

Aonde você quer chegar?
Vai com a



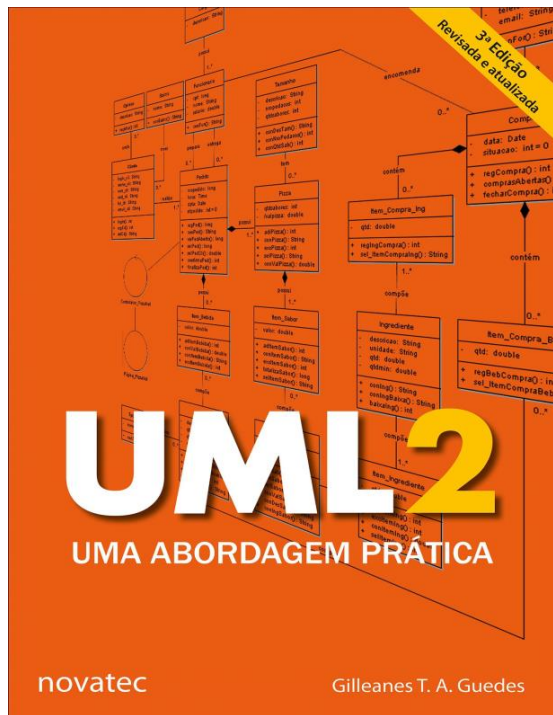


Disciplina: Análise e Projeto OO
Prof. Maurício P. de Freitas MSc.

Aula 06 – 28/03/2024
Diagrama de Sequência

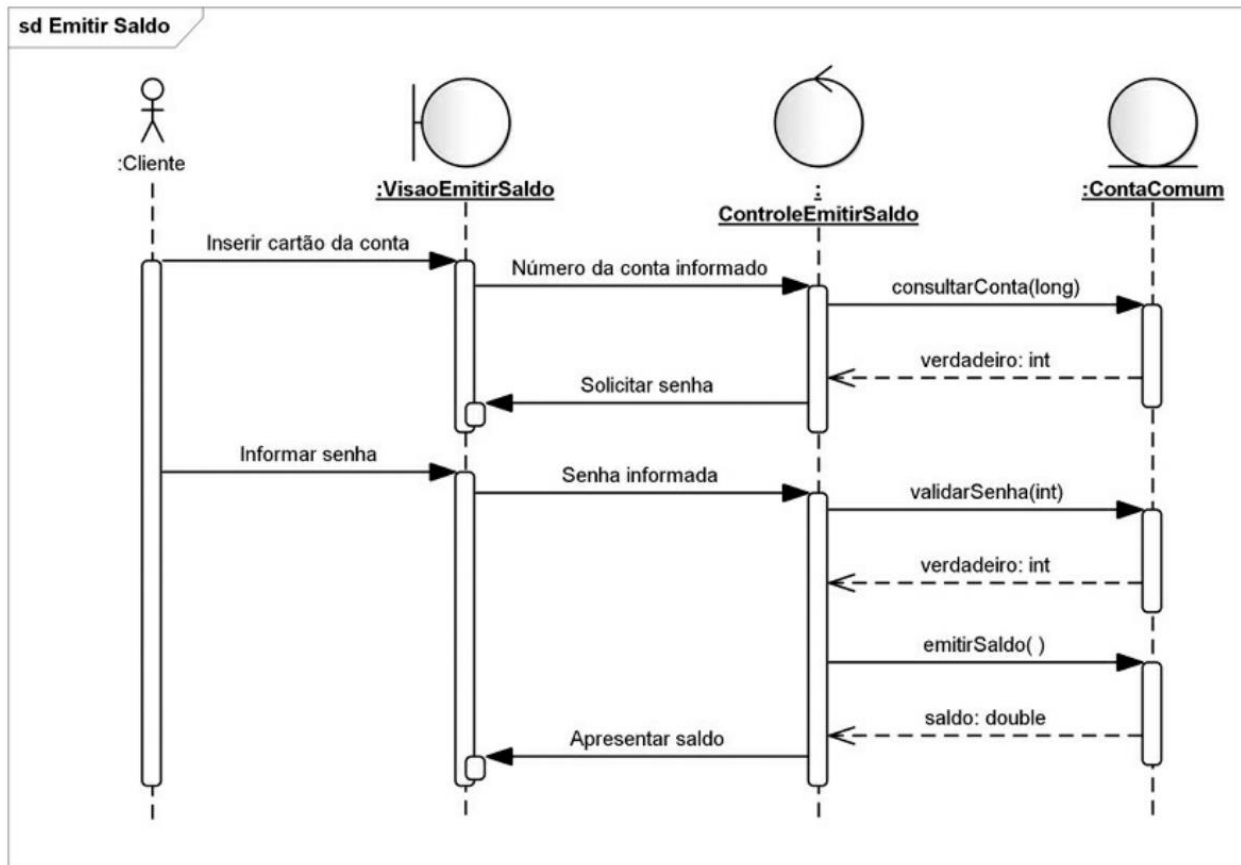


Referência:



GUEDES, Gilleanes TA. **UML 2-Uma abordagem prática**. Novatec Editora, 2018.

Diagrama de Sequência



Análise e Projeto OO

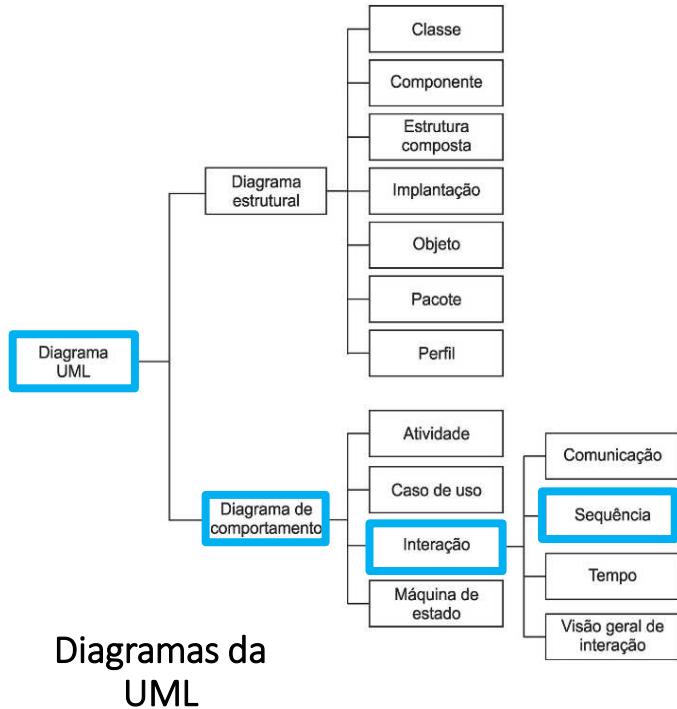


Diagrama de Sequência:

- Procura determinar a **sequência de eventos que ocorrem** em um determinado processo, identificando quais **mensagens devem ser disparadas entre os elementos** envolvidos e em que ordem.

O principal objetivo desse diagrama é:

- Busca determinar a ordem em que os eventos ocorrem;
- As mensagens que são enviadas;
- Os métodos que são chamados;
- Como os objetos interagem dentro de um determinado processo.

