

Processo de Desenvolvimeno de Software

Profº - Dr. Thales Levi Azevedo Valente thales.l.a.valente@ufma.com.br

Sejam Bem-vindos!



Os celulares devem ficar no silencioso ou desligados

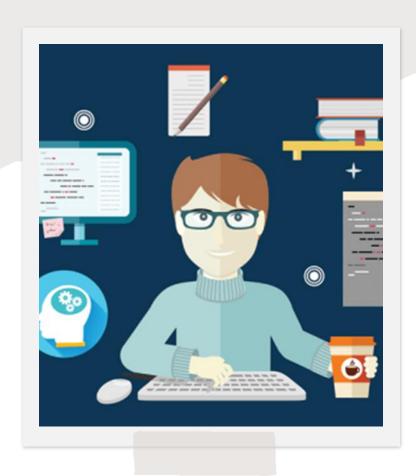
Pode ser utilizado apenas em caso de emergência



Boa tarde/noite, por favor e com licença DEVEM ser usados

Educação é essencial

Roteiro: Tópicos em modelagem e uso da IA Generativa



Introdução — Usando a IA para ajudar a usar a IA —

aula passada...

- Você não precisa ser especialista em prompts
- Use a IA para ensinar você ou dar sugestões sobre prompts

"Você é um excelente programador de IA (LLM), especialista mundial e professor renomado em prompt engineering.. Você quer criar os melhores prompts do mundo para o tópico Tópicos em Engenharia de Requisitos usando IA Generativa - Dinâmica de Entrevista e Elicitação em grupo. Quais prompts voce faria?"

(Thales, 2024)

Introdução — Usando a IA para ajudar a usar a IA —

aula passada...

"Você é um excelente programador de IA (LLM), especialista mundial e professor renomado em prompt engineering.. Você quer criar os melhores prompts do mundo para o tópico Tópicos em Engenharia de Requisitos usando IA Generativa - Dinâmica de Entrevista e Elicitação em grupo. Quais prompts voce faria?"

(Thales, 2024)

"Criar prompts eficazes para um tópico como "Tópicos em Engenharia de Requisitos usando IA Generativa - Dinâmica de Entrevista e Elicitação em grupo" requer uma abordagem estruturada e contextualizada. Aqui estão alguns prompts que poderiam ser usados para explorar esse tema de maneira produtiva"

(GPT4, 2024)

Introdução — prompt inicial

"Você é um renomado professor de engenharia de software e IA generativa e quer dar uma aula sobre um cenario de uso de IA generativa para gerar diagramas como casos de uso, atividades, etc.

Elabore os diagramas via plantUML. Vamos comecar com um caso simples de cinema."

(Thales, 2024)

"Abaixo apresentarei um exemplo de um conjunto inicial de diagramas feitos em PlantUML para ilustrar o uso de IA generativa no contexto de um sistema simples de venda de ingressos de cinema"

(GPT4, 2024)

Introdução – prompt inicial

"Cenário:

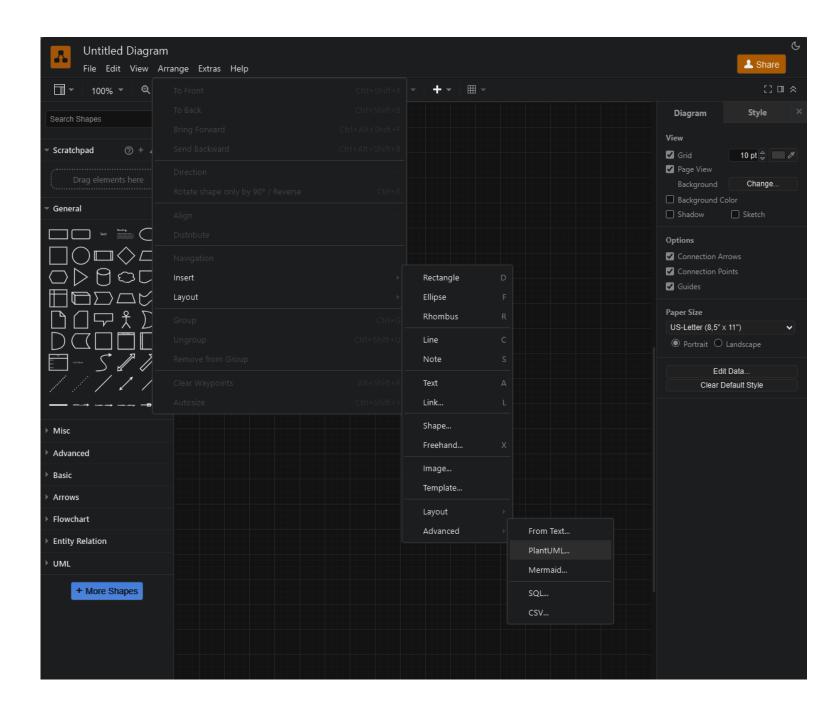
Imagine que temos um sistema de venda de ingressos online para um cinema. Um cliente acessa o sistema, escolhe um filme, uma sessão, um assento, e depois realiza o pagamento. O sistema confirma a compra e emite o ingresso."

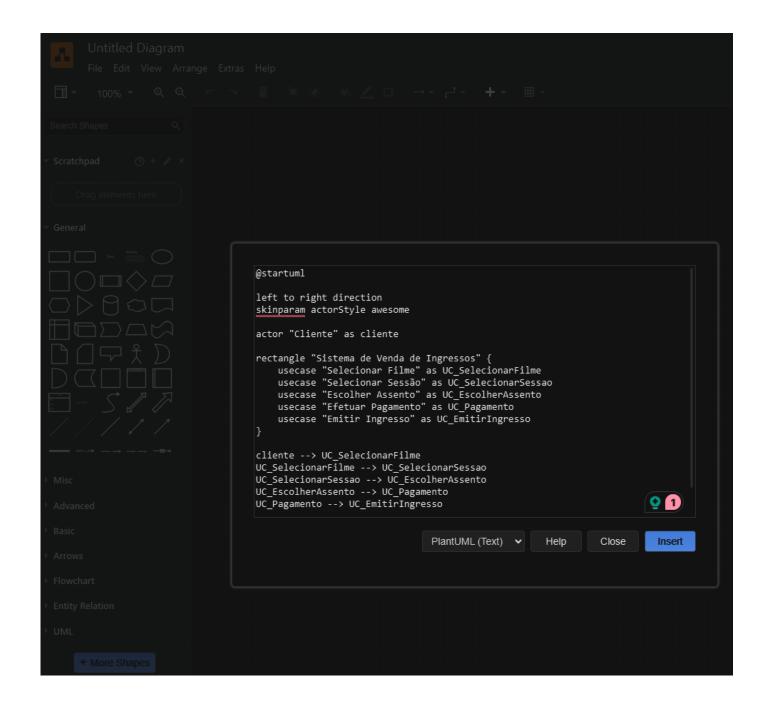
(GPT4, 2024)

"Diagrama de Caso de Uso:

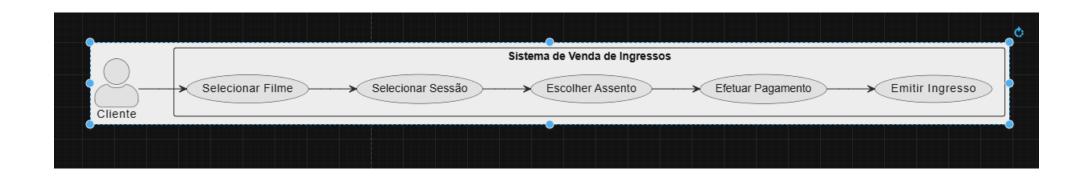
No diagrama de casos de uso, representamos o ator principal (Cliente) interagindo com o sistema (Sistema de Venda de Ingressos), bem como os casos de uso principais."

```
@startuml
left to right direction
skinparam actorStyle awesome
actor "Cliente" as cliente
rectangle "Sistema de Venda de Ingressos" {
    usecase "Selecionar Filme" as UC_SelecionarFilme
    usecase "Selecionar Sessão" as UC SelecionarSessão
    usecase "Escolher Assento" as UC_EscolherAssento
    usecase "Efetuar Pagamento" as UC Pagamento
    usecase "Emitir Ingresso" as UC_EmitirIngresso
cliente --> UC_SelecionarFilme
UC SelecionarFilme --> UC SelecionarSessao
UC_SelecionarSessao --> UC_EscolherAssento
UC EscolherAssento --> UC Pagamento
UC_Pagamento --> UC_EmitirIngresso
@enduml
```





