



Análise e Modelagem de Sistemas

Prof. David S. Tosta

Agenda

- Revisão aula V
- Ferramenta Case – LucidChat
- Classes e Objetos

Revisão Aula V

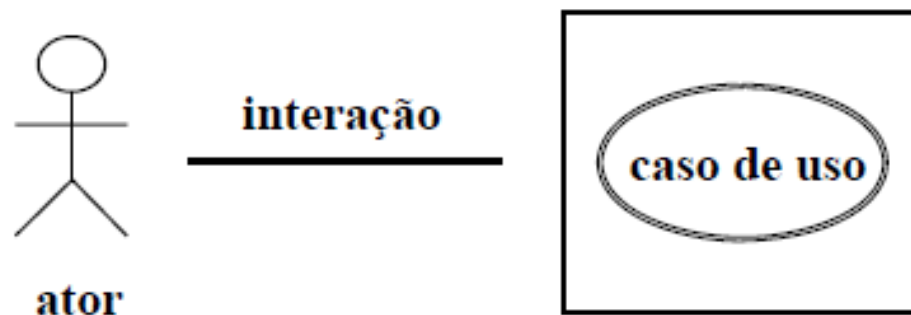
- **Priorização de Requisitos**
 - Garantir que o **essencial** é realizado
 - Expectativas altas
 - Tempo curto
 - Recursos limitados
 - Controlar o escopo do projeto: Síndrome do “já que”
 - Caper Jones reporta que os requisitos que “rastejam para debaixo” do escopo representam
 - risco de 80% a projetos de gerência de informação
 - risco de 70% a projetos militares

Revisão Aula V

- Técnica de Priorização de Requisitos
 - Categorização
 - off line
 - Cada interessado recebe a lista dos requisitos
 - Classifica segundo os critérios
 - crítico - indispensável
 - importante - representa perda de funcionalidade ou perda de fatia do mercado
 - útil - facilita a vida, faz o sistema mais atraente
 - Atribui valores 1,2,3 (onde 1 é crítico)
 - Fazer um ranking dos resultados

Revisão Aula V

- Modelagem de Caso de Uso
 - O conjunto de casos de uso define as diferentes maneiras de interação com o sistema;
 - Os atores e os casos de uso são os principais componentes de um modelo de casos de uso.



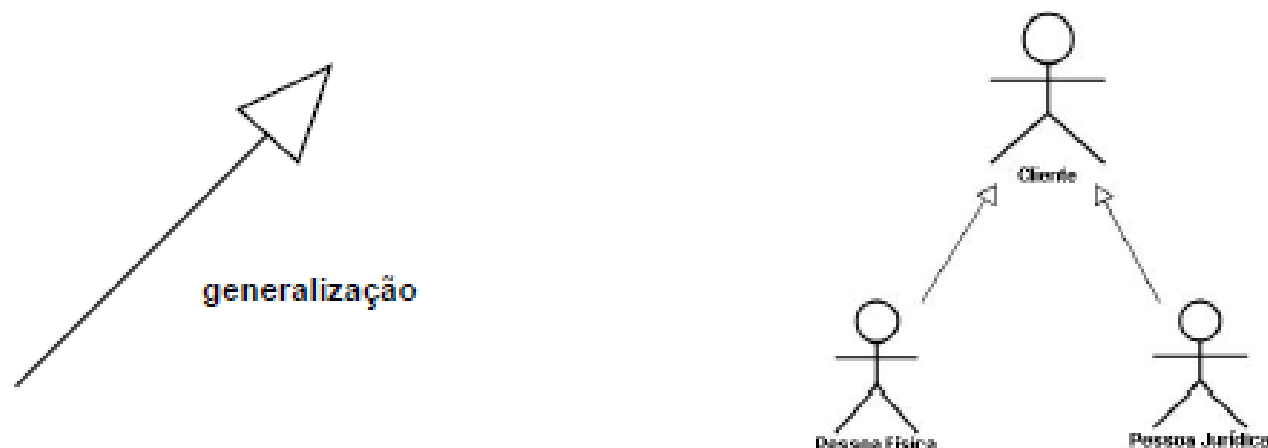
Revisão Aula V

- Ator
 - O seguinte questionário pode ser usado para identificar os atores de um sistema :
 - Quem usará as funções principais do sistema?
 - Quem precisará do sistema para executar suas tarefas diárias?
 - Quem manterá e administrará o sistema?
 - Quais os equipamentos que o sistema irá controlar?
 - Com quais outros sistemas o **SeC** precisará interagir?
 - Quem tem interesse nos resultados que o **SeC** irá produzir?

Revisão Aula V

- Relacionamentos entre Atores

Generalização – Relacionamento hierárquico entre dois atores, indicando que o primeiro representa um conceito mais geral que o segundo. No exemplo, todas as propriedades válidas para um **Cliente** também são válidas para uma **Pessoa Física** ou uma **Pessoa Jurídica**.

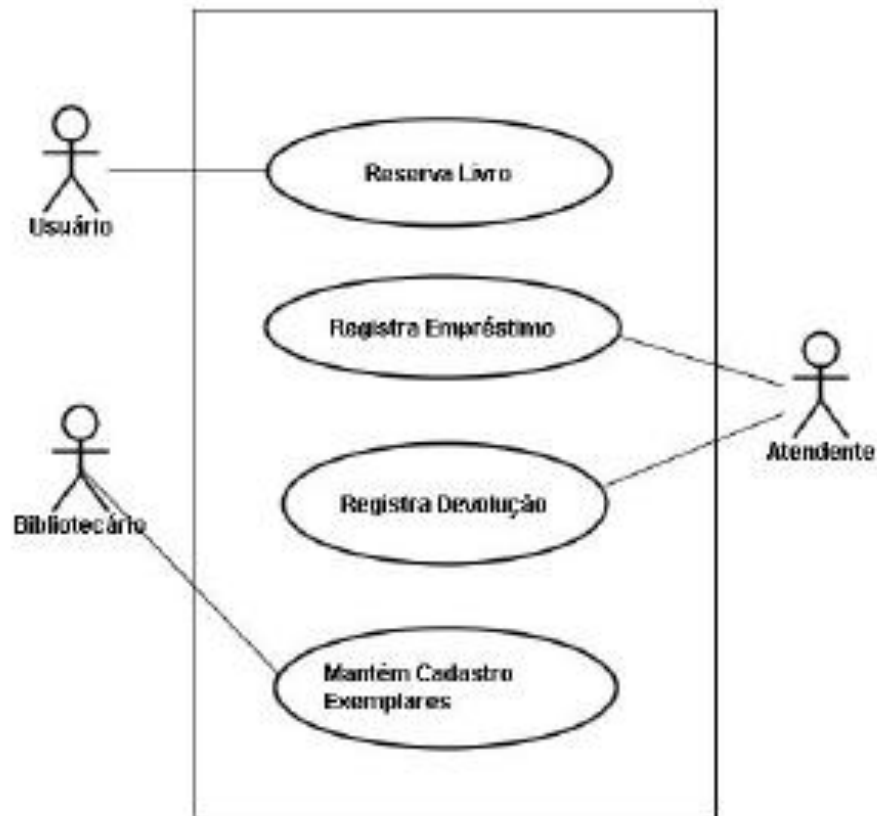


Revisão Aula V

- Casos de Uso
 - Um caso de uso é uma classe e não uma instância;
 - Chamamos de **cenário** a uma instância de um caso de uso;
 - Um caso de uso define um funcionalidade atômica; ou seja, deve ser visto como uma descrição completa do diálogo de um ou mais atores com um sistema;
 - A execução de um caso de uso não estará terminada até que o valor final seja produzido, ou que uma exceção seja levantada.

Revisão Aula V

- Diagrama - Exemplo



Revisão Aula V

- Associação Caso de Uso - Ator
 - Indica que há comunicação entre o caso de uso e o ator;
 - Um ator pode se comunicar com vários casos de uso;
 - O uso de seta de direcionamento (opcional) na associação indica quem iniciou o caso de uso;
 - **Cuidado!!!** As setas de direcionamento **NÃO** representam fluxos de informação;
 - Elas indicam apenas quem iniciou um caso de uso, e isto é tudo. O mais comum é que haja informação fluindo nos dois sentidos.

Diagrama de Caso de Uso

- Ferramenta Case – LucidChat



The image shows a promotional banner for Lucidchart. At the top, there is an orange navigation bar with the Lucidchart logo and links for 'EMPRESA', 'TOUR', 'EXEMPLOS', 'PREÇO', 'CADASTRE-SE DE GRAÇA', and 'LOGIN'. Below the navigation bar, the main content area features a large, semi-transparent image of a laptop displaying a complex flowchart. Overlaid on this image is the text 'DIAGRAMAS BEM FEITOS' in large, white, bold letters. Below this text is a blue play button icon. Further down, there is a white input field for an email address, followed by a blue button labeled 'CADASTRE-SE DE GRAÇA'. Below the email field, there is a small link for 'termos e condições'. At the bottom, there is a white button labeled 'Ou cadastre-se usando Google' with the Google logo.