Análise e Projeto OO

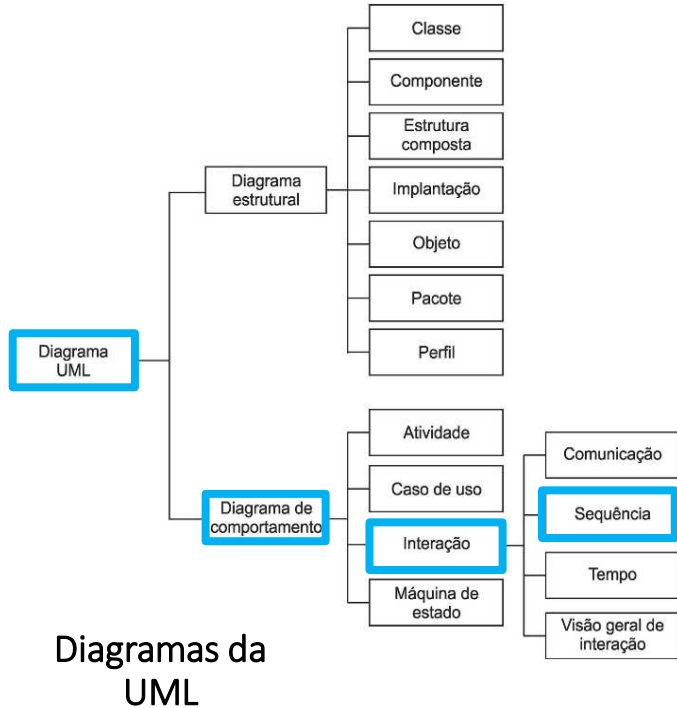
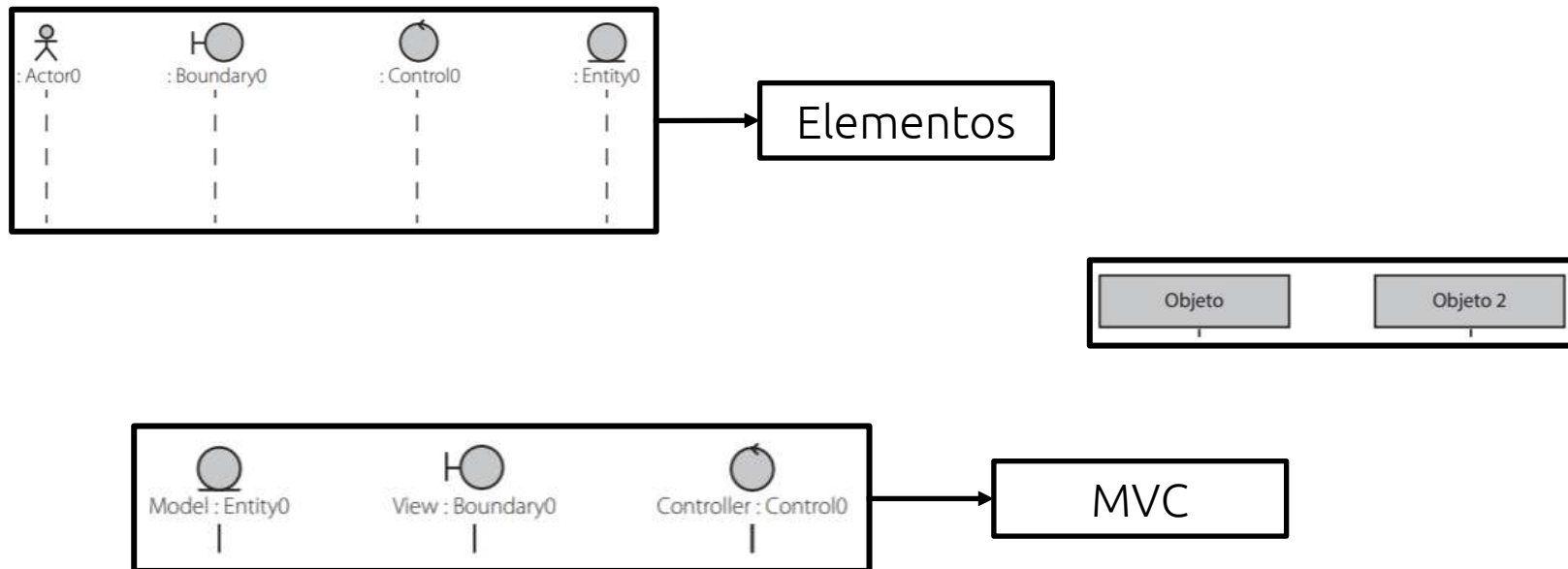


Diagrama de Sequência:

- **Baseia-se no diagrama de casos de uso**, havendo normalmente um diagrama de sequência para cada caso de uso declarado;
- O diagrama de sequência depende **também do diagrama de classes**, uma vez que as classes dos objetos utilizados no diagrama de sequência estão descritas nele.
- Também são chamados de **diagramas de eventos**.

Diagrama de Sequência



Padrão MVC: <https://www.devmedia.com.br/introducao-ao-padrao-mvc/29308>

Elementos:

ATORES:

- São **instâncias dos atores declarados no diagrama de casos de uso**, representando entidades externas que interagem com o sistema e solicitam serviços, gerando, assim, eventos que iniciam processos.
- Os **atores não são realmente obrigatórios** nesse diagrama, mas são utilizados com muita frequência.



Elementos:

LIFELINE:

- Representa a **existência de um participante** (geralmente um objeto ou uma instância de uma classe) **durante um período específico de interação.**
- Ajudam a visualizar a interação entre diferentes objetos, a ordem das mensagens trocadas e a duração da participação de cada objeto na interação geral.



Elementos:

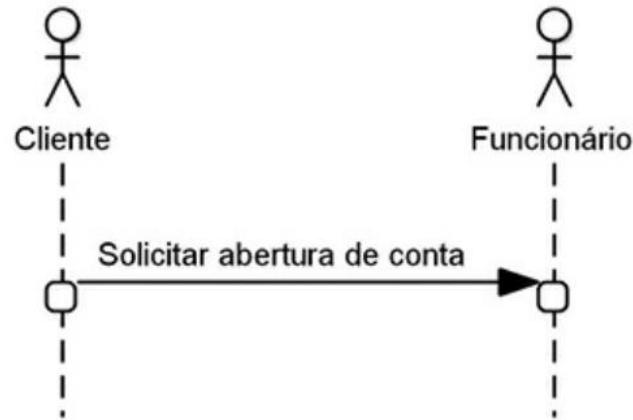
MENSAGENS:

- As mensagens são utilizadas para **demonstrar a ocorrência de eventos**, que normalmente forçam a **chamada de um método em algum dos objetos envolvidos no processo**.
- As mensagens são representadas por **linhas entre dois participantes da interação**, contendo setas indicando qual participante enviou a mensagem e qual a recebeu.
- As mensagens são **apresentadas normalmente na posição horizontal** entre as linhas de vida dos participantes e sua ordem sequencial é demonstrada de cima para baixo.
- Os textos contidos nas mensagens primeiramente **identificam qual evento ocorreu** e forçou o envio da mensagem e **qual método foi chamado**. Sendo estes, separados por “:”.

