

UML

Estrutura básica de um projeto

ROTEIRO:

- 1) Escopo (Visão Geral do Sistema)
- 2) Regras de Negócio (RN)
- 3) Requisitos
 - 3.1) Requisitos Funcionais (RF)
 - 3.2) Requisitos Não Funcionais (RNF)
 - 3.3) Requisitos de Interface com o Usuário (RIU)
 - 3.4) Mensagens do Sistema
- 4) Visões USC: Modelo de Casos de Uso (USC)
 - 4.1) Lista de Casos de uso
 - 4.2) Relação de Atores
 - 4.3) Lista de exceções
 - 4.4) Diagrama de USC
 - 4.5) Detalhamento de Caso de Uso
- 5) Definição da Especificação suplementar

Diagrama de CASOS DE USO (USC)

- b) Modela o problema a ser solucionado.
- c) descreve um cenário - funcionalidades do Stma (ponto de vista do usuário).
- d) O cliente deve ver no diagrama de Use Cases as principais funcionalidades de seu sistema.
- e) Feita a partir de várias discussões entre as pessoas envolvidas com o sistema: Clientes, Usuários, Desenvolvedores

Objetivo

- a) objetivo de auxiliar a comunicação entre os analistas e o cliente.
- b) Visão funcional de tudo que o SW deverá fazer
- c) base para todo o processo de desenvolvimento
 - ✓ Deverá ser aplicado para os testes de validação:
 - ✓ Facilitar a transformação dos requisitos funcionais em classes e operações reais do SW

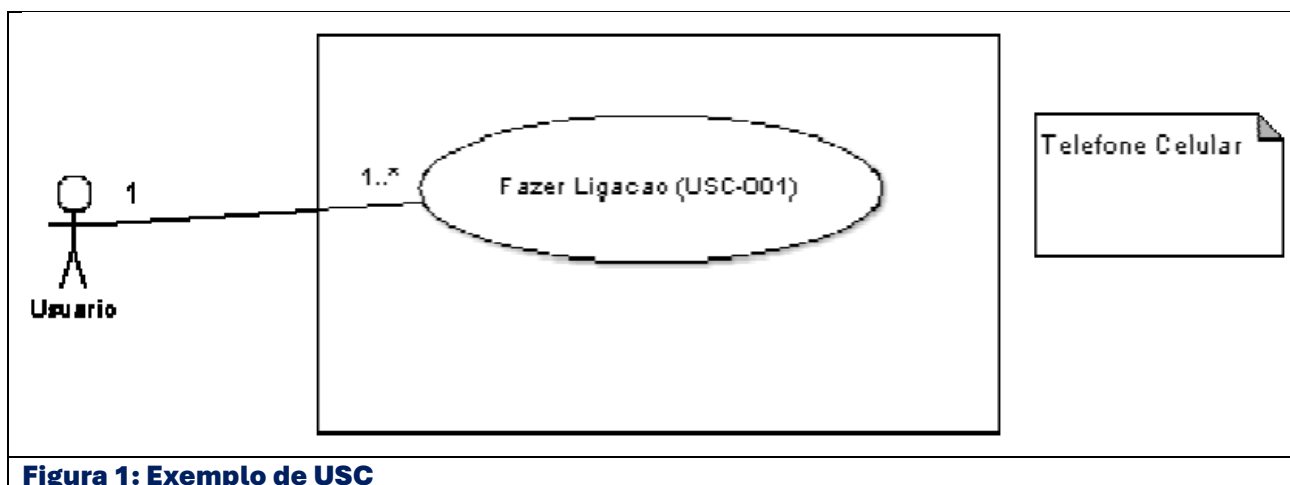


Figura 1: Exemplo de USC

A descrição do modelo deve ser mantida no nível mais simples possível



Figura 2: Exemplo de uma má documentação dos casos de uso

O sistema parece uma caixa preta oferecem funcionalidades

- ☐ Contexto de como será o funcionamento do Stma, sem se preocupar com a implementação do mesmo

Notação:

- 1) Atores;
 - 2) Use case;
 - 3) Relacionamento entre elementos;
 - a) Associação entre atores e casos de uso;
 - b) Generalização entre atores;
 - c) Generalizações entre casos de uso;
- ✓ Extends e Includes

1) Ator

- ☐ é representado por um boneco e um rótulo com o nome do ator.
- ☐ é um usuário do Stma, que pode ser um usuário humano ou um outro Stma computacional.

Formas de representar os atores e relacionamento

- ☐ é representado por um boneco e um rótulo com o nome do ator.
- ☐ é um usuário do Stma, que pode ser um usuário humano ou um outro Stma computacional.