Lucidchart

EMPRESA TOUR EXEMPLOS PREÇO CADASTRE-SE DE GRAÇA LOGIN

DIAGRAMAS BEM FEITOS

Use o Lucidchart para criar fluxogramas

Insira seu endereço de e-mail CADASTRE-SE DE GRAÇA

Ao cadastrar-se, você concorda com nossos termos e condições

Ou cadastre-se usando Google

Diagrama de Caso de Uso

- Ferramenta Case – LucidChat

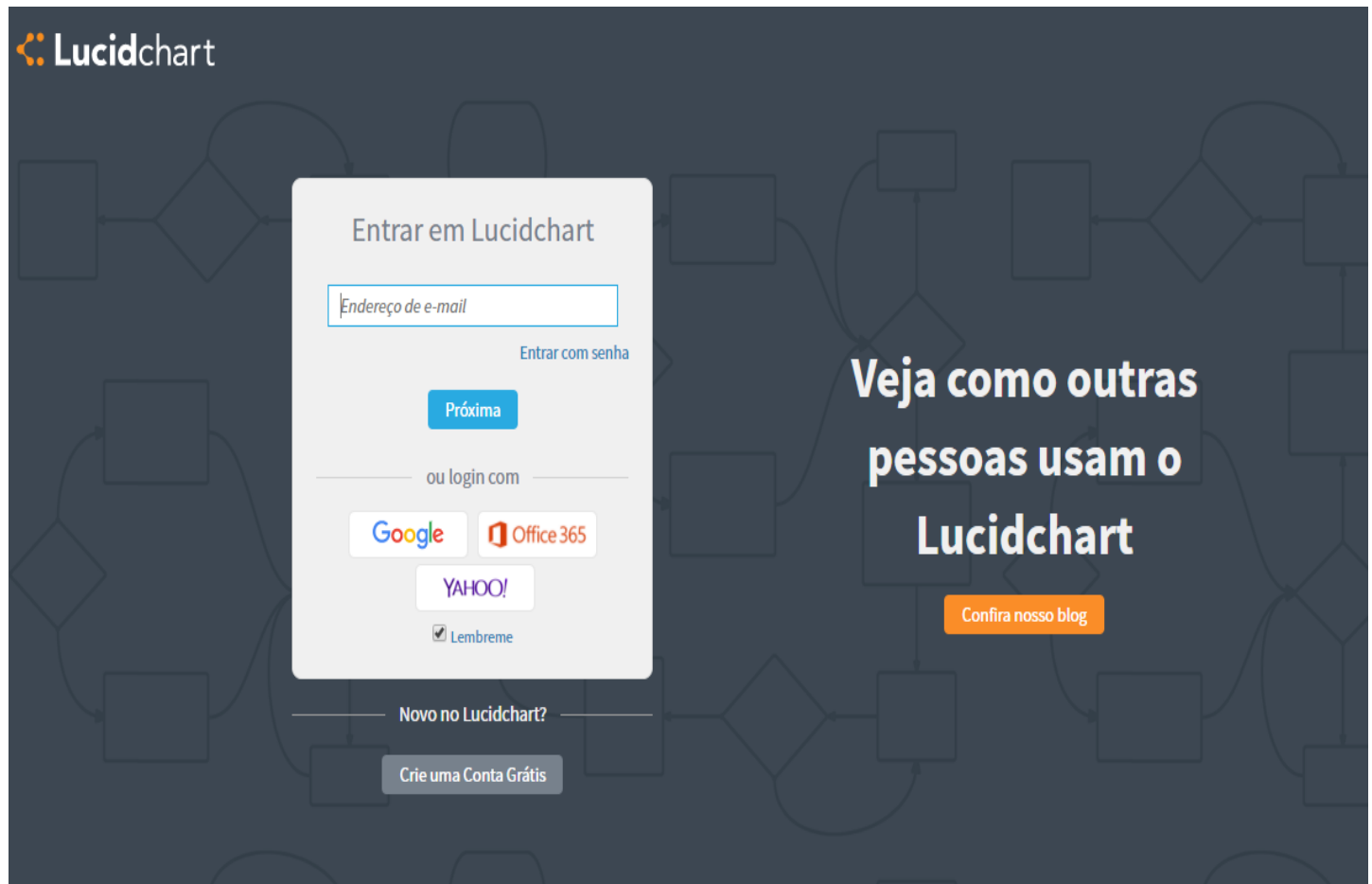


Diagrama de Caso de Uso

- Ferramenta Case – LucidChat

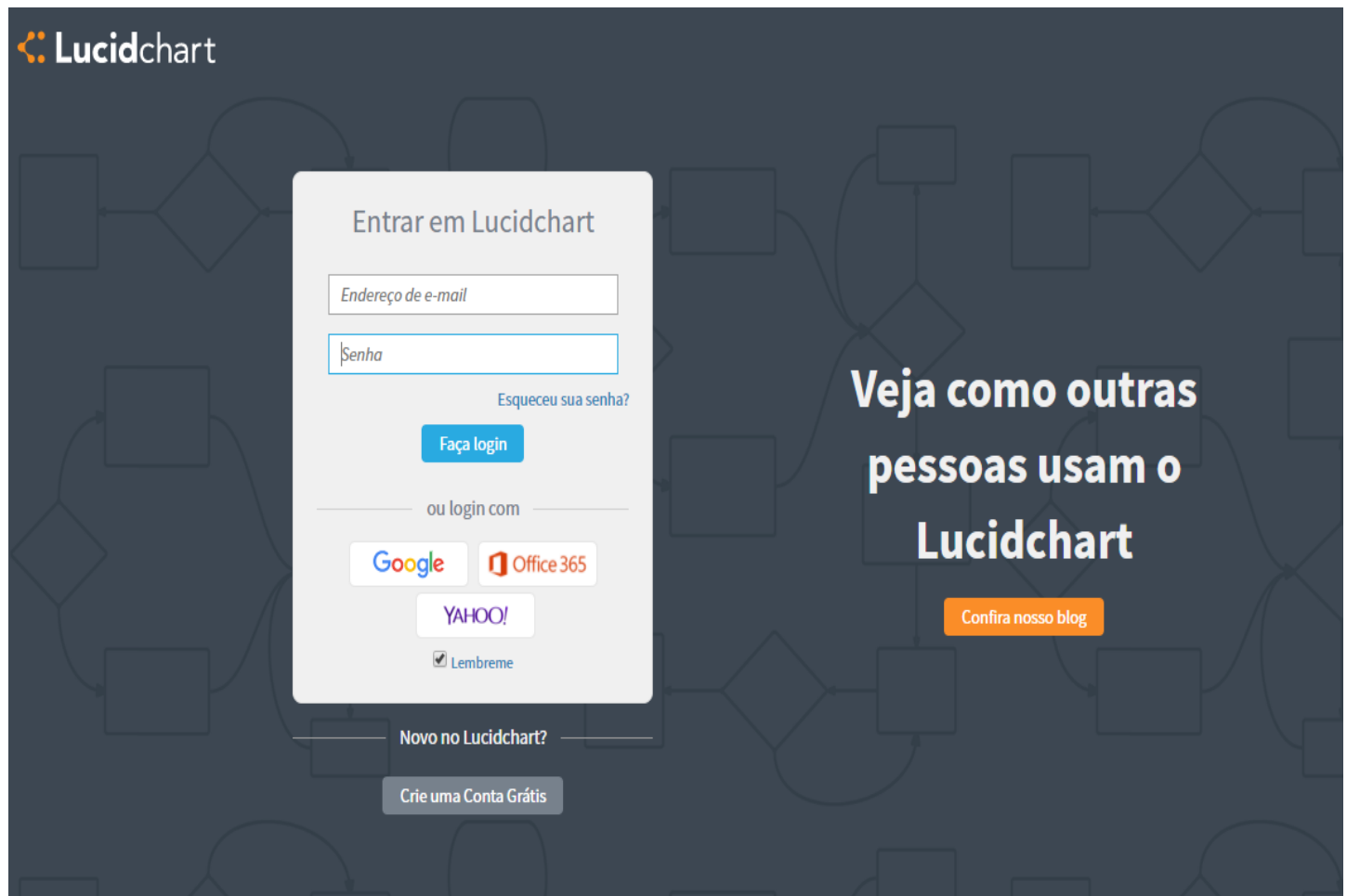
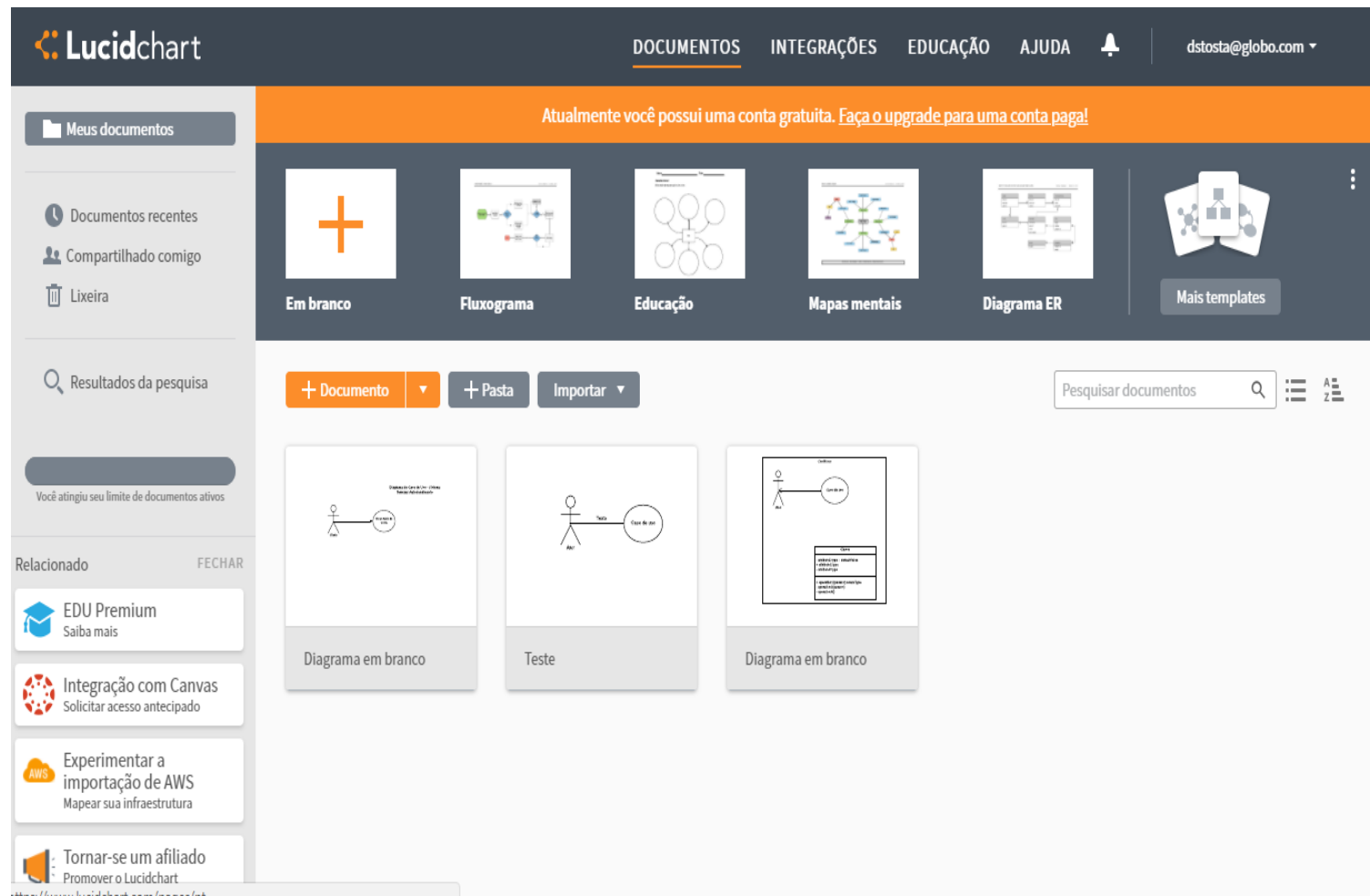


Diagrama de Caso de Uso

- Ferramenta Case – LucidChat



The screenshot displays the Lucidchart web application interface. At the top, the navigation bar includes the Lucidchart logo, a menu with 'DOCUMENTOS', 'INTEGRAÇÕES', 'EDUCAÇÃO', and 'AJUDA', a notification bell, and a user profile dropdown for 'dstosta@globo.com'. Below the navigation bar, a prominent orange banner states: 'Atualmente você possui uma conta gratuita. Faça o upgrade para uma conta paga!'. The main workspace is divided into two sections. The top section features a row of template cards: 'Em branco' (represented by a plus sign), 'Fluxograma' (flowchart), 'Educação' (educational diagram), 'Mapas mentais' (mind map), and 'Diagrama ER' (ER diagram), followed by a 'Mais templates' button. The bottom section contains a search bar labeled 'Pesquisar documentos' and three buttons: '+ Documento', '+ Pasta', and 'Importar'. Below these buttons, three document thumbnails are visible, each with a title: 'Diagrama em branco', 'Teste', and 'Diagrama em branco'. On the left side of the interface, a sidebar provides additional navigation options: 'Meus documentos', 'Documentos recentes', 'Compartilhado comigo', 'Lixeira', and 'Resultados da pesquisa'. At the bottom of the sidebar, a message indicates 'Você atingiu seu limite de documentos ativos'. The 'Relacionado' section at the bottom of the sidebar lists several related services: 'EDU Premium', 'Integração com Canvas', 'Experimentar a importação de AWS', and 'Tornar-se um afiliado'.