Elementos:

MENSAGENS entre ATORES:

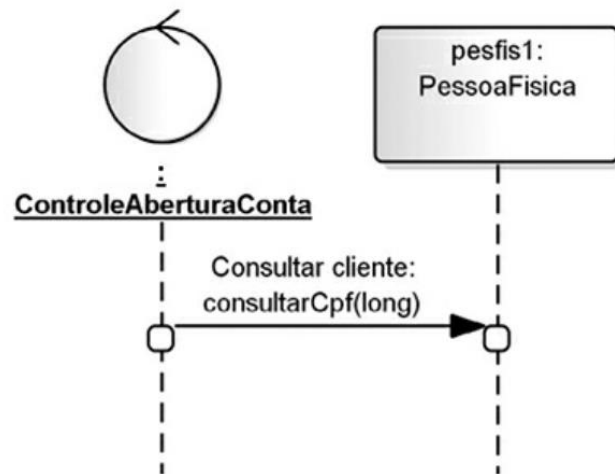
- Esse tipo de mensagem é **opcional** e sua função basicamente é a de **demonstrar interação entre os atores externos** que participam de um processo.



Elementos:

MENSAGENS entre LIFELINES:

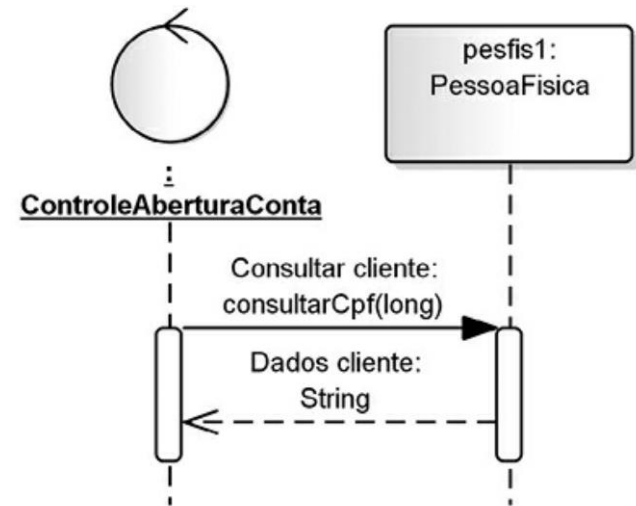
- A **ocorrência mais comum** de troca de mensagens em um diagrama de sequência. Em geral, essas mensagens acarretam a execução de um método.
- Quando uma mensagem é disparada ou recebida, a **linha de vida dos elementos envolvidos se torna mais grossa**. Isto é chamado **Foco de Controle ou Ativação** e determina o momento em que um elemento da interação está participando ativamente do processo.



Elementos:

MENSAGEN DE RETORNO:

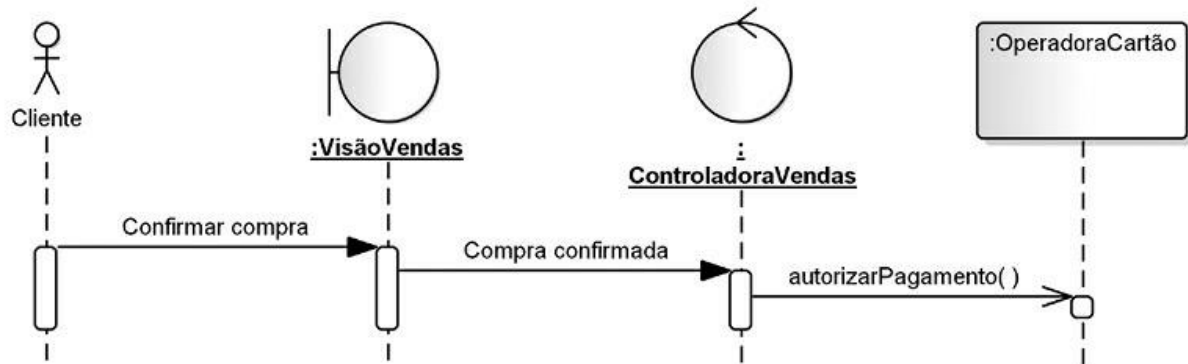
- Esse tipo de mensagem **identifica a resposta a uma mensagem disparada por uma lifeline** (objeto) ou um ator.
- Uma mensagem de retorno **pode retornar informações específicas do método** chamado ou **apenas um valor indicando** se o método foi executado com sucesso ou não.
- As mensagens de retorno são representadas por **uma linha tracejada contendo uma seta fina** que aponta para o elemento que recebe o resultado do método chamado.



Elementos:

MESAGENS SINCRONAS X ASSINCRONAS:

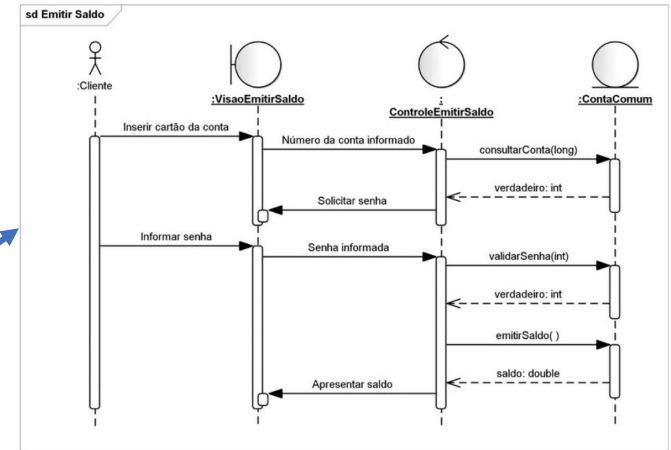
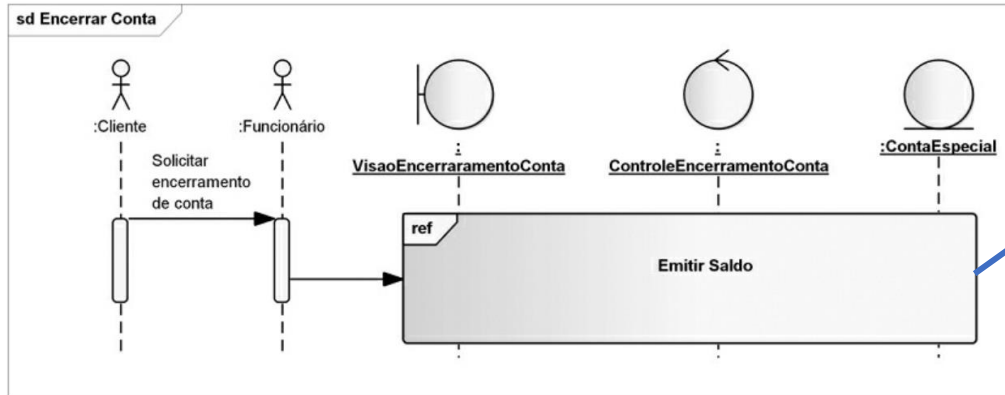
- **Síncronas:** **devem esperar pelo retorno** da mensagem para continuar com o processamento. Representada por uma seta fechada.
- **Assíncronas:** **não esperam pelo retorno** da mensagem para continuar com o processamento. Representada por uma seta aberta.