• Podemos melhorar mais o diagrama abaixo!

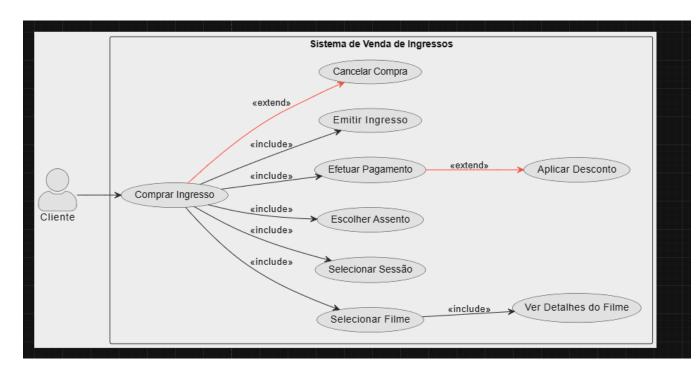


acho que o diagrama de casos de uso pode ser melhorado. Você como professor renomado sabe como fazer isso. por exemplo, casos de uso específicos deveriam estar conectados por extends, includes e etc

```
@startuml
left to right direction
skinparam actorStyle awesome
actor "Cliente" as c
rectangle "Sistema de Venda de Ingressos" {
    usecase "Comprar Ingresso" as UC Comprar
    usecase "Selecionar Filme" as UC_Filme
    usecase "Ver Detalhes do Filme" as UC_DetalhesFilme
    usecase "Selecionar Sessão" as UC Sessão
    usecase "Escolher Assento" as UC_Assento
    usecase "Efetuar Pagamento" as UC_Pagamento
    usecase "Aplicar Desconto" as UC_Desconto
    usecase "Emitir Ingresso" as UC Emitir
    usecase "Cancelar Compra" as UC_Cancelar
c --> UC_Comprar
UC Comprar --> (UC Filme) : <<include>>
UC Comprar --> (UC Sessao) : <<include>>
UC_Comprar --> (UC_Assento) : <<include>>
UC_Comprar --> (UC_Pagamento) : <<include>>
UC_Comprar --> (UC_Emitir) : <<include>>
UC_Filme --> (UC_DetalhesFilme) : <<include>>
UC_Pagamento -[#red]-> UC_Desconto : <<extend>>
UC Comprar -[#red]-> UC Cancelar : <<extend>>
@enduml
```