Diagrama de Caso de Uso

- Ferramenta Case – LucidChat

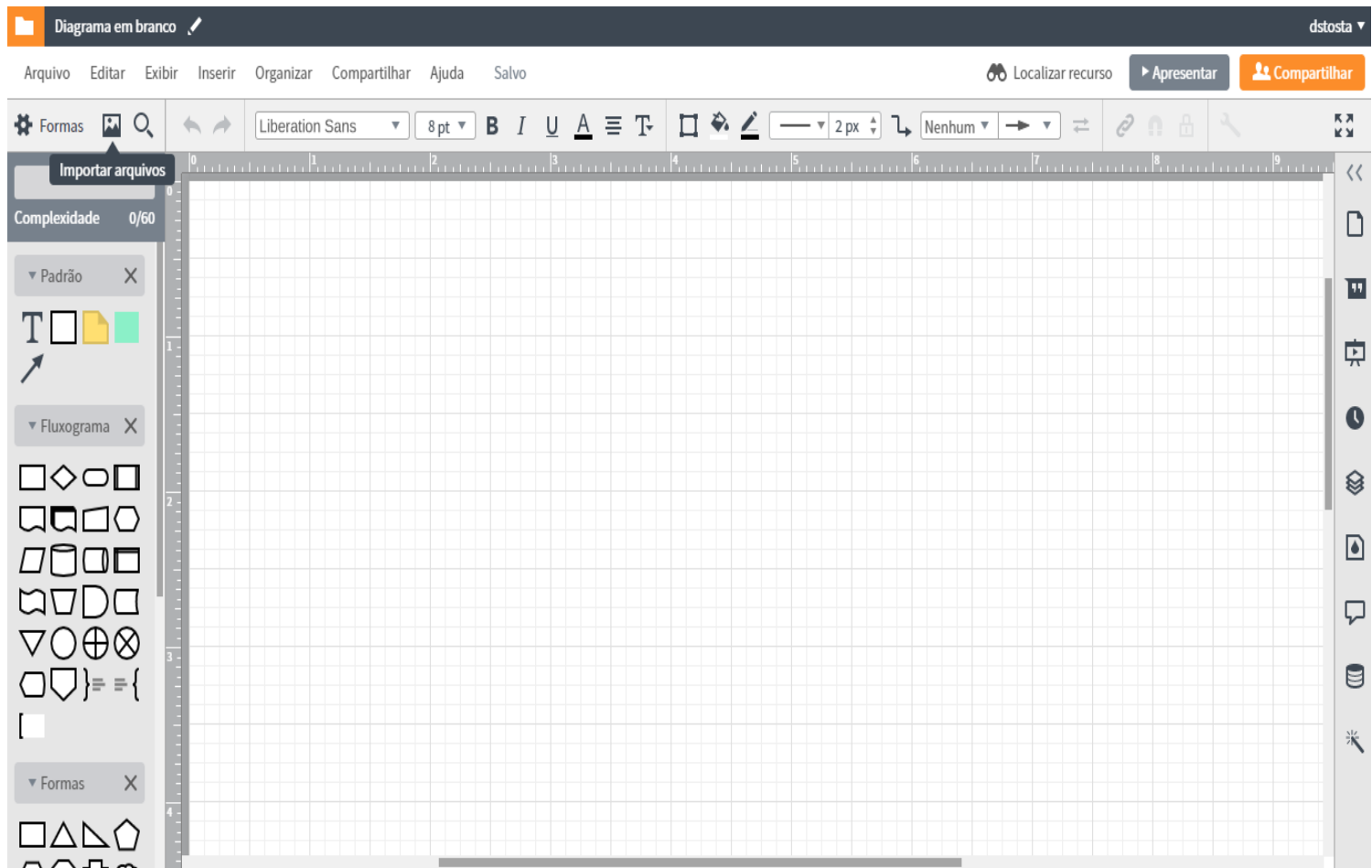


Diagrama de Caso de Uso

- Ferramenta Case – LucidChat

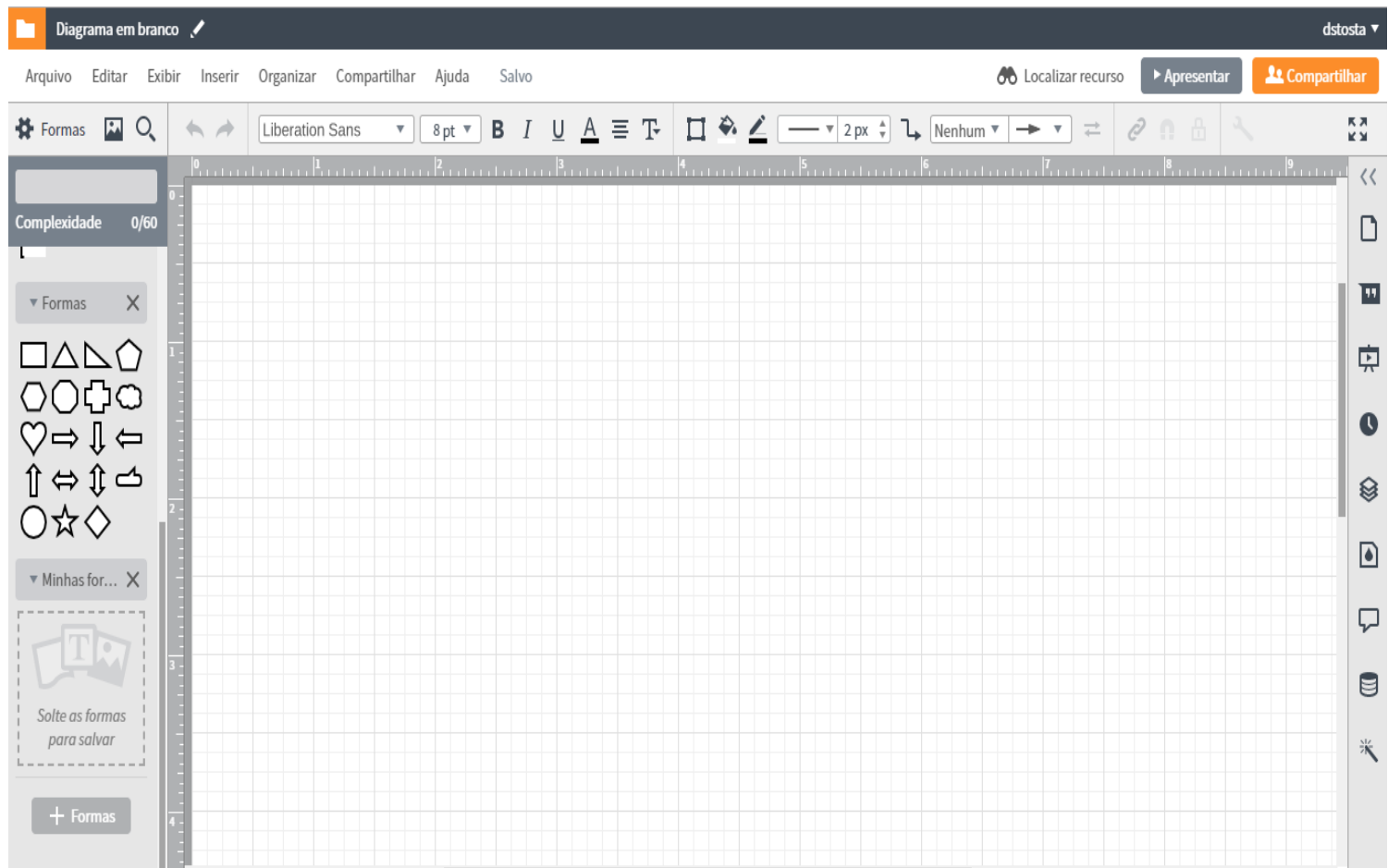


Diagrama de Caso de Uso

- Ferramenta Case – LucidChat

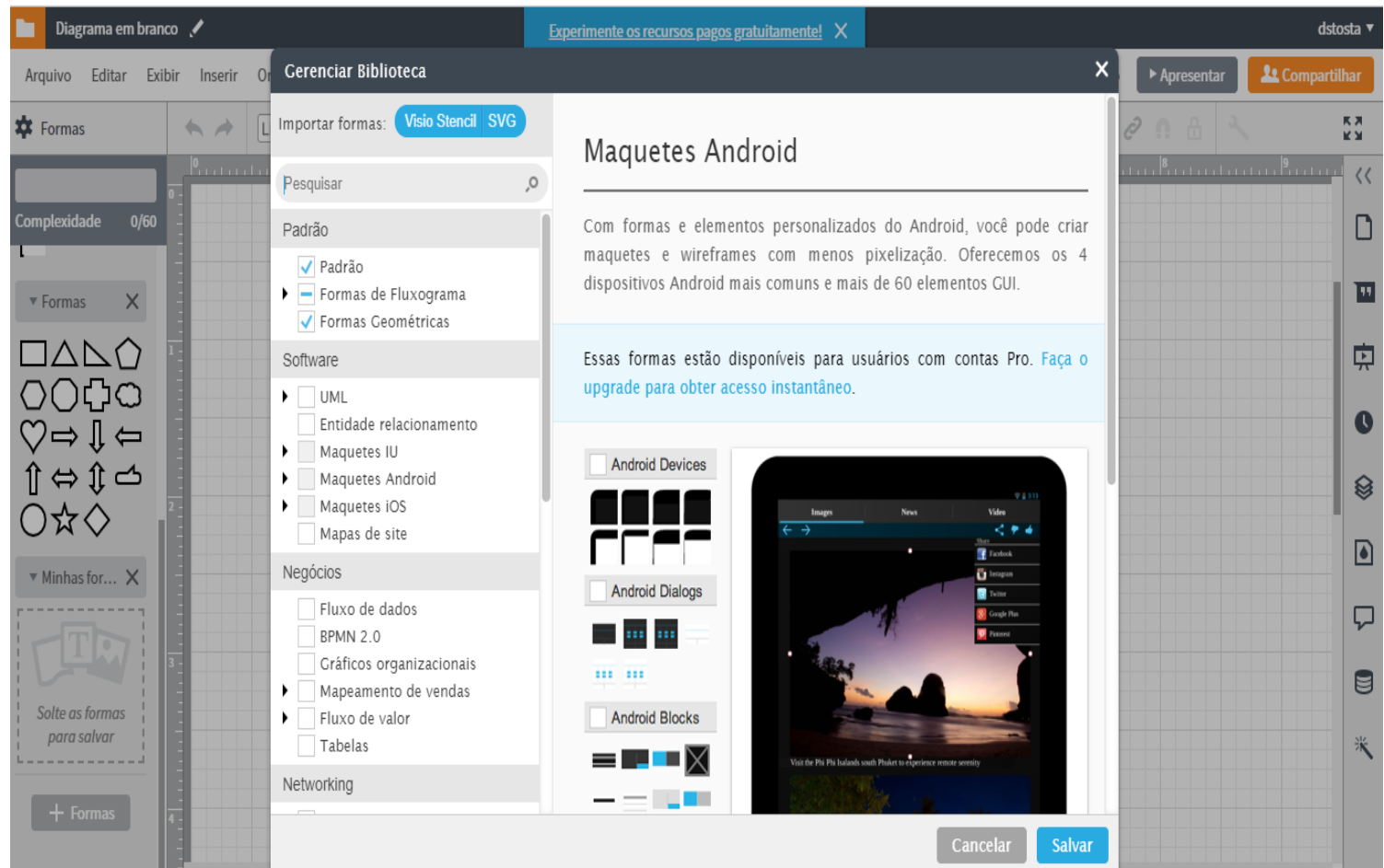
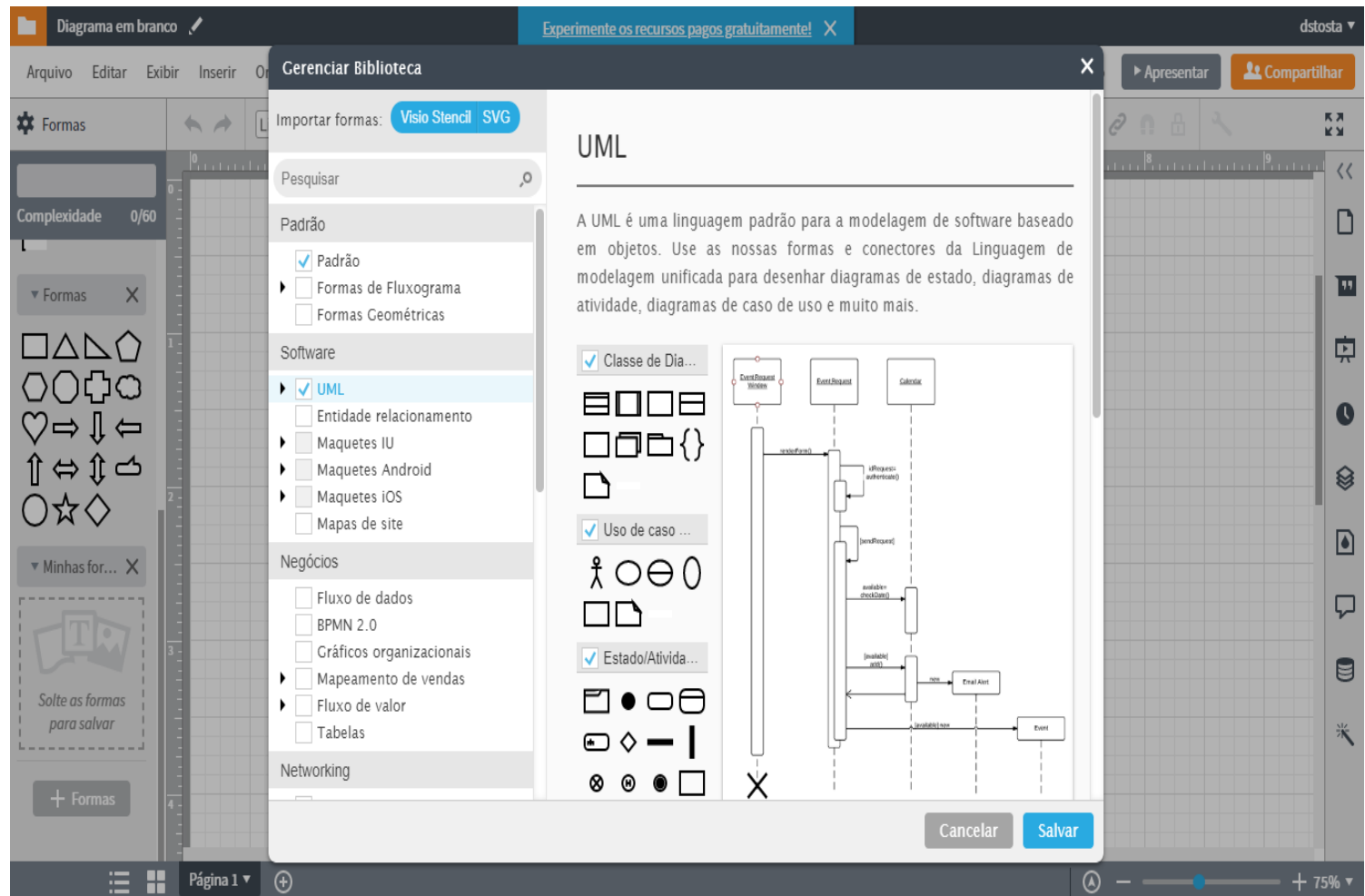


Diagrama de Caso de Uso

- Ferramenta Case – LucidChat



The screenshot displays the LucidChat application interface. On the left, a sidebar contains a 'Formas' (Shapes) panel with various UML symbols and a 'Minhas for...' (My shapes) section. The main workspace is titled 'Diagrama em branco' (Blank diagram). A 'Gerenciar Biblioteca' (Manage Library) window is open, showing the 'UML' category selected. This window lists various UML diagram types, including 'Classe de Diagrama' (Diagram Class), 'Uso de caso' (Use Case), 'Estado/Atividade' (State/Activity), and 'Diagrama de Fluxo' (Flow Diagram). The 'Uso de caso' option is checked. To the right of the library, a text box explains that UML is a standard language for software modeling. Below the text, a sample Use Case diagram is shown, illustrating a process flow involving 'Event Request', 'Event Request', 'Collection', 'Email Alert', and 'Event' components. The interface includes standard menu options like 'Arquivo', 'Editar', 'Exibir', and 'Inserir', as well as a 'Pesquisar' (Search) bar and a 'Complexidade' (Complexity) indicator.