Elementos:

FRAGMENTOS DE INTERAÇÃO:

- Um fragmento de interação é **uma parte de uma interação**, no entanto **cada fragmento de interação é considerado uma interação independente**.



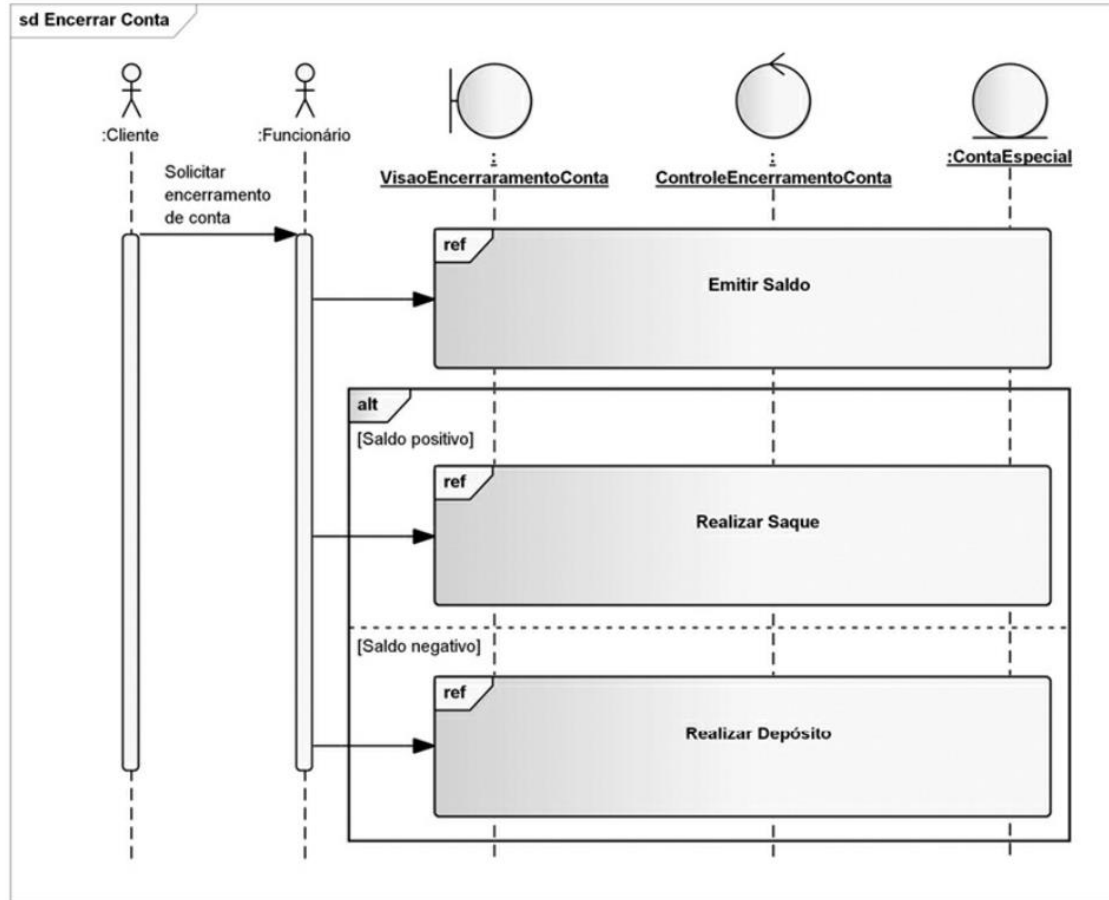
Elementos:

FRAGMENTOS COMBINADOS E OPERADORES DE INTERAÇÃO:

- Eles permitem modelar situações complexas na execução de fragmentos, como escolhas **condicionais**, **loops**, **paralelismo**, e **outros cenários de controle de fluxo** dentro da sequência.
- Os principais tipos de fragmentos combinados incluem: **alt** (alternativas), **opt** (opção), **par** (paralelo), **loop** (laço), **break** (quebra), **critical region** (região crítica), **neg** (negativo), **assertion** (afirmação), **ignore** (ignorar), **consider** (considerar), **seq** (sequência fraca) e **strict** (sequência estrita).

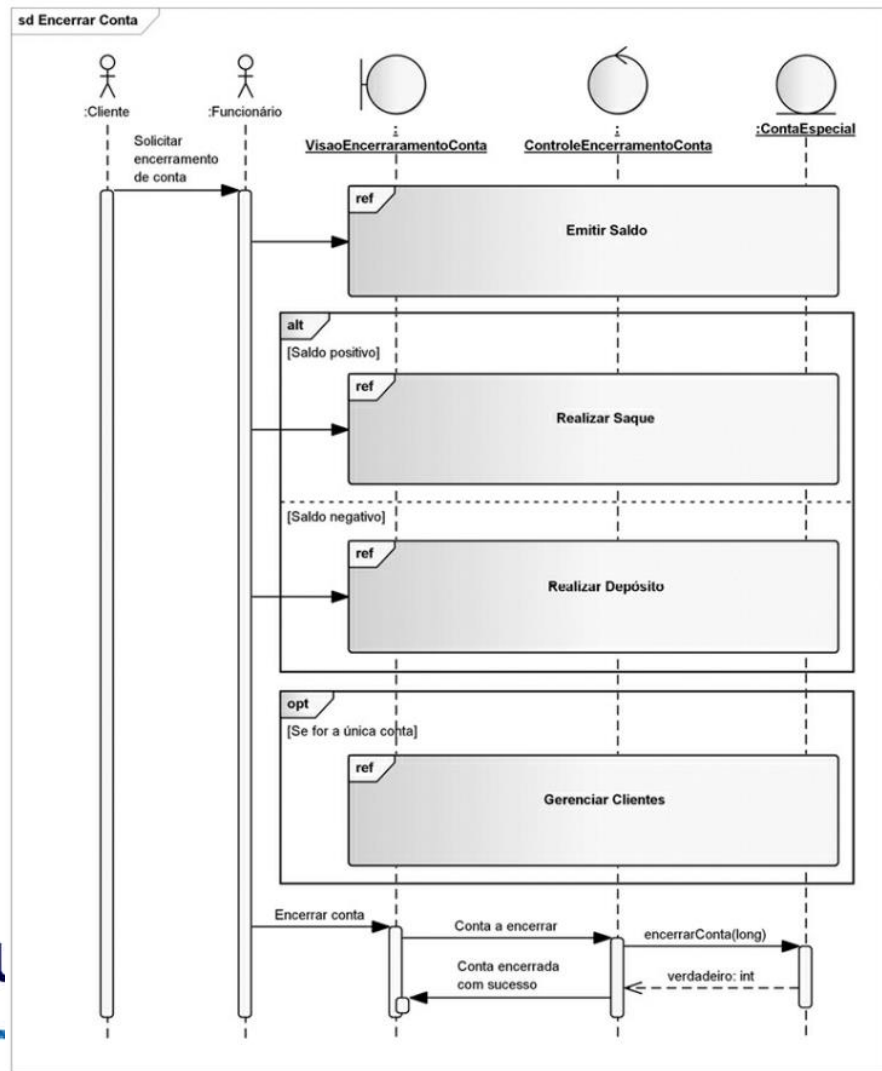
FRAGMENTOS COMBINADOS :

- **alt (Alternativa):** Representa uma escolha condicional (semelhante a um "if-else" em programação). Diferentes sequências de mensagens são mostradas dependendo de condições específicas.