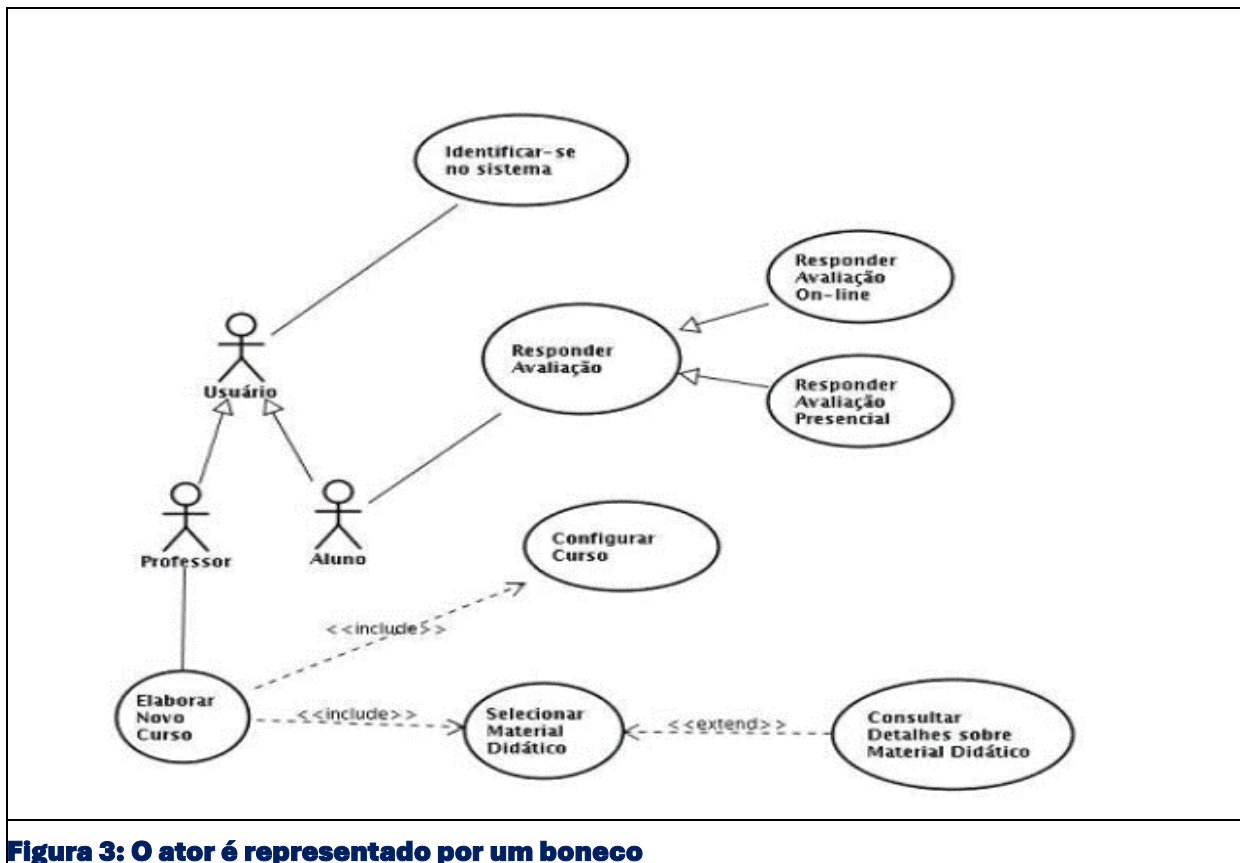


Figura 3: O ator é representado por um boneco

Categorias de atores

- Pessoas: Usuário, secretária, aluno, professor, administrador, empregado, Cliente, Gerente, Almoxarife, Vendedor, etc;.
- Dispositivos: impressoras, máquina ou equipamentos ,atuadores/sensores, etc.
- HW: (Leitora de Código de Barras, Sensor, etc.)
- organizações (Empresa Fornecedora, Agência de Impostos, Administradora de Cartões, etc);
- outros sistemas (Sistema de Cobrança, Sistema de Estoque de Produtos, etc).
- Eventos Externos: Controladores de tempo - Cron

Formas de identificar os atores

- Quem utilizará a funcionalidade principal do Stma?
- Quem precisará de suporte do Stma para fazer suas tarefas diárias ?
- Quem necessita administrar e manter o Stma funcionando ?
- Quais dispositivos de HW o Stma precisará manipular ?
- Com que outros Stmas o Stma precisará interagir ?
- Quem tem interesse nos resultados que o Stma irá produzir ?

3.a) Associação entre atores e casos de uso;

Define uma funcionalidade do Stma do ponto de vista do usuário.

- Um *use case* é representado por uma elipse e um rótulo com o nome do use case.
- Um *use case* é uma funcionalidade do Stma

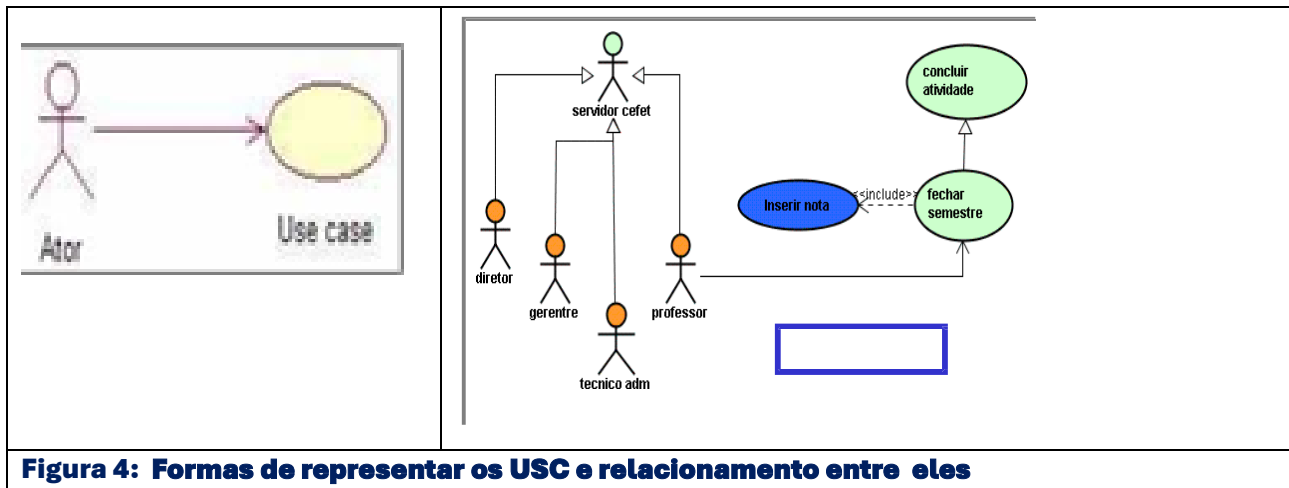


Figura 4: Formas de representar os USC e relacionamento entre eles

3.b) Generalização entre atores;

- ☐ Os use cases de B são também use cases de A
- ☐ A tem seus próprios use cases

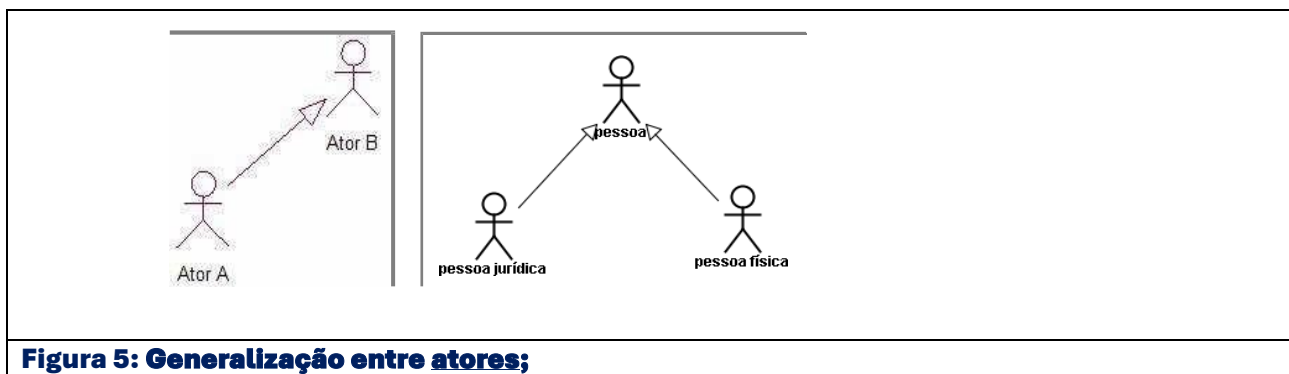


Figura 5: Generalização entre atores;

3) Relacionamentos:

- a) Ajudam a descrever os use cases
- b) Entre um ator e um use case
- c) Define uma funcionalidade do Stma do ponto de vista do usuário.

Formas de Relacionamentos

- a) Associação entre atores e casos de uso;
- b) Generalização entre atores;
- c) Generalizações entre casos de uso;
Extends e Includes

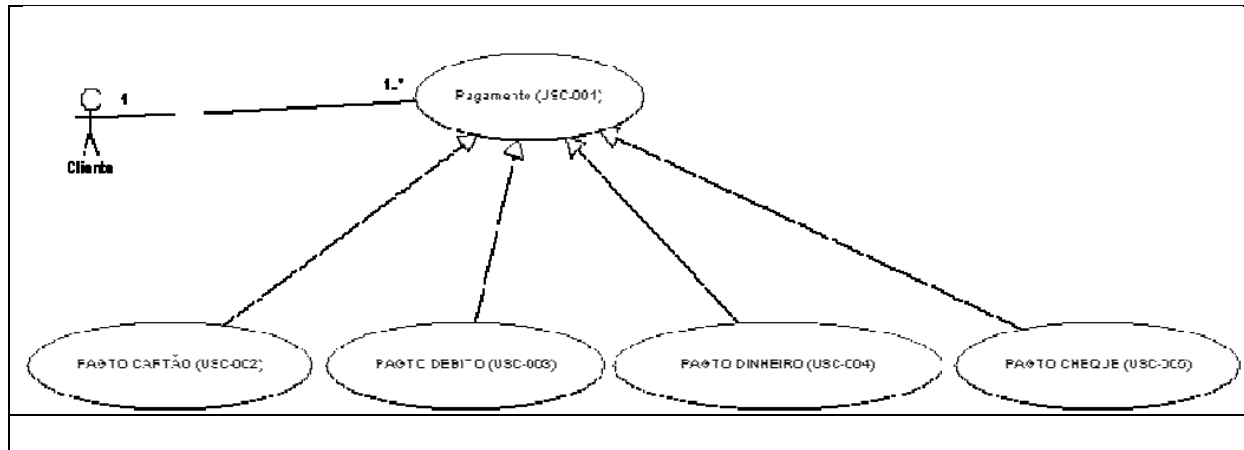
3.c) Generalizações entre casos de uso;