• Podemos melhorar mais o diagrama abaixo! Vamos gerar em um

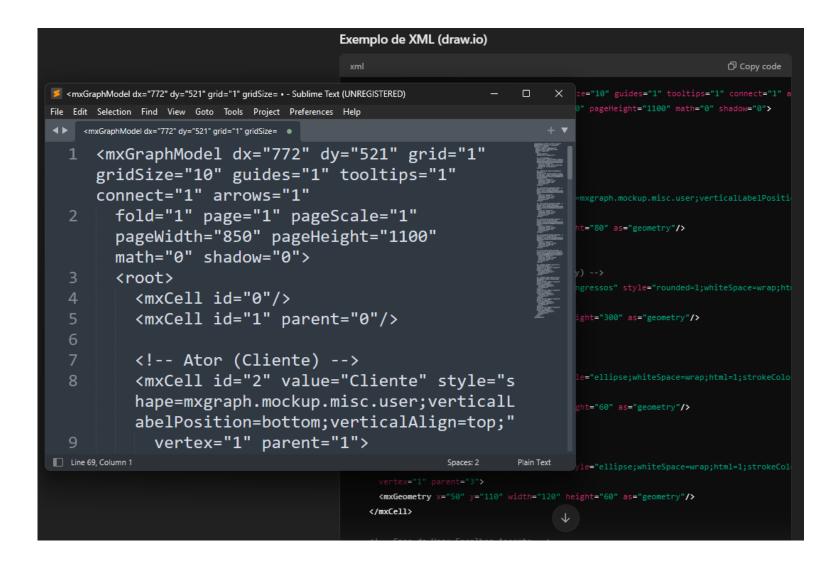
formato editável

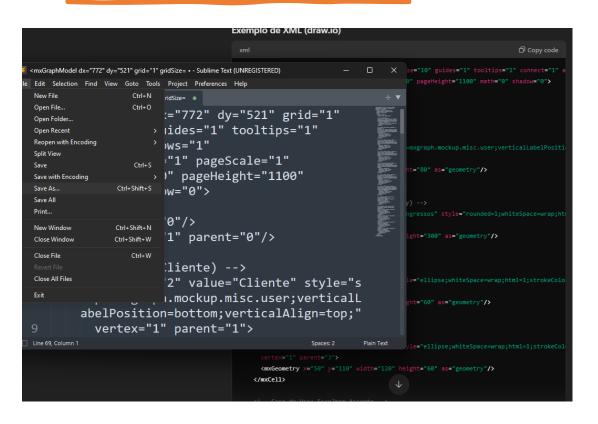


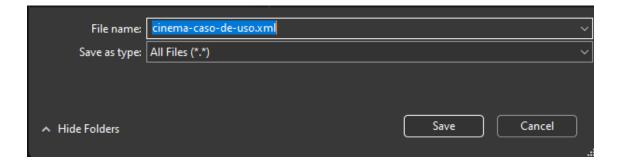
estou achando otimo, mas esse formato está gerando uma imagem do diagrama. Vamos gerar um XML o qual possa ser importado pela ferramenta para gerar o diagrama de casos de uso

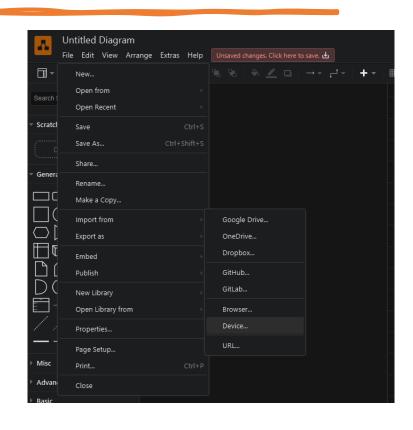
```
Exemplo de XML (draw.io)
                                                                                   (i) Copy code
  <mxGraphModel dx="772" dy="521" grid="1" gridSize="10" guides="1" tooltips="1" connect="1"</pre>
    fold="1" page="1" pageScale="1" pageWidth="850" pageHeight="1100" math="0" shadow="0">
    <root>
      <mxCell id="0"/>
      <mxCell id="1" parent="0"/>
      <!-- Ator (Cliente) -->
      <mxCell id="2" value="Cliente" style="shape=mxgraph.mockup.misc.user;verticalLabelPositi</pre>
        vertex="1" parent="1">
        <mxGeometry x="50" y="80" width="40" height="80" as="geometry"/>
      </mxCell>
      <!-- Sistema de Venda de Ingressos (Boundary) -->
      <mxCell id="3" value="Sistema de Venda de Ingressos" style="rounded=1;whiteSpace=wrap;ht</pre>
        vertex="1" parent="1">
        <mxGeometry x="200" y="50" width="300" height="300" as="geometry"/>
      </mxCell>
      <!-- Caso de Uso: Selecionar Filme -->
      <mxCell id="4" value="Selecionar Filme" style="ellipse; whiteSpace=wrap; html=1; strokeColo</pre>
        vertex="1" parent="3">
        <mxGeometry x="50" y="30" width="120" height="60" as="geometry"/>
      </mxCell>
      <!-- Caso de Uso: Selecionar Sessão -->
      <mxCell id="5" value="Selecionar Sessão" style="ellipse;whiteSpace=wrap;html=1;strokeCol</pre>
        vertex="1" parent="3">
        <mxGeometry x="50" y="110" width="120" height="60" as="geometry"/>
      </mxCell>
```

