

Unified Modeling Language (UML)

Universidade Federal do Maranhão – UFMA
Pós Graduação de Engenharia de Eletricidade
Grupo de Computação
Assunto: Diagrama de Seqüência

Autoria: Aristófanês Corrêa Silva
Adaptação: Alexandre César M de Oliveira

7 Diagrama de Seqüência

7.1 Tipos de diagramas de interação

- Termo genérico que se aplica a vários tipos de diagramas que enfatizam interações de objetos.
- Uma interação é uma especificação comportamental que inclui uma seqüência de trocas de mensagem entre um conjunto de objetos dentro de um contexto para realizar um propósito específico.
- Deve ser utilizado quando se deseja visualizar o comportamento de vários objetos dentro de um único caso de uso, a partir das mensagens que são passadas entre eles.

👉 Diagrama de Seqüência

👉 Diagrama de Colaboração (Comunicação em UML 2.0)

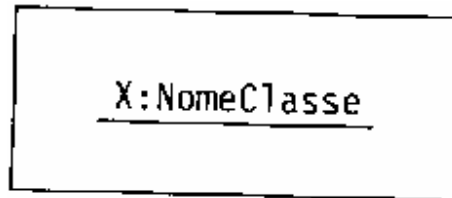
👉 Diagrama de Interação

7.2 Diagrama de Seqüência

- Um diagrama de seqüência mostra a colaboração dinâmica entre os vários objetos de um sistema e que dá ênfase à ordenação temporal das mensagens trocadas entre objetos do sistema.
- Mostra a interação entre os objetos, alguma coisa que acontecerá em um ponto específico da execução do sistema.
- Pode ser usado para mostrar a evolução de uma dada situação em determinado momento do software, mostrar uma dada colaboração entre duas ou mais classes e pode, também, ser usado para mostrar a tradução de um Caso de Uso desde sua interação com o usuário até a finalização daquele dado processo.
- Pode mostrar erros não detectados no diagrama de classes. Ele melhora o diagrama de classes, permitindo que acrescentemos ou retiremos métodos e/ou atributos desnecessários de um conjunto de classes.
- O mais importante aspecto deste diagrama é que a partir dele percebe-se a seqüência de mensagens enviadas entre os objetos.
- A intenção é dar uma demonstração visual da seqüência das ações que o software tomará frente a uma interação com um ator. Serve como norte para o programador e dá-nos maior segurança de que o diagrama de classes atende às necessidades do negócio.