☐ Especialização / Generalização

- ✓ Generalização ou Especialização (é um) Use case B é um use case A (A é uma generalização de B, ou B é uma especialização de A).
- ✓ Um relacionamento entre um use case genérico para um mais específico, que herda todas as características de seu pai.
- ✓ é representada por uma seta que une ao Caso de Uso Geral (para onde a seta aponta)

3.c) Generalizações entre casos de uso;

- ✓ É uma forma de Associação entre casos de uso na qual existe dois ou mais casos de uso com muitas características semelhantes, apresentando contudo algumas diferenças importantes entre si.



3.c.1) Inclusão (<<Include>>):

- ✓ Relacionamento de inclusão indicam uma obrigatoriedade
- ✓ Utilizada quando existe um serviço, situação ou rotina comum a mais de um USC
- ✓ Essa rotina é colocada em um USC específico para que todos outros USC utilizem-se desse serviço
- ✓ Evitando-se descrever uma mesma sequência de USC
- ✓ A execução do 1ro USC obriga a execução do 2do
- ✓ Pode ser comparado à chamada de uma sub- rotina ou função

PARTE 01 - TEORIA

Na documentação oficial da UML, NÃO existe uma forma predefinida de se escrever um caso de uso. Entretanto, existem boas práticas que devem ser seguidas, das quais podemos citar:

- 1) O texto do caso de uso deve ser livre da forma como será implementado, exceto se for um caso de uso de projeto. Sendo assim não diga que o usuário irá selecionar uma determinada informação em um listbox ou combobox; ou que clicará num botão. Diga, apenas, que será selecionado um valor de uma lista; ou que o usuário escolherá uma determinada opção, que pode ser implementada por meio de um botão, de um menu, ou de outra forma.
- 2) Após escrever sua primeira versão de caso de uso, volte e analise-o. Verifique se alguma parte de seu caso de uso é reaproveitável em outro caso de uso. Se existir, devemos separá-la em outro caso de uso e estabelecer um relacionamento de INCLUDE. Verifique se algum cenário alternativo do seu caso de uso está extenso demais. Se existir, devemos separá-lo em outro caso de uso e estabelecer um relacionamento de EXTENDS.
- 3) Um caso de uso pode se transformar numa rotina de um sistema ou apenas em um método de uma classe, dependendo de sua complexidade.

- 4) Para começar a escrever a lista de casos de uso, você pode iniciar pela lista de atores que interagem com a aplicação. Para cada ator você pode questionar:
- qual a influência do ator na aplicação ou
 - o quanto ele é influenciado por ela; em quais ações esse ator participa.

RESUMO DOS PASSOS

- Visão do projeto (brainstorming - tormenta de Idéias)
- Identificação dos atores
- Identificação dos casos de uso
- Pacote de USC

	Comunicação	Extensão	Inclusão	Herança
Caso de uso e caso de uso		X	X	X
Ator e Ator				X
Caso de uso e ator	X			

[Bezerra, 2002]

A tabela a seguir mostra uma comparação mais detalhada entre os três relacionamentos de casos de uso diferentes:

Pergunta	Extensão	Inclusão	Generalização
Qual é a direção do relacionamento?	O caso de uso adicional faz referência ao caso de uso base.	O caso de uso base faz referência ao caso de uso adicional.	O caso de uso adicional (filho) faz referência ao caso de uso base (pai).
O relacionamento tem multiplicidade?	Sim, no lado adicional.	Não. Se você deseja incluir o mesmo segmento de comportamento mais de uma vez, isso deve ser estabelecido no caso de uso base.	Não.
O relacionamento tem uma condição?	Sim.	Não. Se você deseja expressar uma condição na inclusão, é necessário definir isso explicitamente no caso de uso base.	Não.

