

Análise e Projeto de Sistemas



Prof. Dr. Ronaldo C. de Oliveira

ronaldo.co@ufu.br

www.facom.ufu.br/~ronaldooliveira

FACOM - 2018

UML – Linguagem Unificada de Modelagem



Diagramas de Projeto de Software

UML – Linguagem Unificada de Modelagem



Diagramas de Classe Completo

UML

Diagrama de Classe

- Um *diagrama de classe* ilustra as especificações de software para as classes e interfaces do sistema
- Inclui:
 - Classes, associações e atributos;
 - Interfaces (com operações e constantes);
 - Métodos que manipulam os objetos;
 - Informação sobre o tipo dos atributos;
 - Navegabilidade;
 - Dependências;
- UML não diferencia modelo conceitual de diagrama de classe (o termo “classe de implementação” é usado para distinguir o segundo do primeiro)

UML

Diagrama de Classe

- Criando um Modelo Conceitual
 - Passos sugeridos:
 1. Liste os conceitos candidatos para os casos de usos em questão usando a lista de categorias comuns e identificação textual de nomes;
 2. Desenhe-os em um modelo conceitual;
 3. Adicione as associações necessárias para registrar os relacionamentos para os quais é preciso preservar alguma memória;
 4. Adicione os atributos necessários para cumprir os requisitos de informação.

UML

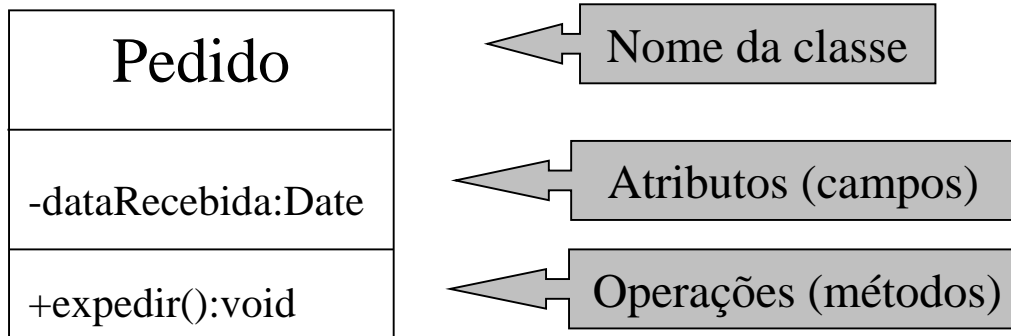
Diagrama de Classe

- Criando um diagrama de classes
 - UML usa o termo genérico “classe” para denotar tanto entidades do domínio da aplicação quanto classes na Programação Orientada a Objetos – POO:
 - Uma classe na POO é chamada mais especificamente de “classe de implementação”
 - Os termos “tipo” e “interface” são usados para denotar *especificações* de classes de implementação;
 - O termo “conceito” denota entidades do mundo real, e “classe” denota componentes de software e suas especificações.

UML

Diagrama de Classe

- Notação UML para classe

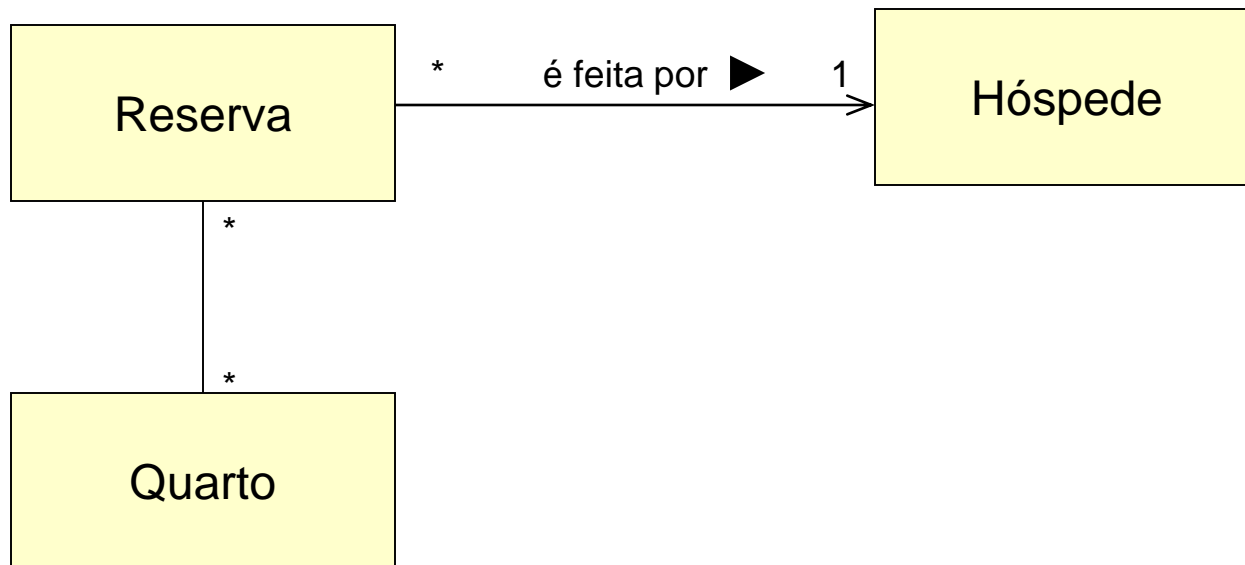


```
Public class Pedido {  
    private Date dataRecebida;  
    public void expedir(){...}  
    ...  
}
```

UML

Diagrama de Classe

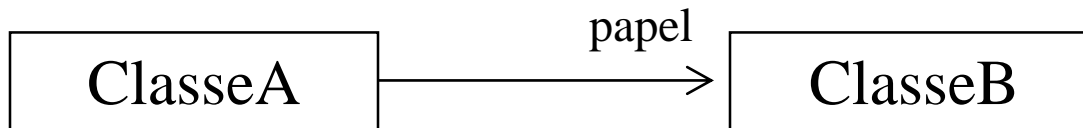
- Associação simples



UML

Diagrama de Classe

- Associações



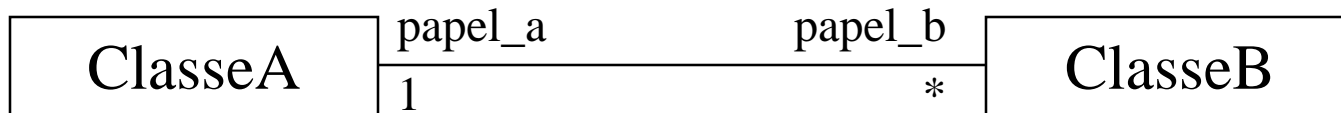
- Java

```
Public class ClasseA {
    private ClasseB papel;
    ...
}
```

UML

Diagrama de Classe

■ Associações



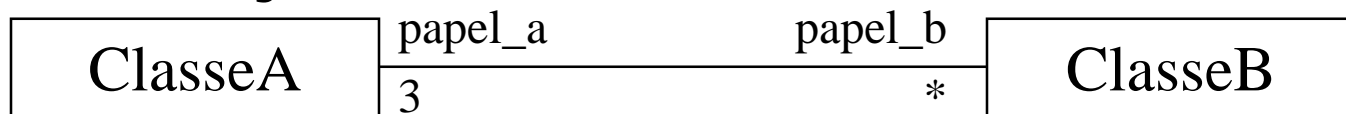
■ Java

```
Public class ClasseA {
    private Vector papel_b;
    ...
}
Public class ClasseB {
    private ClasseA papel_a;
    ...
}
```

UML

Diagrama de Classe

■ Associações



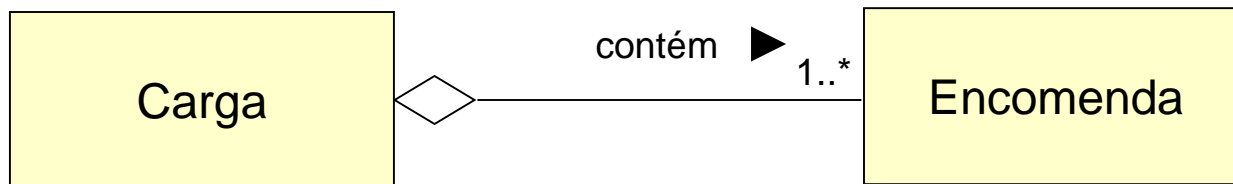
■ Java

```
Public class ClasseA {
    private Vector papel_b;
    ...
}
Public class ClasseB {
    private ClasseA papel_a[];
    ...
    papel_a = new ClasseA[3]
}
```

UML

Diagrama de Classe

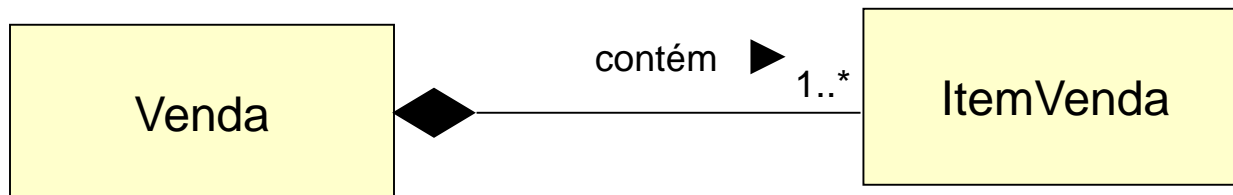
- Relação todo-parte de AGREGAÇÃO
 - Um objeto de uma classe é formada por objetos de outra classe



UML

Diagrama de Classe

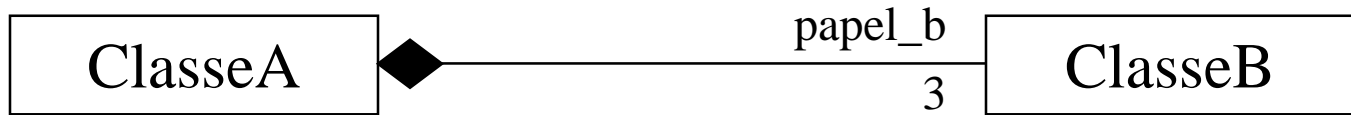
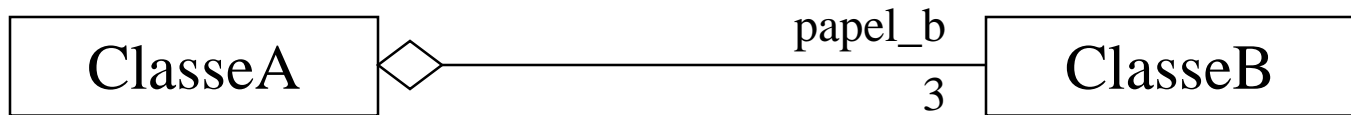
- Relação todo-parte de COMPOSIÇÃO
 - Um objeto de uma classe é formada por objetos de várias classes diferentes



UML

Diagrama de Classe

- Associações – Agregação e Composição



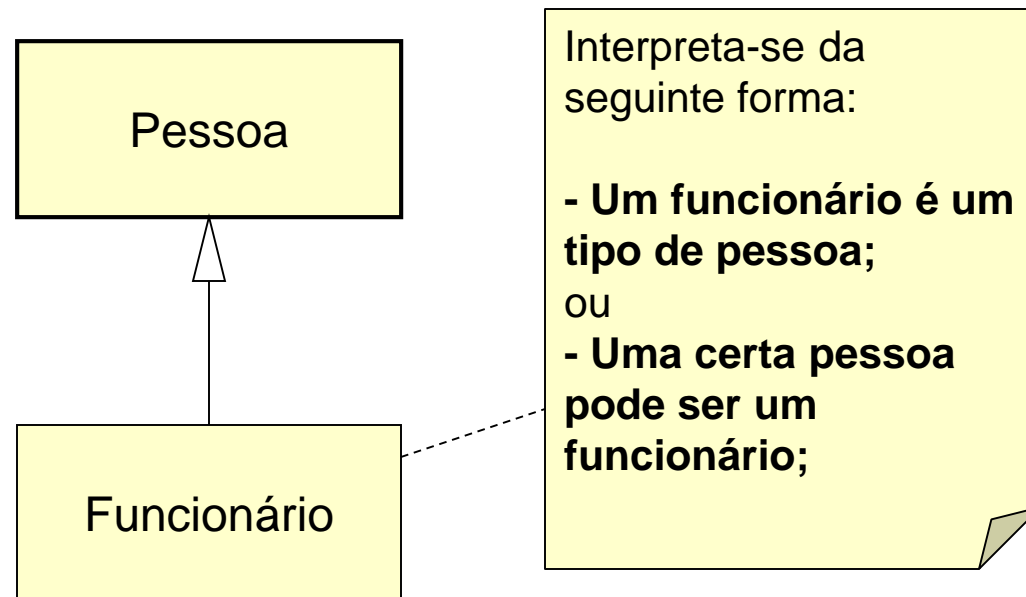
- Java

```
Public class ClasseA {
    private ClasseB papel_b[];
    ...
}
```

UML

Diagrama de Classe

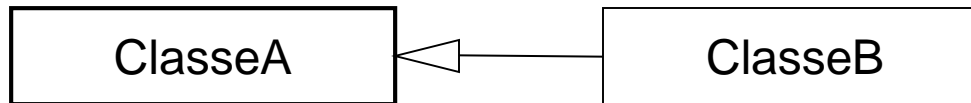
- Relação de generalização/especialização
 - GEN-ESPEC ou Herança



UML

Diagrama de Classe

- Relação de generalização/especialização



- Java

```
Public class ClasseA {
    ...
}
Public class ClasseB extends ClasseA {
    ...
}
```


UML

Diagrama de Classe

- UML – classe abstrata

FormaGeometrica

- Java

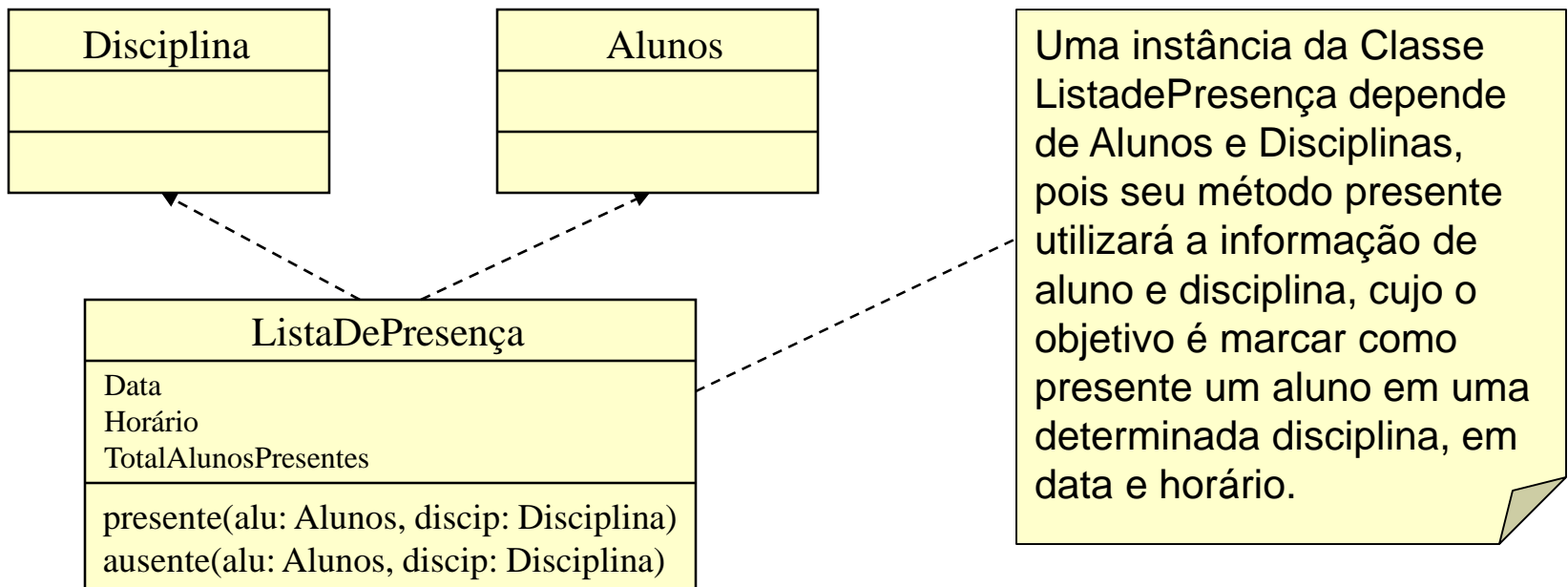
```
public abstract class FormaGeometrica {  
    ...  
}
```

UML

Diagrama de Classe

■ Relação de Dependência entre classes

- Mostra que uma instância de uma classe depende da instância de outra classe, normalmente chamadas de cliente/servidora respectivamente. A dependência é representada por uma seta tracejada.



UML

Diagrama de Classe

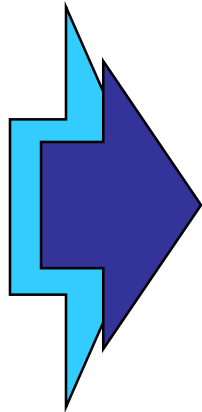
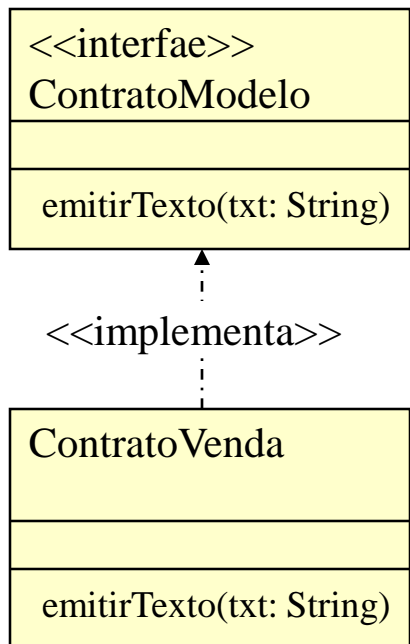
- Interface

- Há um tipo especial de classe a qual não pode ser instanciada, servindo apenas para especificar as operações externamente visíveis para uma classe. Uma interface descreve os padrões legais de interação entre dois objetos. A interface funciona como uma classe modelo, que outras classes poderão fazer uso, implementando as funcionalidades descritas.

UML

Diagrama de Classe

■ Exemplo de Interface



```
public interface ContratoModelo {
    public void emitirTexto (String txt);
}

public class ContratoVenda implements ContratoModelo{
    public void emitirTexto (String txt)
    {
        // Aqui deve ser inserido o código do método
    }
}
```

UML

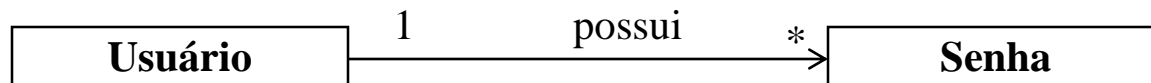
Diagrama de Classe

- Navegação
 - Considerando uma associação simples entre duas classes, é possível navegar de objetos de um tipo até objetos de outro tipo. A menos que seja especificado o contrário, a navegação é bidirecional. Entretanto, em algumas situações, pode ser necessário limitar a navegação em uma só direção.

UML

Diagrama de Classe




- Navegação – Exemplo
 - Ao fazer um controle de acesso ao sistema são encontrados associações entre os objetos ***Usuários*** e ***Senhas***. Considerando um ***Usuário*** devemos ser capaz de encontrar os objetos senhas correspondentes. Agora, a partir de uma ***Senha***, o sistema não deve permitir localizar o ***Usuário*** correspondente.



UML

Diagrama de Classe

■ Visibilidade:

- Público (+ ou   

OBS.: a característica pode ser um atributo ou um método da classe

UML

Diagrama de Classe

- Como Fazer um Diagrama Classe

- Regras úteis:

1. Identificar todas as classes que participam da solução proposta pelos diagramas de interação;
2. Desenhe as classes num diagrama de classe;
3. Inclua os atributos identificados no modelo conceitual;
4. Adicione métodos tal como identificados nos diagramas de interação;
5. Adicione informação sobre o tipo dos atributos e métodos;
6. Adicione as associações necessária para permitir a visibilidade de atributos requisitada;
7. Adicione setas de navegabilidade para indicar a direção da visibilidade de atributos;
8. Adicione relacionamentos de dependência para indicar outros tipos de visibilidade.

UML

Diagrama de Classe

- Os métodos definidos para cada classe são “descobertos” durante a construção do diagramas de interação (seqüência e colaboração). Todas as mensagens que chegam a uma determinada classe, representada no diagrama de Interação, irão representar a construção de um método para classe de mesmo nome da mensagem .

Exemplo - Diagrama de Classe

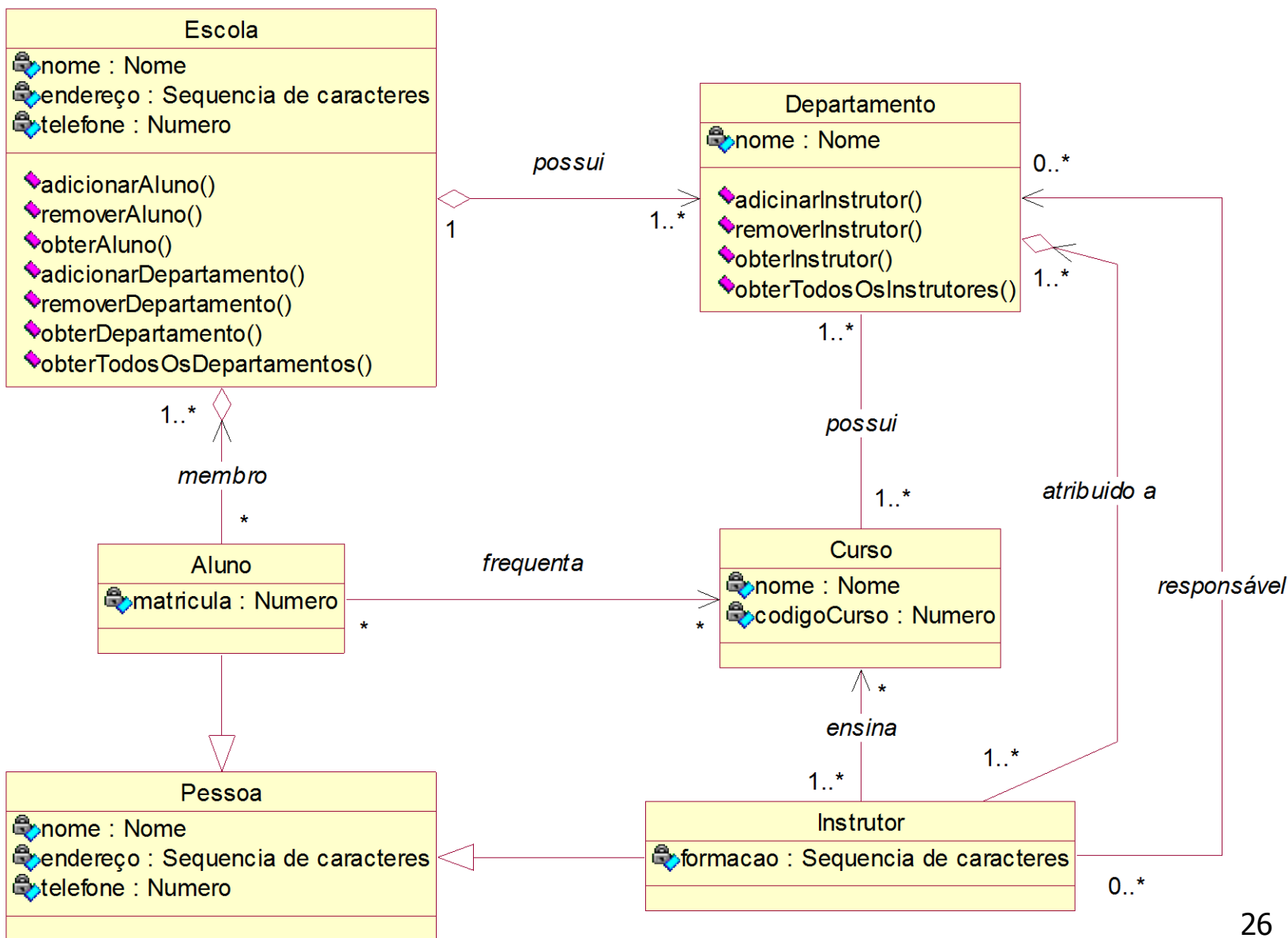




Diagrama de Interação

UML

Diagrama de Interação

- Um *diagrama de interação* ilustra as interações de mensagens entre instâncias (e classes) no modelo de classes
 - Atribuição de responsabilidades aos objetos
 - Ponto de partida é o cumprimento das pós-condições especificadas nos contratos de operação
- A UML defines dois tipos de diagramas de interação:
 - Diagramas de seqüência (faz parte da análise)
 - Diagramas de colaboração entre os objetos (faz parte do projeto)



Diagrama de Seqüência

UML

Diagrama de Seqüência

- Mostra como diversos objetos se relacionam (interagem, colaboram) no ambiente de negócios para a realização de um caso de uso;
- Auxilia na identificação de serviços/métodos e delegação de responsabilidades;
- Elementos:
 - Objetos;
 - Mensagens;
 - Linha da vida;
 - Foco de controle;
 - Retorno.

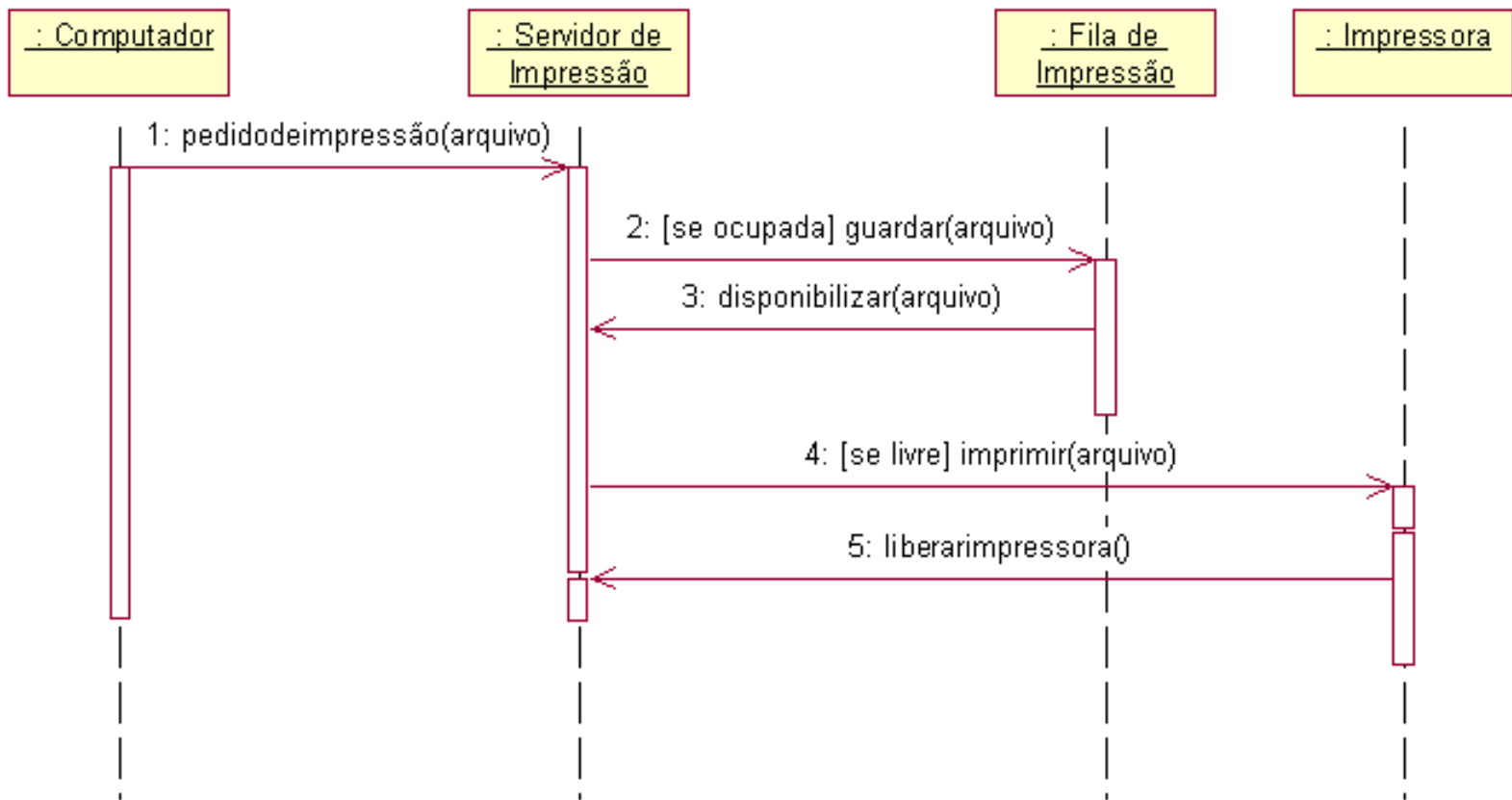
UML

Diagrama de Seqüência

- Como fazer Diagramas de Seqüência
 - Regras úteis:
 1. Identificar os atores que operam diretamente com o sistema. Desenhar uma linha vertical representando cada um desses atores;
 2. Desenhar uma linha vertical representando cada um dos objeto (classes) que o caso de uso manipula;
 3. A partir da descrição das seqüências típicas de eventos dos casos de uso, identificar os eventos de sistema que cada ator gera. Ilustrar os eventos no diagrama através de mensagens.;
 4. Opcionalmente, incluir o texto do caso de uso à esquerda do diagrama.

UML - Diagrama de Seqüência

Exemplo



UML - Diagrama de Seqüência

Exemplo – Clinica Médica

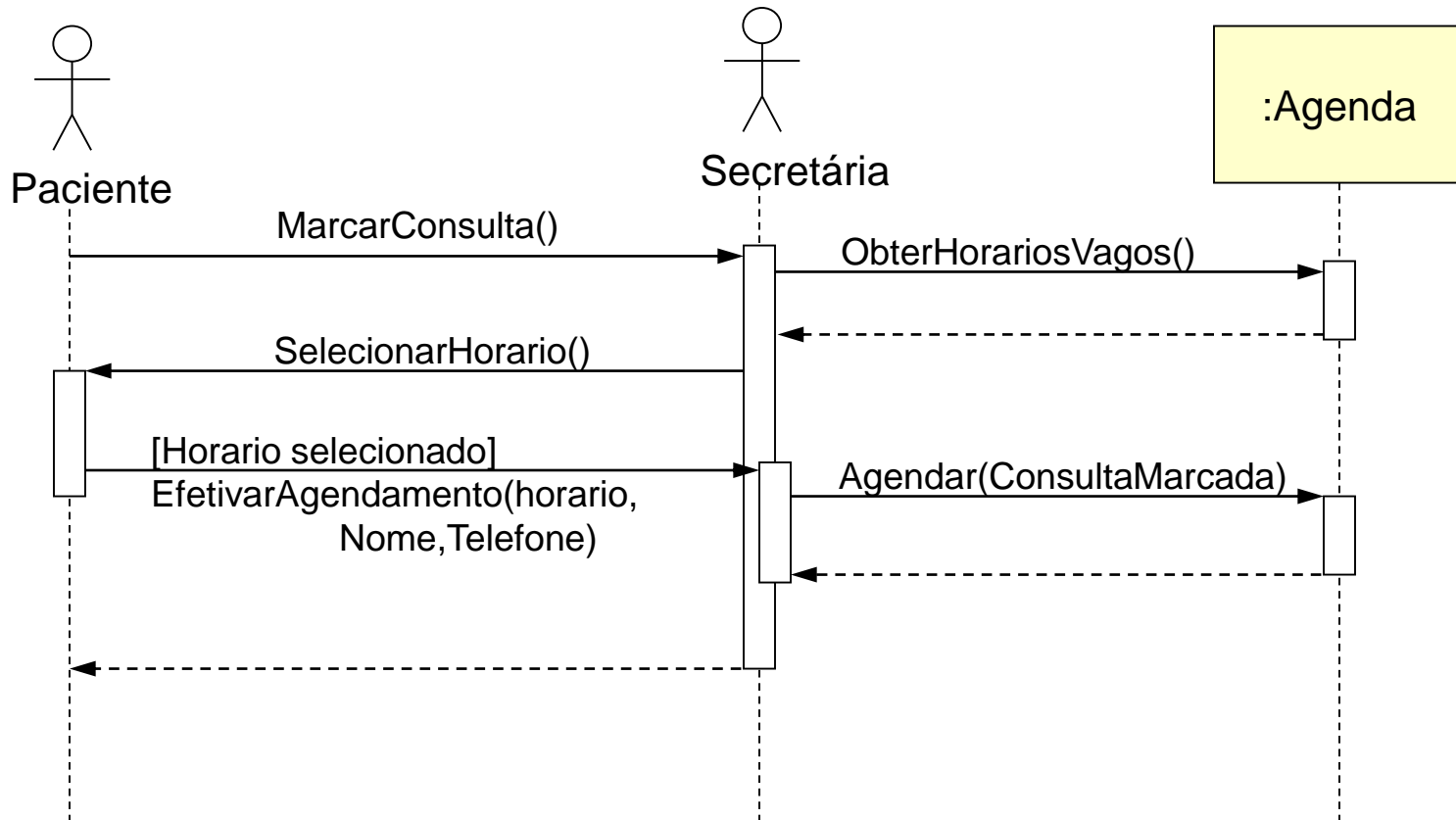


Diagrama de Seqüência – Exemplo – Folha de Pagamento

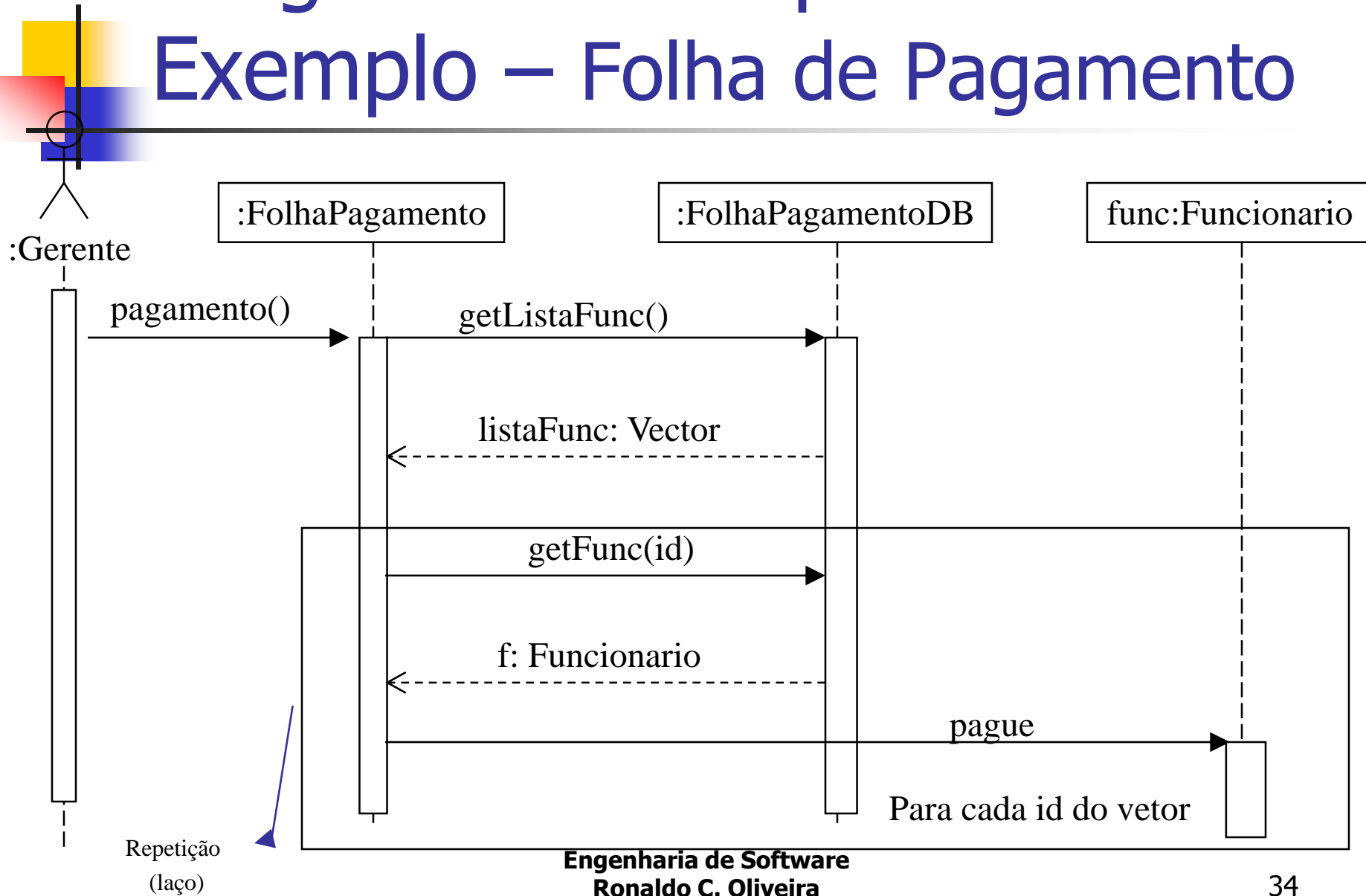


Diagrama de Seqüência - Exemplo

```
public class FolhaPagamento {
    private FolhaPagamentoDB folhaDB;
    private FolhaPagamentoLista listaPagamentos;

    public void pagamento() {
        Vector listaFunc = folhaDB.getListaFunc();
        for (Iterator iterator = listaFunc.iterator(); iterator.hasNext();) {
            String id = (String) iterator.next();
            Funcionario f = folhaDB.getFunc(id);
            if (f.eDiaPagamento()) {
                double pagamento = f.calculatePay();
                double deduções = f.calculaDeduções();
                listaPagamentos.enviaPagamento(pagamento - deduções);
            }
        }
    }
}
//fim do método pagamento
}
//fim da classe FolhaPagamento
```



Diagrama de Colaboração

UML

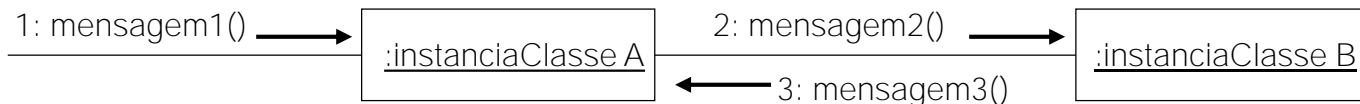
Diagrama de Colaboração

- O diagrama de colaboração apresenta um modo alternativo de representar a troca de mensagens entre um conjunto de objetos, sem a preocupação com a vida útil das mensagens no tempo.
- O diagrama de colaboração não mostra a dimensão do tempo, por isso as seqüências de mensagens e linhas concorrentes devem ser determinadas usando a seqüência de números.

UML

Diagrama de Colaboração

- Diagrama de colaboração
 - Ilustra interações entre objetos (classes) num formato de grafo ou rede, representando a troca de mensagens em uma ordem de execução

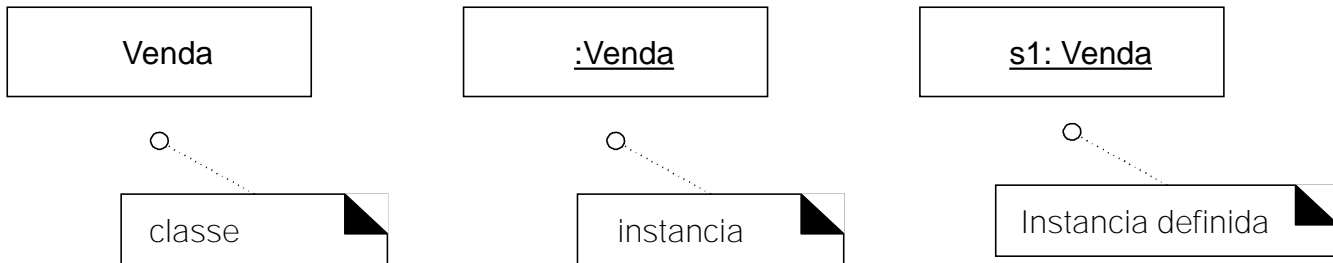


UML

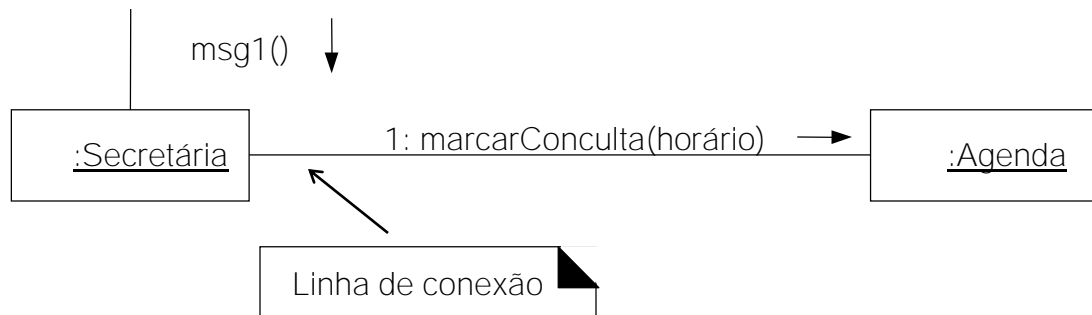
Diagrama de Colaboração

■ Notação Básica

■ Classes e instâncias



■ Conexão entre objetos



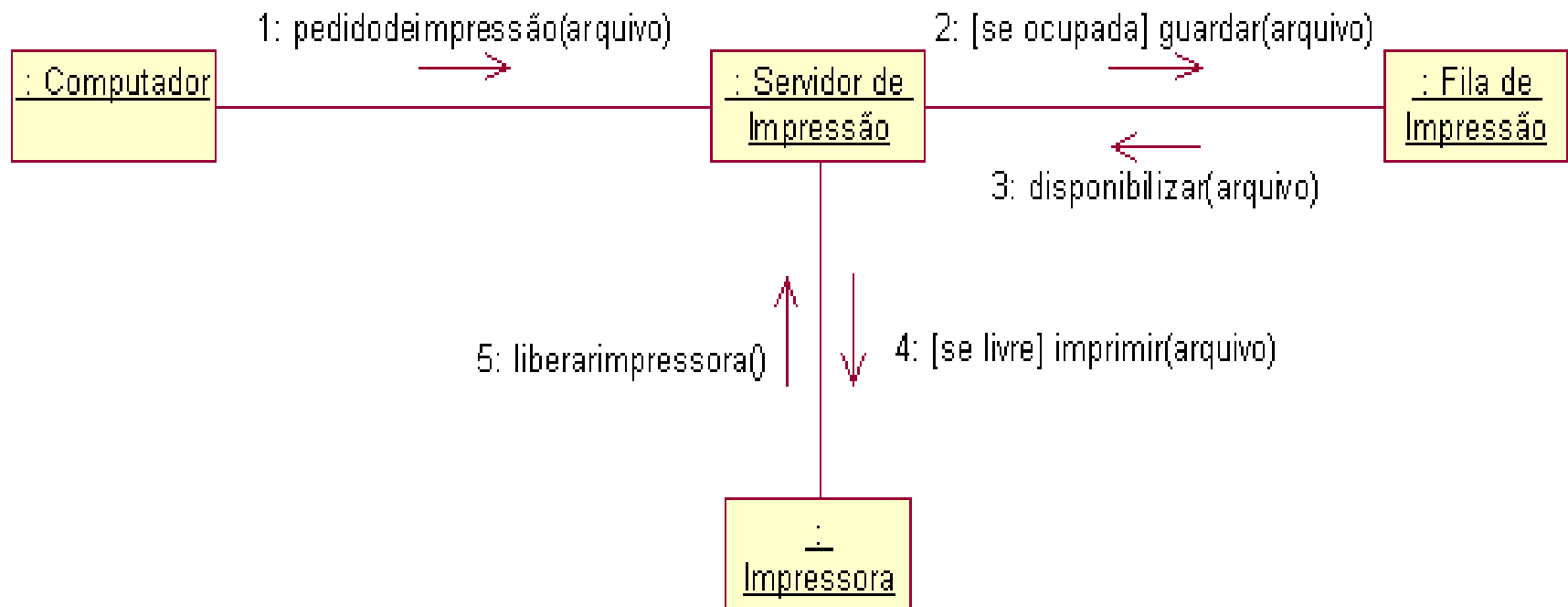
UML

Diagrama de Colaboração

- Como fazer diagrama de colaboração
 - Regras úteis:
 1. Criar um diagrama em separado para cada uma das operações de sistema sendo desenvolvidas no ciclo atual.
 - Para cada mensagem de operação do sistema, criar um diagrama com essa mensagem como mensagem inicial.
 2. Se um diagrama ficar muito complexo (não cabe facilmente num folha de papel A4), o diagrama deve ser dividido em diagramas menores.
 3. Usar as responsabilidades dos atores e a descrição dos casos de uso para projetar um sistema cujo objetos interagem para cumprir as tarefas exigidas.

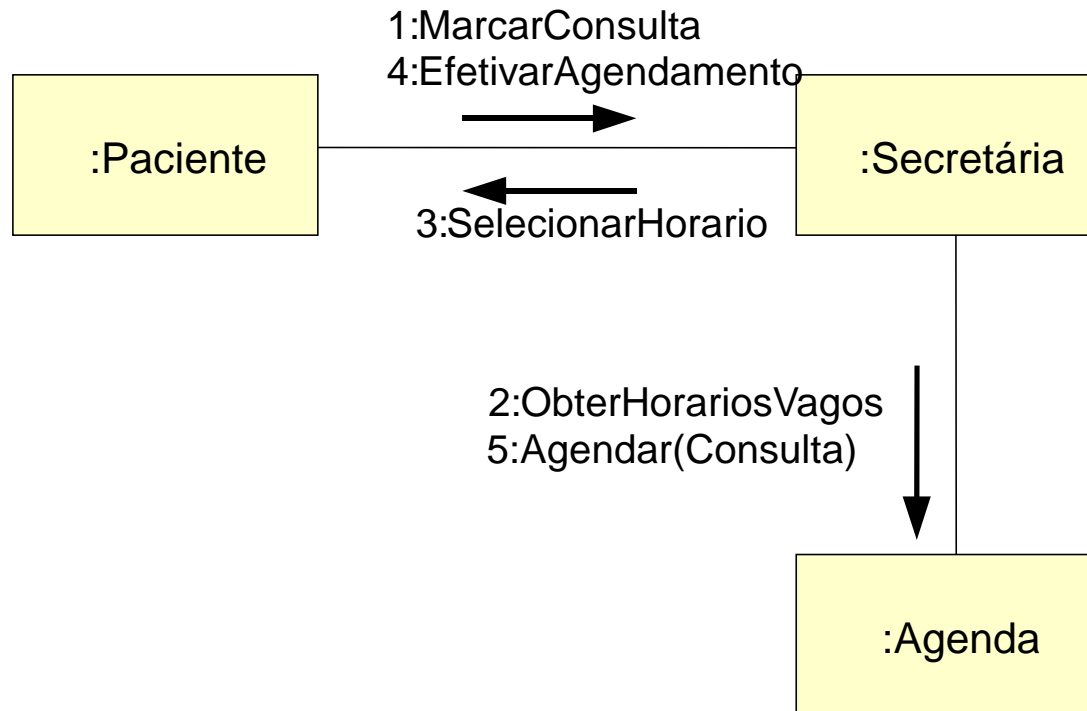
UML - Diagrama de Colaboração

Exemplo



UML - Diagrama de Colaboração

Exemplo – Clínica Médica



UML - Diagrama de Colaboração

Exemplo – Clínica Médica

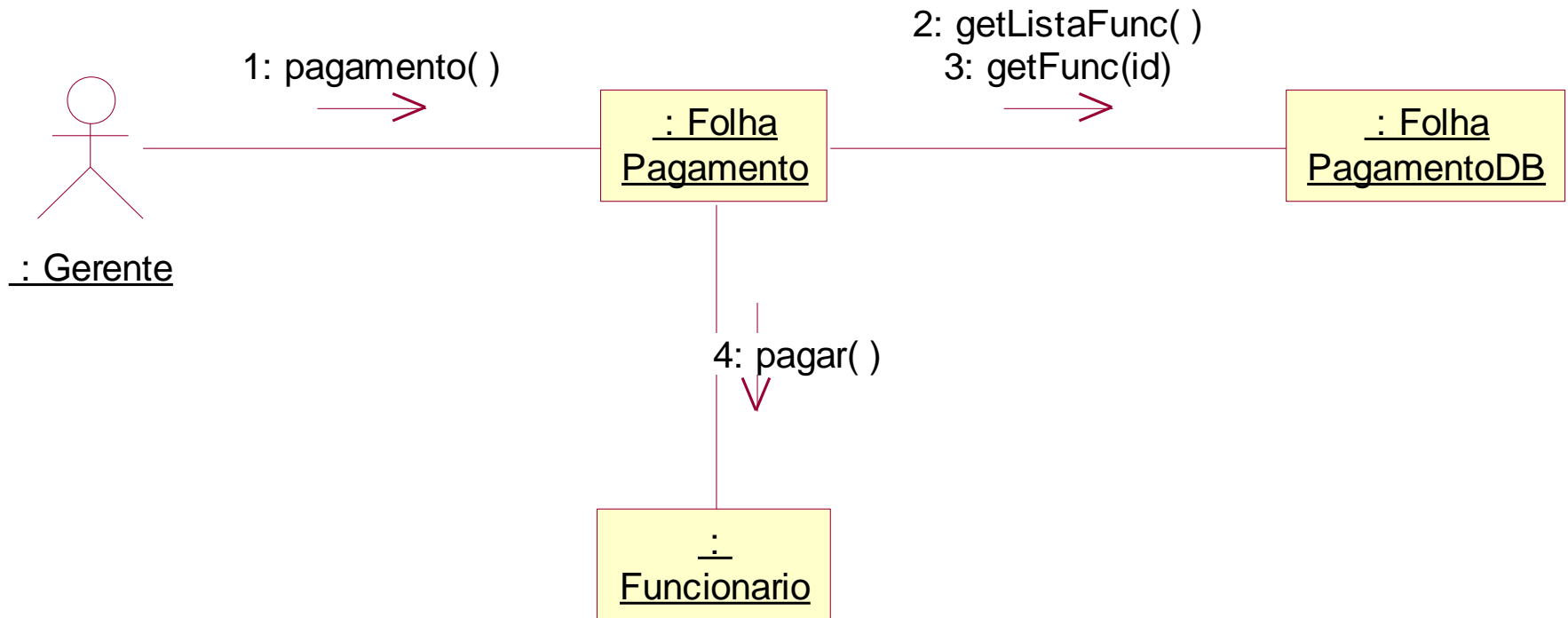




Diagrama de Atividades

UML

Diagrama de Atividades

- O diagrama de atividades modelam também os aspectos dinâmicos do sistema;
- É essencialmente um gráfico de fluxo, representando o controle de atividade para outra;
- Mostra a execução de ações pelos objetos, com ênfase na ordem de execução destas ações. É particularmente útil para modelar sistemas cuja visão atual ou a implementação futura são focados numa estrutura procedural, em detrimento dos aspectos orientados a objetos do sistema.

UML

Diagrama de Atividades

- Enquanto os diagramas de interação dão ênfase no fluxo de controle de um objeto para outro, os diagramas de atividades dão ênfase ao fluxo de controle de uma atividade para outra.
- Uma atividade é uma execução não-atômica em andamento em uma máquina de estados. As atividades acabam resultando em alguma ação, formada pelas computações atômicas executáveis que resultam em uma mudança de estado do sistema ou o retorno de um valor.

UML

Diagrama de Atividades

- Os diagramas de atividades não são importantes somente para a modelagem de aspectos dinâmicos de um sistema, mas também para a construção de sistemas executáveis por meio de engenharia de produção e reversa.

UML

Diagrama de Atividades

- Conteúdo dos diagramas de atividades:
 - Estados de ação:
 - Computações atômicas executáveis que podem chamar uma operação em um objeto, enviar um sinal a um objeto ou até criar ou destruir um objeto.
 - Estados de atividade:
 - Podem ser decompostos e suas atividades podem ser representadas por outros diagramas de atividades. Estes estados são não-atômicos e podem ser interrompidos, e em geral levam algum tempo para serem completados.
 - Transição:
 - Quando a ação ou atividade termina, o fluxo de controle passa imediatamente ao estado seguinte de ação ou atividade, representando a transição.
 - Objetos

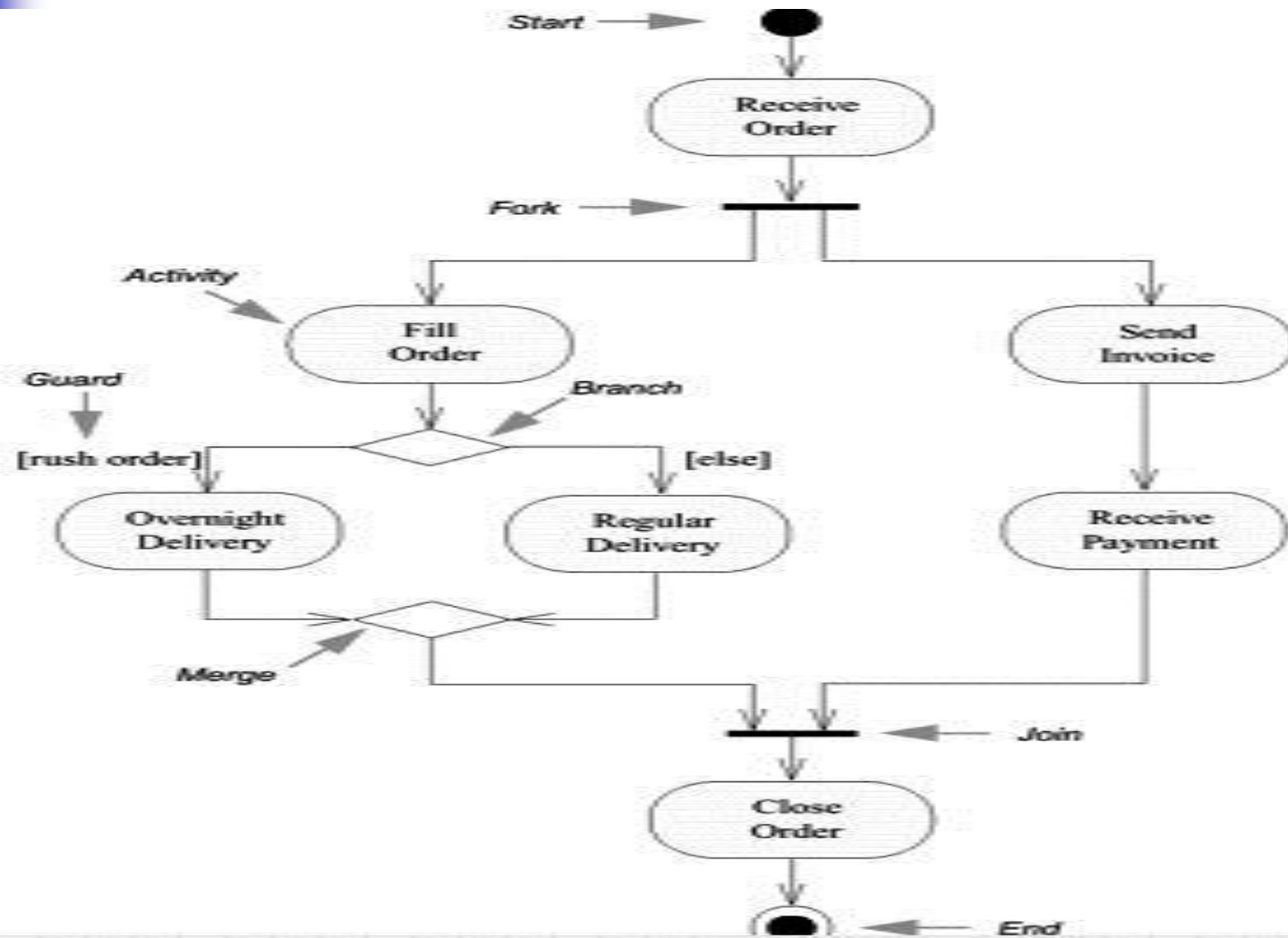
UML

Diagrama de Atividades

- **Uso do diagrama de atividades:**
 - Representar que serão executados quando uma operação (ação) é disparada; (uso mais comum)
 - Representar o trabalho interno de um objeto;
 - Mostrar como um grupo de ações relacionadas podem ser executadas, e como elas vão afetar os objetos em torno delas;
 - Mostrar como uma instância pode ser executada em termos de ações e objetos;
 - Mostrar como um negócio funciona em termos de trabalhadores (atores), fluxo de trabalho, organização, e objetos (fatores físicos e intelectuais usados no negócio).

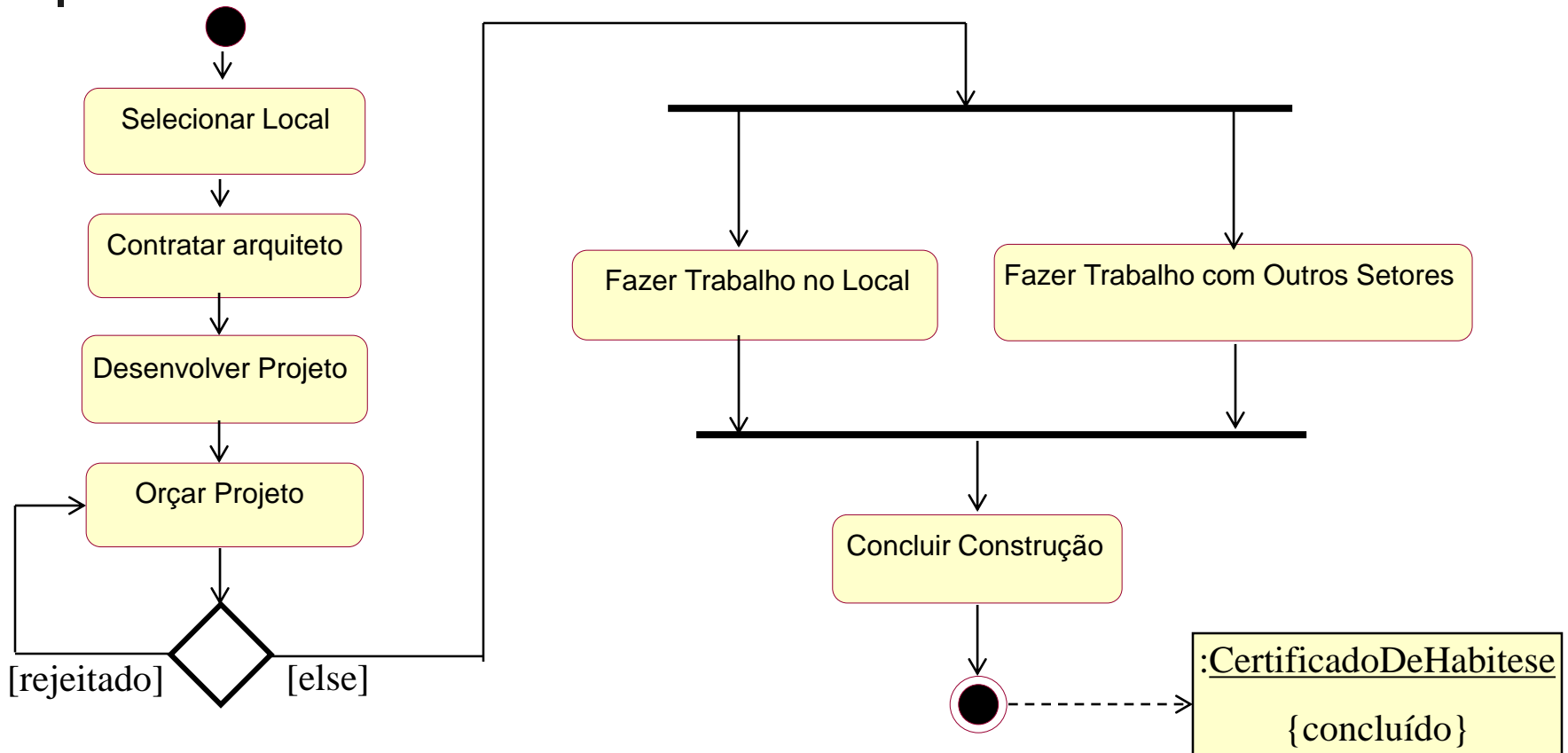
UML - Diagrama de Atividades

Exemplo



UML - Diagrama de Atividades

Exemplo



UML - Diagrama de Atividades

Exemplo

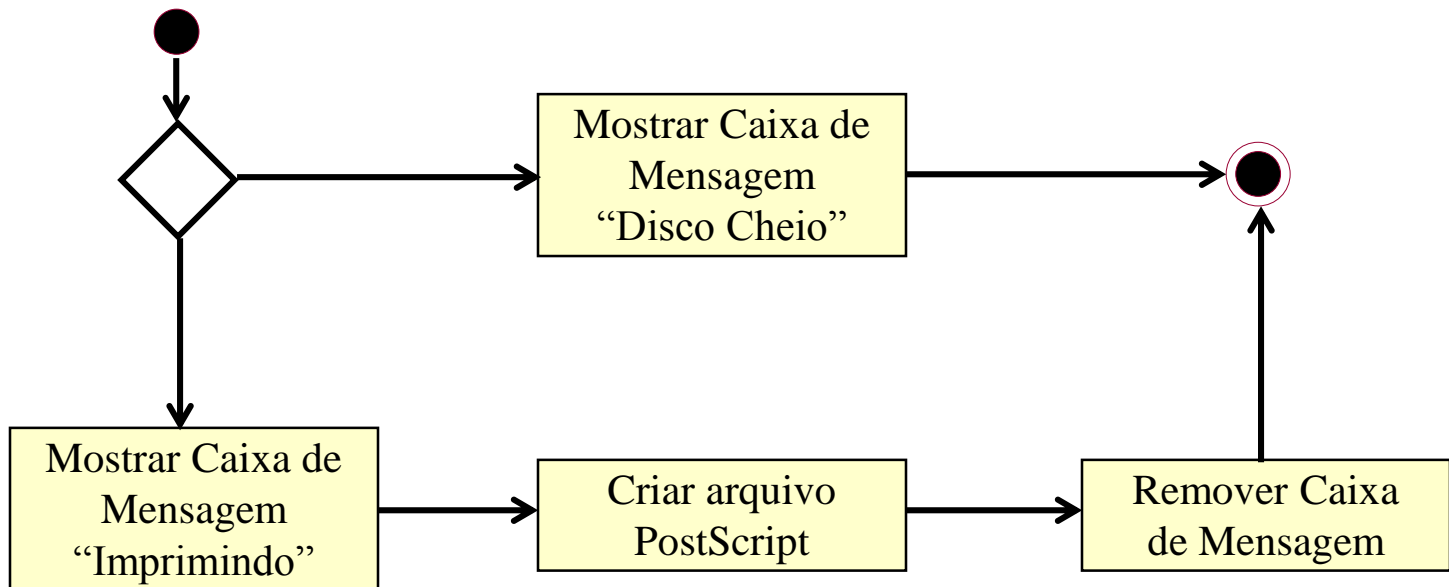




Diagrama de Componentes

UML

Diagrama de Componentes

- O objetivo do diagrama de componentes é mostrar quais são os elementos físicos da aplicação e as relações entre estes componentes, representando como o software deverá ser gerado;
- Apresentam o sistema por um lado funcional, expondo as relações entre os componentes e a organização de seus módulos durante sua execução.

UML

Diagrama de Componentes

- Os componentes são a implementação física dos conceitos e das funcionalidades definidos na arquitetura lógica;
- A UML apresenta uma forma padrão de representação dos componentes. Entretanto pode-se utilizar imagens ou representações específicas para documentos, fontes e imagens (ver "UML-Guia do Usuário").

UML

Diagrama de Componentes

- Um diagrama de componentes contém elementos que representam:
 - Pacotes (packages) de componentes
 - Componentes ou módulos
 - Programa principal
 - Subprogramas
 - Tarefas
 - Dependências
 - DLL 's

UML - Diagrama de Componentes

Exemplo

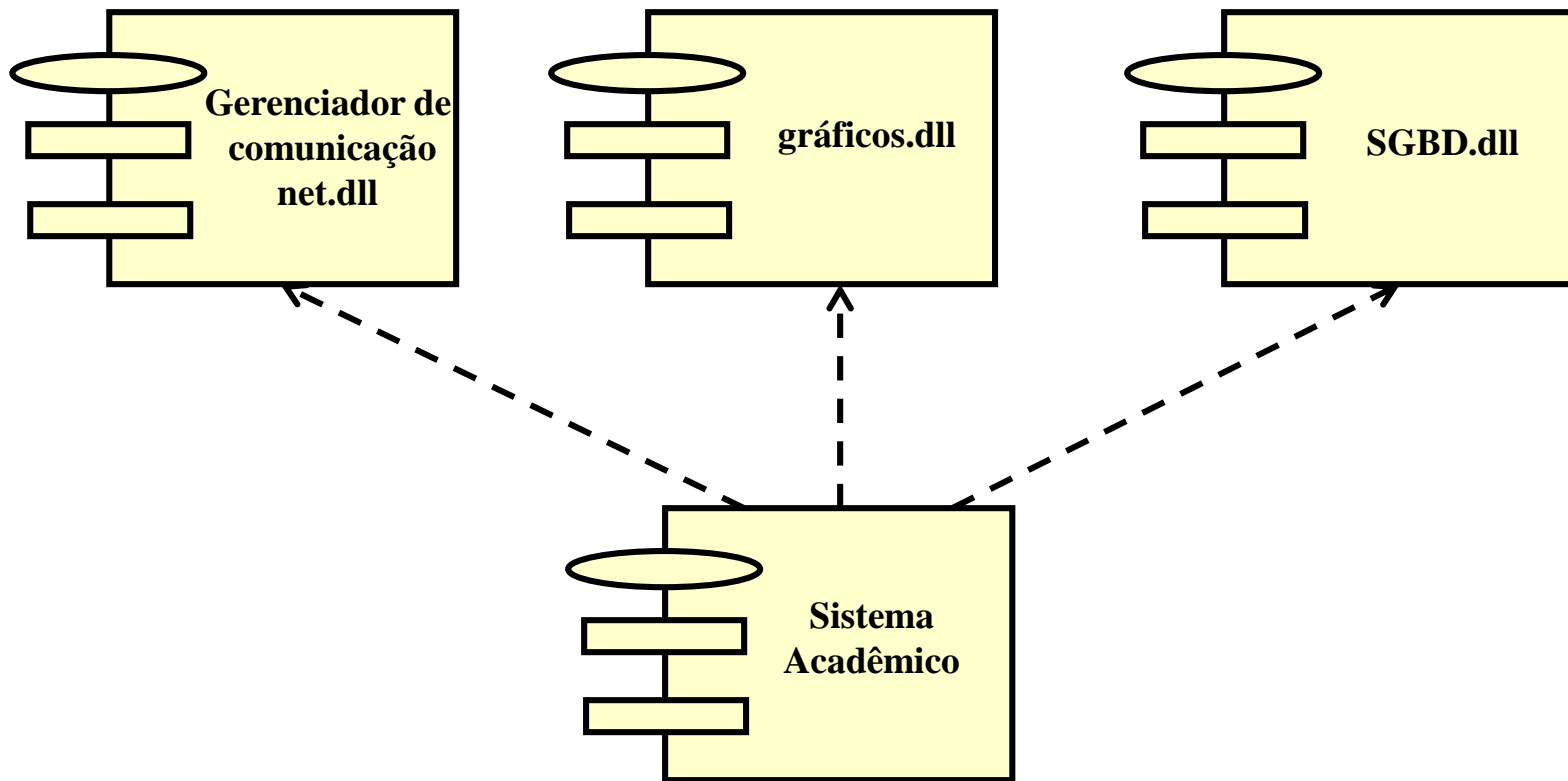




Diagrama de Implantação/Distribuição

UML

Diagrama de Implantação

- Também denominados "diagramas de utilização", são utilizados para descrever a arquitetura física do hardware e do software desejada para o sistema;
- Apresenta, dentro do ambiente de negócios, todos os computadores e periféricos, juntamente com as conexões entre eles;
- Demonstra a arquitetura de execução dos processadores, componentes físicos, e de software que rodam no ambiente que o sistema será implantando.

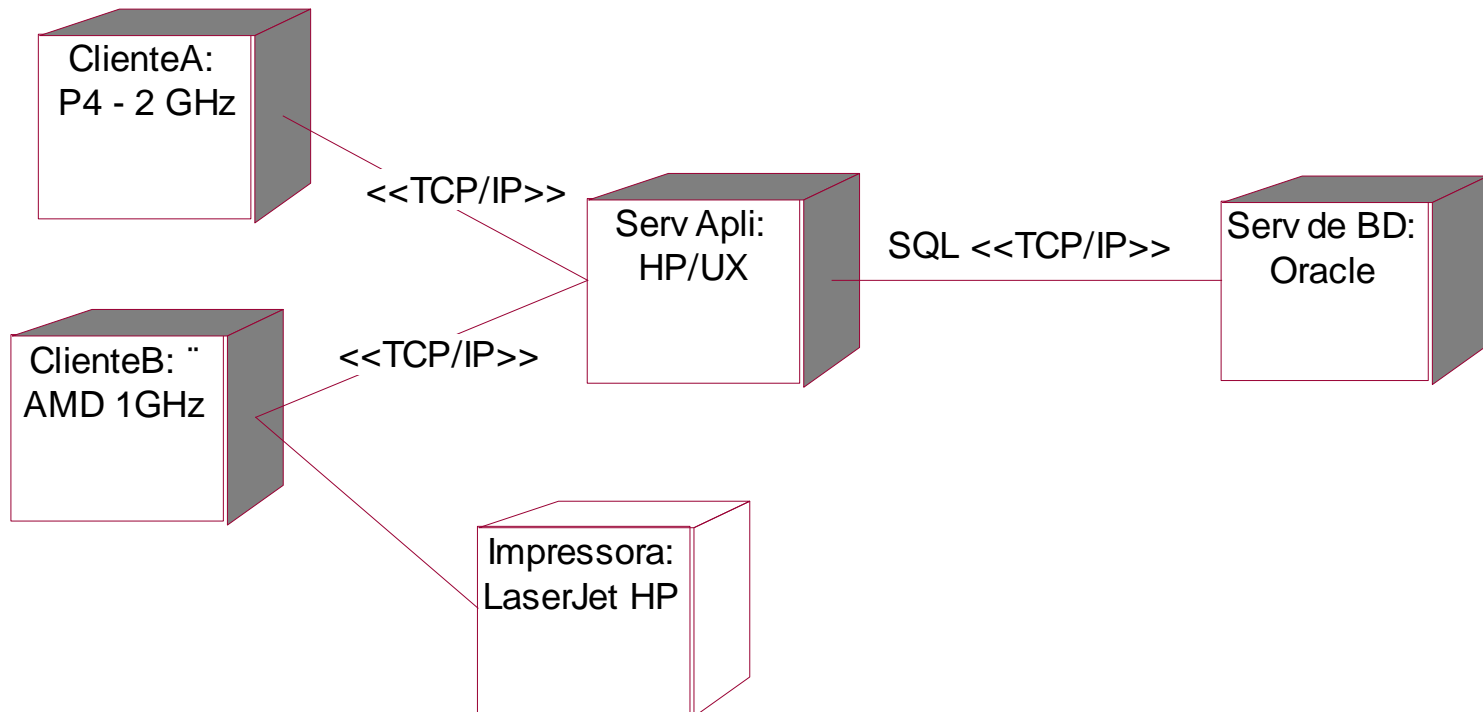
UML

Diagrama de Implantação

- O diagrama inclui os seguintes elementos:
 - **Nós:** É um elemento físico responsável pelo processamento ou transporte ou processamento de informações que contém um ou mais componentes do sistema. São nós os servidores, terminais-clientes, roteadores, um *backbone* etc. Os nós podem ser agrupados em pacotes para fins de organização dos modelos;
 - **Conexões:** ligam os nós. Podem ser documentadas apropriadamente para descrever a natureza da conexão (tipo de protocolo, velocidade, natureza do meio físico que une os nós etc);

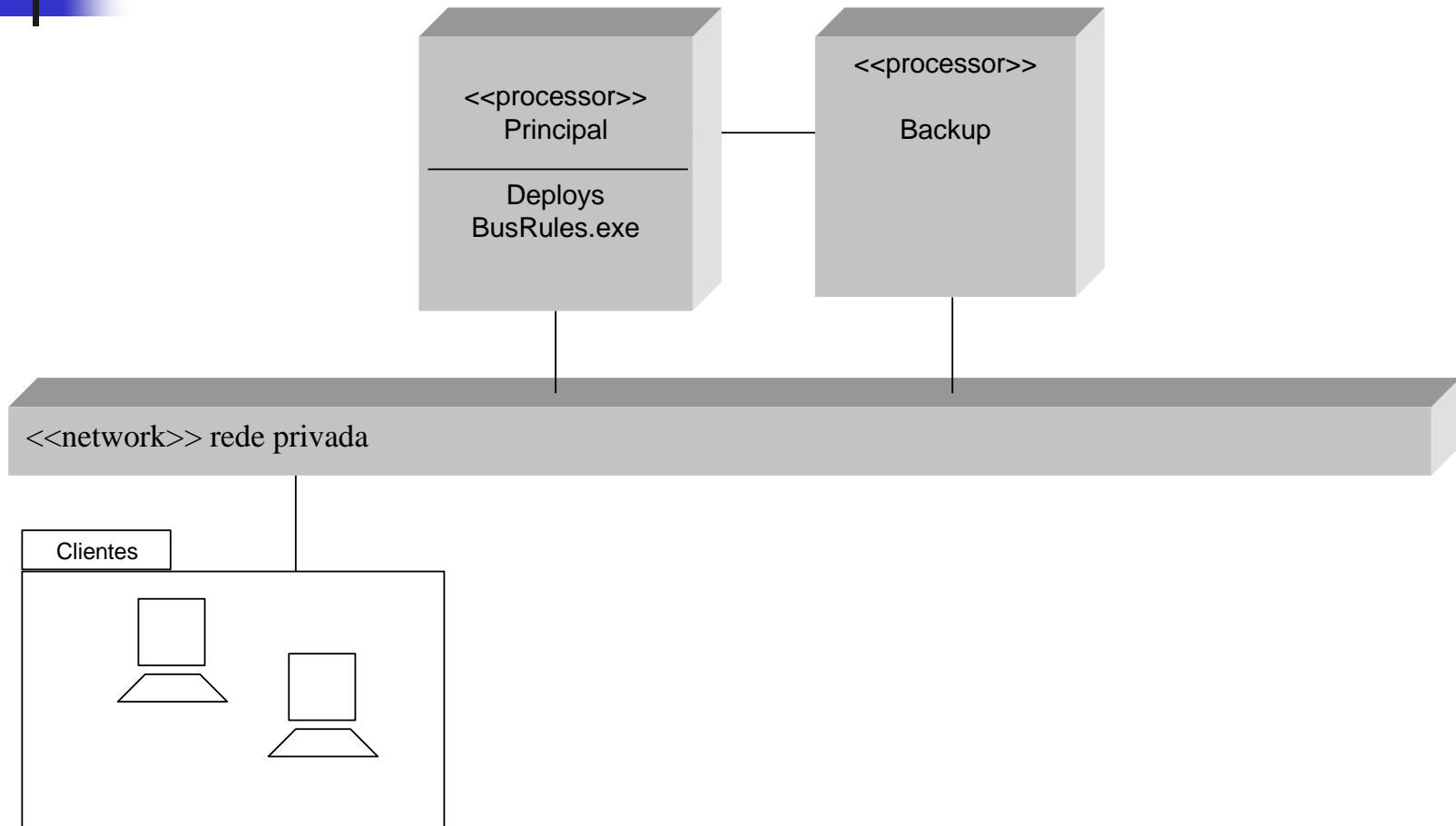
UML - Diagrama de Implantação

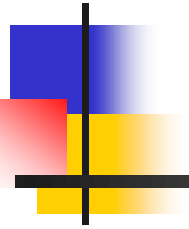
Exemplo



UML - Diagrama de Implantação

Exemplo





Estudo de Caso

Controle de reserva e locação de quartos de hotel

UML - Estudo de Caso

Controle de Hotel

- Escopo do Sistema de Controle de Reserva e Locação de Quartos de Hotel
 - O Sistema de Hotel serve para automatizar o processo de reserva e locação de quartos para clientes. O sistema deve manter os dados dos clientes que reservaram quartos e se hospedaram no hotel. Deverá também controlar os quartos reservados e locados, juntamente com o registro de entrada e saída de hóspedes do hotel. No registro da saída do hóspede deverá ser cobrada a estadia e este valor deverá ficar armazenado no sistema. O sistema deverá controlar também todos os funcionários que trabalham no hotel.

UML - Estudo de Caso

Controle de Hotel

- Informações relevantes:
 - O cliente telefona ou vêm ao hotel e pede para reservar um quarto; o funcionário verifica se existe quarto disponível no período solicitado. Caso afirmativo, é feita a reserva do quarto. Caso negativo, é informado ao cliente a não disponibilidade do quarto. O cliente também poderá optar por fazer uma reserva via WEB, contemplando o uso de internet no hotel;
 - Caso o cliente não mais desejar o quarto reservado, o funcionário providenciará o cancelamento da reserva, disponibilizando o novamente o quarto. O cliente também poderá realizar esta operação pela internet.

UML - Estudo de Caso

Controle de Hotel

- Informações relevantes:
 - Quando o cliente não comparecer ao hotel para hospedar-se até as 12:00 horas no dia da reserva, ela deverá ser cancelada, disponibilizando novamente o quarto;
 - Quando o cliente ocupar um quarto reservado previamente, o funcionário faz o registro do cliente. Caso o quarto não esteja reservado, uma mensagem de rejeição da ocupação será emitida. Caso contrário, um pacote com informações úteis e a confirmação serão fornecidos ao cliente;
 - Quando o cliente deixar o hotel, notificando sua saída, será fornecido a conta, e o quarto será disponibilizado para limpeza;

UML - Estudo de Caso

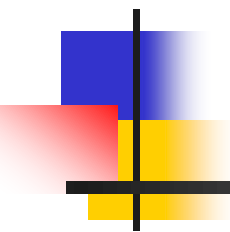
Controle de Hotel

- Informações relevantes:
 - O cliente pode pagar a conta à vista ou usar o cartão de crédito. Pode-se também, no caso de reserva feitas por empresas, emitir uma nota de cobrança contra a empresa;
 - Após uma ocupação, um quarto sofrerá o reabastecimento e limpeza, somente após este fato é que o funcionário o torna disponível para nova locação.

