SpreadGroup (Consumer)	6
3.2	SpreadGroup (Producer) -> ManageServer (Consumer).....	7
4	Workers	8
4.1	Criação e Lógica.....	8
4.2	Eleição do Leader	8

Lista de Figuras

<i>Figura 1 – Arquitetura Geral.....</i>	<i>4</i>
<i>Figura 2 - Contrato entre UserApp e ManageServer.....</i>	<i>5</i>
<i>Figura 3 - PointOfSale -> Worker.....</i>	<i>6</i>

Capítulo I

1 Introdução Arquitetura Geral

O presente relatório descreve o desenvolvimento de um sistema distribuído para a gestão de vendas em cadeias de supermercados. O sistema é composto por diversas componentes, incluindo uma aplicação de ponto de venda (PointOfSale), um Broker Publish/Subscribe, instâncias de trabalhadores (Workers) que processam mensagens de vendas, um sistema de armazenamento distribuído Gluster, um servidor Manager gRPC e uma aplicação de utilizador (UserApp). O sistema tem como objetivo simular uma caixa de supermercado, gerando mensagens de vendas e processando essas vendas de forma distribuída.

A arquitetura geral do projeto segue o seguinte esquema.

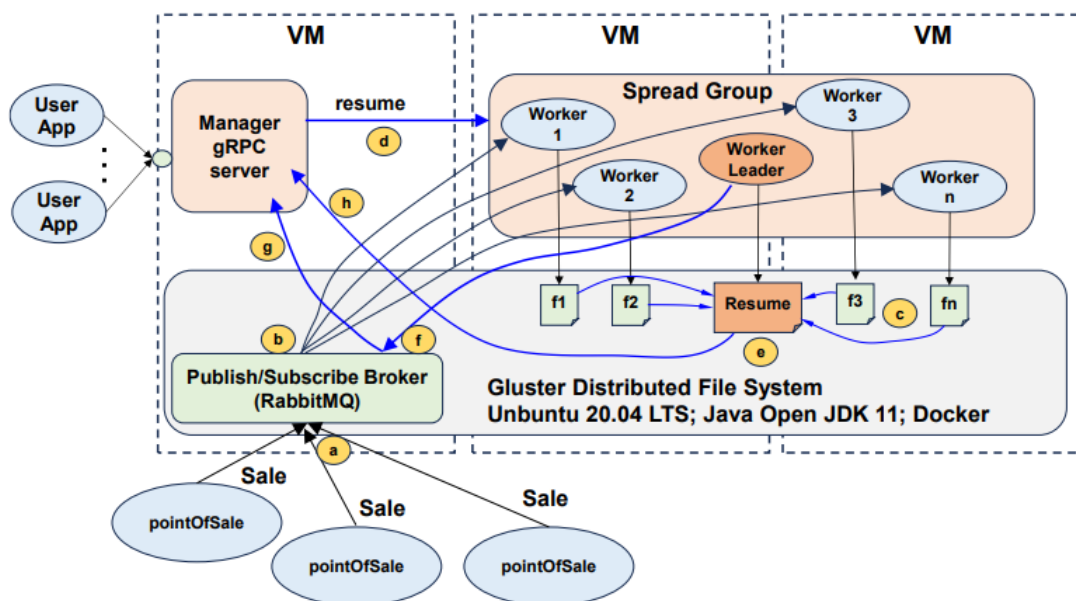


Figura 1 – Arquitetura Geral

Capítulo II

2 Grpc Server

O servidor Grpc deste projeto é responsável por responder aos pedidos realizados pela aplicação UserApp, nomeadamente a funcionalidade de obter o resumo das “Sales” do tipo “ALIMENTAR” ou “CASA”.

O fluxo de processamento deste pedido envolve passagens por todos os componentes do projeto, nomeadamente de acordo com a seguinte estrutura:

- Utilizador realiza um pedido de obtenção de um ficheiro de Resume;
- ManageServer realiza o recolhimento dos parâmetros necessários e envia uma mensagem em multicast para todos os workers membros do Spread Group
- Os Workers param o seu processamento normal e realizam a eleição de um leader que irá ser responsável pelo processamento e criação do ficheiro de resumo
- Quando o ficheiro estiver concluído é produzida uma mensagem de notificação que é publicada na respetiva fila
- O servidor consome essa mensagem, e utiliza o “fileID” presente na notificação para ir ao sistema gluster e obter o ficheiro correspondente.
-

Atentando a este processamento o Servidor Grpc deve apenas disponibilizar 2 operações, a operação de “resumeFile” e “downloadFile”, de acordo com o seguinte contrato. Quando o servidor recebe uma operação do tipo downloadFile este obtém o ficheiro correspondente da diretoria do gluster e envia-o em chunks de bytes para a diretoria previamente escolhida do lado do utilizador.

```
service UMService {  
    rpc resumeSales(ResumeInfo) returns (EmptyResponse);  
    rpc downloadFile(FileIdentifier) returns (stream FileResponse);  
}
```