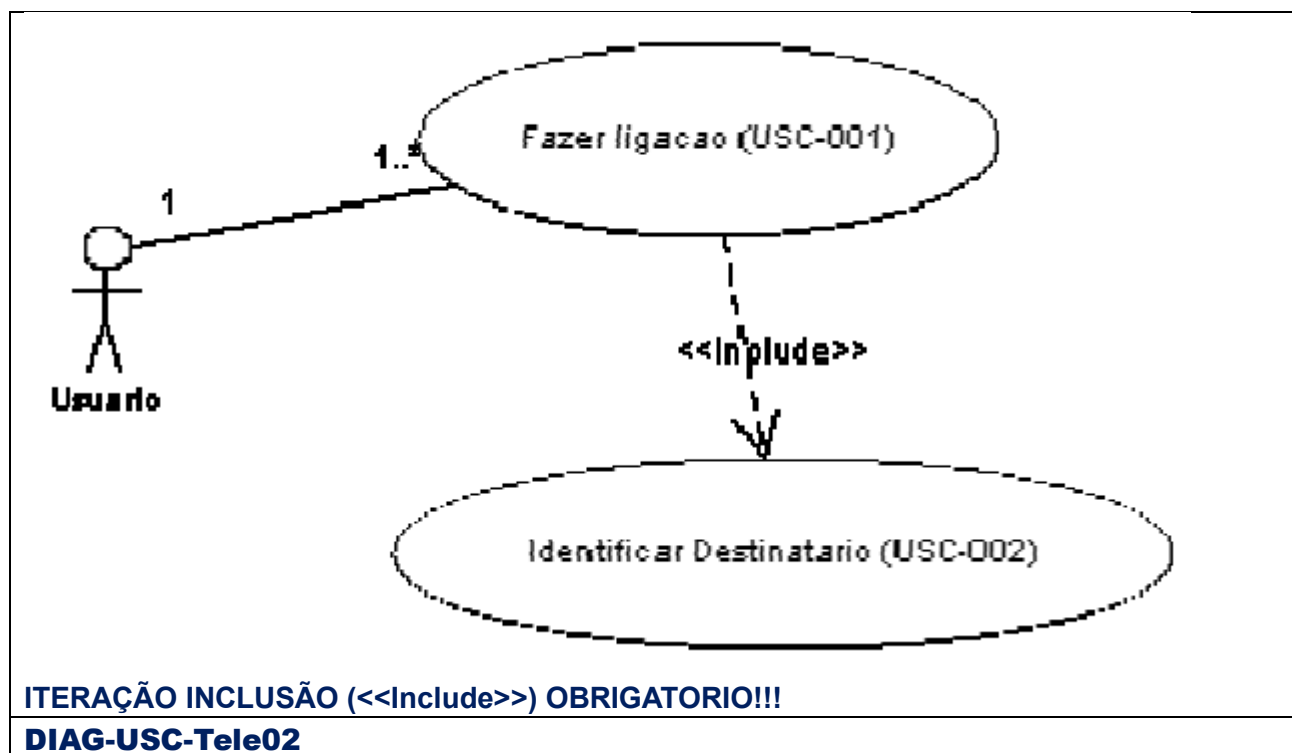
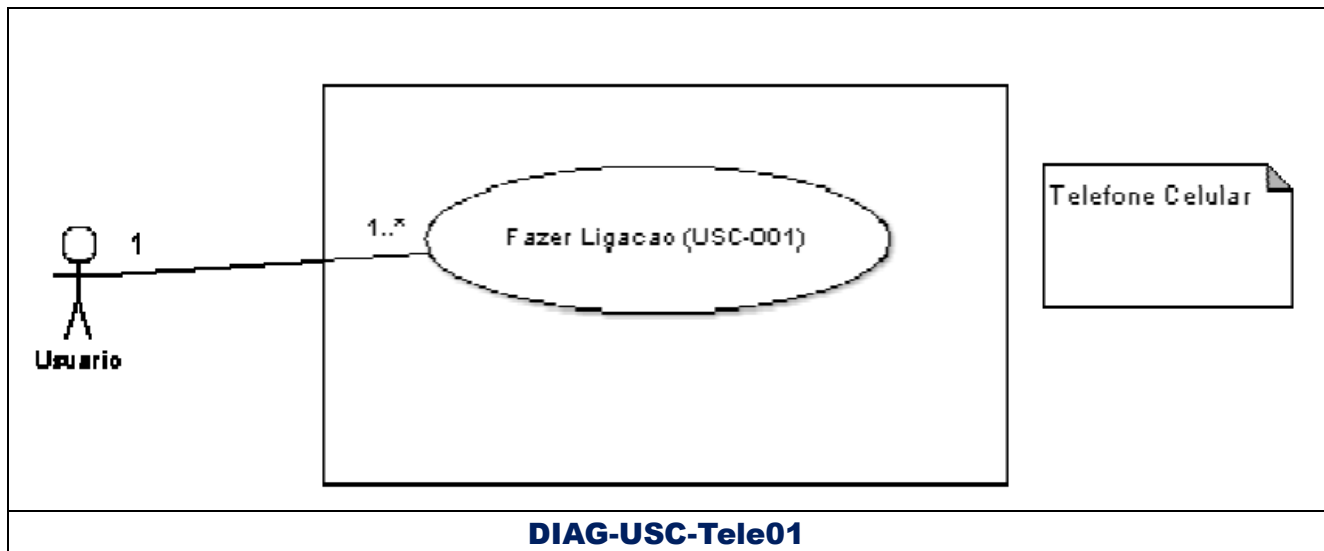
O caso de uso adicional é abstrato?	Normalmente sim, mas não necessariamente.	Sim.	Normalmente não, mas pode ser.
O caso de uso base é modificado pela adição?	A extensão implicitamente modifica o comportamento do caso de uso base.	A inclusão explicitamente modifica o efeito do caso de uso base.	Se o caso de uso base (pai) estiver instanciado, ele não será afetado pelo filho. Para obter os efeitos da adição, o caso de uso adicional (filho) deve estar instanciado.
O caso de uso base deve estar completo e significativo?	Sim.	Junto com as adições, sim.	Se for abstrato, não.
O caso de uso adicional deve estar completo e significativo?	Não.	Não.	Junto com o caso de uso base (pai), sim.
O caso de uso adicional pode acessar os atributos do caso de uso base?	Sim.	Não. A inclusão é encapsulada e só "vê" a si mesma.	Sim, pelos mecanismos normais de herança.
O caso de uso base pode acessar os atributos do caso de uso adicional?	Não. O caso de uso base deve ser bem formado na ausência da adição.	Não. O caso de uso base só sabe sobre o efeito da adição. A adição é encapsulada.	Não. O caso de uso base (pai) nesse sentido deve ser bem formado na ausência da adição (filho).

Exemplos de USC

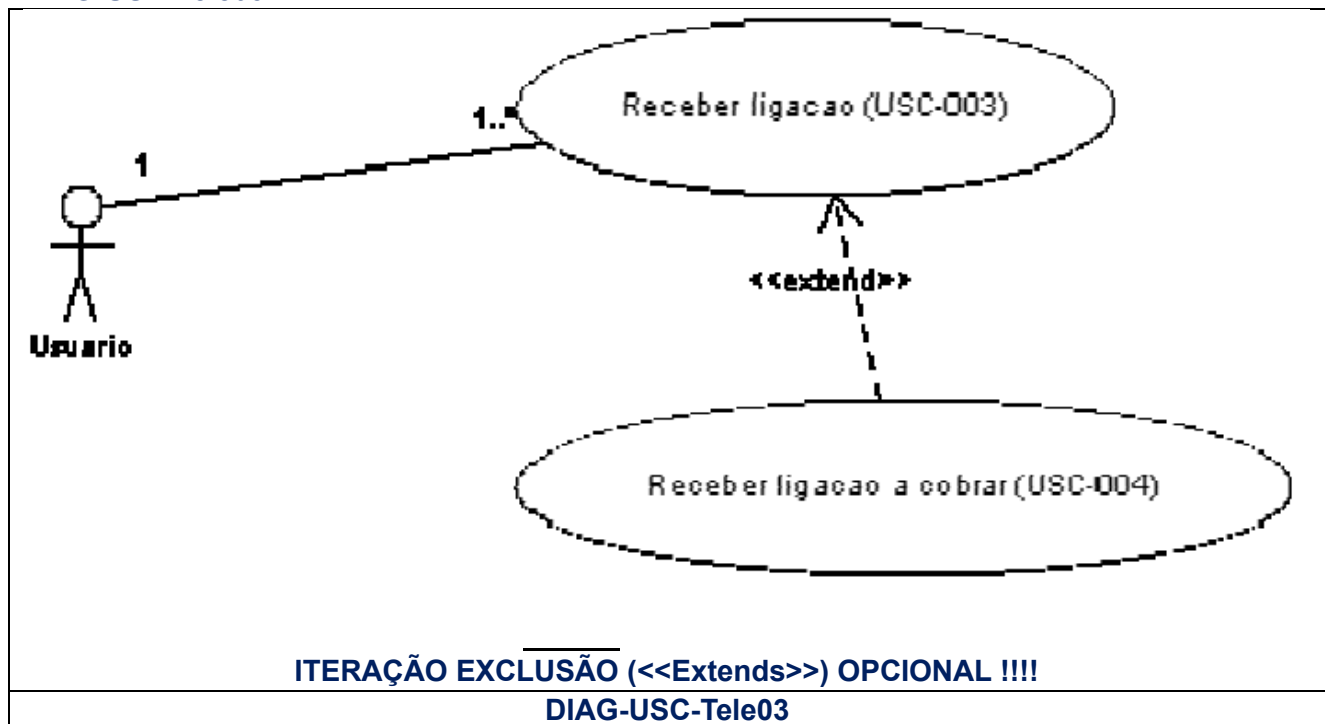
Para os seguintes exemplos criar um pacote(pasta) com o nome **DIAG-PROJ-TELEFONIA** e dentro do pacote criar os seguintes USC (exemplos)