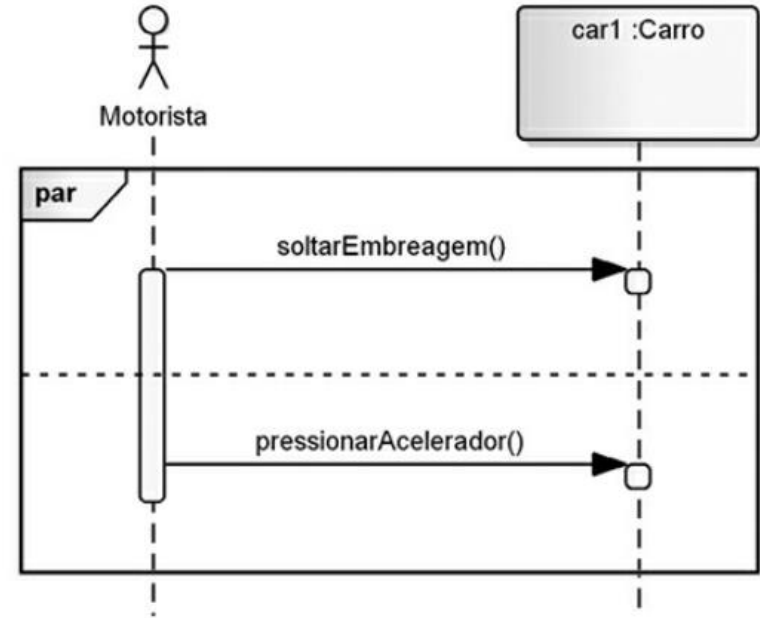
FRAGMENTOS COMBINADOS :

- **opt (Opcional):** Um caso especial de "alt" onde há apenas uma condição. É usado para mostrar **uma sequência de mensagens que só ocorre sob uma determinada condição**.



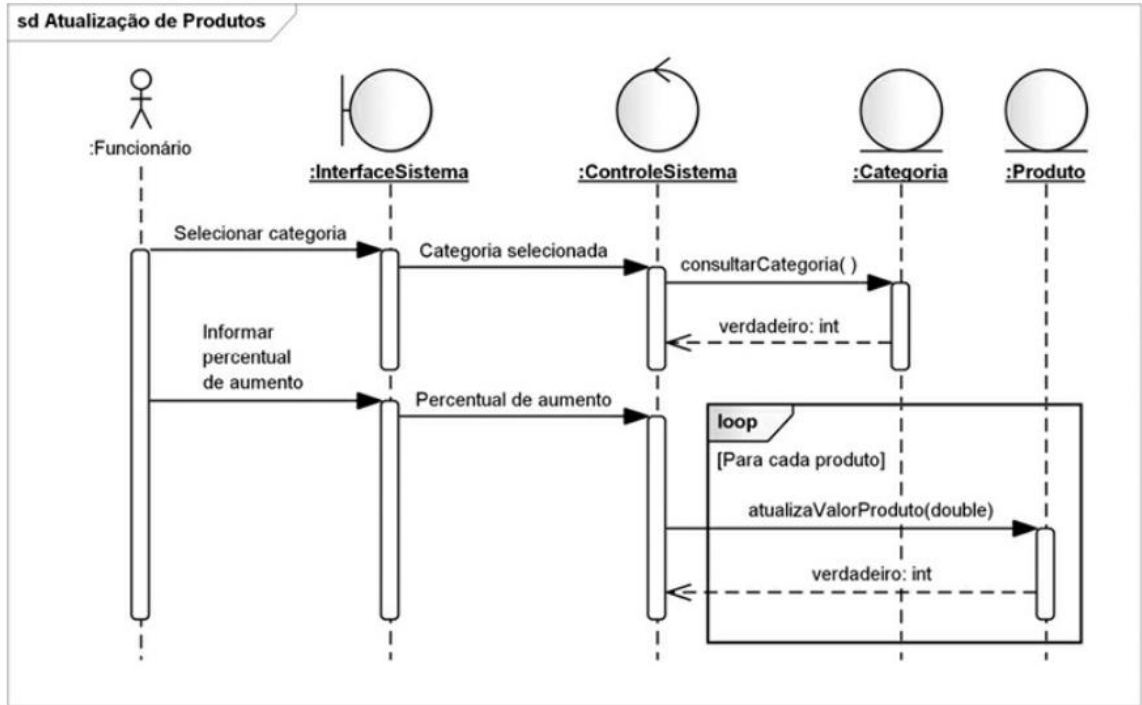
FRAGMENTOS COMBINADOS :

- **par (Paralelo):** Permite mostrar **seqüências de mensagens que ocorrem em paralelo**. É útil para modelar situações de multitarefa ou processamento simultâneo.



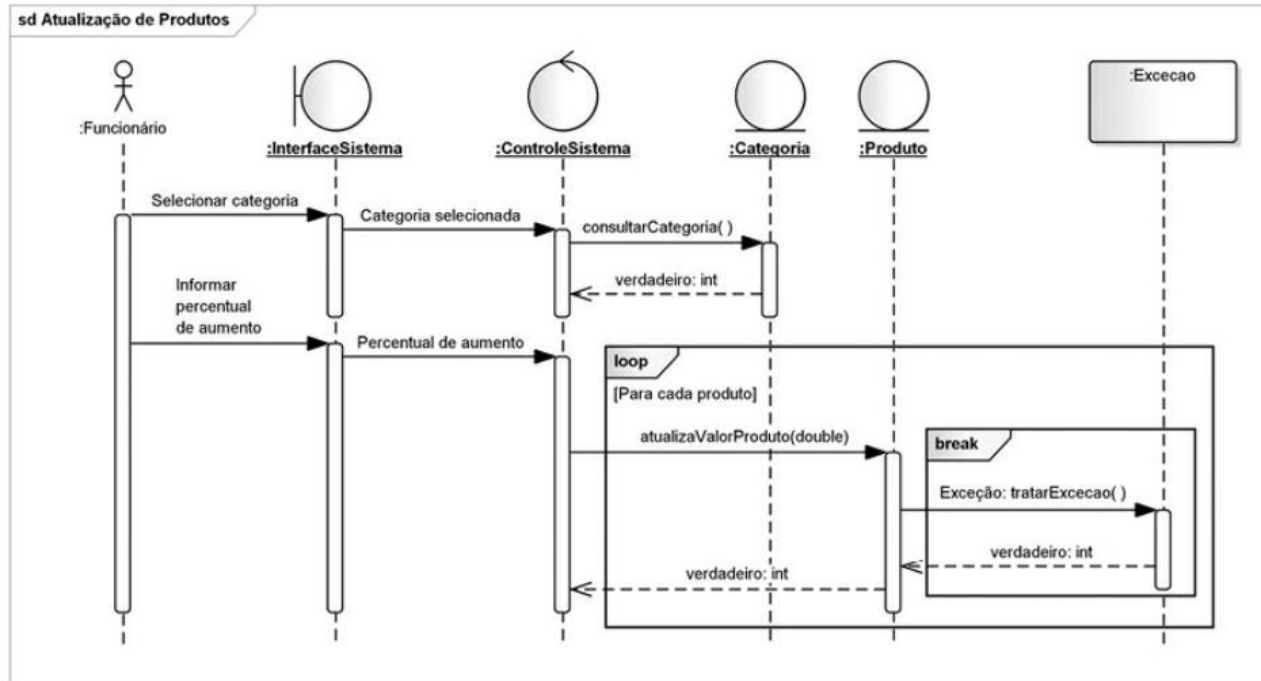
FRAGMENTOS COMBINADOS :