Diagrama de Caso de Uso

- Ferramenta Case – LucidChat

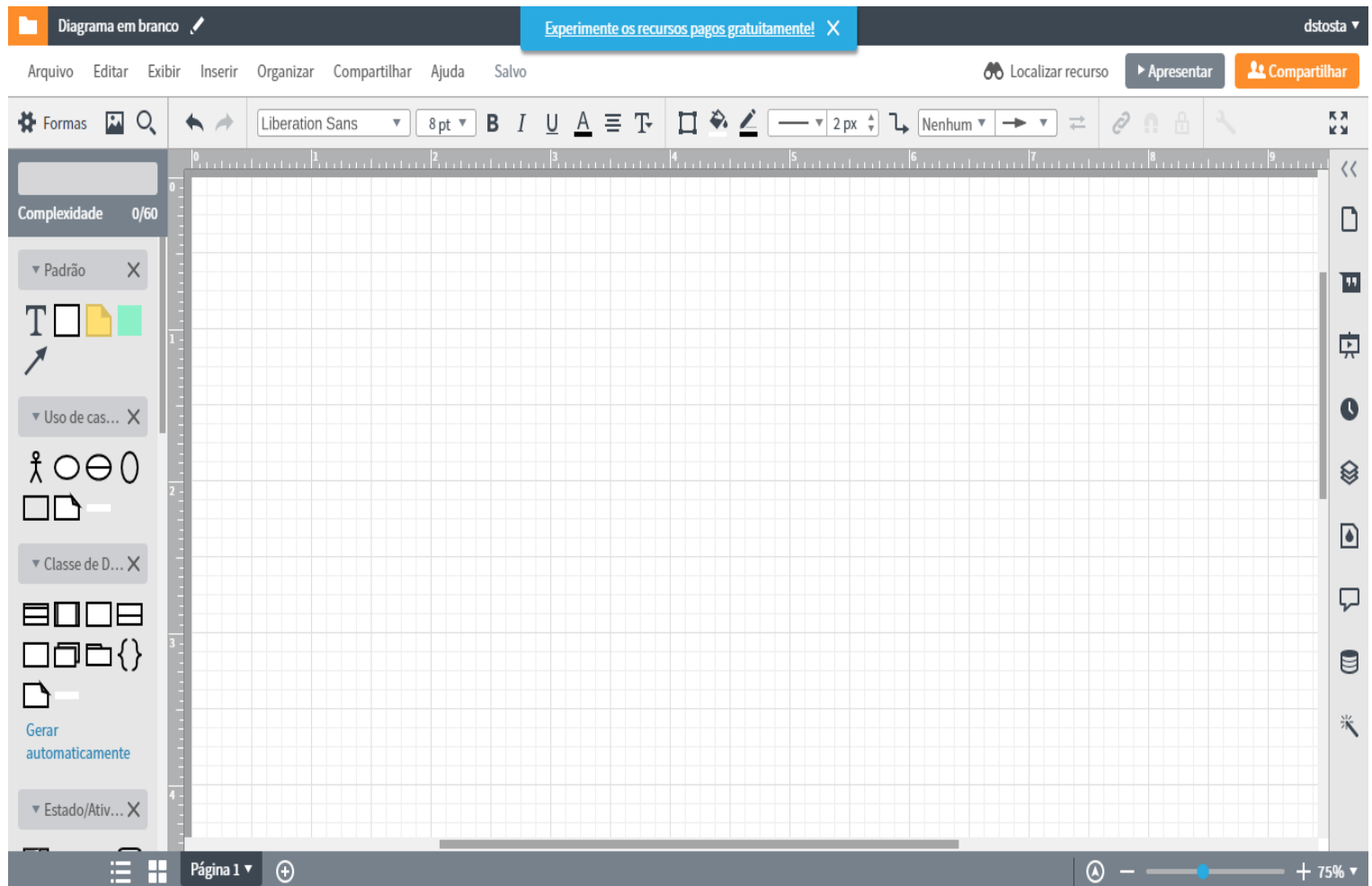


Diagrama de Caso de Uso

- Ferramenta Case – LucidChat

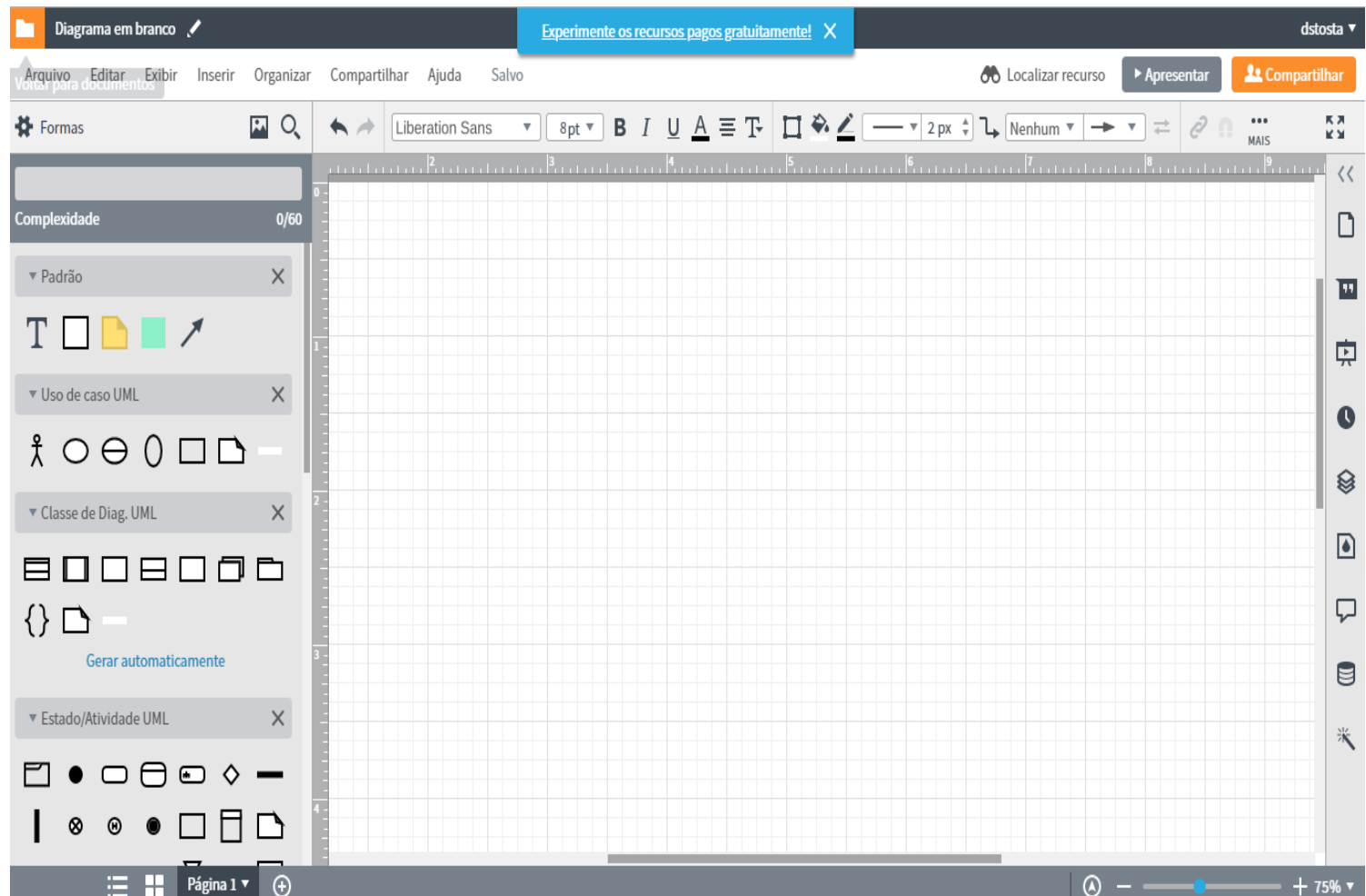


Diagrama de Caso de Uso

- Ferramenta Case – LucidChat

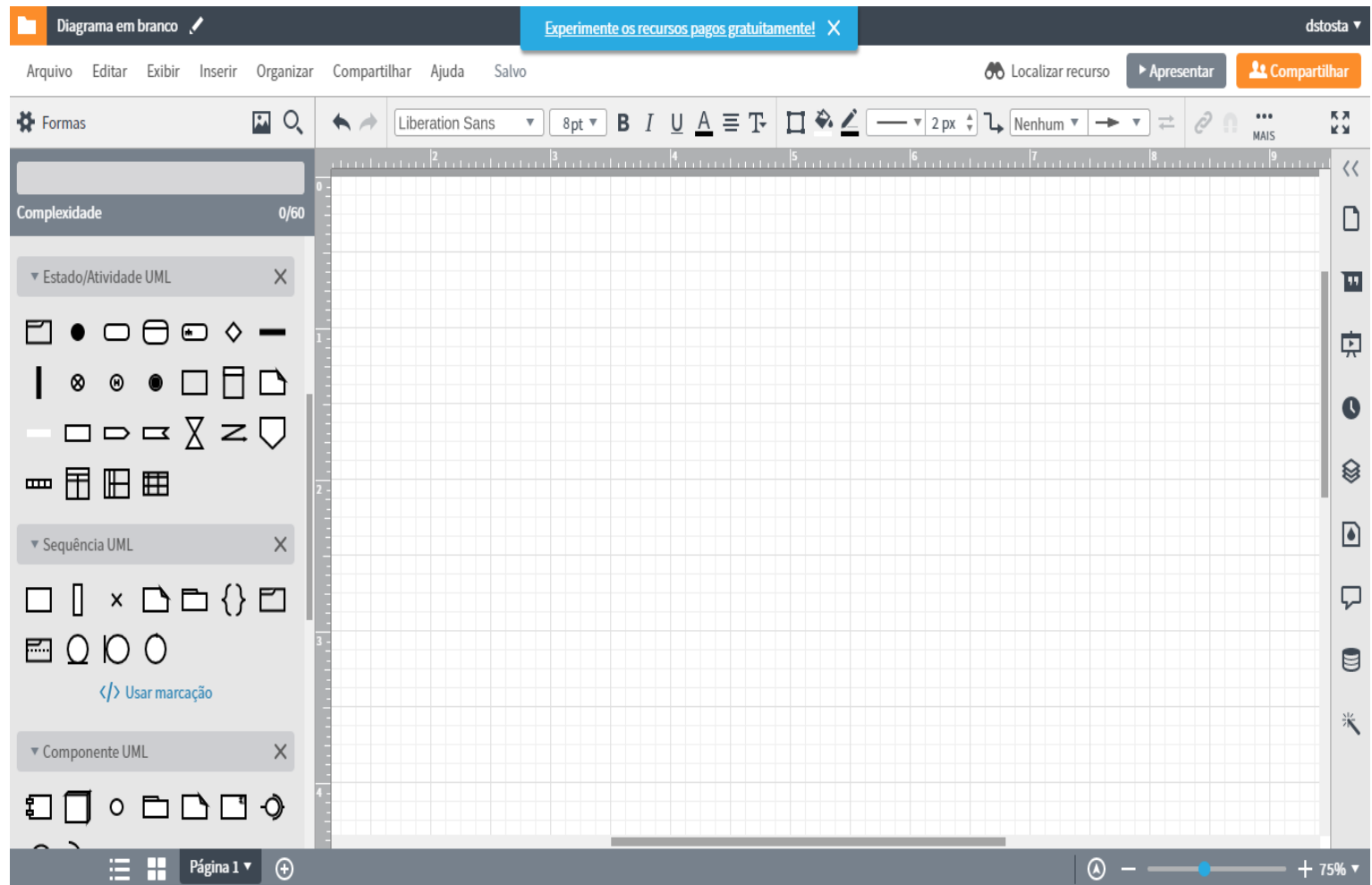


Diagrama de Caso de Uso

- Ferramenta Case – LucidChat

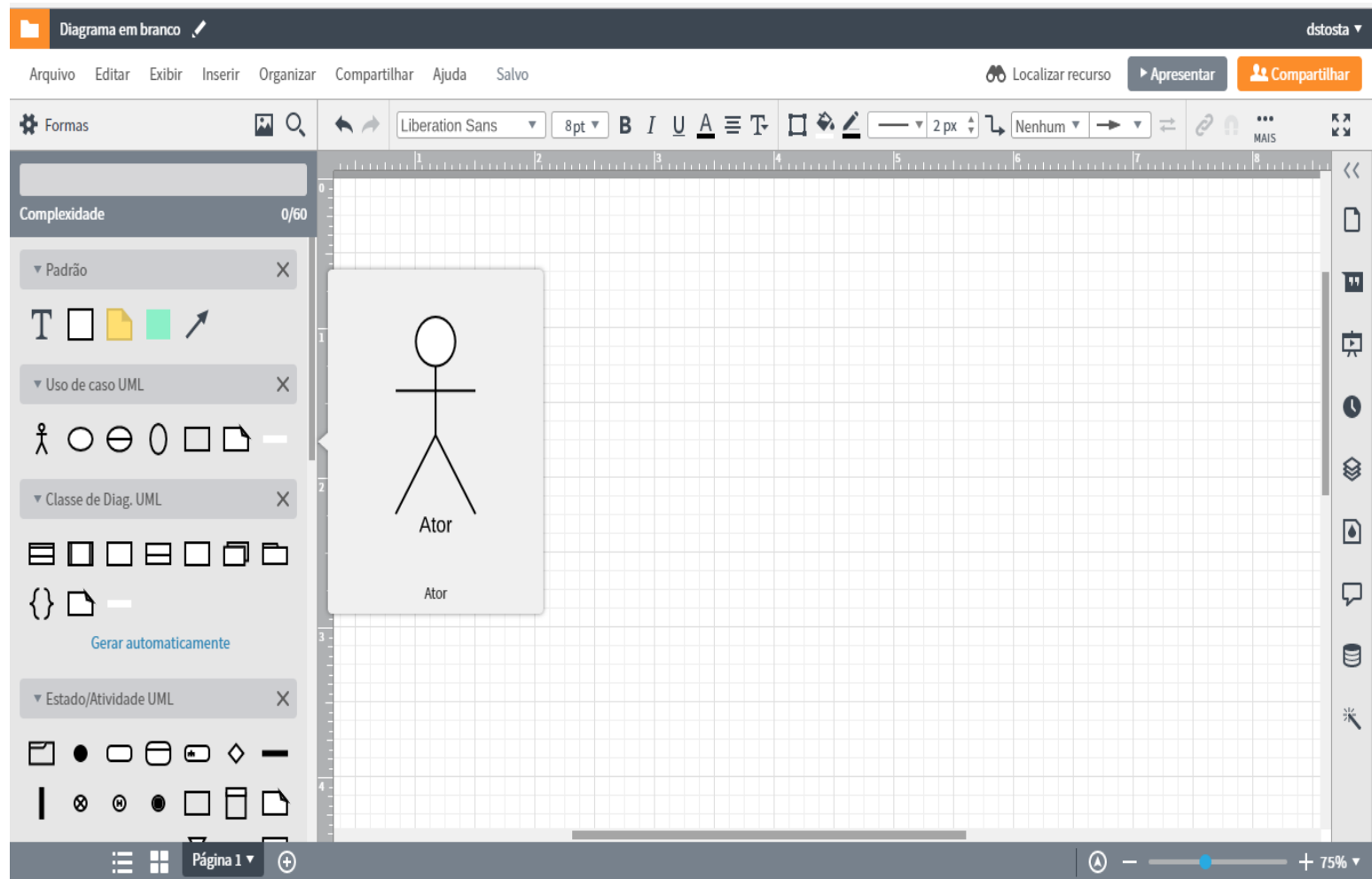


Diagrama de Caso de Uso