A estrutura hierárquica será **DIAG-PROJ-TELEFONIA**

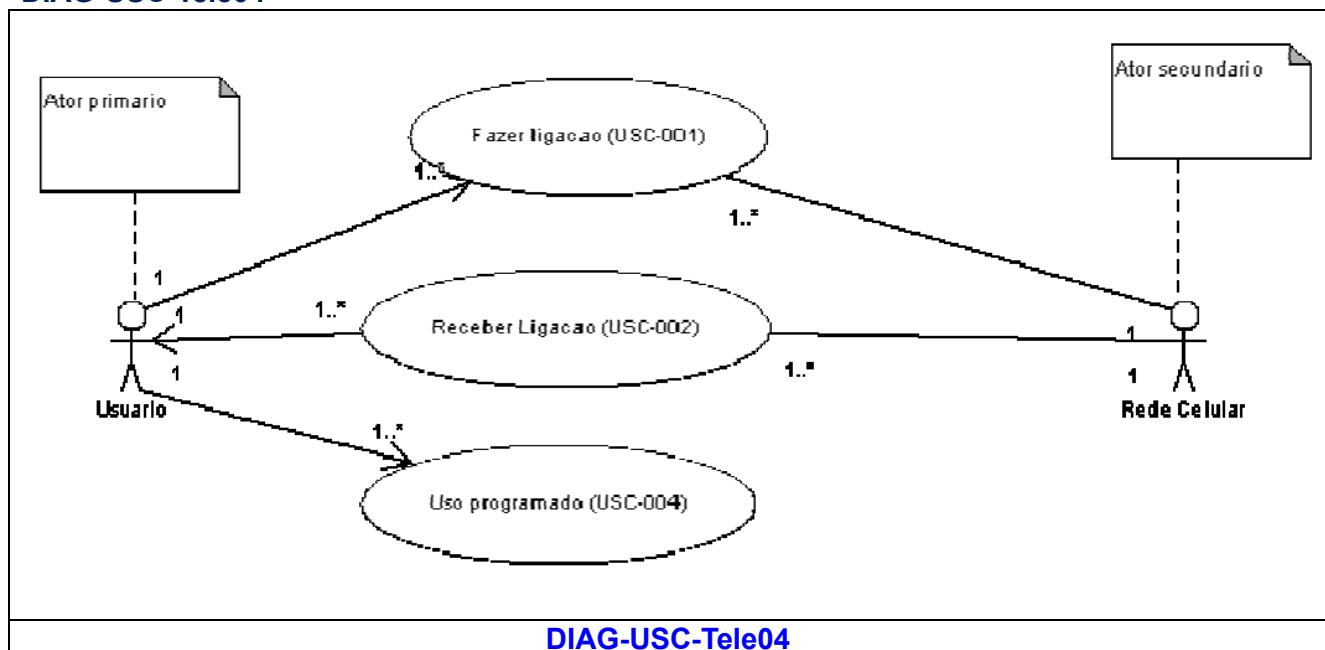
DIAG-USC-Tele01 **DIAG-USC-Tele02**
DIAG-USC-Tele03 **DIAG-USC-Tele04**
DIAG-USC-Tele05
DIAG-USC-Tele06