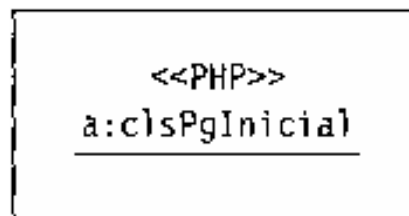
7.3 Notação

- Objeto

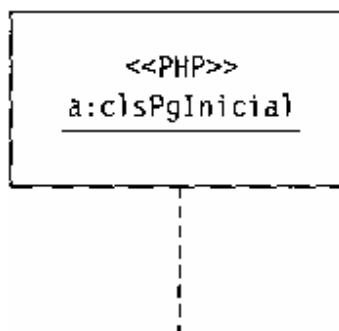


- Estereótipos

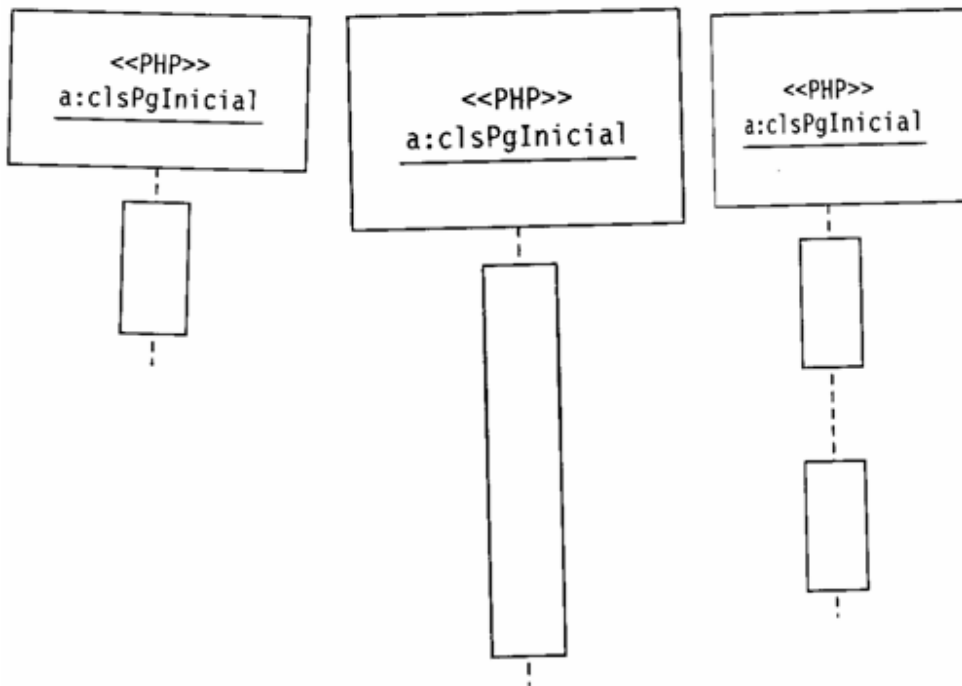
- bastante utilizados, pois precisamos representar interações com usuário, gravação e recuperação de informações em banco de dados, textos ou XML
- A letra “a” antes do dois pontos será o nome da página PHP em questão.



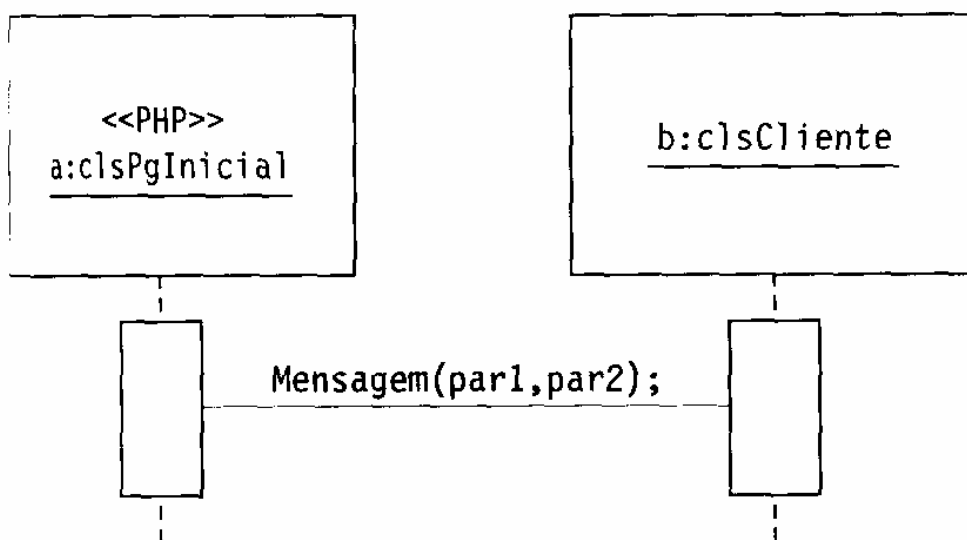
- Uma linha tracejada significa a linha de vida da solução em questão, proporcional ao tamanho que a solução exigir.