- Ferramenta Case – LucidChat

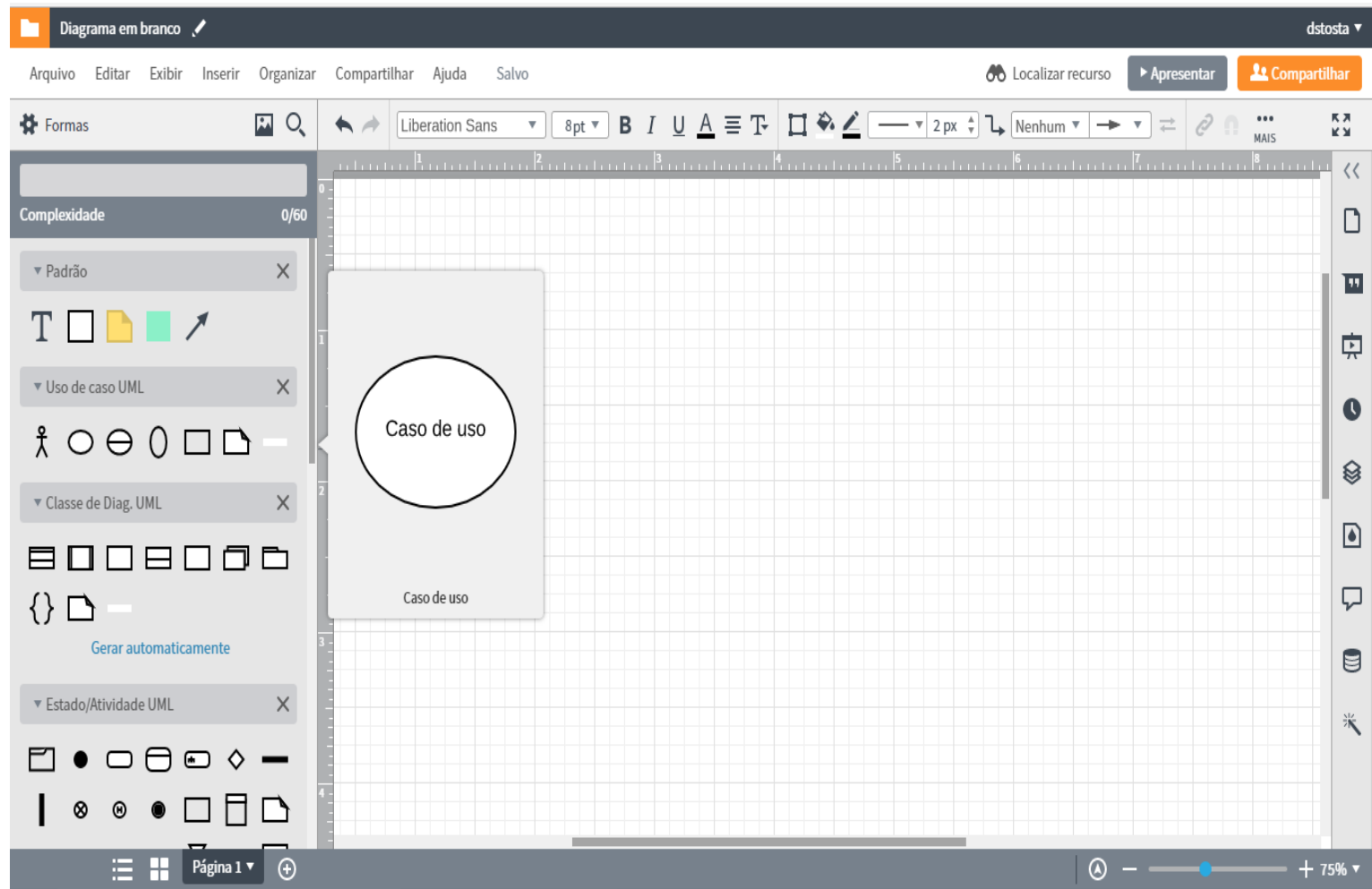


Diagrama de Caso de Uso

- Ferramenta Case – LucidChat

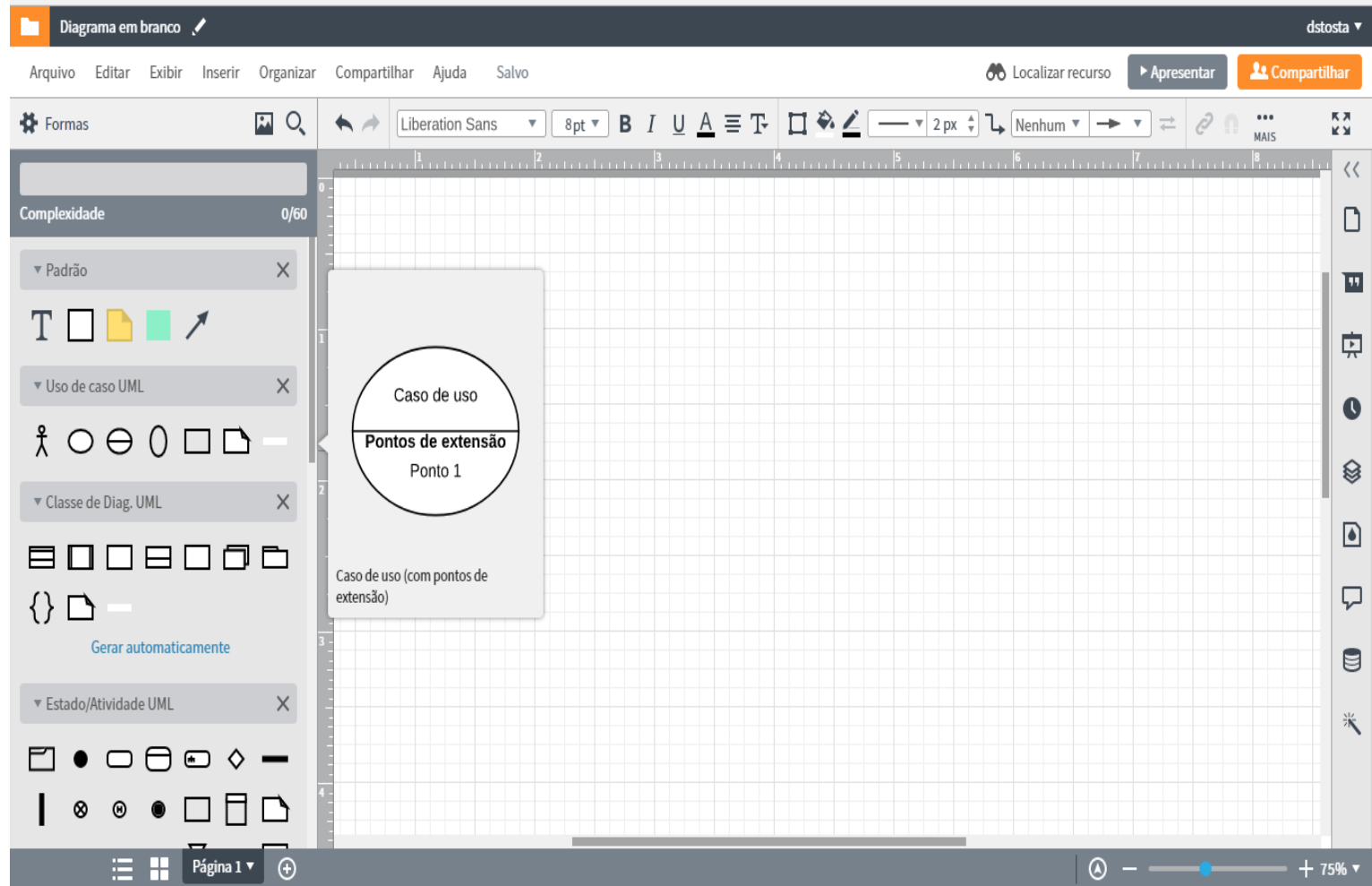


Diagrama de Caso de Uso

- Ferramenta Case – LucidChat

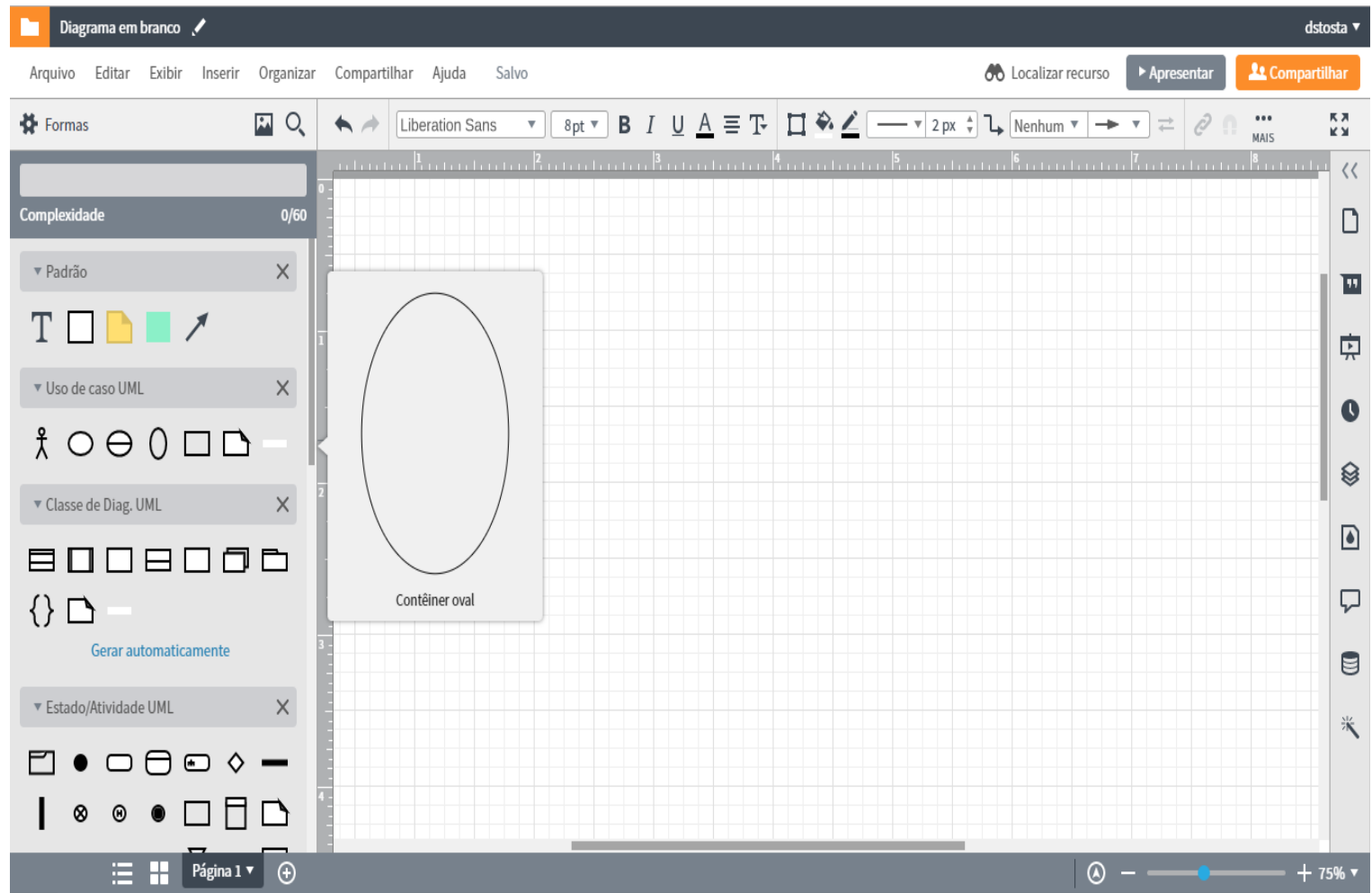


Diagrama de Caso de Uso

- Ferramenta Case – LucidChat

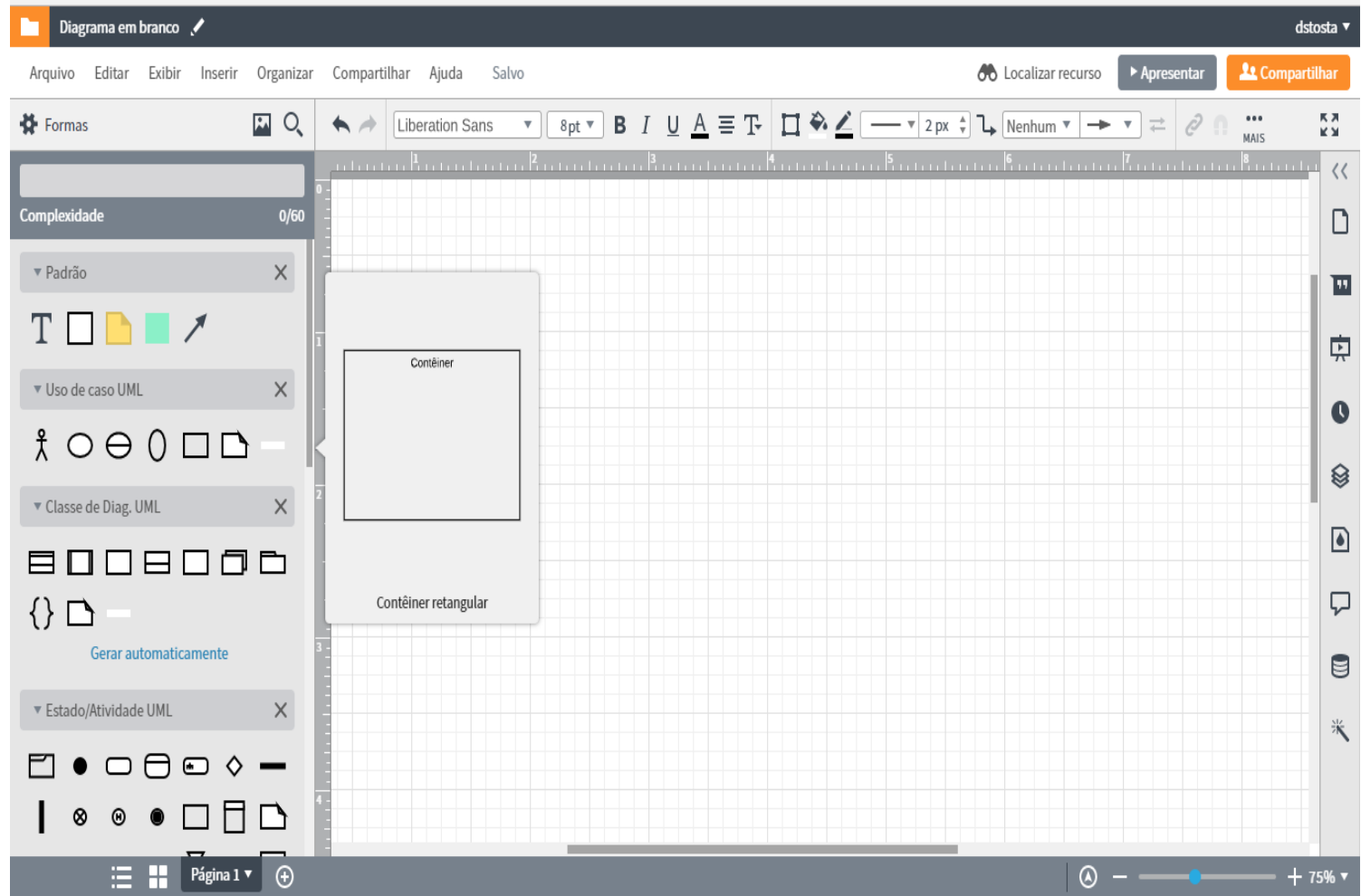
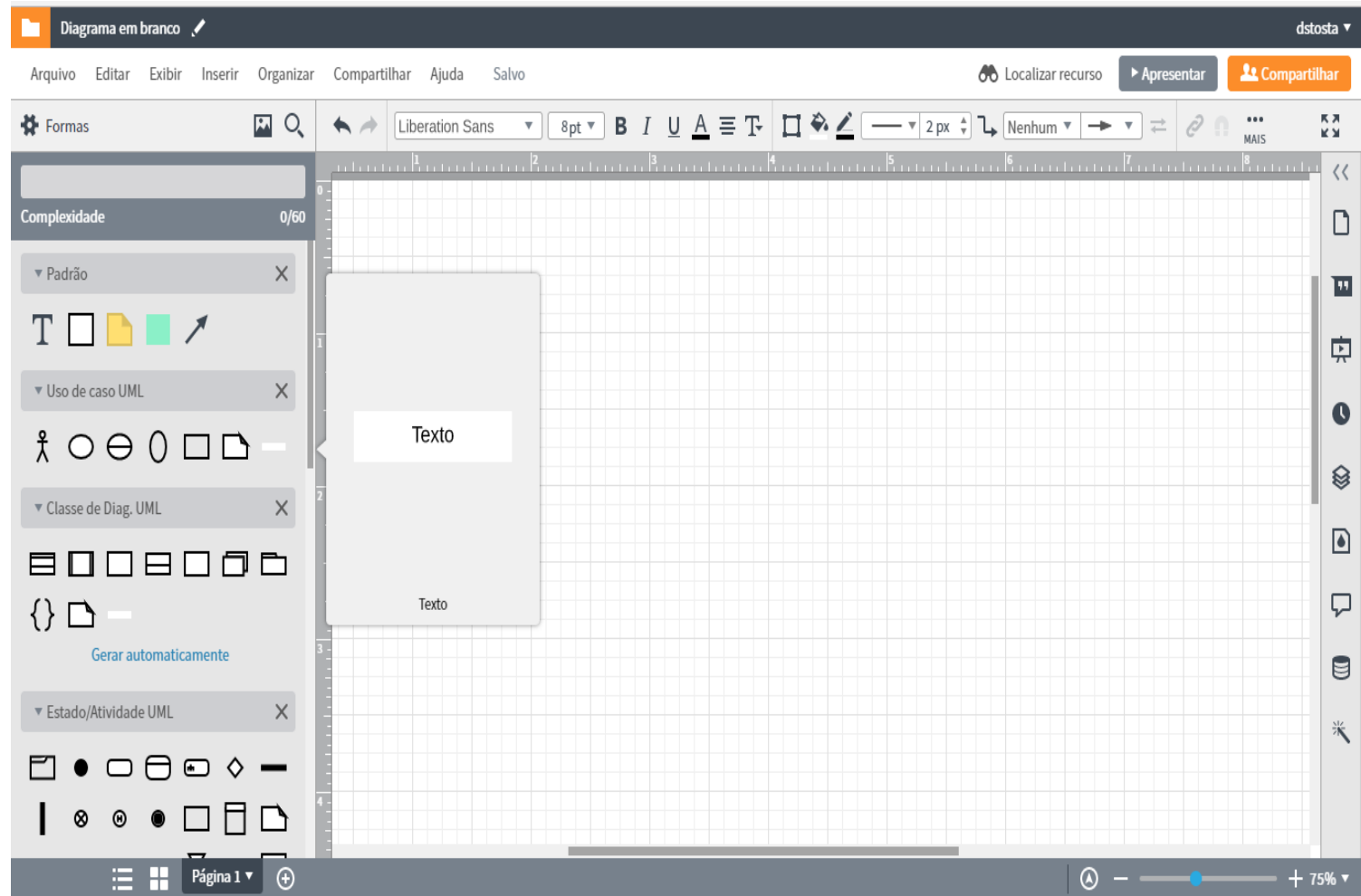