- **loop (Laço):** Indica uma **repetição de mensagens**. Condições para iniciar e terminar o loop podem ser especificadas



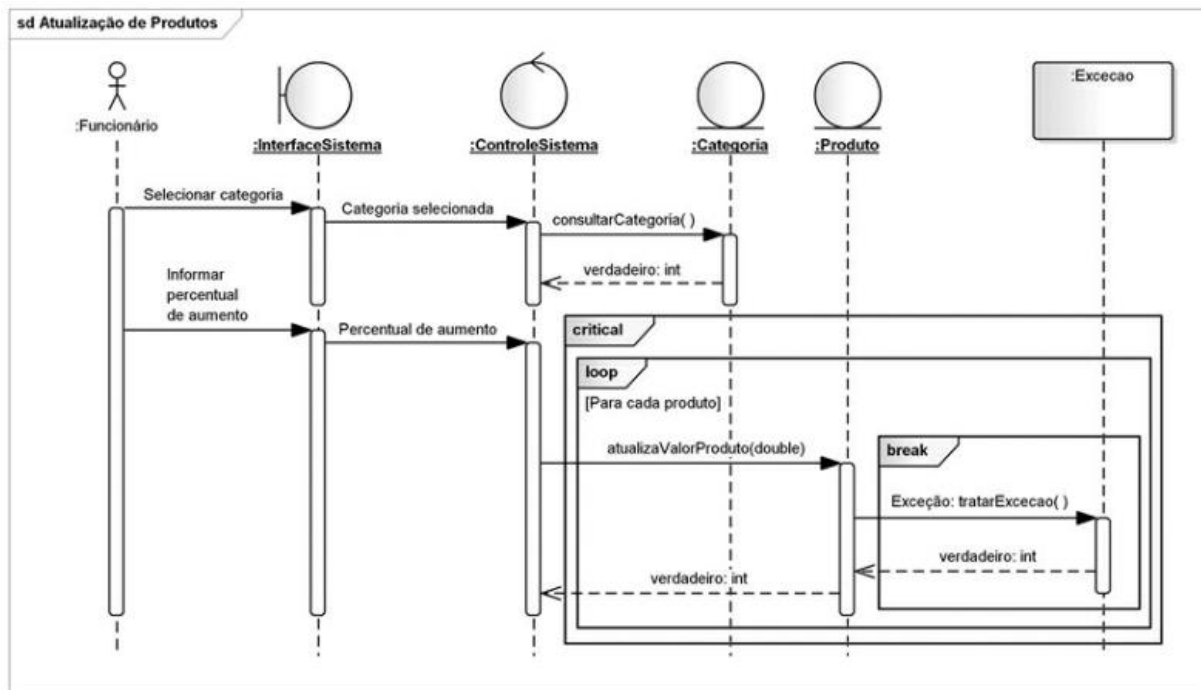
FRAGMENTOS COMBINADOS :

- **break (Interrupção):** Esse operador de interação indica uma “**quebra**” na execução normal do processo. É usado principalmente para modelar o tratamento de exceções.



FRAGMENTOS COMBINADOS :

- **critical** (Região Crítica) – esse operador de interação **identifica uma operação atômica que não pode ser interrompida** por outro processo até ser totalmente concluída.

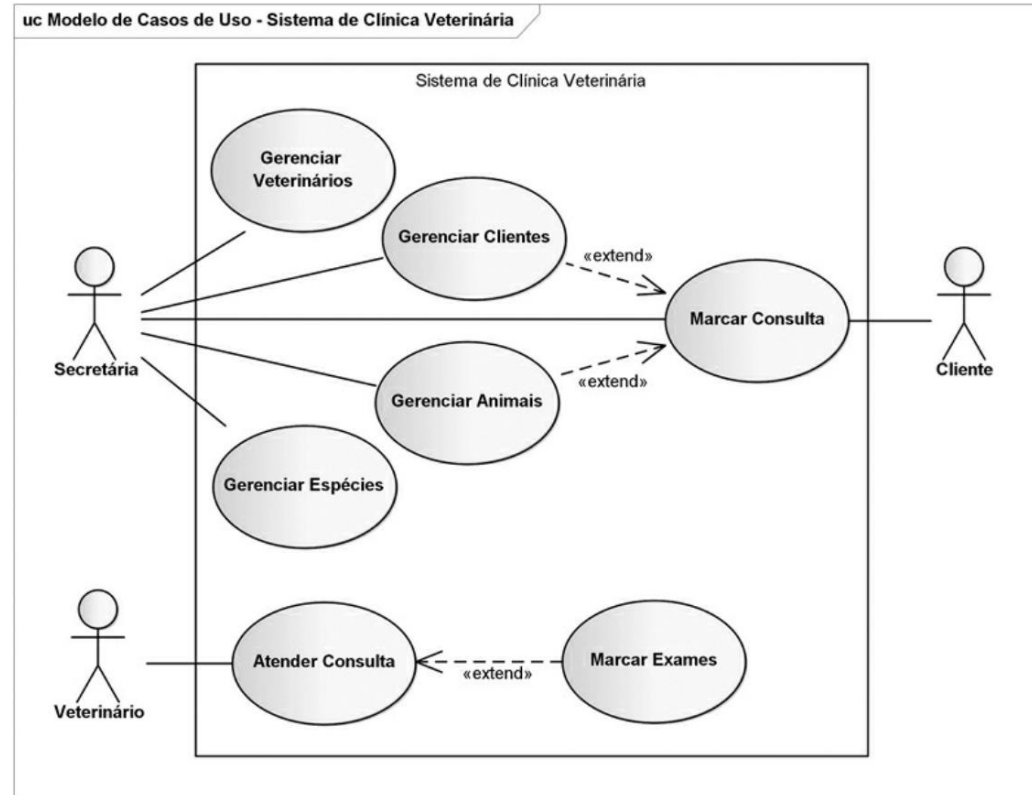


Exemplo Clínica Veterinária.