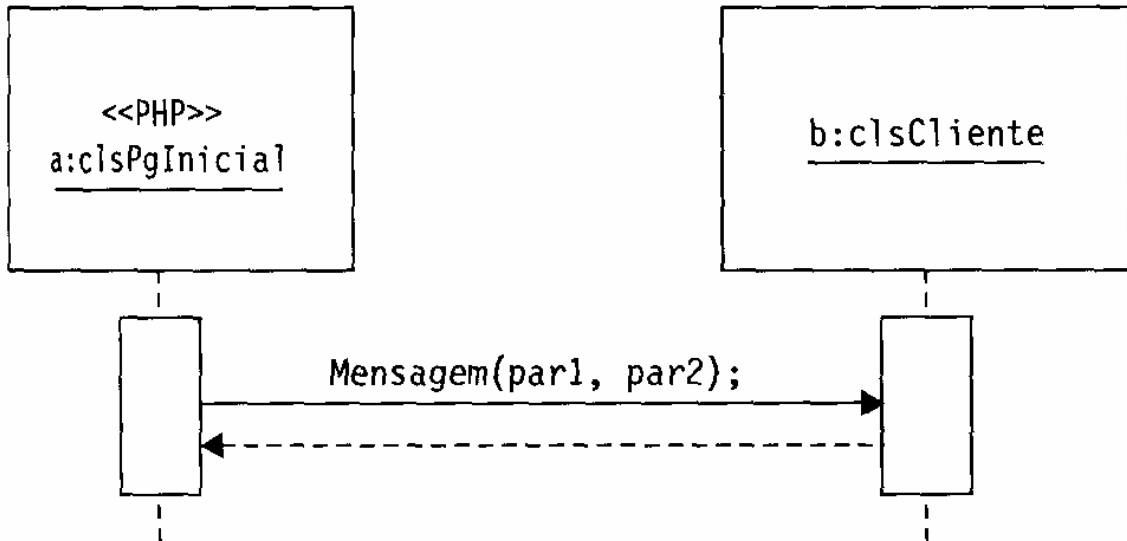
- Um retângulo que ocupa qualquer área da linha tracejada é o tempo de vida do objeto.



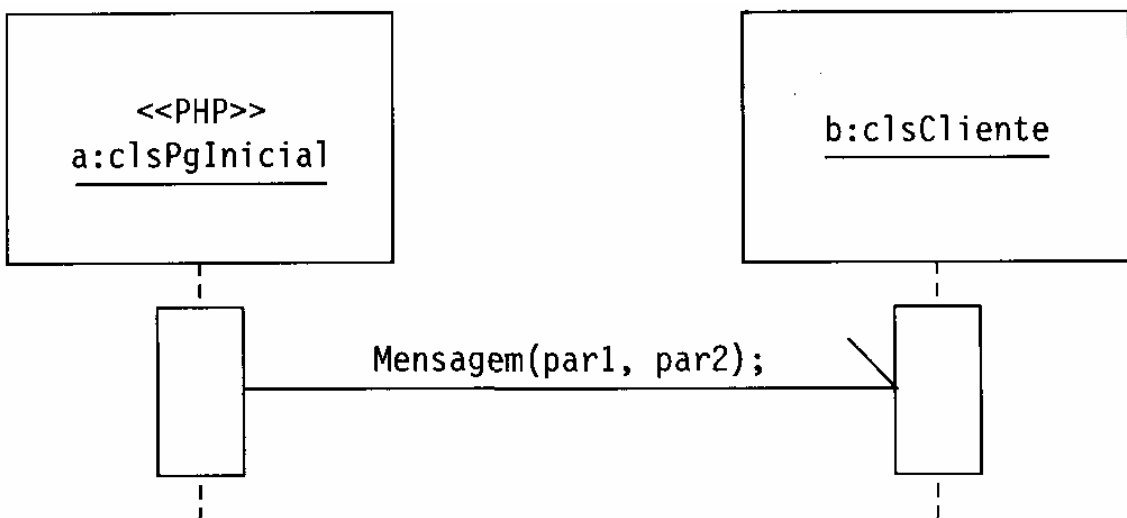
- Um objeto enviando uma mensagem para outro objeto. Quando um objeto envia uma mensagem está executando um método no outro objeto. É dessa forma que as classes colaboram