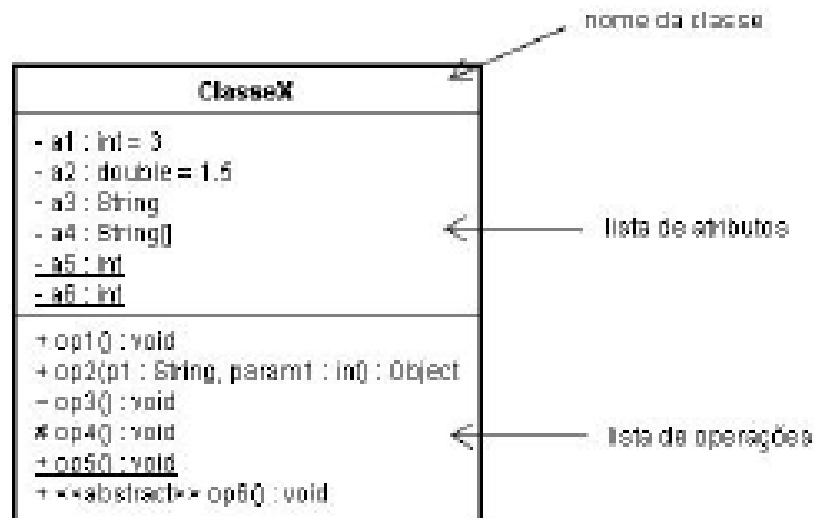
Diagrama de Caso de Uso

- Ferramenta Case – LucidChat



Classes e Objetos

- O que é uma classe ?
 - Uma classe descreve um conjunto de objetos que têm os mesmos atributos, as mesmas operações e a mesma **semântica**;
 - Graficamente uma classe é representada por um retângulo dividido em três seções: nome, lista de atributos e lista das operações.



