

Professor:	Esp. Marcos Morais de Sousa
Curso:	Sistemas de informação
Disciplina:	Análise e Modelagem de Sistemas II
Aula 01	Dinâmica da disciplina, plano de curso e avaliação
Turma:	04º & 05º semestre

# Apresentação

---

Professor:	<b>Marcos Morais de Sousa</b>
Contato:	<b>marcosmoraisdesousa@gmail.com</b>


---

Resumo profissional:	Desde 1995 atua profissionalmente na área de infraestrutura, desenvolvimento, automação comercial, consultoria em gestão da informação e reengenharia de processos; conhecimentos das linguagens CLIPPER, DELPHI, JAVA e C# bem como na área de banco de dados, redes e segurança da informação; Experiência na área de coordenação estratégica e gerencial de equipes; atua em mais de 10 empresas na região de Jequié, Ipiaú e Itabuna.
----------------------	---

---

Formação acadêmica e titulação:	Especialista em Gestão da Tecnologia e Informação. Bacharel em Sistemas de Informação.
---------------------------------	---

---

		<b>PLANO DE ENSINO</b>			
		<b>CURSO DE SISTEMAS DE INFORMAÇÃO</b>			
<b>Disciplina</b>	<b>Análise e Modelagem de Sistemas II</b>			<b>Código</b>	<b>SIS416</b>
<b>Docente</b>	Marcos Morais de Sousa	<b>Semestre</b>	5º	<b>Carga horária</b>	60
<b>1 EMENTA</b>					
Metodologias de desenvolvimento de software. Implementação, revisão e reutilização de software. Padrões de Projeto. Frameworks e componentes.					
<b>2 OBJETIVOS DO COMPONENTE CURRICULAR</b>					
<b>OBJETIVO GERAL</b> <ul style="list-style-type: none"> <li>Compreender os conceitos da Análise e Projeto Orientado a Objetos;</li> </ul> <b>OBJETIVOS ESPECIFICOS</b> <ul style="list-style-type: none"> <li>Analisar problemas do mundo real e produzir uma abstração orientada a objetos utilizando UML;</li> <li>Projetar soluções do mundo real e produzir uma abstração orientada a objetos usando UML.</li> </ul>					

3 CONTEUDOS PROGRAMATICOS / CRONOGRAMA DAS AULAS			
SEMANA	Dias da Aula (nº aula)	ASSUNTO	METODOS E TECNICAS DE APRENDIZAGEM (indicar as estratégias didáticas que serão utilizadas)
1.	06/02 – 07/02 (1 – 3)	Apresentações: Dinâmica da disciplina, plano de curso e avaliação.	Aula Expositiva
2.	13/02 – 14/02 (4 – 6)	Diagrama de Sequência: aplicação, componentes, regras de formação. Aplicação no projeto em desenvolvimento.	Aula Expositiva
3.	20/02 – 21/02 (7 – 9)	Diagrama de Colaboração: aplicação, componentes, regras de formação. Aplicação no projeto em desenvolvimento.	Aula Expositiva
4.	27/02 – 28/02 (10 – 12)	Diagrama de Estados: aplicação, componentes, regras de formação. Aplicação no projeto em desenvolvimento.	Aula Expositiva
5.	05/03 – 06/03 (13 – 15)	Diagrama de Componentes: aplicação, componentes, regras de formação. Aplicação no projeto em desenvolvimento.	Aula Expositiva
6.	12/03 – 13/03 (16 – 18)	Diagrama de Implantação: aplicação, componentes, regras de formação. Aplicação no projeto em desenvolvimento.	Aula Expositiva
7.	19/03 – 20/03 (19 – 21)	Apresentação do andamento da documentação.	Aula Expositiva
8.	27/03 – 27/03 (22 – 24)	Processo Simplificado de Desenvolvimento de Software.	
9.	02/04 – 03/04 (25 – 28)	Análise de Requisitos.	Aula Expositiva

10.	09/04 – 10/04 (29 – 31)	Revisão do Conteúdo <b>Avaliação da I Unidade</b> (conteúdo lecionado da 2ª a 9ª Semana de aula)	
11.	16/04 – 17/04 (32 – 35)	Requisitos Funcionais e não Funcionais. Outras classificações.	Aula Expositiva
12.	23/04 – 24/04 (36 – 38)	Modelagem do comportamento do Sistema Projeto Orientado a Objeto	Aula Expositiva
13.	06/02 – 07/02 (39 – 41)	Fluxo de trabalho. Padrões para a atribuição de responsabilidade	Aula Expositiva
14.	30/04 – 01/05 (42 – 44)	Modelos conceituais.	Aula Expositiva
15.	07/05 – 08/05 (45 – 47)	Modelo estático do sistema.	Aula Expositiva
16.	14/05 – 15/05 (48 – 50)	Modelo dinâmico do sistema.	Aula Expositiva
17.	21/05 – 22/05 (51 – 53)	Modelagem do comportamento do Sistema.	Aula Expositiva
18.	28/05 – 29/05 (54 – 56)	Revisão do Conteúdo <b>Avaliação da I Unidade</b> (conteúdo lecionado da 9ª. A 17. Semana de aula)	
19.	04/06 – 05/06 (57 – 59)	Apresentação dos trabalhos em Grupo. Devolutiva da avaliação.	Aula Expositiva
20.	11/06 – 12/06 (60)	Apresentação dos trabalhos em Grupo.	Aula Expositiva

OBS: 1) Este cronograma poderá ser alterado durante o período letivo, desde que não cause prejuízo das atividades pedagógicas e dos conteúdos da disciplina.

2) Os registros acima correspondem a 60 horas/aula de 50 minutos.



4 CRONOGRAMA DAS ATIVIDADES DISCENTES EXTRACLASSE (Relacionar as Atividades Discentes Extraclasse previstas)			
UNIDADE	ATIVIDADES	CH	CONTEUDO
1ª	Apresentação do trabalho	06	Análise de Requisitos.
2ª	Apresentação do trabalho de	06	Projeto de sistema

OBS: Os registros acima correspondem a horas de atividades acadêmicas efetivas.

Considerando a necessidade de um uso correto da linguagem, será considerado, na correção dos trabalhos, o uso adequado da linguagem escrita - correção gramatical e ortográfica, coesão e coerência da linguagem escrita:

- Os docentes procederão a correção devida, descontando 0,1 (um décimo) por incorreção na linguagem escrita, não devendo ultrapassar 10% do valor total da avaliação.

Os critérios gerais de avaliação atendem ao Regulamento aprovado no Conselho Superior Acadêmico.

### Unidade I e II

$MU = (\text{peso AI} \times \text{MAI} + \text{peso AG} \times \text{MAG}) / 10$

Sendo:

MU = Média da Unidade

MAI = Média aritmética simples da(s) avaliação(ões) individual(ais) realizada(s) na unidade

MAG = Média aritmética simples da(s) avaliação(ões) em grupo realizada(s) na unidade

Peso AI = Peso da Avaliação Individual

Peso AG = Peso da Avaliação em Grupo

### Verificação Multidisciplinar - VMD

Após o período de avaliações da segunda Unidade Letiva, o aluno deverá responder a Verificação Multidisciplinar - VMD, que tem como objetivo identificar o nível de aprendizagem dos estudantes no que concerne às competências e habilidades estabelecidas nas diretrizes curriculares dos cursos. A Verificação Multidisciplinar será realizada em todos os cursos e disciplinas, excluindo-se as disciplinas de trabalho de conclusão de curso, estágio supervisionado, tópicos especiais e demais disciplinas integralmente práticas.

A Verificação Multidisciplinar terá peso 1 (um) e comporá a Média Curricular (MC).

### Cálculo da Média Curricular (MC) do Semestre

O aluno será aprovado sem final quando obtiver Média Curricular (MC) do período letivo (semestre) igual ou superior a 7,0 (sete).

$MC = (MU1 \times \text{peso } 4,5 + MU2 \times \text{peso } 4,5 + VMD \times \text{peso } 1) / 10$

Sendo:

MC = Média Curricular

MU1 = Média da Unidade Letiva 1

MU2 = Média da Unidade Letiva 2

VMD = Nota obtida na Verificação Multidisciplinar

### Cálculo da Média com Avaliação Final (AF) do Semestre

A Média Curricular (MC) mínima para que o aluno tenha direito à realização da Avaliação Final (AF) do semestre passa a ser 4,0 (quatro). Assim, o aluno que obtiver Média Curricular (MC) menor que 4,0 (quatro) estará reprovado, sem direito à realização da Avaliação Final (AF) do semestre.

Tendo realizado a Avaliação Final (AF) do semestre, o aluno será aprovado quando obtiver Média Final Curricular (MFC) igual ou superior a 5,0 (cinco) conforme a seguinte fórmula:

$MFC = (MC \times \text{peso } 6 + AF \times \text{peso } 4) / 10$

Sendo:

MFC = Média Final Curricular

MC = Média Curricular

AF = Avaliação Final

6 RECURSOS	
1.	<b>Auxiliares:</b> Livros, Revistas, Publicações, etc.
2.	<b>Visuais:</b> Projetor Multimídia, Computador, Kit Computador +TV, etc.
7 REFERENCIAS BASICAS	
1.	PRESSMAN, Roger S. <b>Engenharia de software</b> . 6 ed. São Paulo: McGraw-Hill, 2006. 720 p. ISBN 8586804576 (broch.)
2.	GUEDES, Gilleanes T. A. <b>UML: uma abordagem prática</b> . 2. 4d. São Paulo: Novatec Editora, 2006. 319 p ISBN 8575220527 (broch.).
3.	BOOCH, Grady.; RUMBAUGH, James.; JACOBSON, Ivar.. <b>UML: guia do usuário</b> / Rio de Janeiro: Campus, 2000. 472 p ISBN 8535205624 (broch.).
8 REFERENCIAS COMPLEMENTARES	
1.	PESSÔA, André de Almeida. <b>Projeto de sistemas</b> . Rio de Janeiro: Book Express, 2000. 227 p. ISBN 858684649X
2.	GAMMA, Erich. <b>Padrões de projeto: soluções reutilizáveis de software orientado a objetos</b> . Porto Alegre: Bookman, 2000. 364p. ISBN 8573076100 (broch.)
3.	FERNANDES, Aguinaldo Aragon; TEIXEIRA, Descartes de Souza. <b>Fábrica de software: implantação e gestão de operações</b> . 1.ed. - 3. reimpr. São Paulo: Atlas, 2007. 304p. ISBN 9788522436903 (broch.)
4.	COAD, Peter.; YOURDON, Edward,. <b>Análise baseada em objetos</b> . 2. 4d. Ver. -. Rio de Janeiro: Campus, c1991. 225 p ISBN 8535200428 (broch.).
5.	HAY, David C., <b>Princípios de modelagem de dados</b> . São Paulo: Makron Books, 1999. 271 p. ISBN 8534608709



# Obrigado!



Professor: Esp. Marcos Morais de Sousa

---

Curso: Sistemas de informação

---

---

---

---

**Professor:** Esp. Marcos Morais de Sousa

---

**Curso:** Sistemas de informação

---

**Disciplina:** Análise e Modelagem de Sistemas II

---

**Aula 2-3:** Diagrama de sequencia

---

**Turma:** 05-06 semestre

---

# Revisão - introdução

- Na última década, a Orientação a Objetos firmou-se definitivamente como paradigma padrão de desenvolvimento de sistemas.
- Entre as diversas vantagens em relação à sua antecessora, a Análise Estruturada, a Orientação a Objetos trouxe uma vantagem considerada fundamental: a adoção de uma linguagem unificada de modelagem, acabando com a verdadeira Babel de metodologias que a precederam.

# Revisão – UML

- A UML disponibiliza um conjunto de 13 diagramas diferentes, cada um deles capturando uma abstração do sistema, e cujo conjunto constitui um excelente conjunto de artefatos de análise e projeto.
- Nem todos os diagramas são aplicáveis e necessários ao desenvolvimento todo de qualquer sistema.

# Visões UML 2

<b>Visão</b>	<b>Descreve</b>	<b>Diagramas</b>
<b>Visão de Requisitos Funcionais</b>	Requisitos funcionais do sistema pelo ponto de vista do usuário.	diagramas de casos de uso
<b>Visão Estrutural Estática</b>	Estrutura estática do sistema.	diagrama de classes diagrama de estruturas
<b>Visão de Comportamento Dinâmico</b>	Comportamento dinâmico do sistema, mostrando suas interações.	diagramas de seqüências diagramas de atividades diagramas de estados

# UML 2

- Todavia, de uma forma quase absoluta, três dos tipos disponíveis de diagramas devem sempre fazer parte da documentação de projeto de um sistema:
- os **Diagramas de Casos de Uso**, juntamente com as respectivas descrições detalhadas,
- os **Diagramas de Classes**, e um de dois tipos de **Diagramas de interação**:
  - os Diagramas de Comunicação ou
  - os Diagramas de Sequência.

# UML2

A UML 2.0 define 13 tipos de diagramas divididos em 2 categorias:

- **Diagramas Estruturais**

- Pacotes
- Classes
- Objetos
- Estrutura Composta
- Componentes
- Instalação

- **Diagramas Comportamentais**

- Casos de uso
- Atividades
- Máquina de estado
- Colaboração ou Comunicação
- Sequência
- Tempo
- Interatividade

- Estudem o material complementar no blog do professor e nos livros indicados.

[marcosmorais.blogspot.com](http://marcosmorais.blogspot.com)



# Diagramas de Sequência

- Diagramas de Sequência são muito úteis em fornecer suporte real à implementação e em constituir uma rica documentação de alto nível.
- Eles representam os objetos participantes de uma colaboração enquanto emitem e recebem mensagens no intuito de realizar um caso de uso. As mensagens são apresentadas em sua ordem temporal, o que facilita a compreensão do fluxo de controle do caso de uso.

# Diagramas de Sequência

- Diagramas de sequência possuem dois eixos: o eixo vertical, que mostra o tempo e o eixo horizontal, que mostra os objetos envolvidos na sequência de uma certa atividade.
- Eles também mostram as interações para um cenário específico de uma certa atividade do sistema.

# Diagramas de Sequência

- O diagrama de sequência mostra a troca de mensagens entre diversos objetos, em uma situação específica e delimitada no tempo. Coloca ênfase especial na ordem e nos momentos nos quais mensagens para os objetos são enviadas.
- Em diagramas de sequência, objetos são representados através de linhas verticais tracejadas (denominadas como linha de existência), com o nome do objeto no topo.
- O eixo do tempo é também vertical, aumentando para baixo, de modo que as mensagens são enviadas de um objeto para outro na forma de setas com a operação e os nomes dos parâmetros.

# Diagramas de Sequência

- Em alguns sistemas, objetos rodam concorrentemente, cada um com sua linha de execução (thread). Se o sistema usa linhas concorrentes de controle, isto é mostrado como ativação, mensagens assíncronas, ou objetos assíncronos.

# Notação



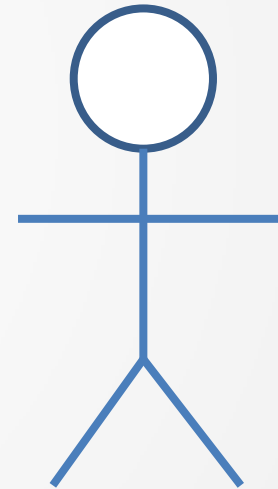
OBJETO



MENSAGENS

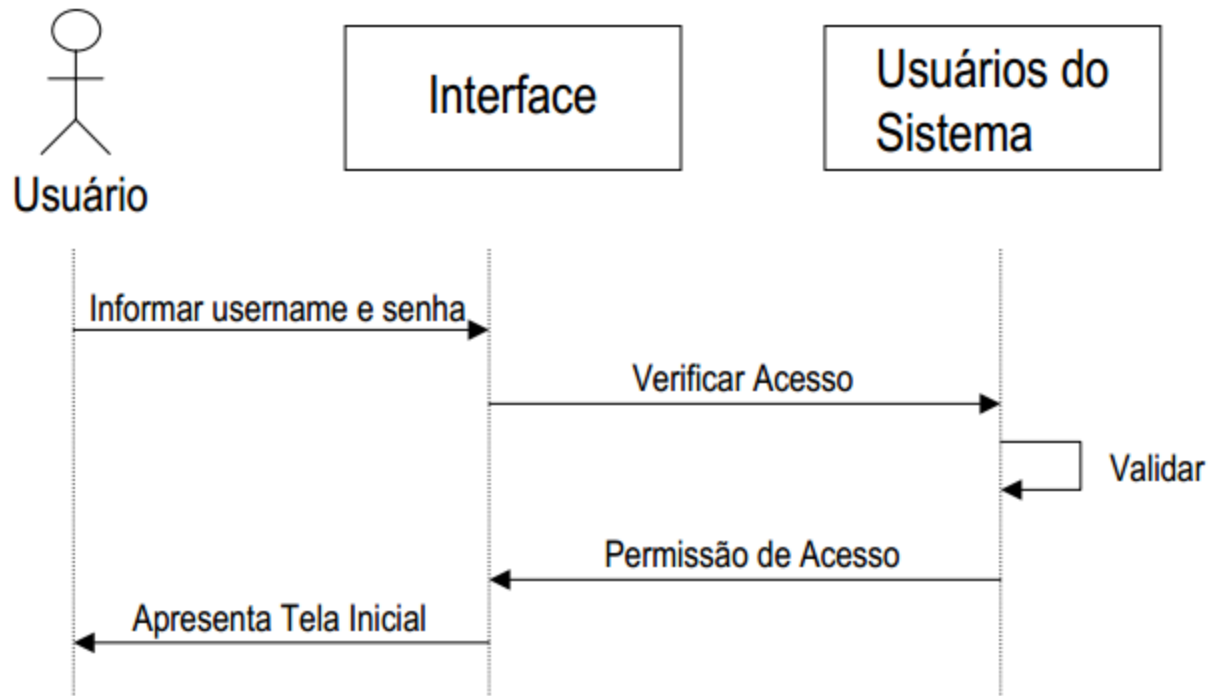


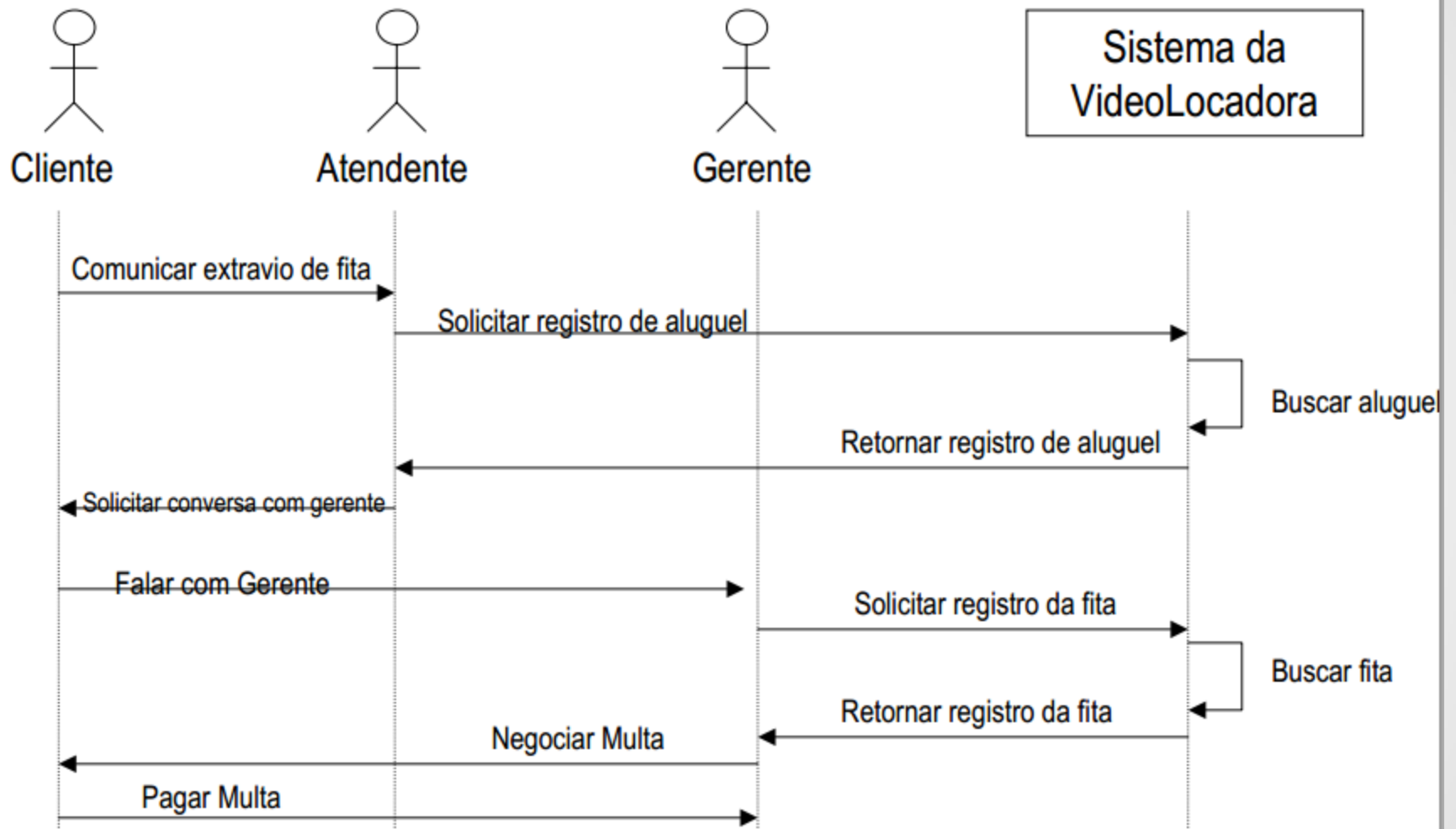
TEMPO



ATOR

## Exemplo – Cenário Normal do Caso de Uso Efetuar Login:

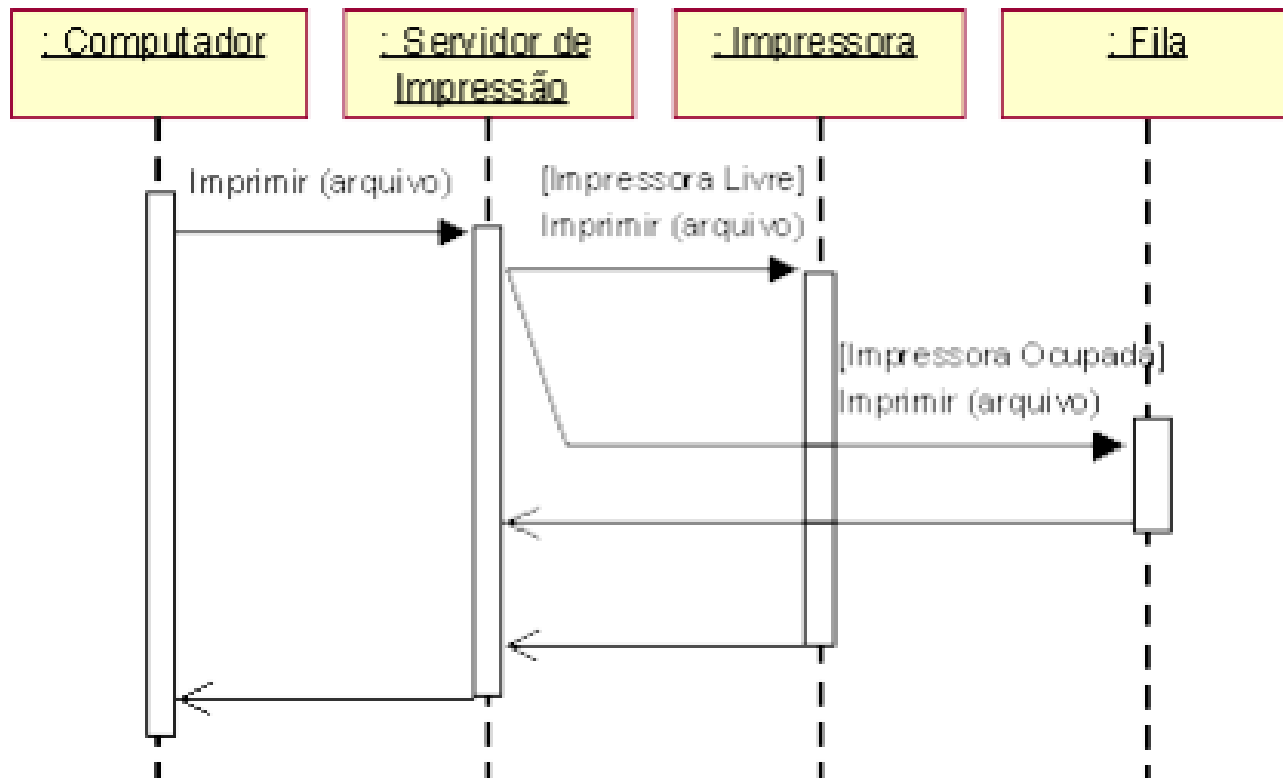




Os diagramas de sequência podem mostrar objetos que são criados ou destruídos como parte do cenário documentado pelo diagrama.

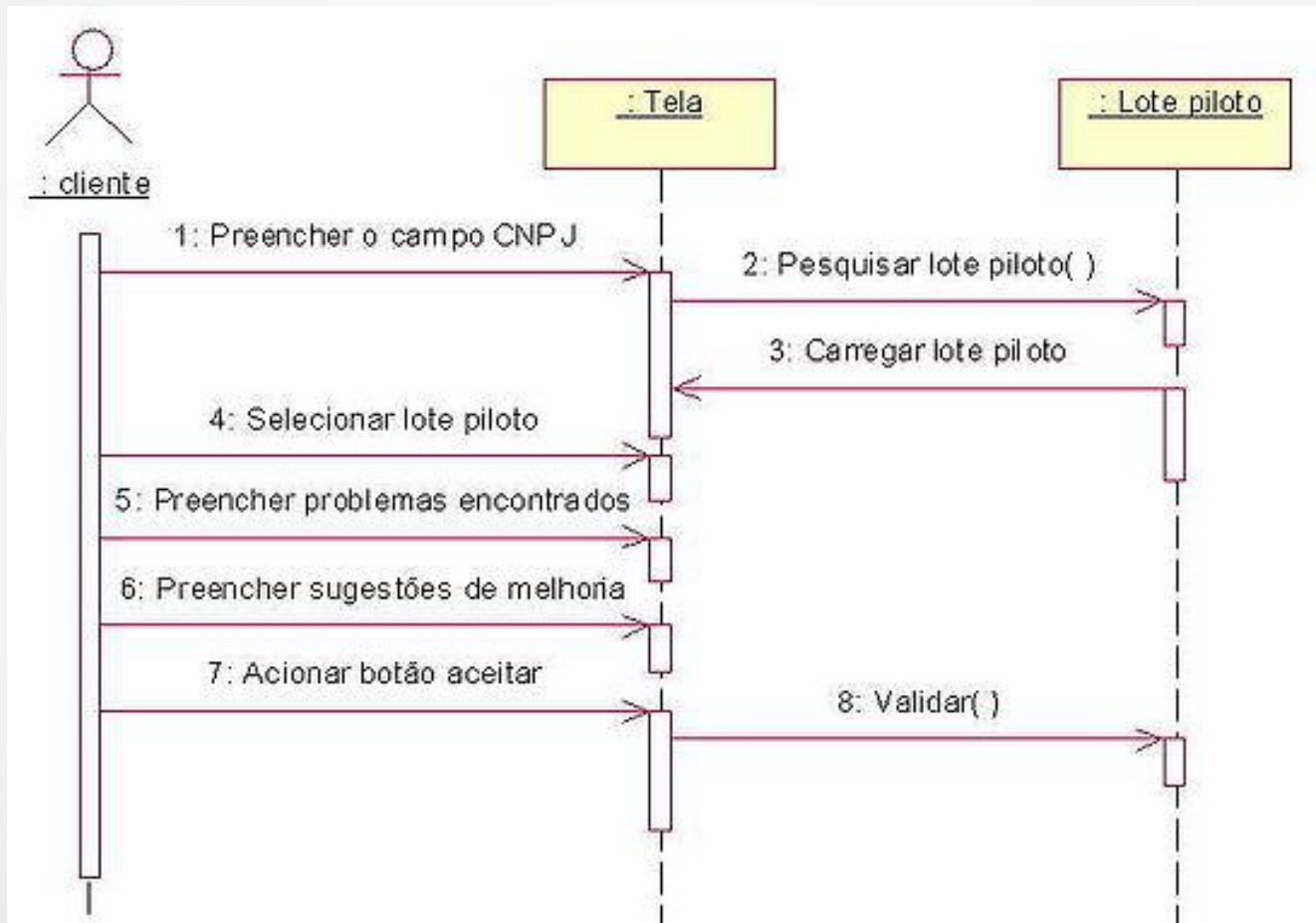
Um objeto pode criar outros objetos através de mensagens.

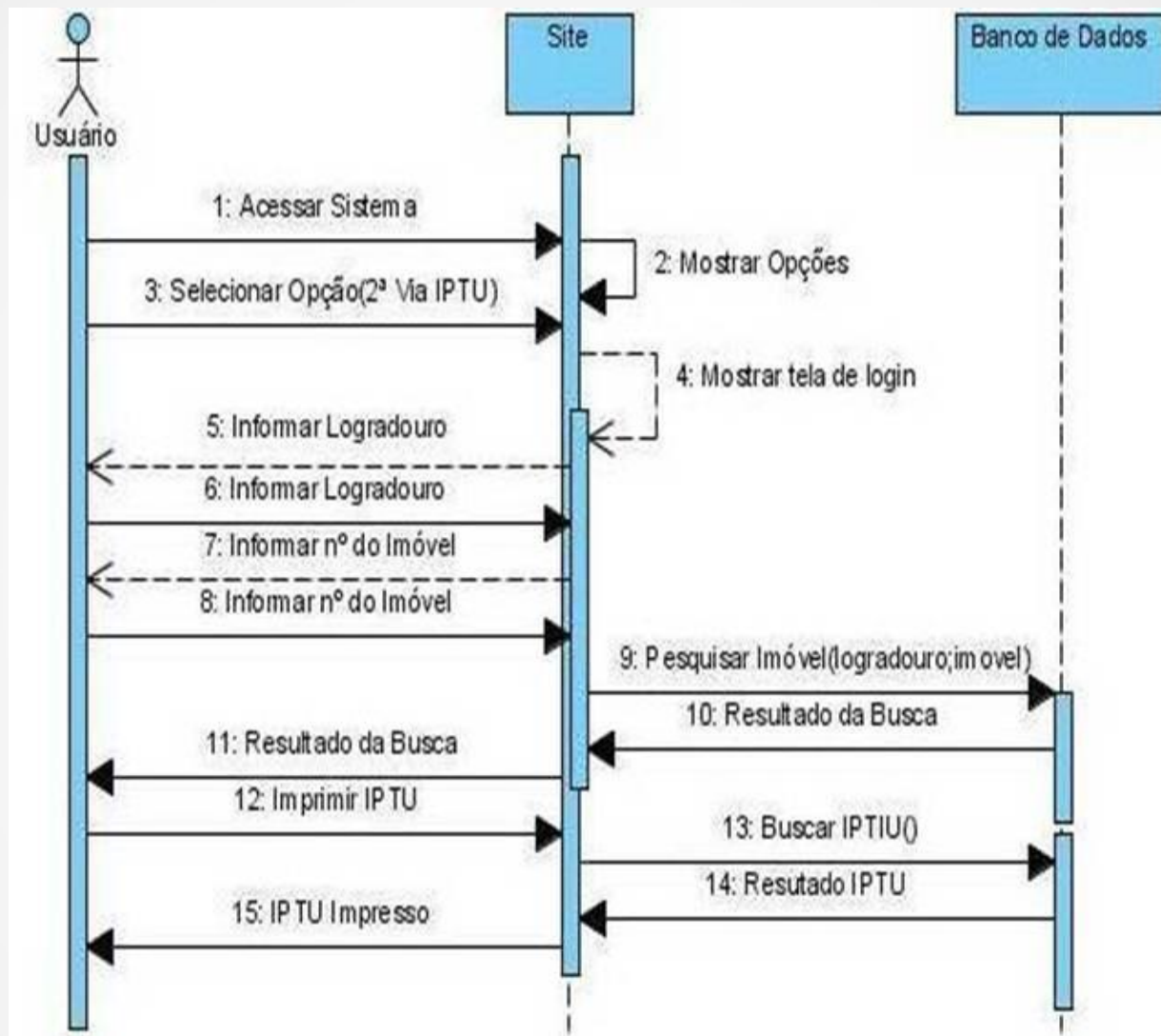
A mensagem que cria ou destrói um objeto é geralmente síncrona, representada por uma seta sólida.





# exemplo

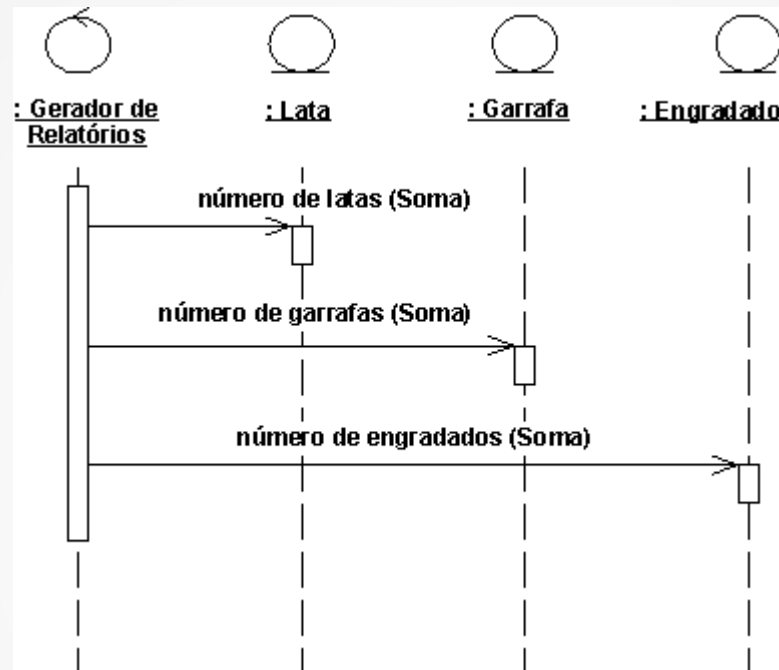




Fonte: <http://iptucf.sqlweb.com.br/modelagem.php>

## Exemplo – controle centralizado

O **controle centralizado** de um fluxo de eventos ou de parte do fluxo de eventos significa que poucos objetos guiam o fluxo trocando mensagens com outros objetos. Esses objetos controladores decidem a ordem em que outros objetos serão ativados no caso de uso. A interação entre os objetos restantes é mínima ou inexistente.

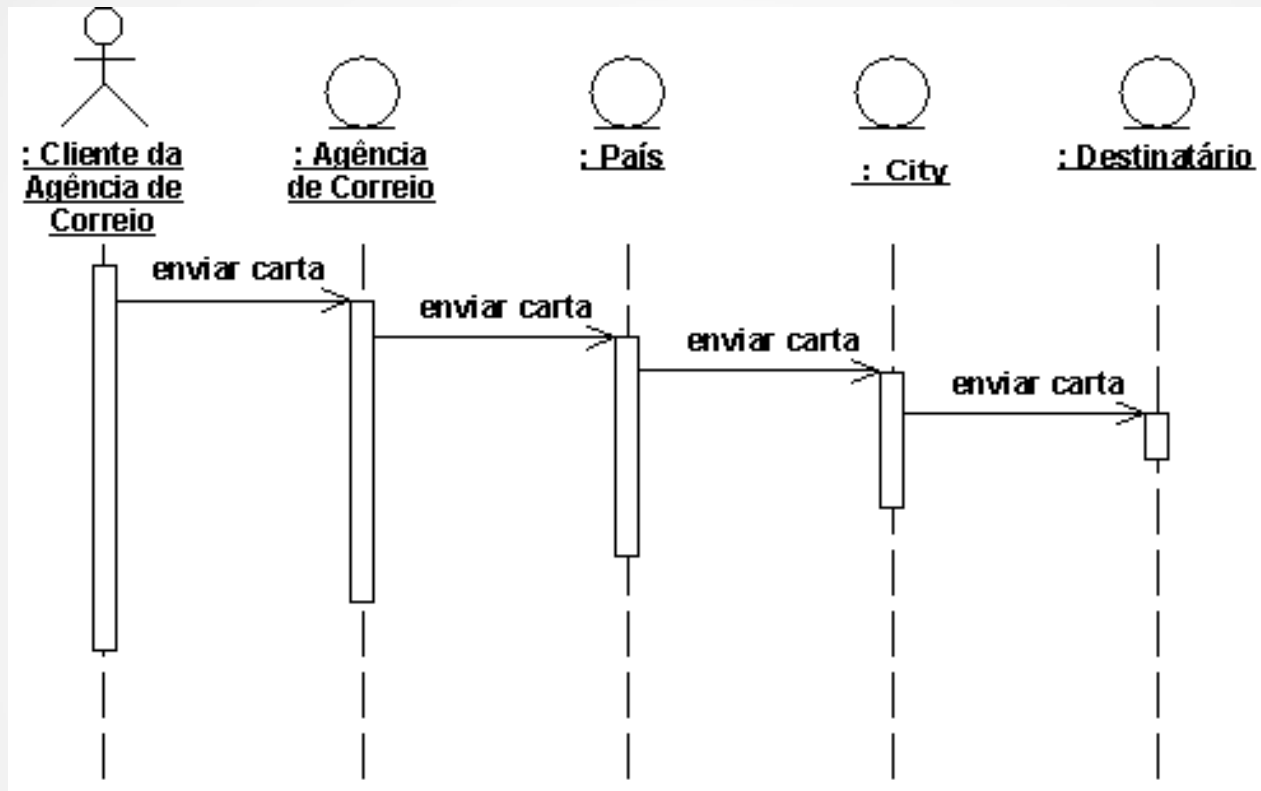


No **Sistema da Máquina de Reciclagem**, o caso de uso **Imprimir Relatório Diário** controla, entre outros, o número e o tipo de objetos retornados, e escreve a contagem em um recibo. O objeto de controle **Gerador de Relatório** decide a ordem em que os totais serão extraídos e escritos.

A estrutura de comportamento do caso de uso **Imprimir Relatório Diário** é centralizada no objeto de controle **Gerador de Relatório**.

## Exemplo – controle descentralizado

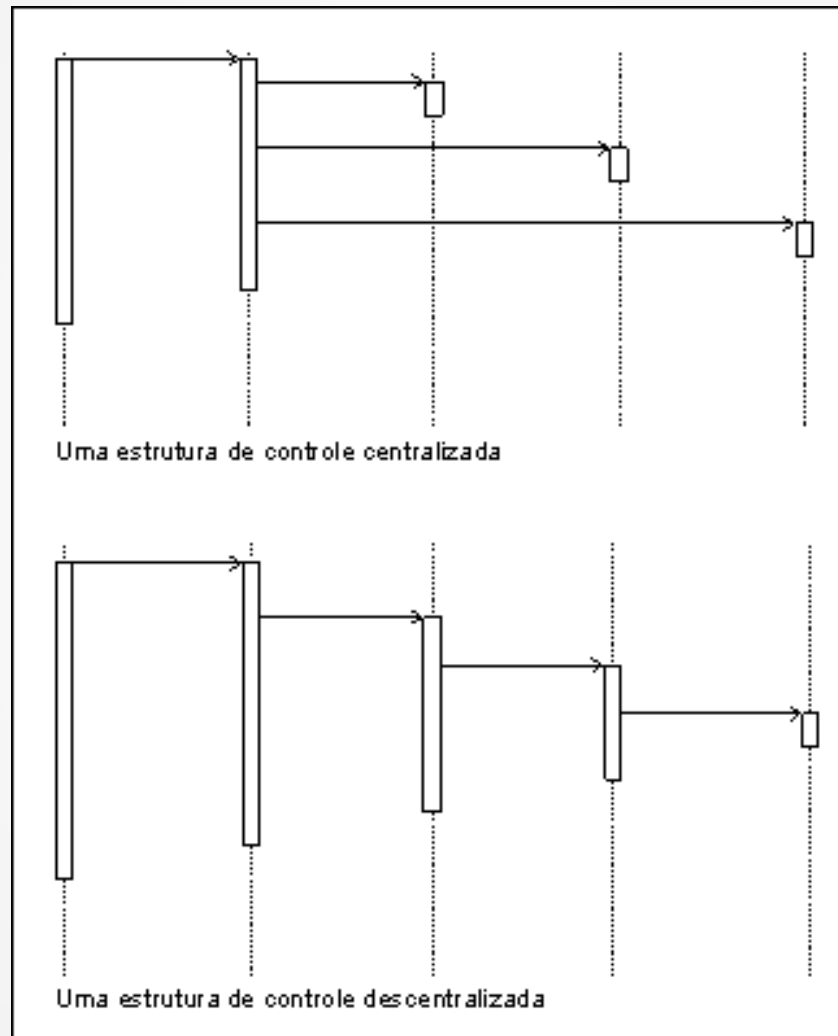
O **controle descentralizado** surge quando os objetos participantes se comunicam diretamente entre si, e não através de um ou mais objetos controladores.



No caso de uso **Enviar Carta** alguém remete uma carta para outro país através de uma agência de correio. Primeiro, a carta é enviada para o país do destinatário. No país, a carta é enviada para uma cidade específica. A cidade, por sua vez, envia a carta para a residência do destinatário.

Um fluxo de eventos com controle centralizado terá um diagrama de sequência "em forma de forquilha". Por outro lado, um diagrama de sequência "em forma de escada" ilustra que a estrutura de controle foi descentralizada para os objetos participantes.

Uma estrutura de controle centralizada em um fluxo de eventos produz um diagrama de sequência "em forma de forquilha". Uma estrutura de controle descentralizada produz um diagrama de sequência "em forma de escada".



Uma estrutura de controle centralizada em um fluxo de eventos produz um diagrama de sequência "em forma de forquilha". Uma estrutura de controle descentralizada produz um diagrama de sequência "em forma de escada".

**A estrutura de comportamento da realização de um caso de uso frequentemente consistem em uma mistura de comportamento centralizado e descentralizado.**

**Uma estrutura descentralizada será adequada:**

- Se as fases de subeventos estiverem intrinsecamente acopladas. Esse será o caso se os objetos participantes:
  - Formarem uma hierarquia de partes ou constituintes, como País - Estado - Cidade;
  - Formarem uma hierarquia de informações, como CEO - Gerente de Divisão - Gerente de Seção;
  - Representarem uma progressão cronológica fixa (a sequência de fases de subeventos será sempre realizada na mesma ordem), como Anúncio - Pedido - Fatura - Remessa - Pagamento; ou
  - Formarem uma hierarquia de herança conceitual, como Animal - Mamífero - Gato.
- Se você desejar encapsular a funcionalidade e, portanto, fazer abstrações dela. Isso é bom para alguém que deseje utilizar sempre a funcionalidade inteira, porque a funcionalidade pode se tornar desnecessariamente de difícil compreensão caso a estrutura de comportamento seja centralizada.

**Uma estrutura centralizada será adequada:**

- Se a ordem em que as fases de subeventos serão executadas puder ser mudada.
- Se você espera inserir novas fases de subeventos.
- Se você deseja manter partes da funcionalidade reutilizáveis como peças separadas.

# Diagramas de Sequência – importante!

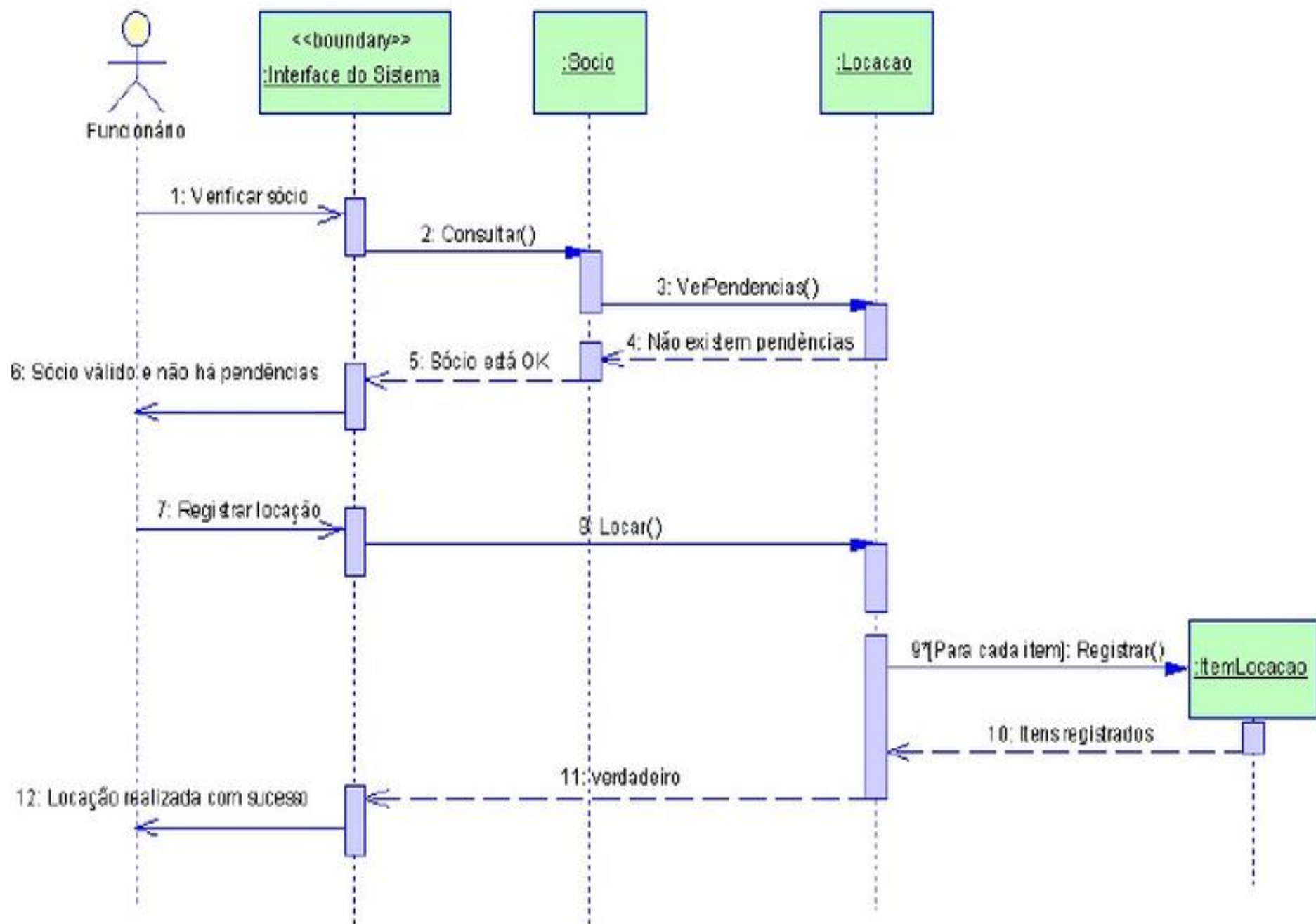
## Lembre-se:

- Diagramas de Sequência Preocupa-se com a ordem temporal em que as mensagens são trocadas entre os objetos de determinado processo;
- Diagramas de Sequência baseia-se em um caso de uso definido;
- Diagramas de Sequência apoia-se no diagrama de classes.

# Exercício1

- Desenvolva um **Diagrama de Sequência** para um sistema de vídeo locadora, de acordo com seguintes fatos:
- Primeiramente o atendente deve verificar se o sócio está cadastrados e não a locação será recusada.
- Em seguida deve verificar se o sócio possui alguma locação pendente, caso em que também recusará o empréstimo.
- Se o sócio existir e não tiver locações pendentes, então a locação deverá ser registrada e as cópias emprestadas ao sócio.
- Durante o registro da locação deverão ser registrados também todos os itens da locação.