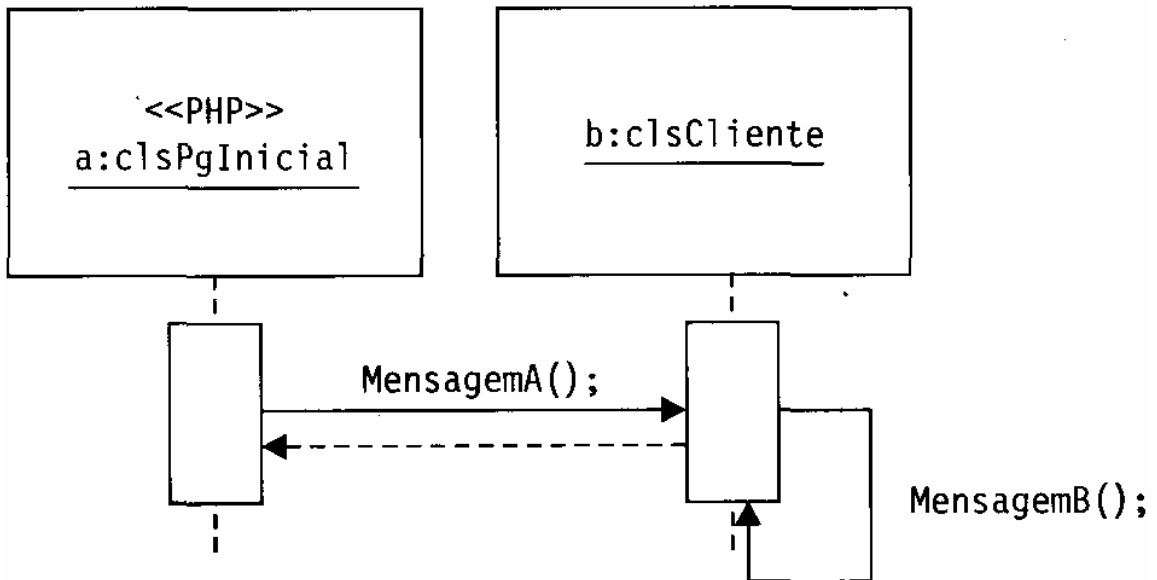
- A seta tracejada representa o retorno que a MensagemA pode dar. Quando métodos não tem retorno simplesmente não represente esta seta



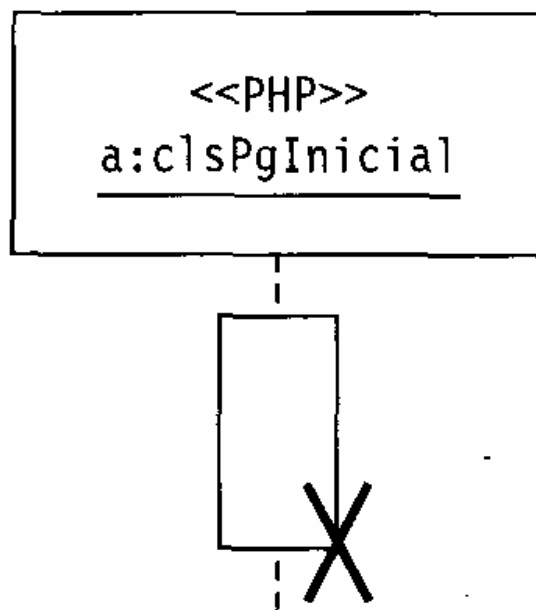
- A representação de mensagem assíncronas, aquelas que são enviadas e, das quais, não se espera retorno. Elas ficam resolvendo seus assuntos até o término. Ex.: impressão de um relatório.



- Método recursivo é aquela mensagem que é enviada ao próprio objeto. Ex.: uma classe de senha e ela própria cuida de criptografar essa senha.