Classes e Objetos

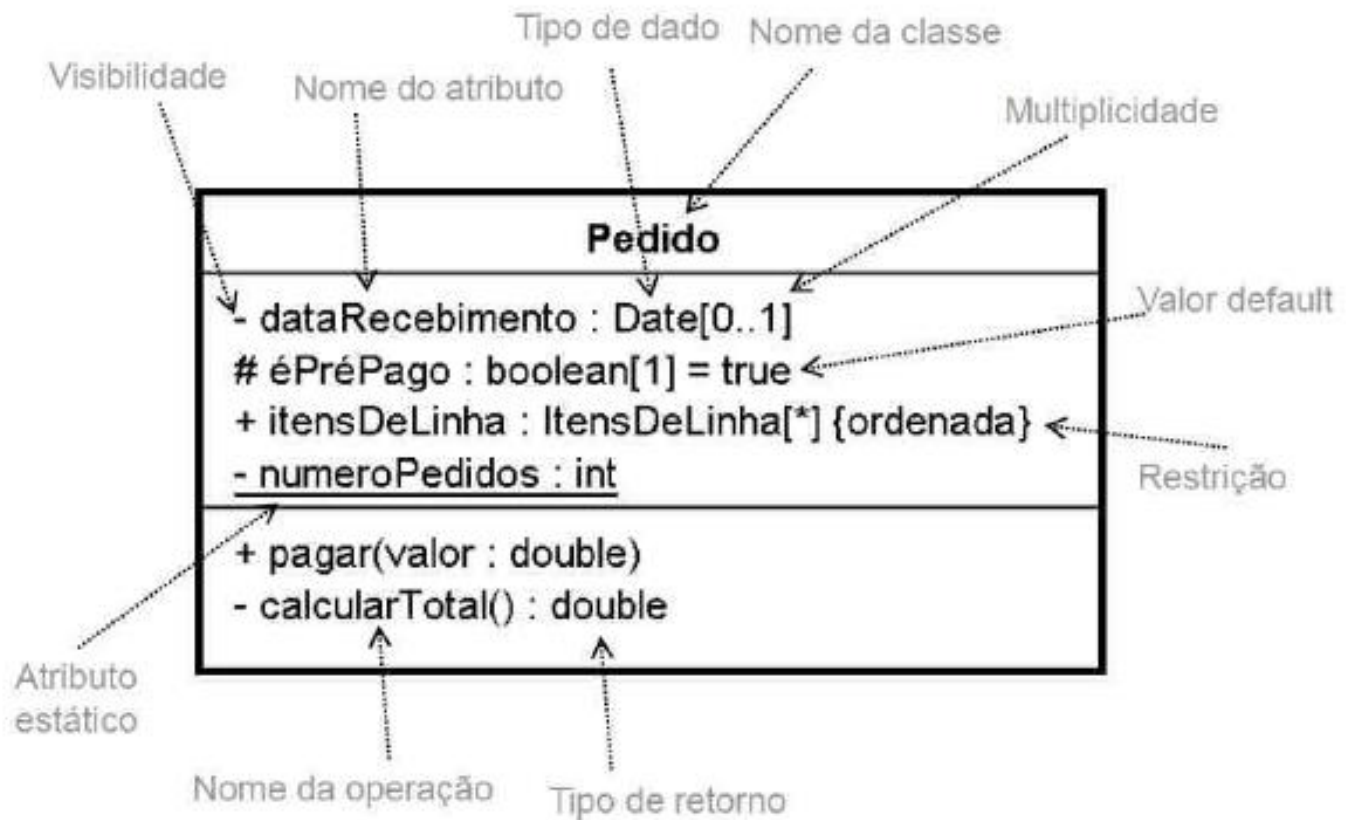
- O que é uma classe ?
 - A UML não define nenhuma regra para a semântica do nome de uma classe;
 - O nome deve ser escolhido de acordo com o vocabulário do domínio do problema ou com a função exercida pela classe;
 - Normalmente, os nomes são substantivos no singular com a primeira letra em caixa alta.

Aluno
- nome : String - matricula : String
+ getCR() : double + getTurmasPeriodo() : Vector

ListaAlunos
- alunos : Vector
+ insere(x : Aluno) : boolean + altera(x : Aluno) : boolean + exclui(matricula : int) : boolean

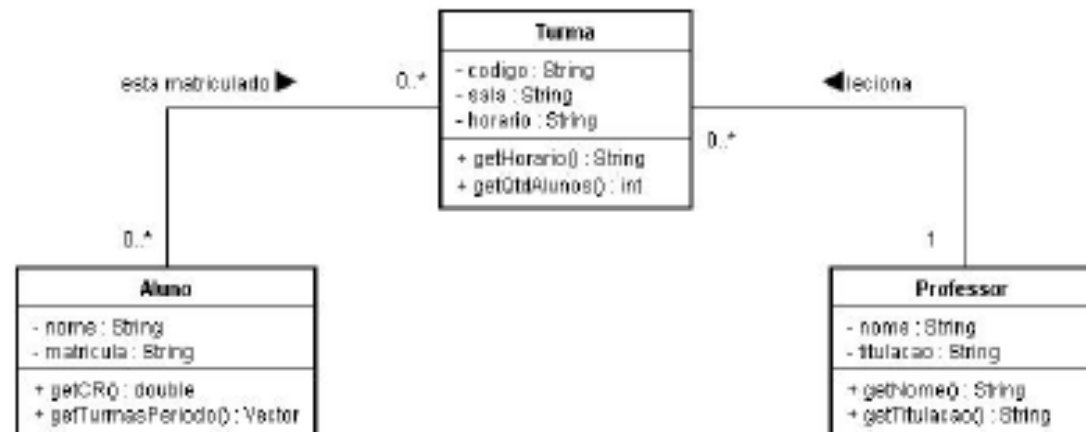
Classes e Objetos

- Estrutura de uma Classe



Classes e Objetos

- Diagrama de classe
 - Descreve a visão estática de um sistema através das classes e dos relacionamentos entre elas;
 - É composto de um ou mais diagramas de classes;
 - É o modelo central da abordagem orientada a objetos.



Classes e Objetos

- Exemplo de uma classe em Java

```
public class Pessoa {  
    String nome;  
    | double dinheiroNaCarteira;  
  
    public Pessoa(String nome) {  
        this.nome = nome;  
    }  
  
    public void gastar(double valor) {  
        dinheiroNaCarteira -= valor;  
    }  
  
    public void receber(double valor) {  
        dinheiroNaCarteira += valor;  
    }  
}
```


Classes e Objetos

- O que é um objeto ?
 - Um objeto, em programação orientada a objetos, é uma instância (ou seja, um exemplar) de uma classe
 -
 - Chama-se instância de uma classe, um objeto cujo comportamento e estado são definidos pela classe.
 - As instâncias de uma classe compartilham o mesmo conjunto de atributos, embora sejam diferentes quanto ao conteúdo desses atributos.

Classes e Objetos

- Exemplo de um objeto (classe pessoa)

```
public static void main(String args[]) {  
  
    // Cria um Objeto de Pessoa.  
    Pessoa pVitor;  
    pVitor = new Pessoa("Vitor Fernando Pamplona");  
}
```

Classes e Objetos

- **Atributo**

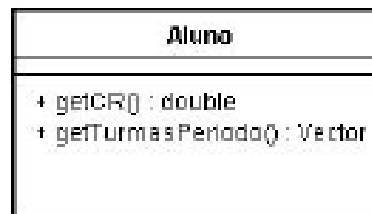
- É uma propriedade nomeada de uma classe que define um domínio de valores que um objeto – instância da classe – pode armazenar;
- A definição de um atributo é válida para todos os objetos de uma classe;
- Os valores de todos os atributos de um objeto determinam o estado do mesmo.

Aluno
- nome : String - matricula : String

Classes e Objetos

- Operações - Métodos

- É a declaração de um serviço que pode ser solicitado às instâncias de uma classe;
- A declaração de uma operação é válida para todos os objetos de uma classe;
- Uma implementação de uma operação é chamada de **método**.



Classes e Objetos

- Visibilidade
 - Determina o nível de restrição de acesso (referência) a alguns elementos de modelagem da UML;
 - Aplicável às classes, às operações e aos atributos;
 - Tipos de visibilidade:
 - pública "+"
 - protegida "#"
 - privada "-"
 - pacote "~"

Classes e Objetos

- **Visibilidade**
 - **público** – pode ser referenciado por qualquer elemento de modelagem externo ao classificador em questão;
 - **protegido** – pode ser referenciado pelos elementos de modelagem internos ao classificador em questão e pelos descendentes deste (generalização);
 - **privado** – pode ser referenciado **apenas** pelos elementos de modelagem internos ao classificador em questão;
 - **pacote** – pode ser referenciado por qualquer elemento de modelagem declarado no mesmo pacote do classificador em questão. Para os elementos externos, o elemento é tratado como **privado**.

Classes e Objetos

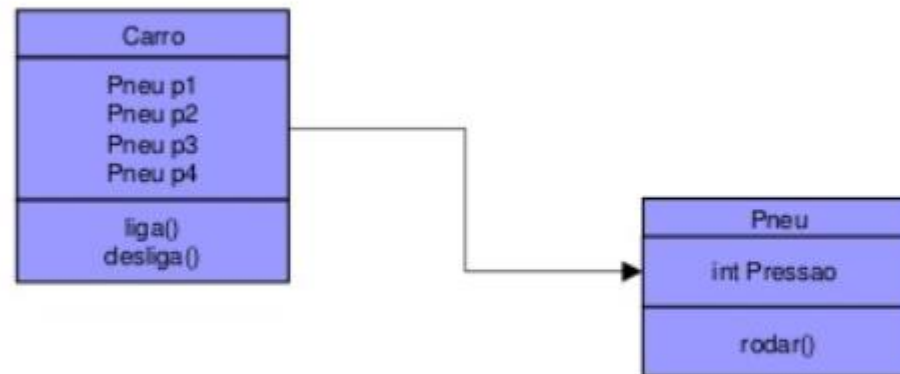
- Associação

- Uma associação é um relacionamento estrutural cujo significado – semântica – é representativo para o problema sendo modelado;
- Uma associação representa um conjunto de ligações entre os objetos das classes que participam da associação;
- Ela define as regras de conexão entre estes objetos;
- Uma associação binária é representada por uma linha sólida que conecta as duas classes envolvidas.



Classes e Objetos

- Associação – Entendendo melhor
 - Associação ocorre quando uma classe possui atributos do tipo de outra classe.



Nota : Neste caso estamos dizendo que carro possui pneu (4 pneus)

Classes e Objetos

- Associação – Entendendo melhor

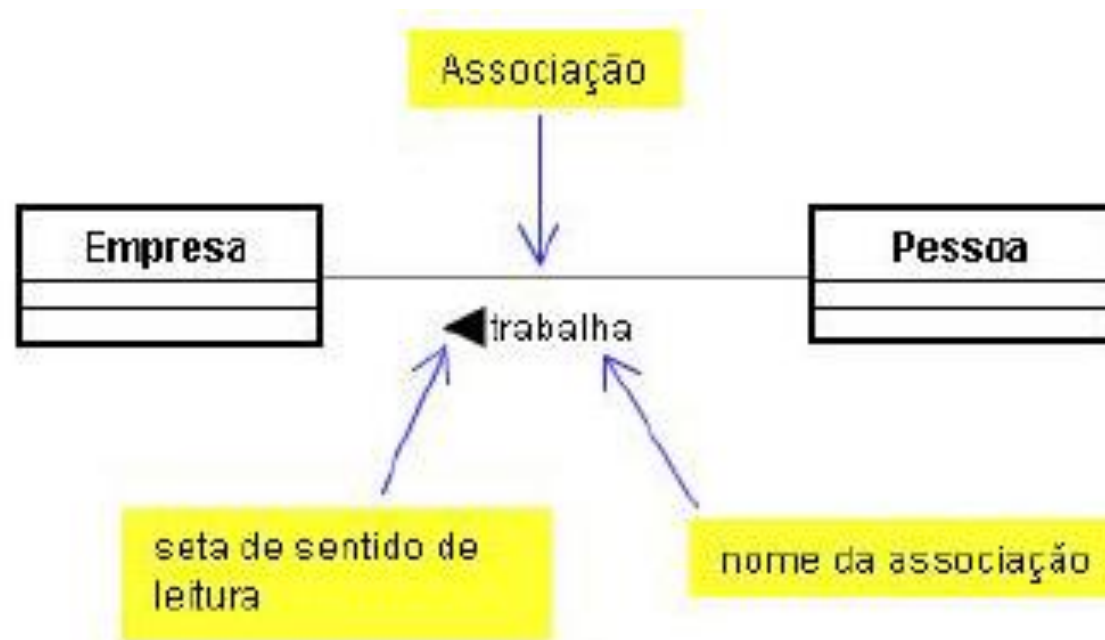
- A associação pode ser representada em Java da seguinte forma:

```
public class Pneu {  
    int Pressao;  
  
    void roda() {  
        System.out.println("Pneu em movimento");  
    }  
}
```

```
public class Carro {  
    Pneu p1;  
    Pneu p2;  
    Pneu p3;  
    Pneu p4;  
  
    void liga() {  
        System.out.println("Carro ligado");  
    }  
  
    void desliga() {  
        System.out.println("Carro desligado");  
    }  
}
```

Classes e Objetos

- Associação - Exemplo

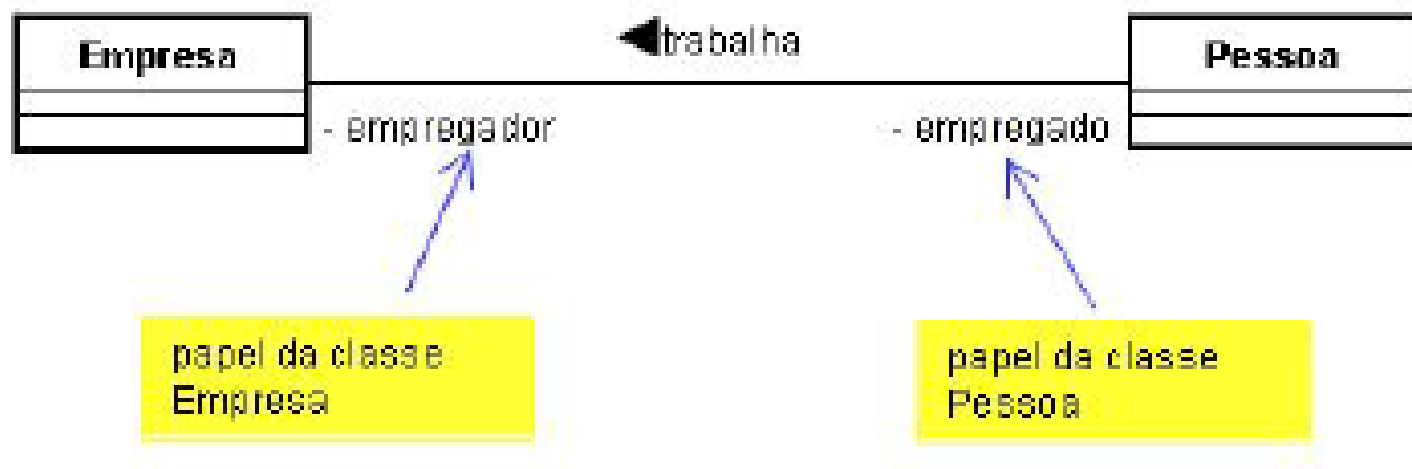


Classes e Objetos

- **Papeis**
 - Quando uma classe participa de uma associação, ela desempenha um **papel** nesse relacionamento;
 - O papel evidencia a finalidade ou função de cada classe de uma associação;
 - Uma mesma classe pode desempenhar vários papéis em diversas associações;
 - Na maioria da vezes, o próprio nome da classe define o seu papel na associação.