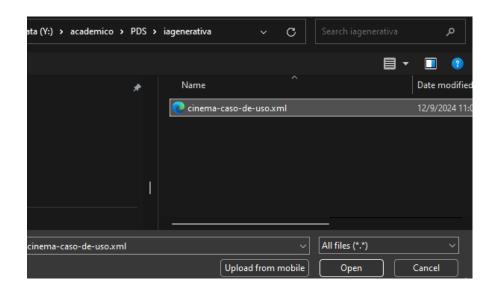
```
<!-- Caso de Uso: Escolher Assento -->
   <mxCell id="6" value="Escolher Assento" style="ellipse;whiteSpace=wrap;html=1;strokeCol</pre>
     vertex="1" parent="3">
     <mxGeometry x="50" y="190" width="120" height="60" as="geometry"/>
   </mxCell>
   <!-- Caso de Uso: Efetuar Pagamento -->
   <mxCell id="7" value="Efetuar Pagamento" style="ellipse; whiteSpace=wrap; html=1; strokeCol</pre>
     vertex="1" parent="3">
     <mxGeometry x="200" y="110" width="120" height="60" as="geometry"/>
   </mxCell>
   <!-- Caso de Uso: Emitir Ingresso -->
   <mxCell id="8" value="Emitir Ingresso" style="ellipse; whiteSpace=wrap; html=1; strokeColor</pre>
     vertex="1" parent="3">
     <mxGeometry x="200" y="190" width="120" height="60" as="geometry"/>
   </mxCell>
   <!-- Relações (Linhas entre Ator e "Selecionar Filme") -->
   <mxCell id="9" edge="1" parent="1" target="4" source="2" style="endArrow=none;">
     <mxGeometry relative="1" as="geometry"/>
   </mxCell>
   <!-- Relações entre casos de uso (simplesmente linhas representando sequência) -->
   <mxCell id="10" edge="1" parent="3" source="4" target="5" style="endArrow=block;">
     <mxGeometry relative="1" as="geometry"/>
   </mxCell>
   <mxCell id="11" edge="1" parent="3" source="5" target="6" style="endArrow=block;">
     <mxGeometry relative="1" as="geometry"/>
   </mxCell>
   <mxCell id="12" edge="1" parent="3" source="6" target="7" style="endArrow=block;">
     <mxGeometry relative="1" as="geometry"/>
   </mxCell>
   <mxCell id="13" edge="1" parent="3" source="7" target="8" style="endArrow=block;">
     <mxGeometry relative="1" as="geometry"/>
   </mxCell>
 </root>
</mxGraphModel>
```

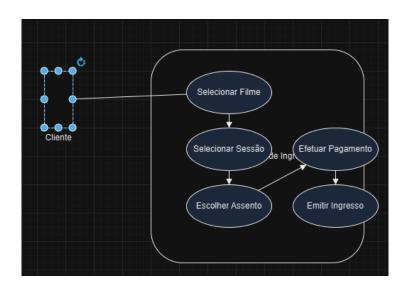


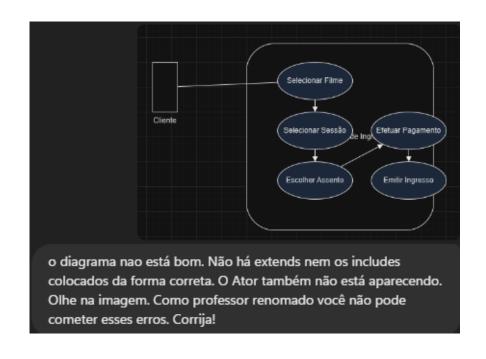


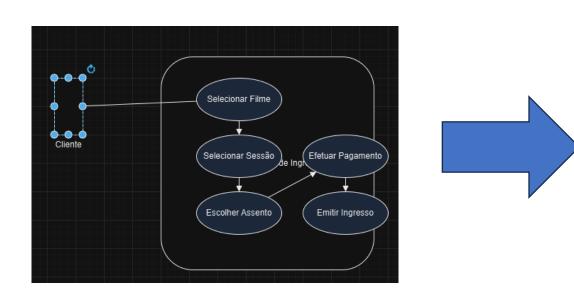


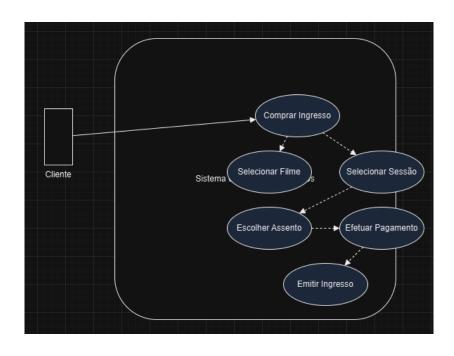


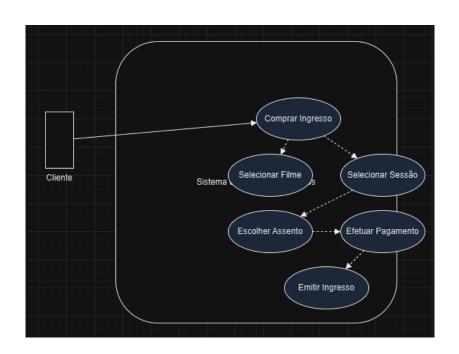












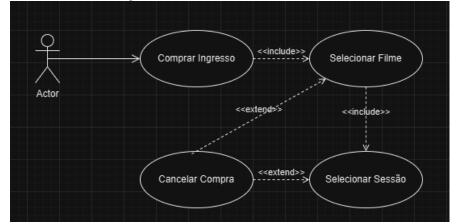
ok. Vamos melhorar o diagrama. Vamos focar nos relacionamentos entre os casos de uso.

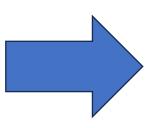
Selecionar filme include Selecionar Sessao Selecionar sessao includie Escolher assento Escolher assento include Efetuar pagamento Efetuar pagamento include Emitir ingresso TOdos devm ter um extend para cancelar compra, menos o caso de uso Emitir Ingresso

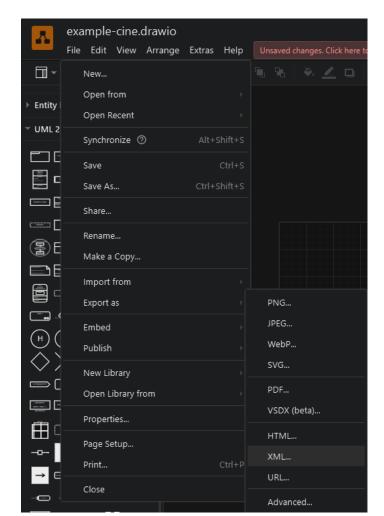
Não esqueca dos textos "include" e extend" nas setas tracejadas



 Erros foram gerados. Nós podemos ajudar a IA gerando um exemplo para direcionar melhor. Por exemplo, vamos criar um pequeno pedaço do diagrama, exportar e dar como modelo para a IA..



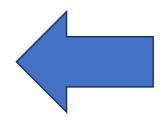




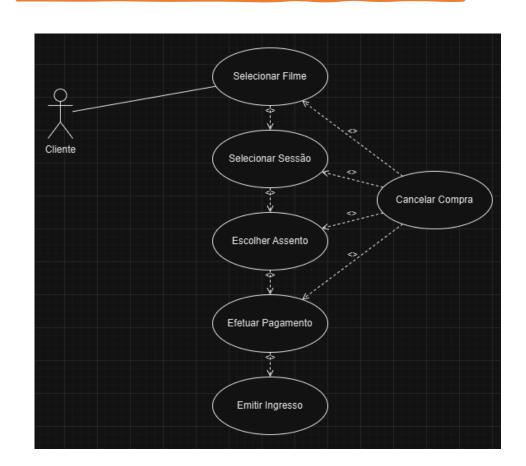
 Erros foram gerados. Nós podemos ajudar a IA gerando um exemplo para direcionar melhor. Por exemplo, vamos criar um pequeno pedaço do diagrama, exportar e dar como modelo para a IA..