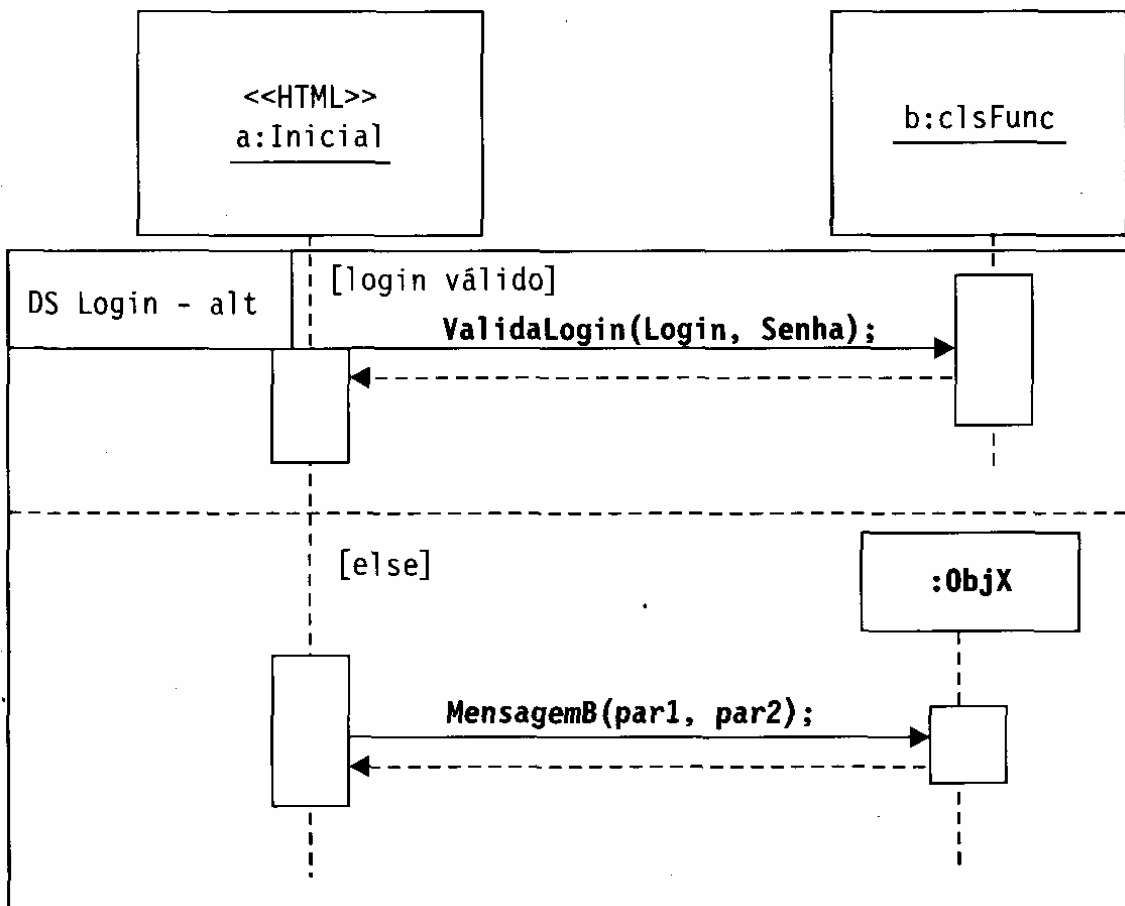


- Um “X”, no final de uma ocorrência de execução, indica que o objeto foi destruído.



- Operador de interação.

- O operador deve estar dentro de um retângulo maior, chamado de Fragmento Combinado, dividido por uma ou mais linhas pontilhada.



- Formas de representar alternativas (**Alt** ou **alt**), else, switch, cases entre outras. Uma ou outra opção é escolhida
- Utilizando **opt** temos uma opção. Pelo menos um é escolhido ou nada deverá acontecer.
- Quando o operador de interação for o **break**, isso indica que o fragmento combinado mostra um cenário que deve acontecer

em vez de simplesmente terminarmos, é um ponto de parada obrigatória.

- Quando o operador de interação for **Paralelo** ou **par**, designa que o fragmento combinado executa um intercalação entre os operandos ou cenários desenhados no fragmento combinado
- Quando o operador de interação for **Região Crítica**, designa que o fragmento combinado não pode ser intercalado por outro e seu tratamento é especial
- Quando o operador de interação for **loop**, significa que aquele fragmento passa por um loop na linguagem. O loop tem uma sintaxe própria, assim como a condição,

*loop [(**<minimo>**, [**<maximo>**])]*

- Quando o operador de interação for **ignorar**, significa que existem alguns tipos de mensagem que não são mostrados naquele fragmento combinado. Por exemplo, mensagens que são intuitivamente construídas, como a exibição de uma tela que informa: por favor, se seu login ou sua senha apresentam problemas, você tem mais duas chances. Inversamente **considerar** indica que existem mensagens que devem ser consideradas naquele fragmento.
- Quando o operador de interação for **asserção**, indica que a seqüência de ações, dentro do fragmento combinado, é somente válida se executado continuamente.

- Existe, ainda, os gates (portões), cuja única finalidade é promover uma forma de interação entre fragmentos combinados alcançando outra mensagem em outro fragmento combinado. Para representar um gate, basta enviar uma mensagem, uma seta, para a borda do fragmento combinado.
- Exemplo:

