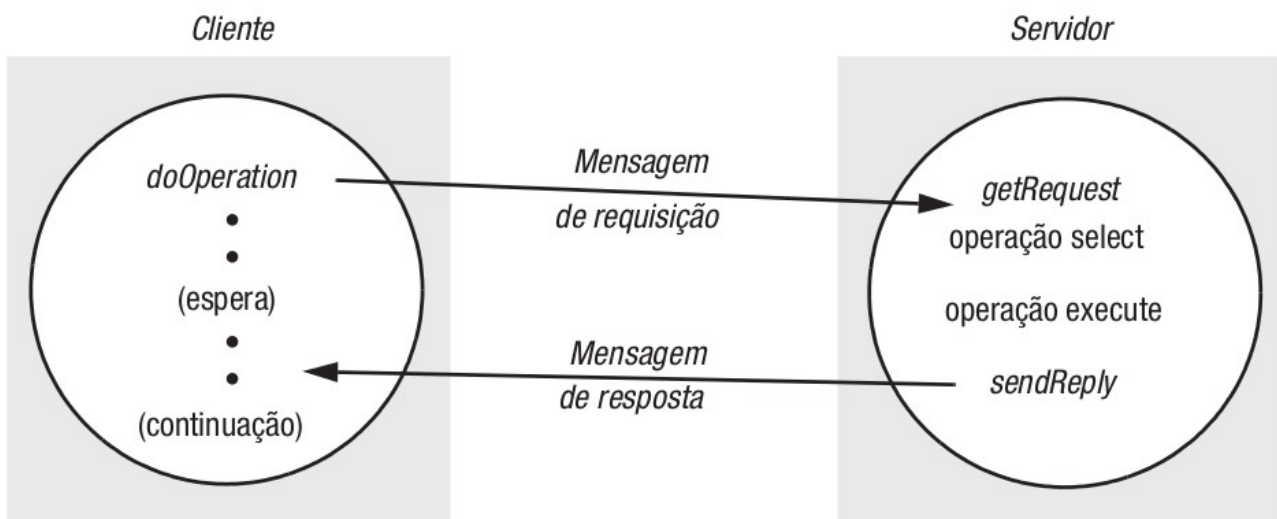




- Informações importantes:
 - equipe: máximo de 2 alunos
 - **cronograma:**
 - 22/10/19: Definição da equipe e do serviço remoto;
 - 30/10/19: Definição dos argumentos e troca de mensagens entre cliente e servidor;
 - 06/11/19: Troca de mensagens empacotadas;
 - 20/11/19: Entrega da versão final.

Cada equipe deve entregar o código fonte e um relatório descrevendo o serviço remoto implementado. A entrega deve ser feita via upload no SIPPA. Não há necessidade de prover Interface Gráfica com o Usuário (GUI). Modo texto é suficiente.

O serviço remoto deve ser implementado através da comunicação cliente-servidor e organizado num protocolo de requisição-resposta como descrito nas seções 5.1 e 5.2 do livro texto e ilustrado na figura abaixo. A comunicação entre cliente e servidor deve ser implementada via sockets TCP que trocam fluxos de bytes (mensagens empacotadas).



O uso dos métodos sugeridos pelo autor, que implementam o protocolo requisição-resposta, deve ser seguido, mas alterações em suas assinaturas são permitidas:

- **`public byte[] doOperation (RemoteObjectRef o, int methodId, byte[] arguments):`**
envia uma mensagem de requisição para o objeto remoto e retorna a resposta. Os argumentos especificam o objeto remoto, o método a ser chamado e os argumentos para aquele método.
- **`public byte[] getRequest ():`** obtém uma requisição de um cliente através de uma porta servidora.
- **`public void sendReply (byte[] reply, InetAddress clientHost, int clientPort):`**
envia a mensagem de resposta para o cliente, endereçando-a a seu endereço IP e porta.

messageType	int (0=Request, 1= Reply)
requestId	int
objectReference	RemoteObjectRef
methodId	Int
arguments	array of bytes

Os elementos *objectReference* e *methodID* podem ser Strings que representam, respectivamente, o nome do objeto que fornece o serviço e o nome do método a ser invocado.

Regras para a implementação:

- a equipe deve desenvolver o cliente e o serviço (servidor) em linguagens diferentes. Foi sugerido em sala **Python** e **Java**.
- para a representação externa dos dados as equipes deverão utilizar o **JSON** (JavaScript Object Notation);

Material de apoio:

- Sistemas Distribuídos: conceitos e projetos. Capítulo 5;
- <https://github.com/google/gson> (Json Java)
- https://www.w3schools.com/python/python_json.asp (Json Python)

Avaliação:

- seguir o cronograma: 1 ponto;
- implementação: 6 pontos;
- apresentação: 3 pontos.

Entrega (via sippa) e apresentação: 20/11/2019