UML - Estudo de Caso

Controle de Hotel

- Roteiro de Tarefas do estudo de caso:
 - Definir Casos de Uso do sistema, com suas descrições, e definir atores que interagem com o caso de uso
 - Gerar os cartões CRC (desenvolvido em papel)
 - Desenvolver o modelo conceitual do sistema
 - Implementar o diagrama de caso de uso
 - Analisar e desenvolver os digramas de estado de objetos
 - Desenvolver os diagramas de Interação (seqüência e colaboração)
 - Desenvolver o diagrama de classe
 - Desenvolver o diagrama de componentes e suas dependências
 - Desenvolver o diagrama de Implantação física do sistema



Discussão em sala da Análise – Controle de um Hotel

UML - Estudo de Caso

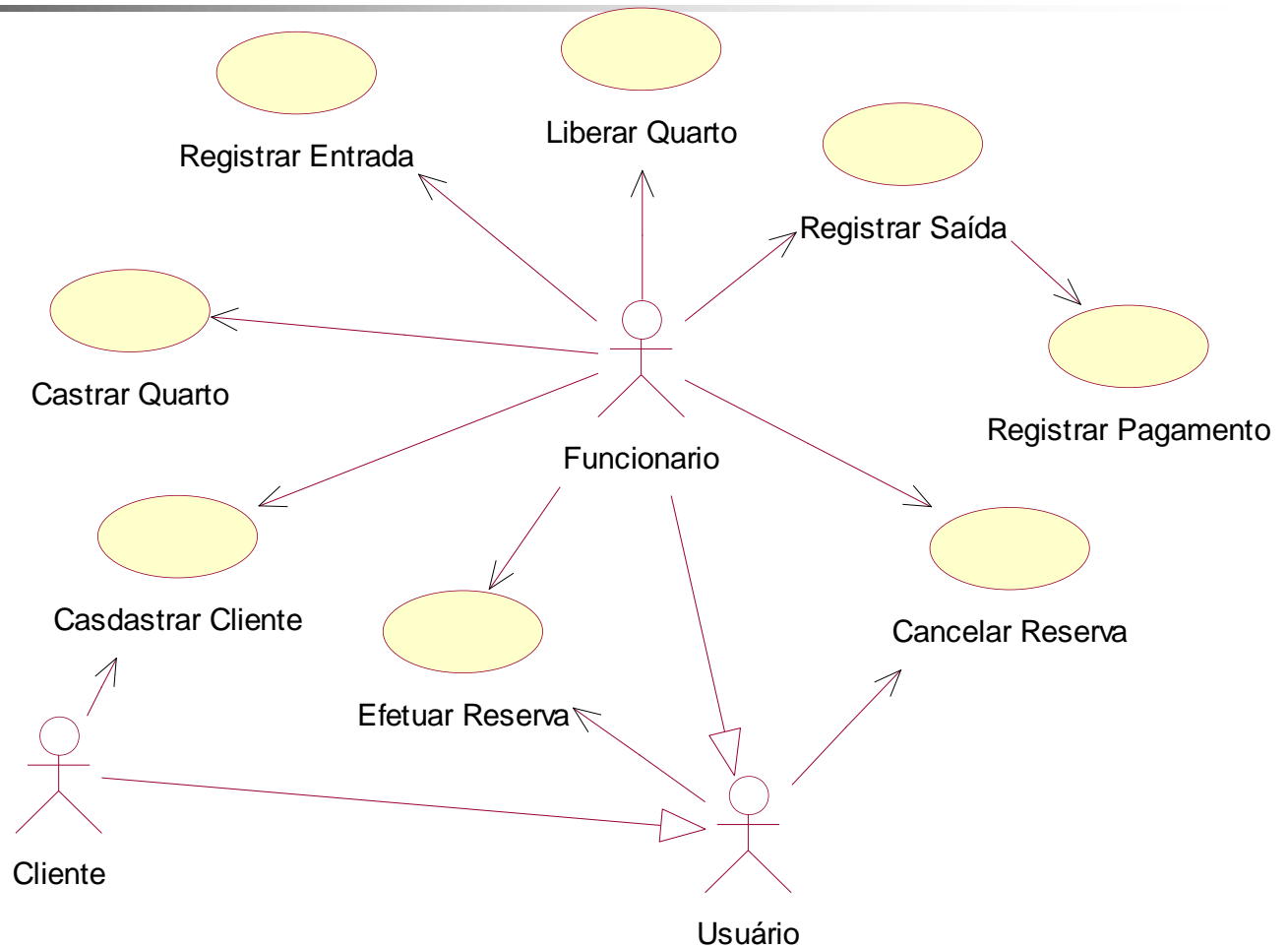
Controle de Hotel

- Escopo do Sistema de Controle de Reserva e Locação de Quartos de Hotel
 - O Sistema de Hotel serve para automatizar o processo de reserva e locação de quartos para clientes. O sistema deve manter os dados dos clientes que reservaram quartos e se hospedaram no hotel. Deverá também controlar os quartos reservados e locados, juntamente com o registro de entrada e saída de hóspedes do hotel. No registro da saída do hóspede deverá ser cobrada a estadia e este valor deverá ficar armazenado no sistema. O sistema deverá controlar também todos os funcionários que trabalham no hotel.

UML - Estudo de Caso

Controle de Hotel

Diagrama de Caso de Uso

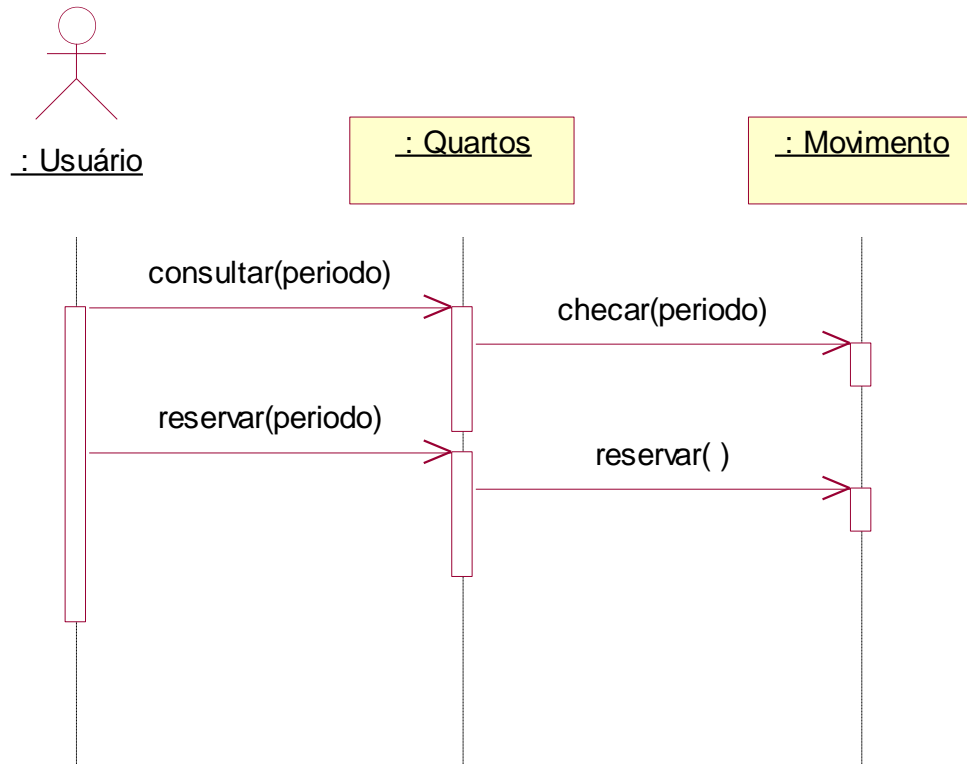


UML - Estudo de Caso

Controle de Hotel

Efetuar Reserva

- Quando o Cliente telefona, ou vem até o Hotel, e pede para reservar um quarto, o funcionario executa um procedimento padrão que registra a ocupação futura do quarto dentro de um limite de datas de entrada e saída previstos. Este registro deve tornar indisponível, para qualquer operação, o referido quarto, dentro do limite de datas informado. O cliente poderá efetuar a reserva via internet.

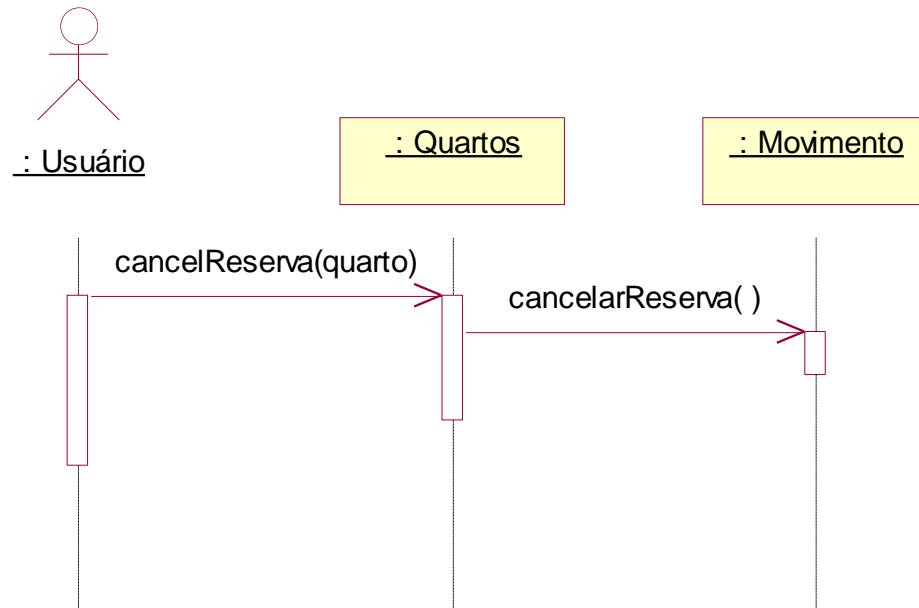


UML - Estudo de Caso

Controle de Hotel

Cancelar Reserva

- Quando o cliente não mais desejar o quarto reservado e comunicar o fato, será cancelada a reserva, disponibilizando o quarto novamente. Deverá ser informado os motivos do cancelamento mantendo um histórico do cliente. Este procedimento pode ser realizado pela internet. Quando o cliente não comparecer ao hotel para hospedar-se até às 12:00 do dia da reserva deve-se proceder o cancelamento da reserva anotando o motivo também.

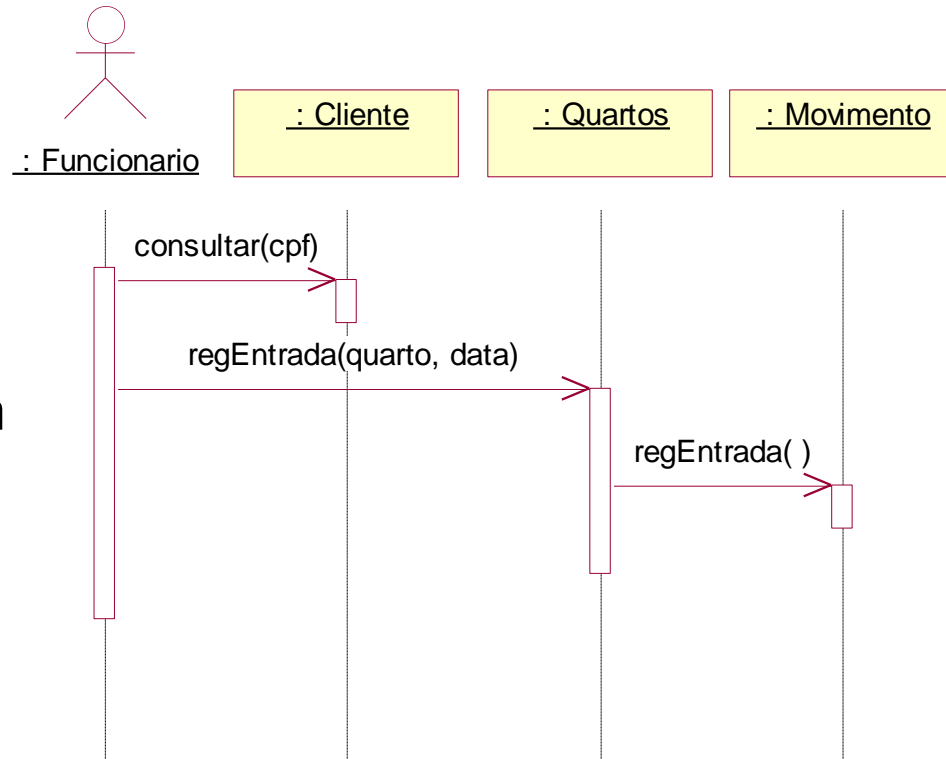


UML - Estudo de Caso

Controle de Hotel

Registrar Entrada

- O cliente faz o registro para a ocupação do quarto reservado previamente. O funcionário confere a existência da reserva para o requerido cliente e registra a entrada, indisponibilizando o quarto. Caso não exista uma reserva, uma mensagem de rejeição será emitida. Se a reserva existir um pacote de informações será emitida para o cliente.

