

## Guião de Demonstração – Grupo A63

### Instruções de instalação e configuração:

- Abrir um terminal na pasta onde irão ficar os ficheiros do projeto.
  - Executar nesse terminal o comando “git clone <https://github.com/tecnico-distsys/A63-Komparator.git>”
  - Executar nesse terminal o comando “mvn clean install -DskipTests”
- 

## 2ª Entrega

### Instruções para correr testes da 2ª Entrega:

- Seguir Instruções de instalação e configuração em cima descritas.
  - Abrir um terminal na pasta “supplier-ws” e executar comando “mvn compile exec:java”
  - Abrir outro terminal na pasta “supplier-ws” e executar comando “mvn exec:java -Dws.i=2”
  - Abrir outro terminal na pasta “supplier-ws” e executar comando “mvn exec:java -Dws.i=3”
  - Abrir um terminal na pasta “mediator-ws” e executar comando “mvn compile exec:java -Dws.i=2”
  - Abrir outro terminal na pasta “mediator-ws” e executar comando “mvn exec:java”
  - Abrir um terminal na pasta “mediator-ws-cli” e executar comando “mvn verify”
- 

## Replicação

### (R1) Instruções para testar funcionamento normal:

- Seguir Instruções de instalação e configuração em cima descritas.
- Abrir um terminal na pasta “supplier-ws” e executar comando “mvn compile exec:java”
- Abrir outro terminal na pasta “supplier-ws” e executar comando “mvn exec:java -Dws.i=2”
- Abrir outro terminal na pasta “supplier-ws” e executar comando “mvn exec:java -Dws.i=3”
- Abrir um terminal na pasta “mediator-ws” e executar comando “mvn compile exec:java -Dws.i=2”
- Abrir outro terminal na pasta “mediator-ws” e executar comando “mvn exec:java”
- É possível verificar que periodicamente (5 em 5 segundos) é impressa uma mensagem SOAP do método imAlive (implementado para esta entrega) no terminal do servidor secundário. Isto significa que o servidor primário está a dar uma prova de vida ao secundário dentro do tempo esperado.
- Abrir um terminal na pasta “mediator-ws-cli” e executar comando “mvn verify” (Se preferível executar comando “mvn verify -Dit.test=BuyCartIT#buyCartCompleteTest” para executar somente um teste)
- É possível verificar que são imprimidas mensagens SOAP no terminal secundário do método updateShopHistory/updateCart. Isto significa que o servidor primário está a propagar as alterações do seu estado ao servidor secundário, de forma a que este esteja sempre atualizado.
- Está assim assegurado o funcionamento normal do projeto.

## (R2) Instruções para testar tolerância a faltas:

- Seguir Instruções de instalação e configuração em cima descritas.
- Abrir um terminal na pasta “supplier-ws” e executar comando “mvn compile exec:java”
- Abrir outro terminal na pasta “supplier-ws” e executar comando “mvn exec:java –Dws.i=2”
- Abrir outro terminal na pasta “supplier-ws” e executar comando “mvn exec:java –Dws.i=3”
- Abrir um terminal na pasta “mediator-ws” e executar comando “mvn compile exec:java –Dws.i=2”
- Abrir outro terminal na pasta “mediator-ws” e executar comando “mvn exec:java”
- Abrir um terminal na pasta “mediator-ws-cli” e executar comando “mvn verify” após ser lida a mensagem “Publishing ‘A63\_Mediator’ (...)” que indica que o servidor principal está registado no UDDI.
- Causar a paragem súbita do servidor primário através de sigkill (CTRL+C) durante a execução dos testes.
- É possível verificar que após a paragem do servidor primário, o servidor secundário torna-se primário e regista-se no UDDI. Depois disto, verificamos que os testes recomeçam e acabam com sucesso.
- Está assim assegurada a tolerância a faltas.

---

## Timeouts

### Instruções para testar funcionamento dos timeouts:

- No ficheiro “MediatorClient.java” do módulo “mediator-ws-cli” configurar os timeouts do cliente com os valores desejados: timeout de conexão com servidor na linha 122; timeout de receção de resposta na linha 132. (isto é opcional visto que os timeouts já vêm com um valor por defeito realista).
- No ficheiro “MediatorPortImpl.java” do módulo “mediator-ws” descomentar as linhas 351 a 357 (para que o servidor faça *sleep*) e configurar a duração do *sleep* na linha 353 (isto é opcional visto que já existe um valor por defeito realista).
- Seguir Instruções de instalação e configuração em cima descritas.
- Abrir um terminal na pasta “supplier-ws” e executar comando “mvn compile exec:java”
- Abrir outro terminal na pasta “supplier-ws” e executar comando “mvn exec:java –Dws.i=2”
- Abrir outro terminal na pasta “supplier-ws” e executar comando “mvn exec:java –Dws.i=3”
- Abrir um terminal na pasta “mediator-ws” e executar comando “mvn compile exec:java –Dws.i=2”
- Abrir outro terminal na pasta “mediator-ws” e executar comando “mvn exec:java”
- Abrir um terminal na pasta “mediator-ws-cli” e executar comando “mvn verify” (Se preferível executar comando “mvn verify –Dit.test=BuyCartIT#buyCartCompleteTest” para executar somente um teste)
- É possível verificar que o servidor primário “adormece” periodicamente, ao ser imprimido no terminal “YOU FELL ASLEEP!”. Enquanto estiver “adormecido”, verifica-se que o servidor recebe mensagens SOAP com o mesmo identificador (reenviadas pelo cliente por se ter esgotado o tempo para receber resposta do servidor), e que estas são rejeitadas, lendo-se “MENSAGEM REJEITADA” no terminal do servidor (visto que não são processados pedidos repetidos).
- Está assim assegurado o front-end com a semântica pretendida no enunciado