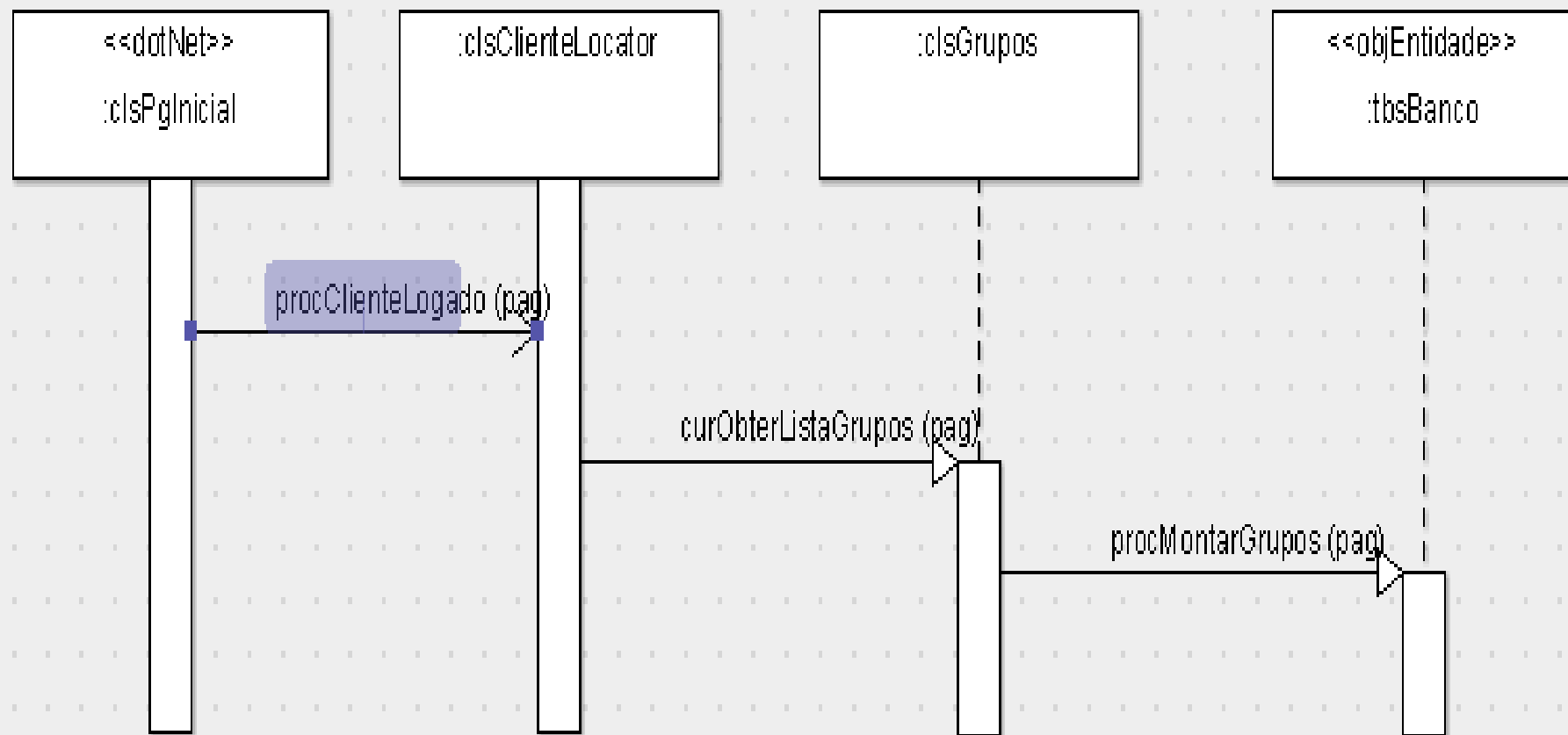


# Exercício2

- O cliente entra no na **Loja Virtual** e visualiza os grupos de produtos, para fazer uma pesquisa e a classe responsável é **procClienteLogado**.
- Uma lista criada indica que esses grupos estão salvos em um banco de dados e são recuperados por uma página em **dotNET**. Quando desejarmos retirar ou inserir algo no banco de dados vamos estereotipar a entidade do banco como **tbsBanco**.
- As descrições do banco de dados serão através da classe **clsGrupos**, no seu método **curObterListaGrupos()**.
- Então, quando a URL for solicitada, a página dotNET **clsPagInicial** pede a classe **clsGrupos** que execute **curObterListaGrupos()** e esta vai até o banco de dados e executa uma procedure que retorna grupos possíveis (**procMontarGrupos**). Este método esta na classe **clsClienteLocator**.

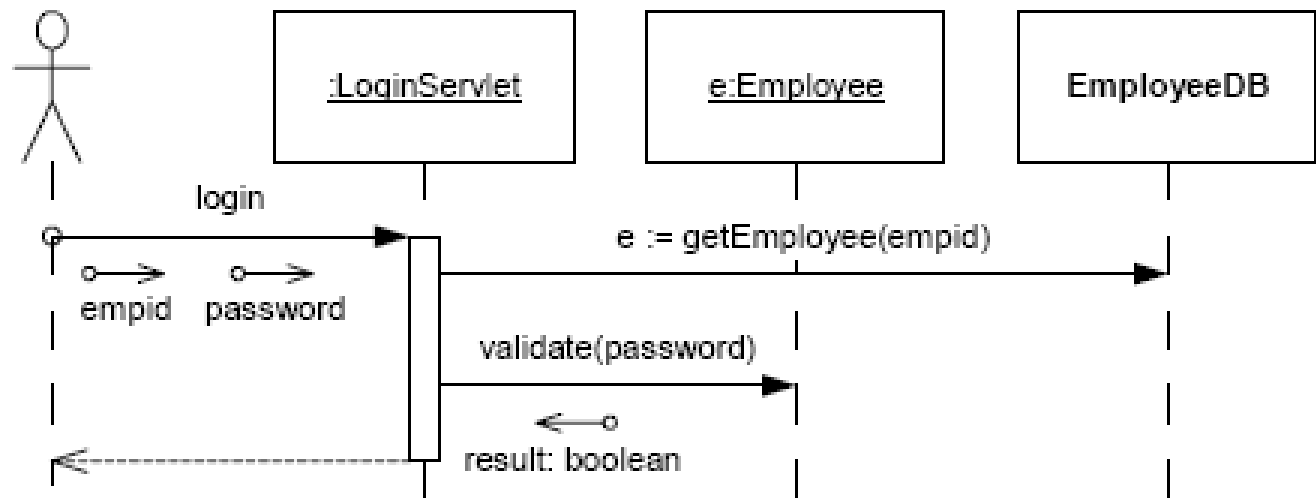
# Sequencia planejada/esperada

1. Página dotNet
2. Classe Cliente
3. Classe de Grupos
4. Entidades do banco de dados



# Exercício 3

Desenvolva a narrativa adequada para o diagrama de sequencia apresentado.



# Até a próxima



Professor: Esp. Marcos Morais de Sousa

---

Curso: Sistemas de informação

---

---

---

---