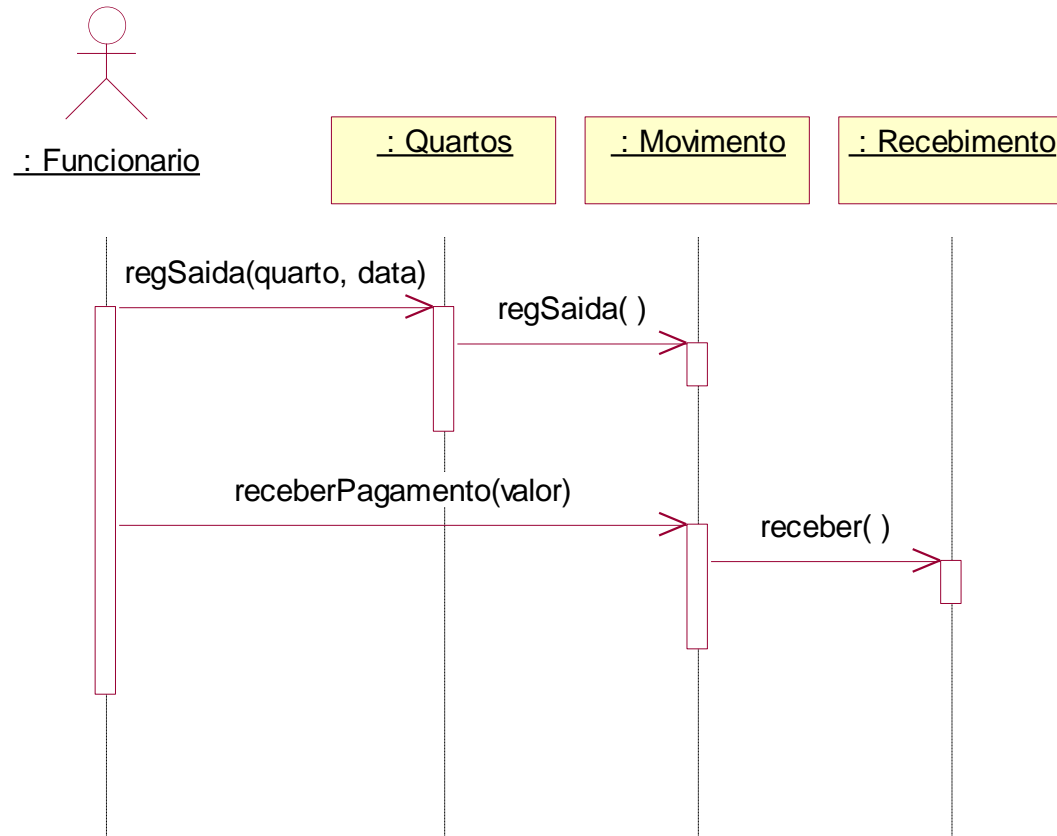


UML - Estudo de Caso

Controle de Hotel

Registrar Saída

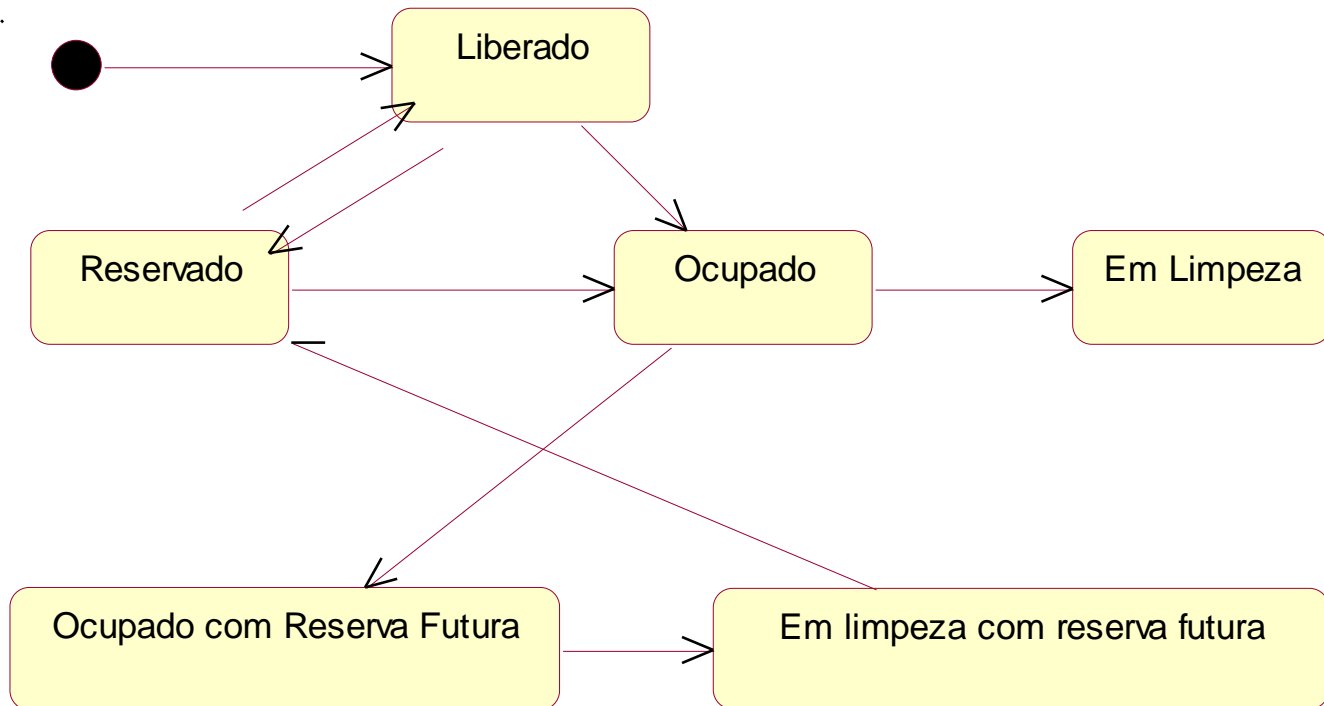
- Quando o cliente solicitar a saída do hotel, através das informações repassadas pelo próprio cliente será providenciado o fechamento da sua conta, e ao mesmo tempo será disponibilizado o quarto para a limpeza. O fechamento do quarto deverá gerar uma conta a receber referente a estadia fechada.



UML - Estudo de Caso

Controle de Hotel

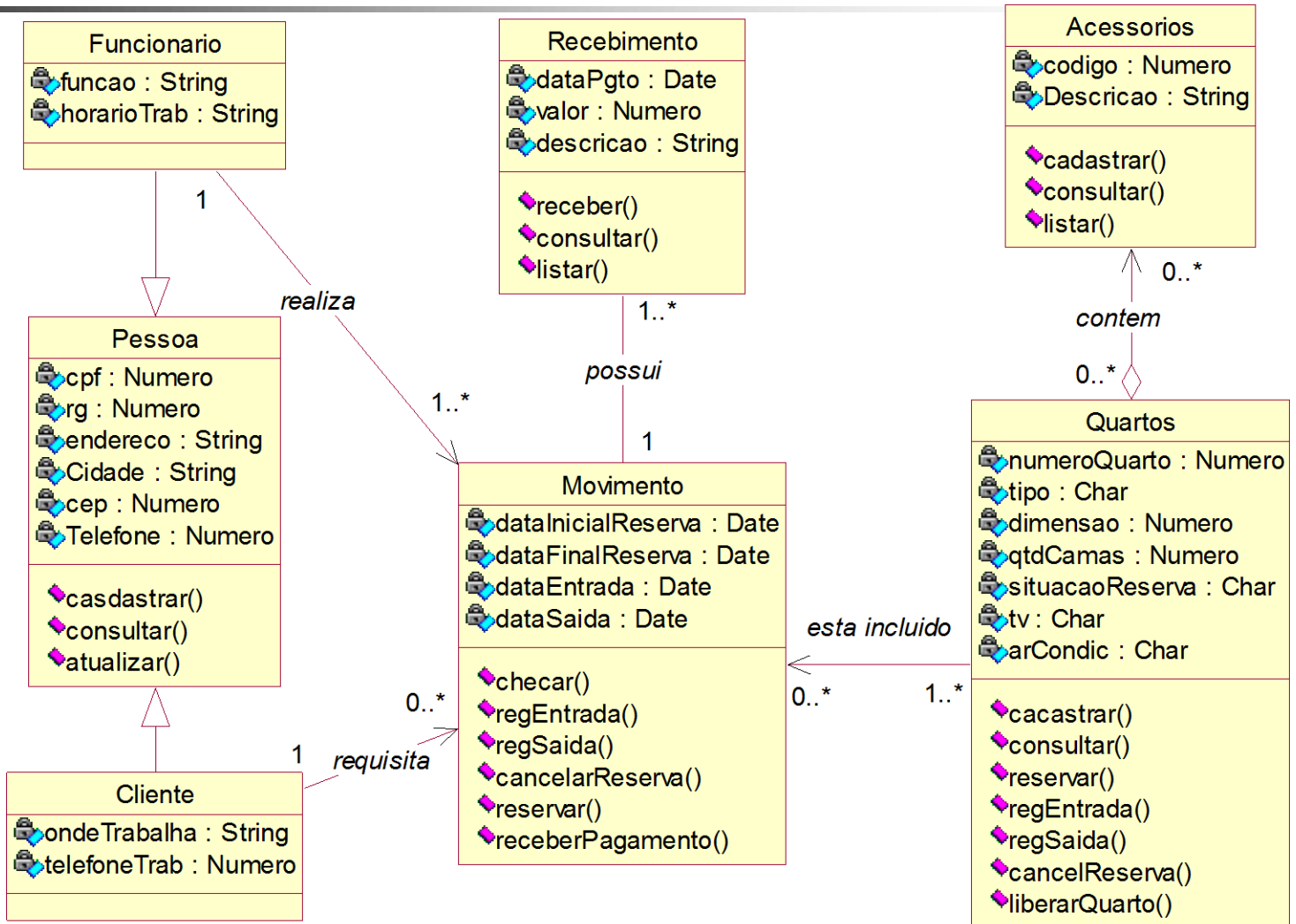
Diagrama de Estado – Objeto Quarto



UML - Estudo de Caso

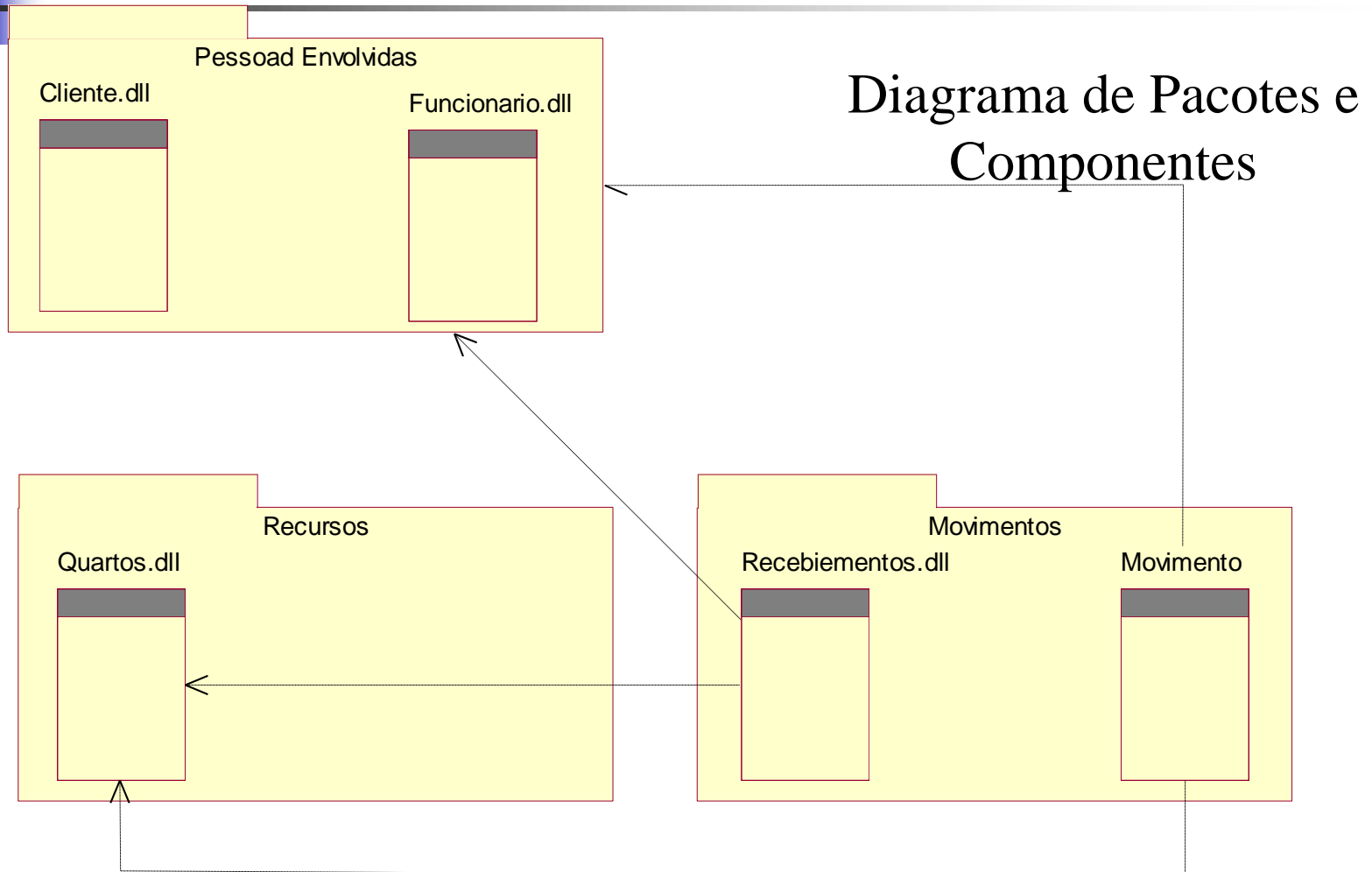
Controle de Hotel

Diagrama de Classes



UML - Estudo de Caso

Controle de Hotel



UML - Estudo de Caso

Controle de Hotel

Diagrama de
Implantação e
Distribuição