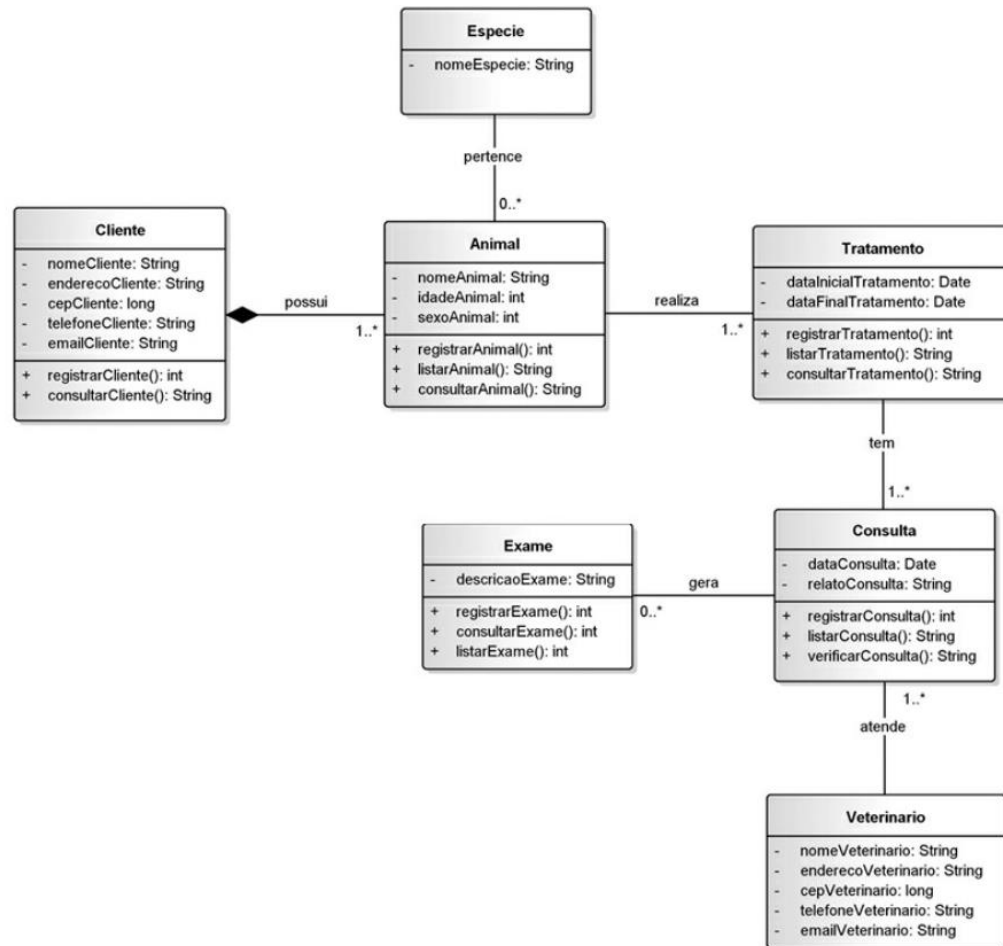
GUEDES, Gilleanes TA. **UML 2-Uma abordagem prática**. Novatec Editora, 2018.



Exemplo Clínica Veterinária: Diagrama de Caso de Uso.

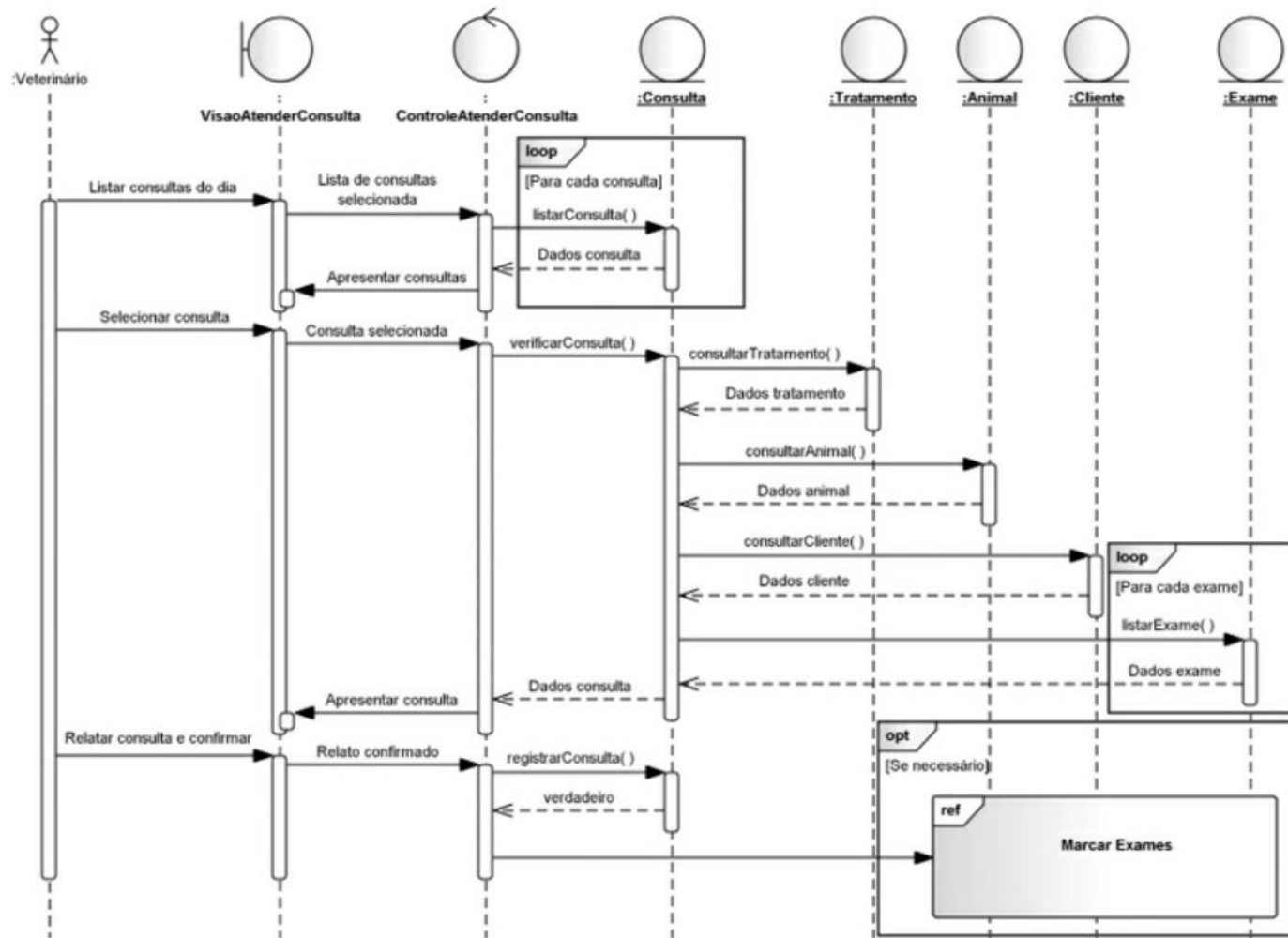


Exemplo Clínica Veterinária: Diagrama de Classe.



