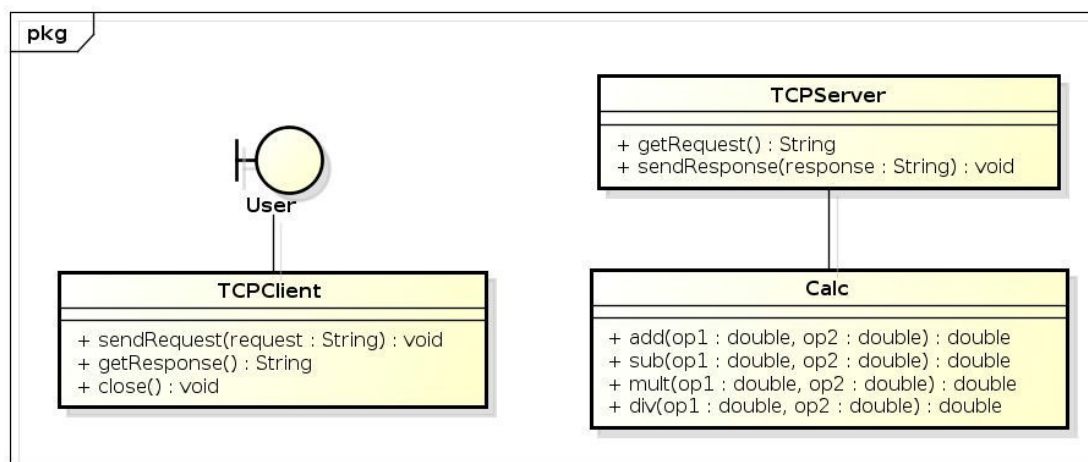




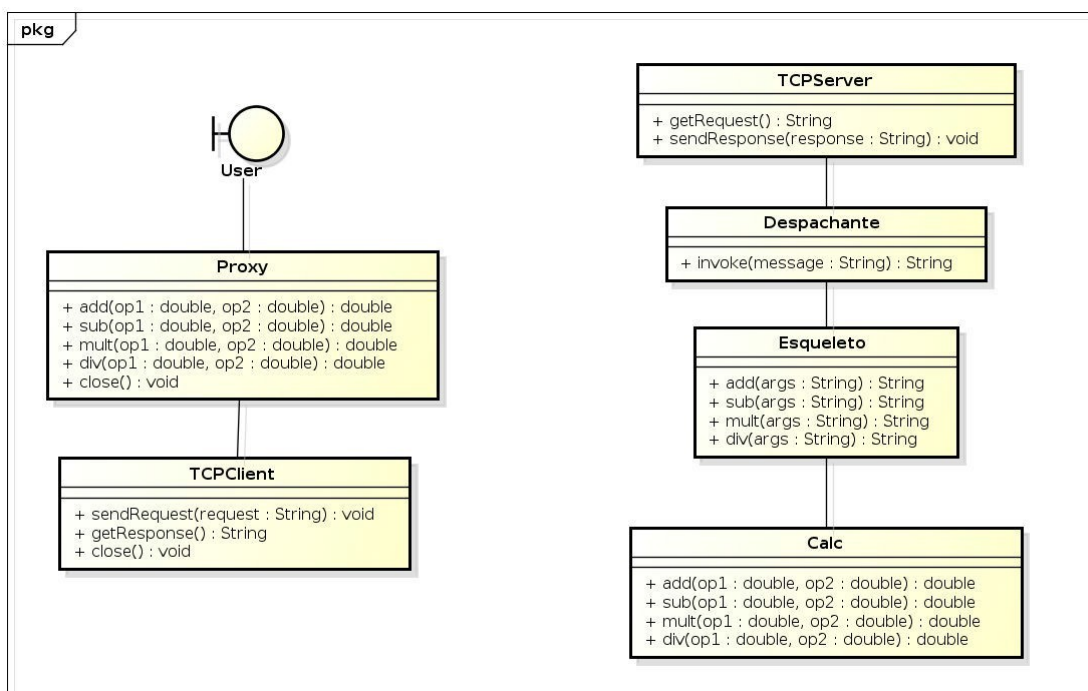
**Universidade Federal do Ceará (UFC)**  
**Disciplina: Sistemas Distribuídos**  
**Trabalho 2**  
**Entrega: 05/10/2018**

**Questão 1** - Continuando a evolução da arquitetura Cliente-Servidor vista no trabalho 1, devemos retirar a interação com o usuário da classe TCPClient e o serviço da classe TCPServer. Dessa forma, os objetos de TCPClient e TCPServer tornam-se coesos com o único objetivo de prover comunicação (estabelecimento de conexão e trocas de mensagens através dos fluxos de entrada (IN) e saída (OUT)).



powered by Astah

**Questão 2** - Tornar a invocação remota a método transparente para o usuário.



powered by Astah

User:

- Objeto de fronteira que implementa a interação com o usuário. Permite que o usuário escolha uma operação, atribua os parâmetros necessários e receba a resposta.

Proxy:

- Tornar a invocação remota a método transparente para o usuário. O objeto de fronteira (User) “acha” que o Proxy é o objeto que realmente fornece o serviço.
- Empacota a requisição remota (operação + parâmetros) em uma mensagem e solicita o envio via `sendRequest(String request)` de `TCPClient`.
- Recebe a resposta via `getResponse` de `TCPClient`, desempacota o resultado e o retorna para User.

Despachante:

- Recebe a requisição, desempacota a operação e invoca o método correspondente do esqueleto.
- Recebe o retorno do esqueleto e o retorna para o `ServidorTCP` para ser enviado ao cliente.

Esqueleto:

- Semelhante ao Proxy para o lado Servidor.
- Desempacota os parâmetros, faz as conversões necessárias e chama o método apropriado do serviço (`Calc`).
- Recebe a resposta e a empacota para retorná-la ao despachante.

Calc:

- Realmente implementa as funções da calculadora.