DIAG-USC-Tele01

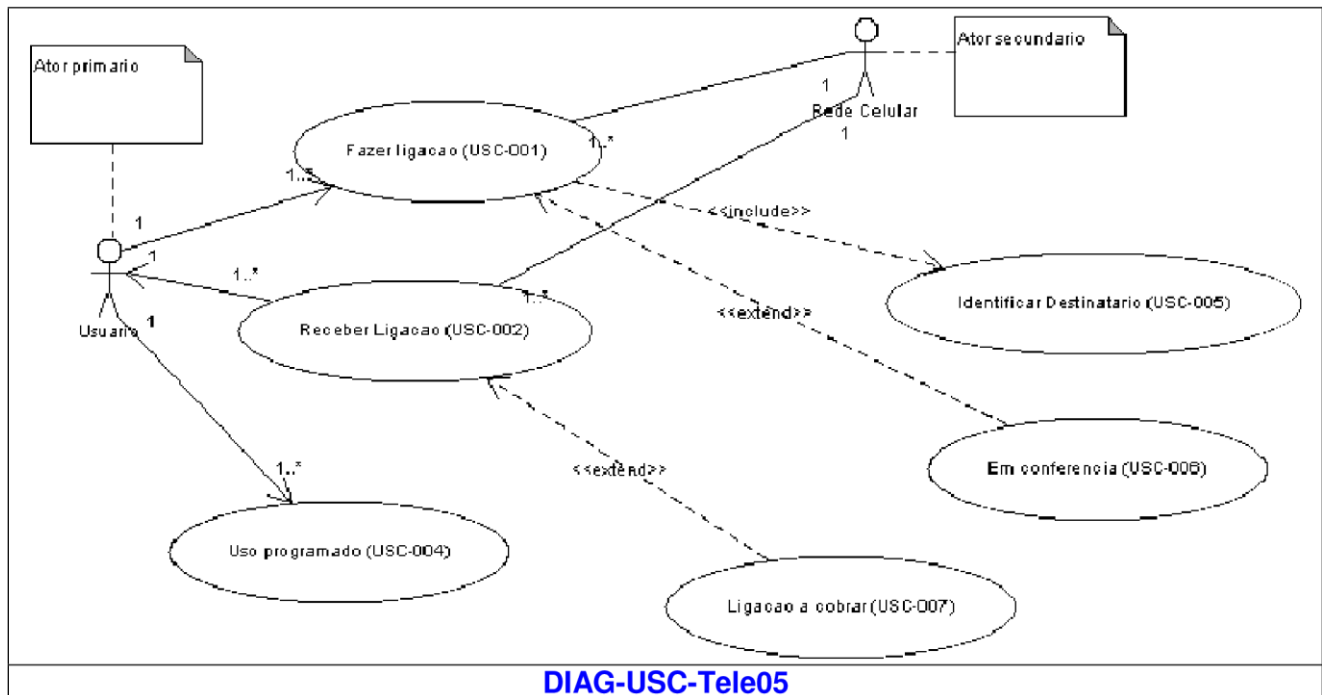
DIAG-USC-Tele03



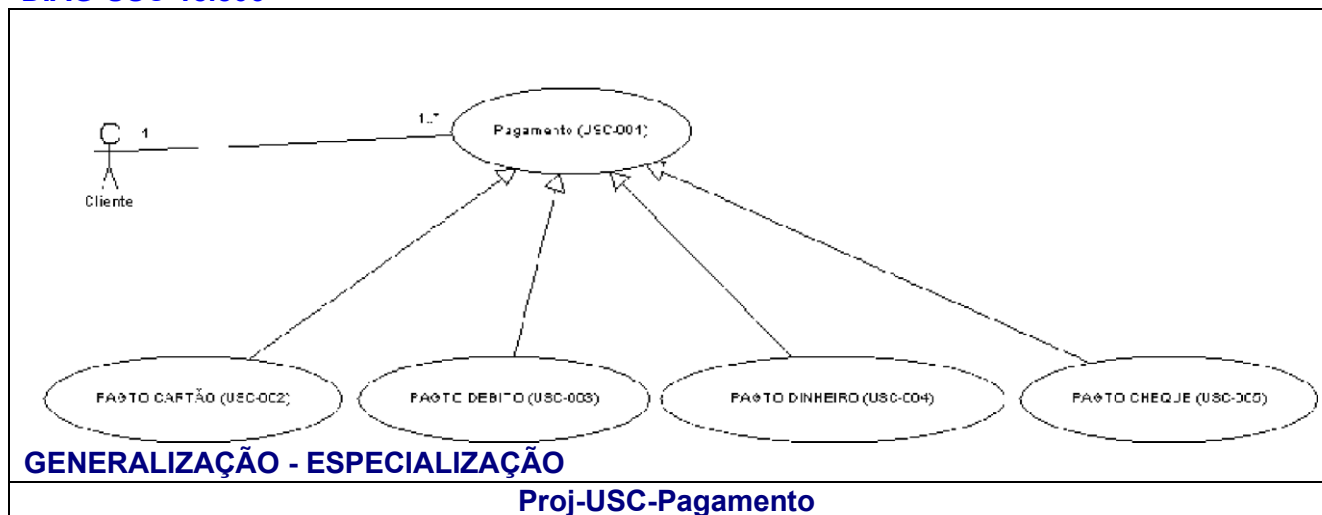
DIAG-USC-Tele04



DIAG-USC-Tele05

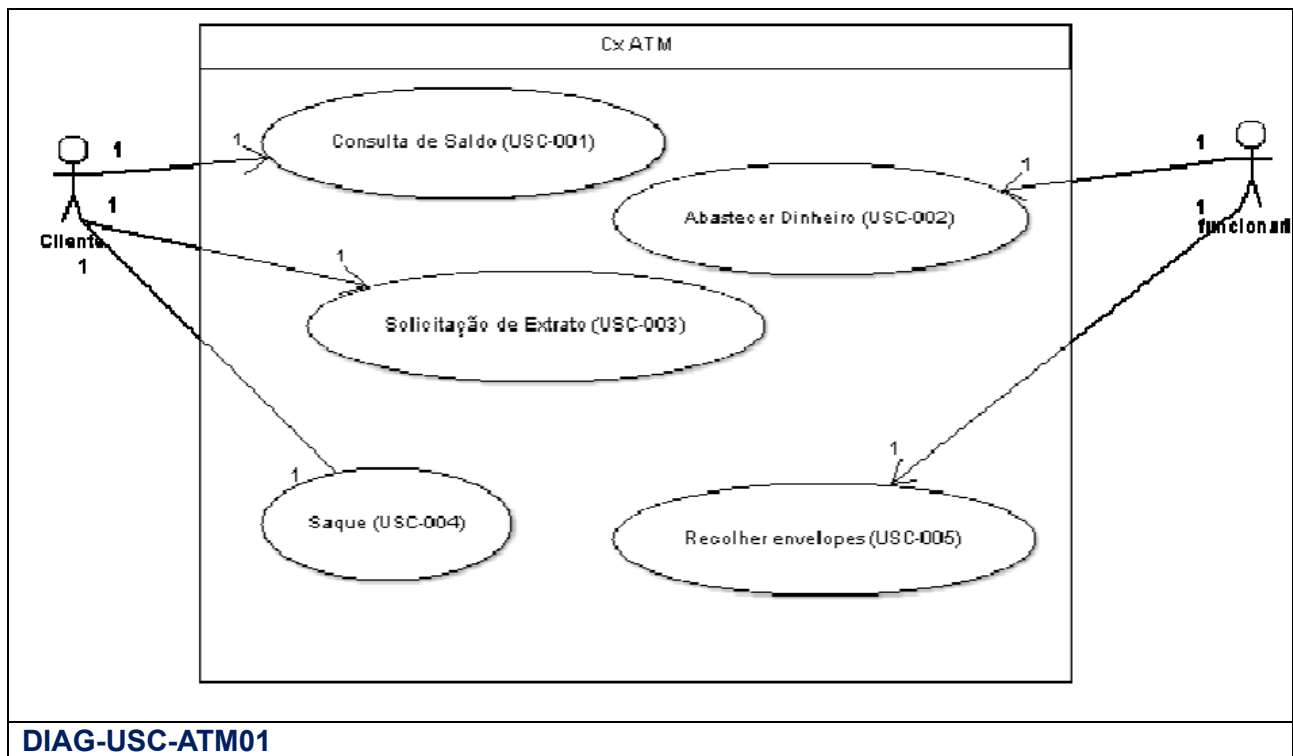


DIAG-USC-Tele06



DIAG-PROJ-BANCO

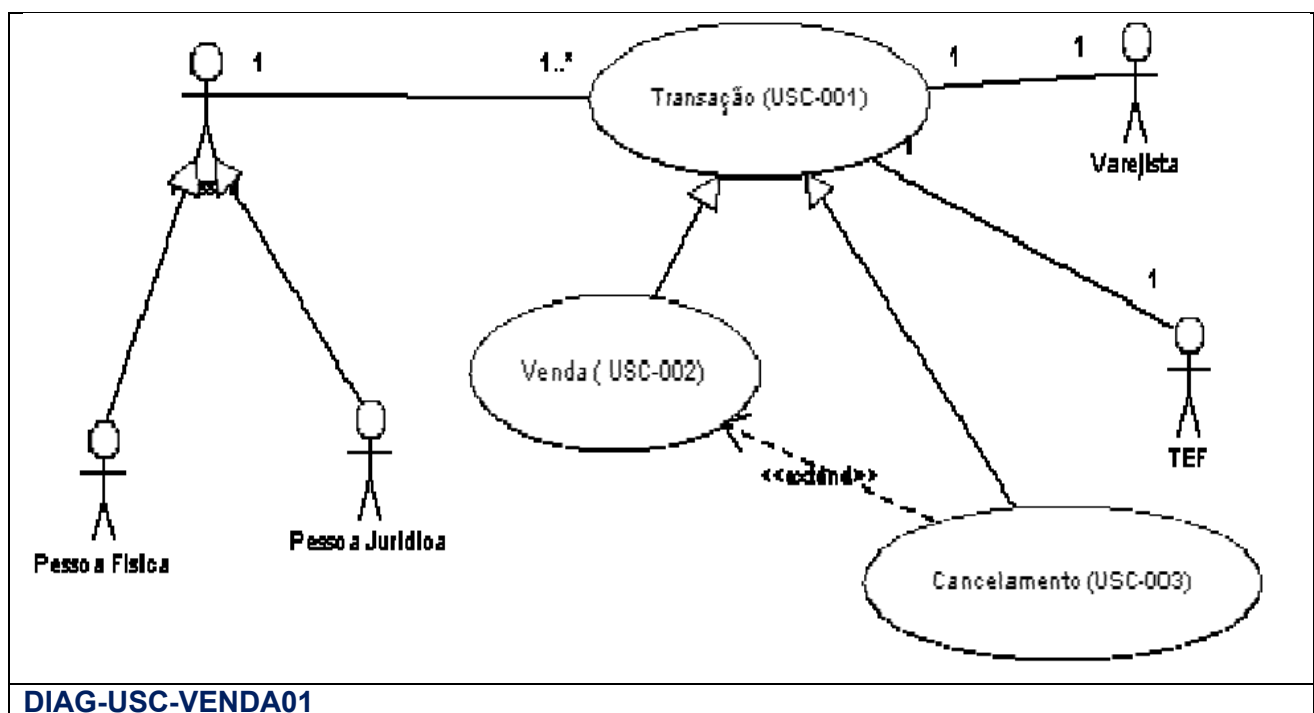