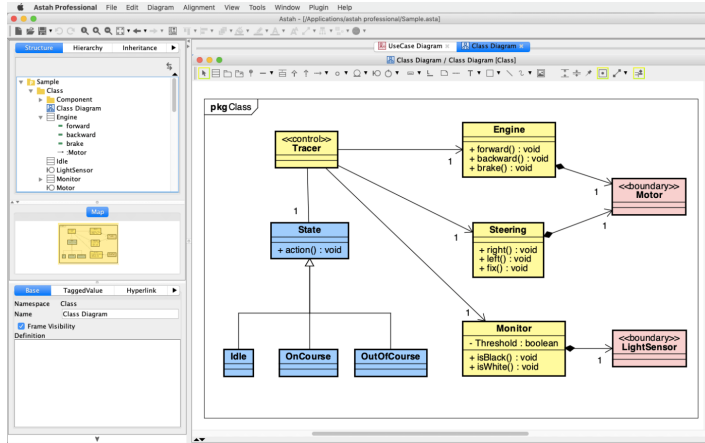
Exemplo Clínica Veterinária: Diagrama de Sequência.

sd Atender Consulta

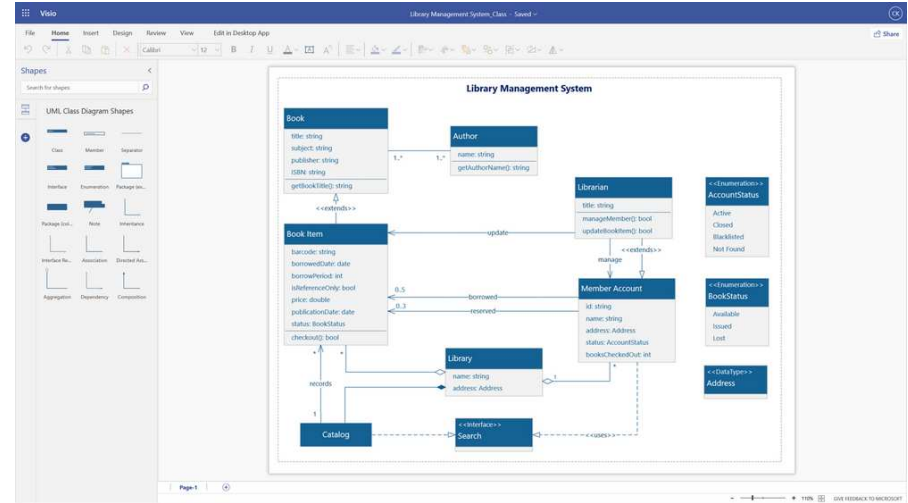


UML - Ferramentas CASE

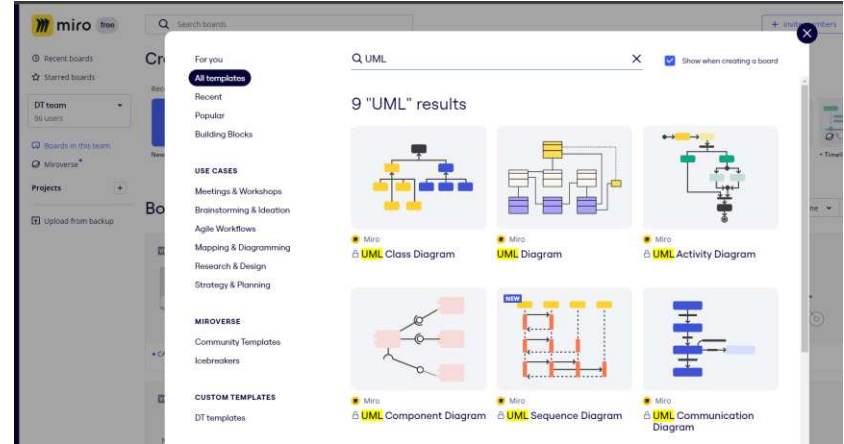
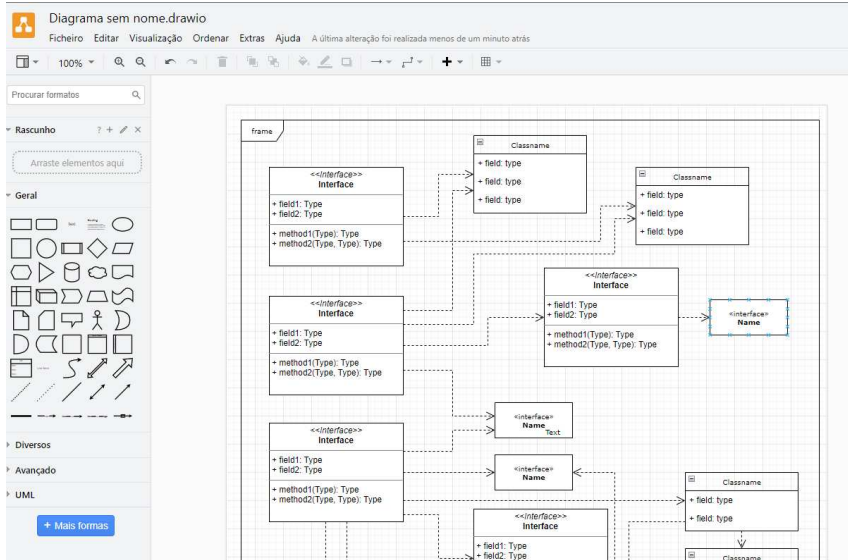
UML - Ferramentas CASE

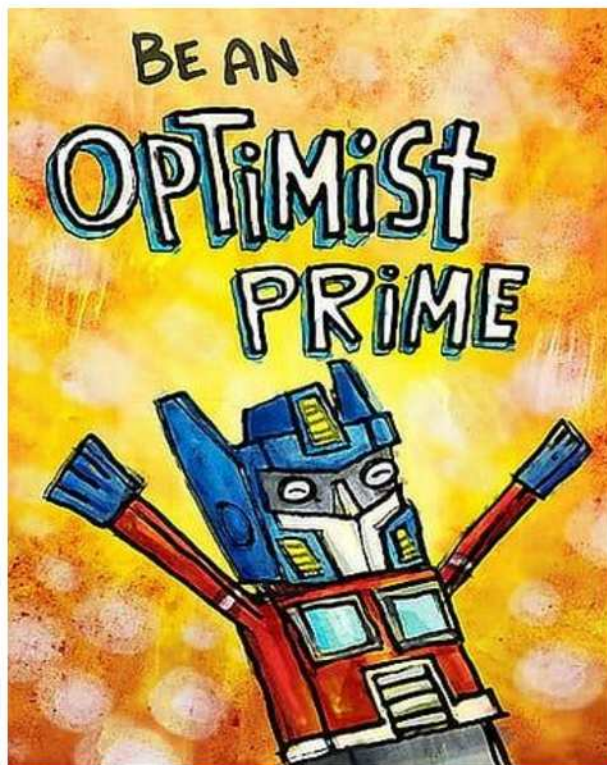


astah



UML - Ferramentas CASE







UniCesumar

EDUCAÇÃO PRESENCIAL E A DISTÂNCIA