Classes e Objetos

- Papeis



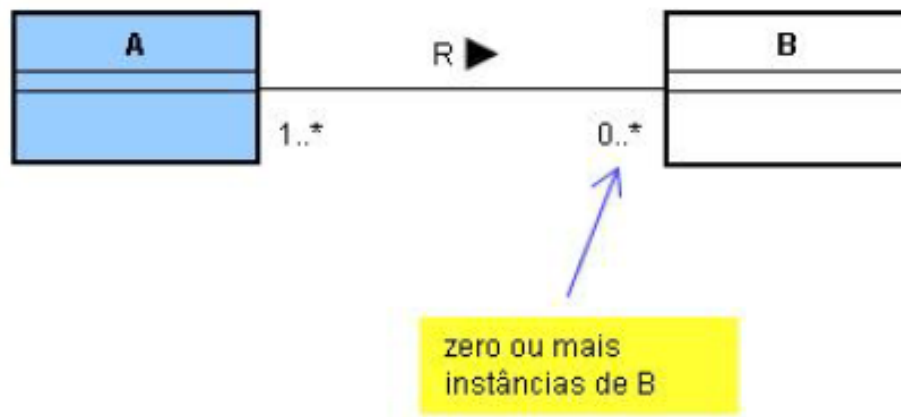
Classes e Objetos

- Multiplicidades
 - O conceito de **multiplicidade** é muito importante na definição das regras que governam as associações:

A **multiplicidade** de uma extremidade de uma associação define o número de objetos da classe associada a esta extremidade que podem estar conectados a **uma** dada instância da classe associada à extremidade oposta.

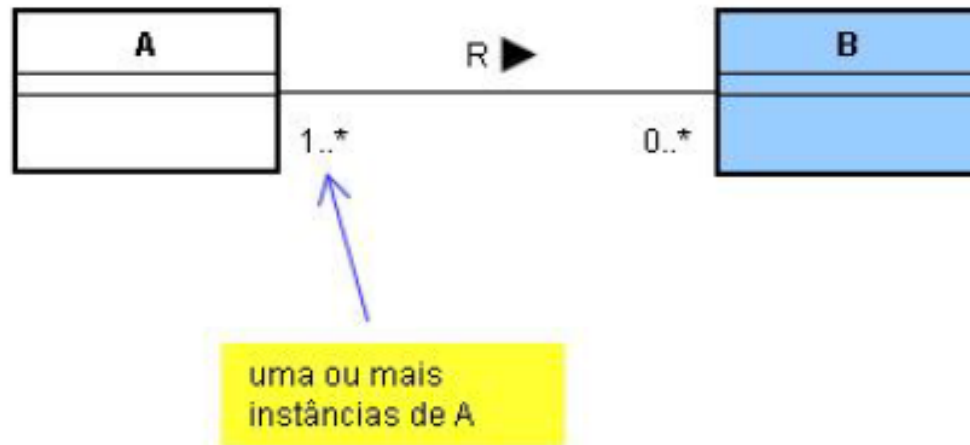
Classes e Objetos

- Multiplicidades
 - No exemplo abaixo **uma instância de A** pode estar conectada, através da relação **R**, a **zero ou mais instâncias de B**.



Classes e Objetos

- Multiplicidades
 - No sentido inverso, **uma instância de B** pode estar conectada, através da relação **R**, a **uma ou mais instâncias de A**.



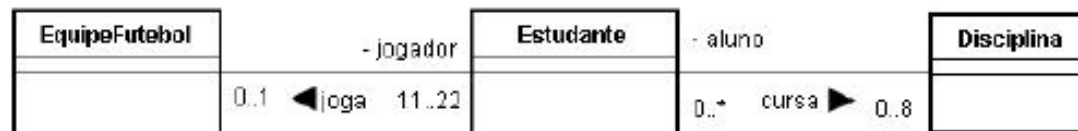
Classes e Objetos

- Indicadores de Multiplicidades
 - Indicadores de multiplicidade mais comuns:
 - **1** exatamente um
 - **1..*** um ou mais
 - **0..*** zero ou mais (muitos)
 - ***** zero ou mais (muitos)
 - **0..1** zero ou um
 - **m..n** faixa de valores (por exemplo: 4..7)

Classes e Objetos

- Multiplicidades - Exemplo

- Um **estudante** pode pertencer ou não a uma **equipe de futebol**;
- Um **equipe de futebol** tem entre 11 e 22 **estudantes** jogando;
- Um **estudante** pode cursar no máximo oito **disciplinas**, ou estar com a matrícula trancada;
- Uma **disciplina** pode ser cursada por um número indeterminado de **estudantes**.

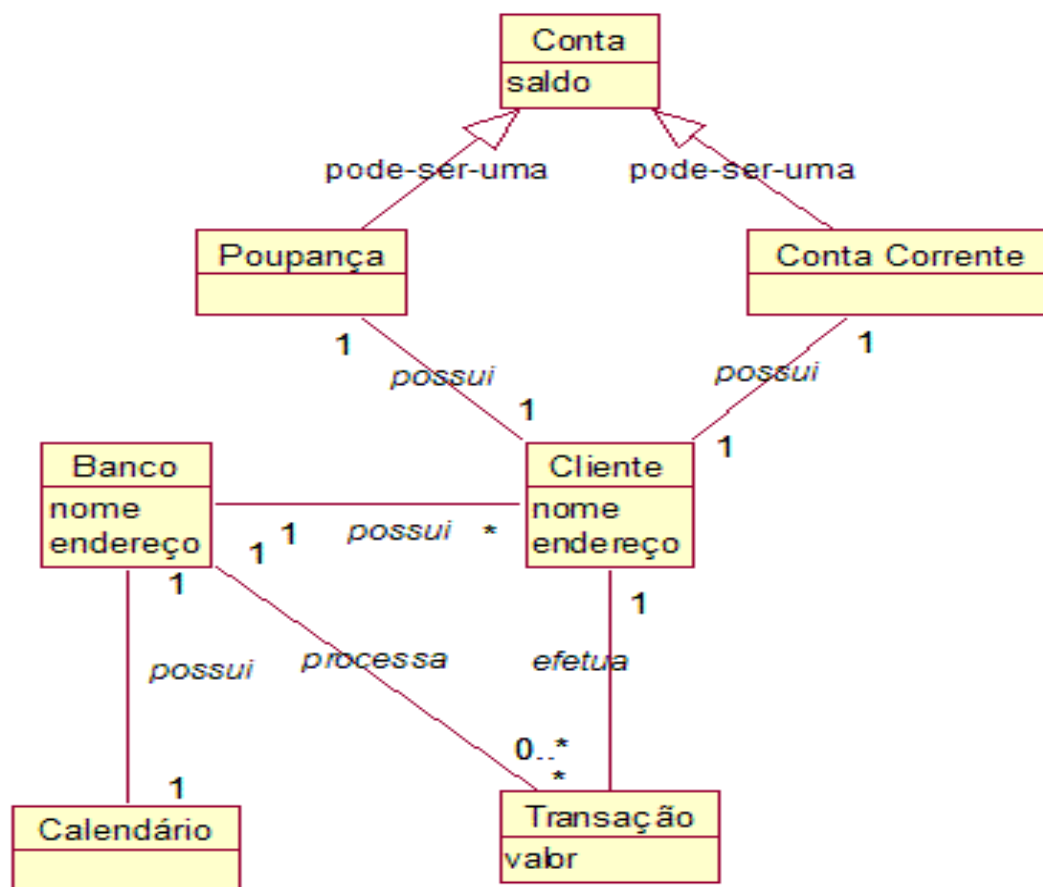


Classes e Objetos

- Diagrama de Classes
 - Um diagrama de classes pode oferecer três perspectivas, cada uma para um tipo de observador diferente. São elas:
 - Conceitual - Representa os conceitos do domínio em estudo.
Perspectiva destinada ao cliente.

Classes e Objetos

- Diagrama de Classes

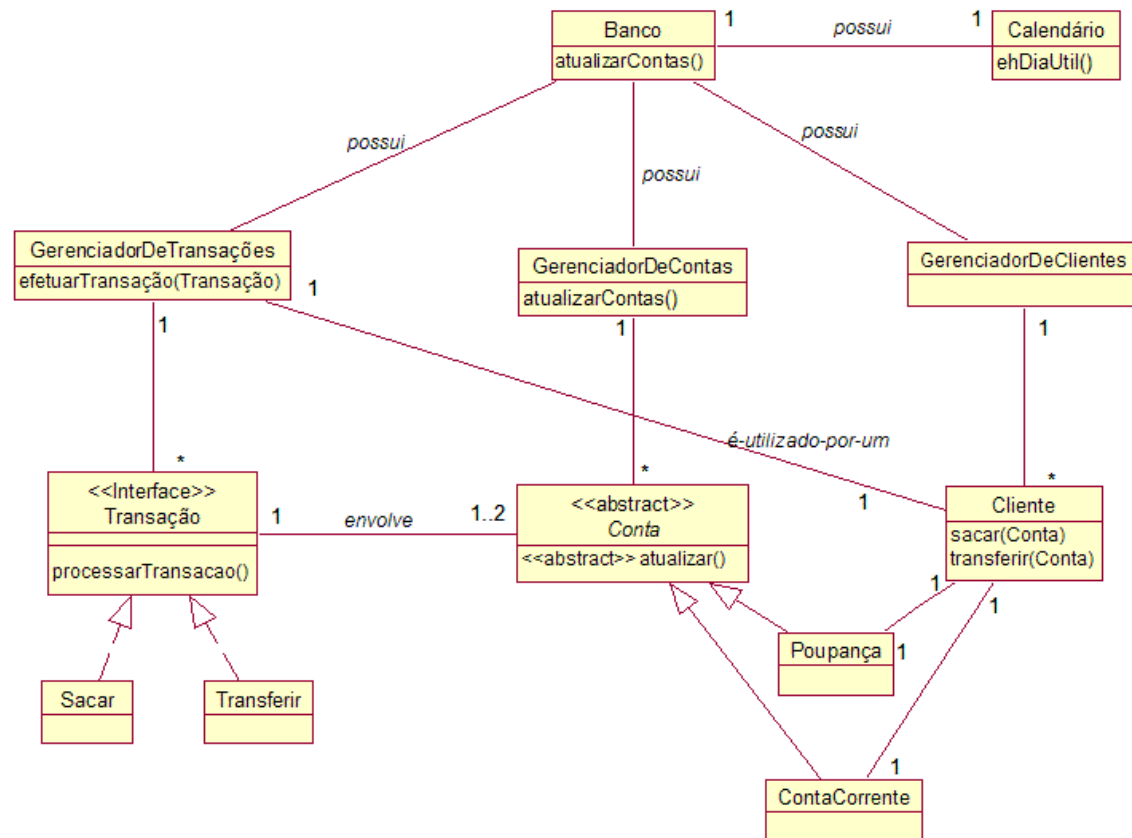


Classes e Objetos

- Diagrama de Classes
 - Especificação
 - Tem foco nas principais interfaces da arquitetura, nos principais métodos, e não como eles irão ser implementados.
 - Perspectiva destinada as pessoas que não precisam saber detalhes de desenvolvimento, tais como gerentes de projeto.

Classes e Objetos

- Diagrama de Classes
 - Especificação



Classes e Objetos

- Diagrama de Classes
 - Implementação - a mais utilizada de todas (exemplo)
 - Aborda vários detalhes de implementação, tais como navegabilidade, tipo dos atributos, etc.
 - Perspectiva destinada ao time de desenvolvimento.

Classes e Objetos

- Diagrama de Classes