DIAG-USC-ATM01

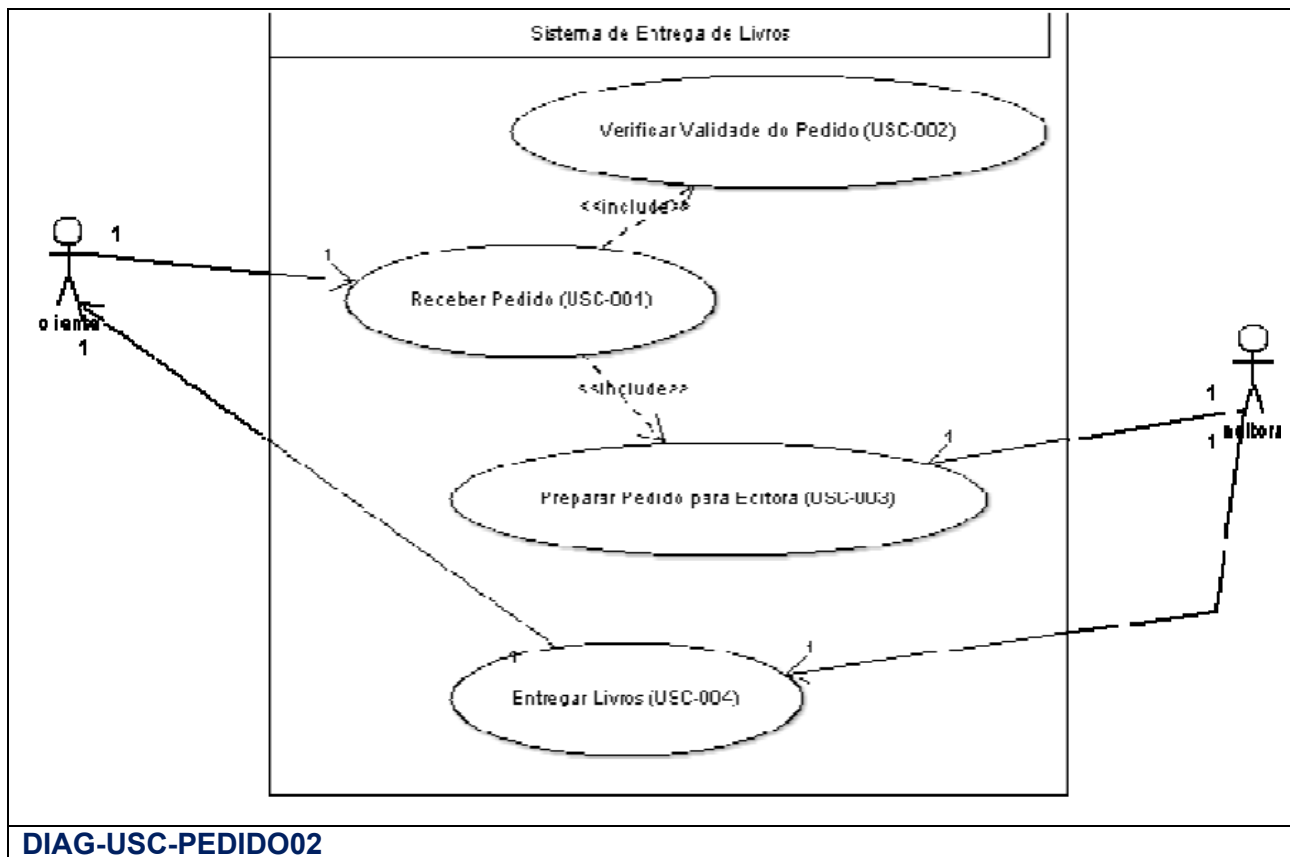


DIAG-PROJ-SISTEMA-VENDA

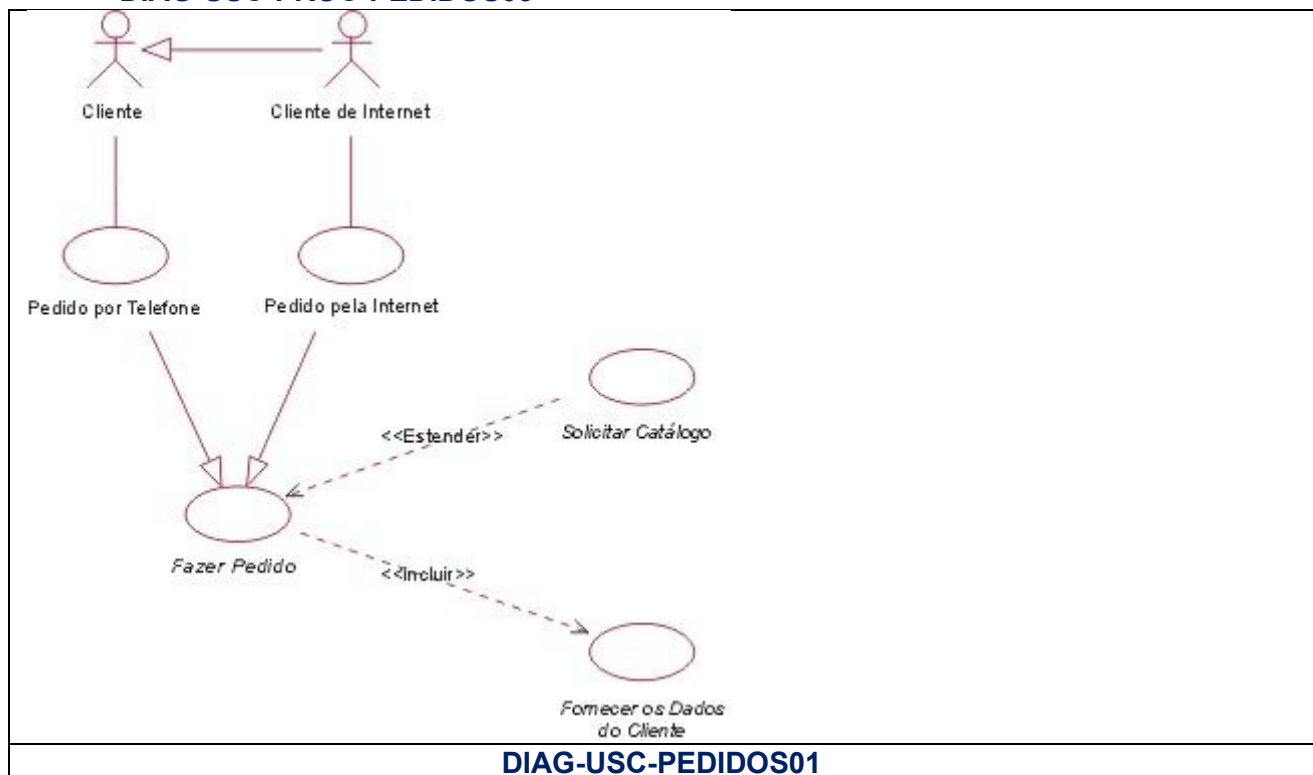
DIAG-USC-VENDA01

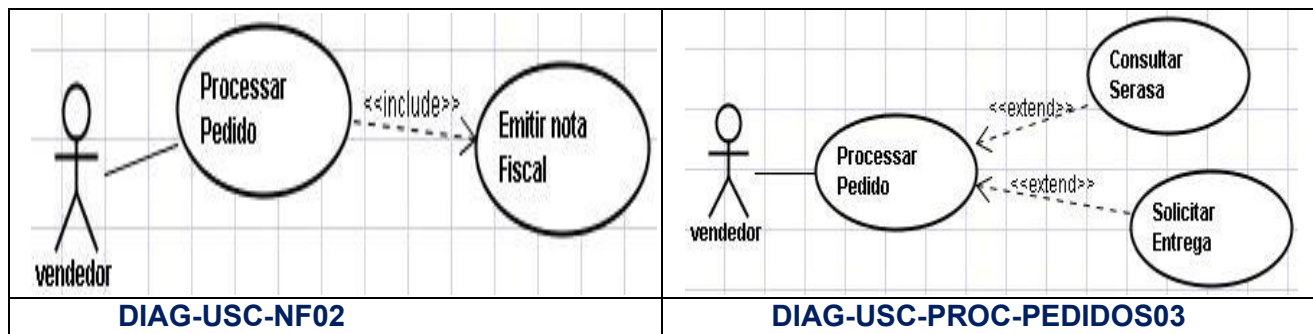
DIAG-USC-PEDIDO02



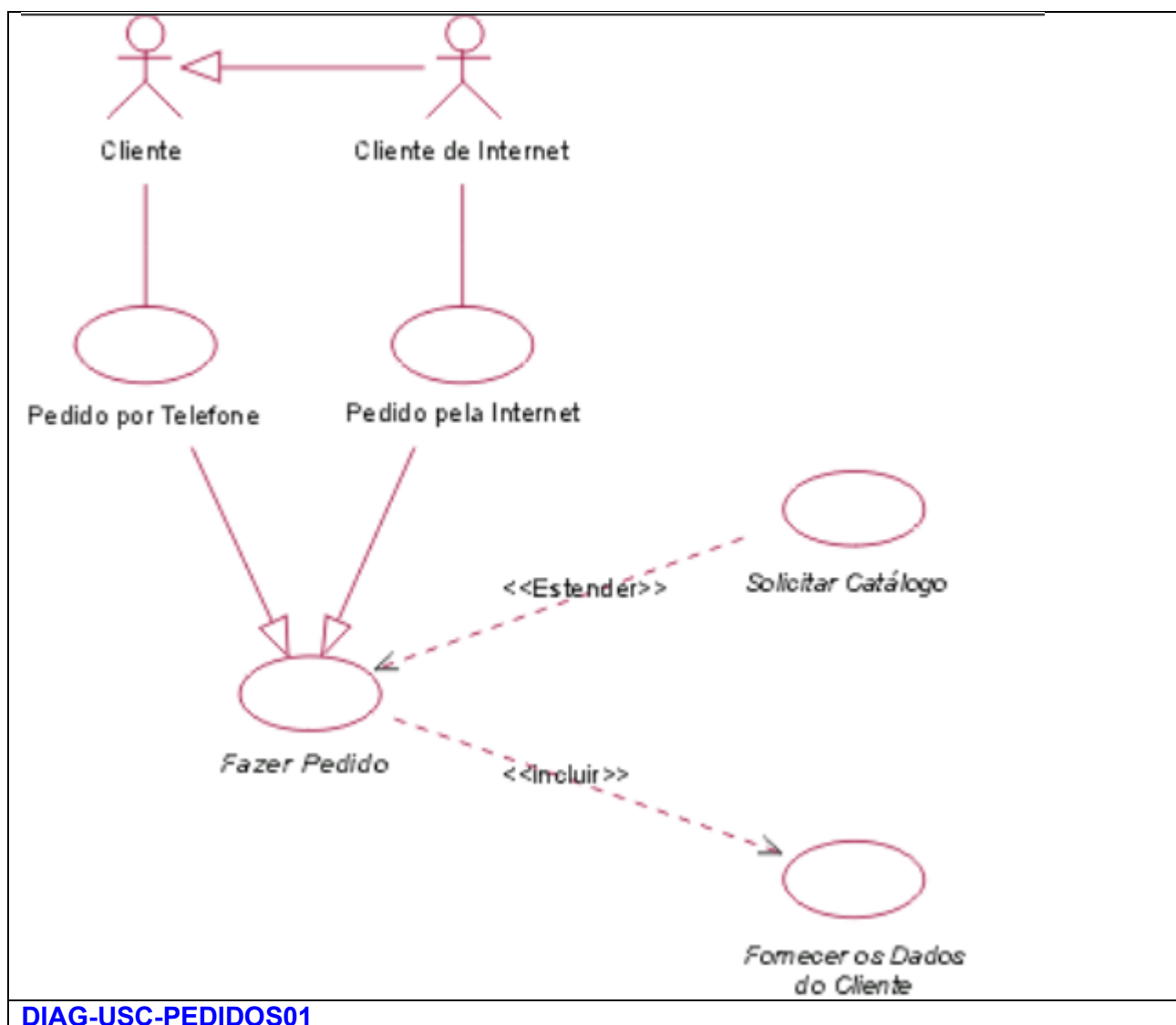


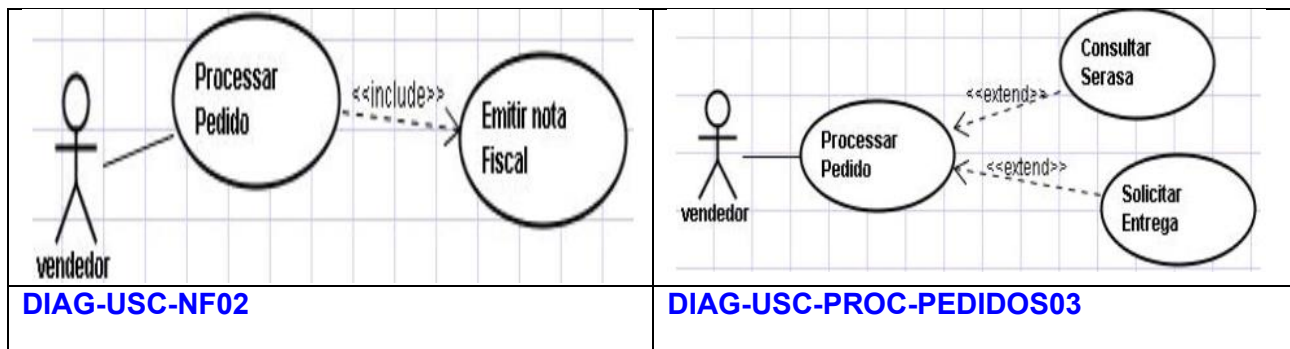
DIAG-PROJ-SISTEMA-PEDIDOS
USC-PEDIDOS01
DIAG-USC-NF02
DIAG-USC-PROC-PEDIDOS03





DIAG-PROJ- Sistema-Acadêmico
 USC-PEDIDOS01
 DIAG-USC-NF02
 DIAG-USC-PROC-PEDIDOS03



**DIAG-PROJ- Sistema-Acadêmico****DIAG-USC-PEDIDOS01****DIAG-USC-NF02****DIAG-USC-PROC-PEDIDOS03**