Figura 2 - Contrato entre UserApp e ManageServer

Capítulo III

3 Rabbit

Para estabelecer a conexão com o daemon do rabbit é necessário conhecer o seu ip e porto e criar uma palavra passe e utilizador para possibilitar o acesso. Para possibilitar a correta conexão entre Produtor e consumidor, ambos os extremos tem de efetuar a ligação ao daemon a também tem de definir as filas de mensagem. Desta forma é necessário definir no

- No ManageServer a fila de mensagens de notificação;
- Na aplicação pointOfSale as duas filas de mensagens de "Sale"
- Na instância de Worker a fila de notificação e as duas filas de mensagens de "Sale";

3.1 Point-Of-Sale (Producer) -> SpreadGroup (Consumer)

Na aplicação pointOfSale quando o utilizador escolhe a opção de publicar uma venda, a aplicação constrói uma mensagem de venda (Sale) e a pública no Exchange ExgSales com uma routing key associada à categoria do produto (ALIMENTAR ou CASA).

O exchange utilizado é do tipo direto, ou seja, encaminha as mensagens para message queues cuja binding key corresponde à routing key existente na mensagem.

Desta forma a aplicação pointOfSale comporta-se como um produtor que publica mensagens "Sales" em filas, por outro lado as instâncias de Worker irão consumir essas mensagens, como evidenciado na figura 3.

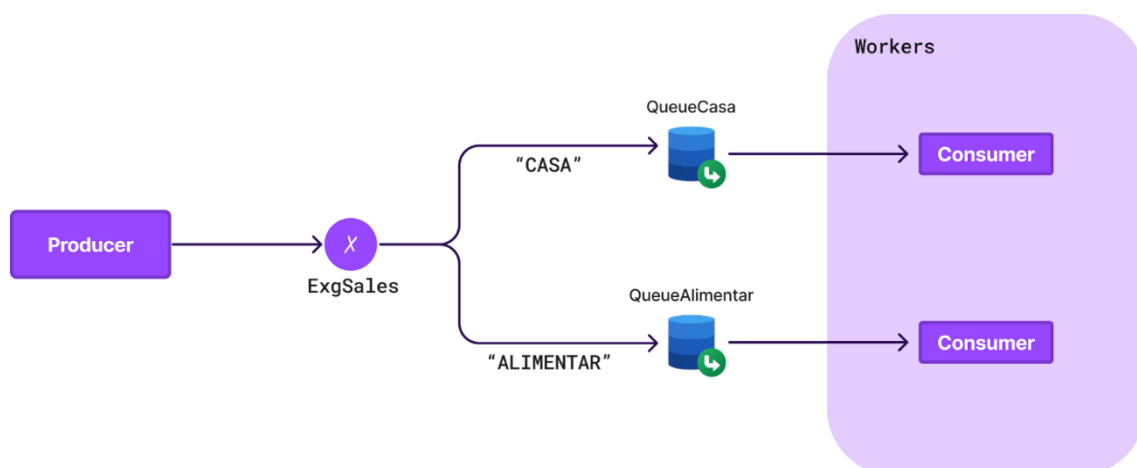


Figura 3 - PointOfSale -> Worker

3.2 SpreadGroup (Producer) -> ManageServer (Consumer)

Os Workers para além de consumirem mensagens de “Sale” como previamente ilustrado estes simultaneamente atuam como produtores de mensagens de notificação. Quando um ficheiro de resumo está pronto deve ser enviada uma mensagem de notificação a avisar o servidor que o ficheiro pode ser transferido.

O exchange utilizado também é do tipo direto, ou seja, encaminha as mensagens para message queues cuja binding key corresponde à routing key existente na mensagem.

Capítulo V

4 Workers

4.1 Criação e Lógica

Na iniciação/criação de um Worker este é logo adicionado ao spread group “SalesWorkers”, indicando com o ip e porta do daemon do Spread, aquando a criação do Worker também é imediatamente associado um listener responsável por capturar as mensagens recebidas pelos Workers. Notando que um Worker é do tipo “Alimentar” ou “Casa”

Após a adição do Worker ao spreadGroup é associada uma classe de callBack às filas do rabbit a qual vai ser chamada quando existirem mensagens prontas a serem consumidas na fila. Este callBack é responsável por processar uma mensagem da fila e escrever o seu conteúdo num ficheiro.

4.2 Eleição do Leader

Quando existe o processamento de um pedido de resume no manager server é enviada uma mensagem em multicast para todos os Workers de modo a que estes possam parar o seu processamento normal e que procedam á escolha do leader. Desta forma criamos um algoritmo que elege o leader tendo em conta o seu identificador, tornando o algoritmo mais simples e tomando proveito das operações já existentes no Spread.

Ó algoritmo funciona da seguinte forma:

Quando um Worker recebe uma mensagem inicialmente vai obter a lista de todos os membros deste grupo, posteriormente é obtido o elemento cuja identificação tem o maior valor. Sendo que o Worker associado a essa identificação é que deve ser nomeado leader.

Mas surge o problema, como saber que a que Worker pertence aquela identificação? Para isso é enviada uma mensagem em multicast para todos os workers passando como informação a identificação de maior valor, quando os Workers receberem essa mensagem vao comparar o valor recebido com a sua própria identificação para averiguarem se são o leader, em caso afirmativo é colocada a flag associada a cada Worker a true, e iniciado o processamento do ficheiro de resumo de vendas.