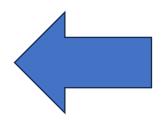
Com base no exemplo fornecido, eu ajustei o XML para seguir corretamente o formato exibido no seu exemplo e incorporei as relações *include* e *extend* entre os casos de uso. Abaixo está o XML corrigido para que seja totalmente funcional no **draw.io**.

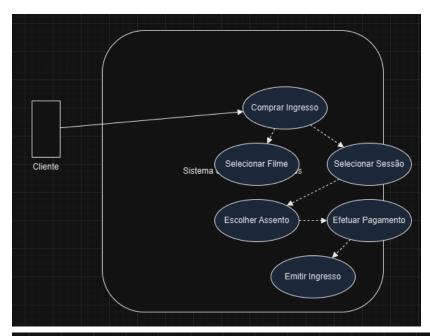
XML Corrigido e Aprimorado

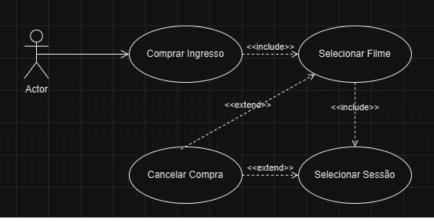


```
Olha, eu gerei um exemplo pequeno com includes e extends. Use-
o para corrigir o seu
<mxfile host="app.diagrams.net" agent="Mozilla/5.0 (Windows
NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/131.0.0.0 Safari/537.36 Edg/131.0.0.0" version="25.0.3">
 <diagram name="Page-1" id="IQh7eSPuvrQETuiB3fF5">
  <mxGraphModel dx="1364" dy="1132" grid="1" gridSize="10"
guides="1" tooltips="1" connect="1" arrows="1" fold="1" page="1"
pageScale="1" pageWidth="850" pageHeight="1100" math="0"
shadow="0">
   <root>
    <mxCell id="0"/>
    <mxCell id="1" parent="0" />
    <mxCell id="9cRPYVZAUiLj59y8L_VH-35" value="Actor"
style="shape=umlActor;verticalLabelPosition=bottom;verticalAlign
=top;html=1;" vertex="1" parent="1">
     <mxGeometry x="170" y="245" width="30" height="60"
as="geometry" />
    </mxCell>
    <mxCell id="9cRPYVZAUiLj59y8L_VH-36" value="Comprar
Ingresso" style="ellipse;whiteSpace=wrap;html=1;" vertex="1"
     <mxGeometry x="300" y="240" width="140" height="70"
as="geometry" />
    </mxCell>
    <mxCell id="9cRPYVZAUiLj59y8L_VH-37"
value="<&amp;lt;include&amp;gt;&amp;gt;"
style="html=1;verticalAlign=bottom;labelBackgroundColor=none;
endArrow=open;endFill=0;dashed=1;rounded=0;exitX=1;exitY=0.5;
exitDx=0;exitDy=0;" edge="1" parent="1"
source="9cRPYVZAUiLj59y8L_VH-36"
target="9cRPYV" \iLi59y8L_VH-38">
```

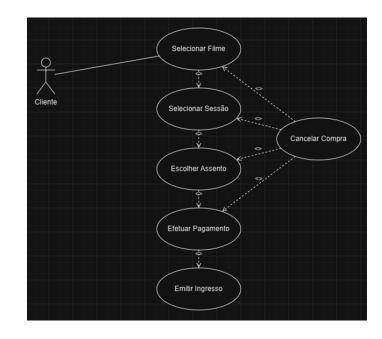


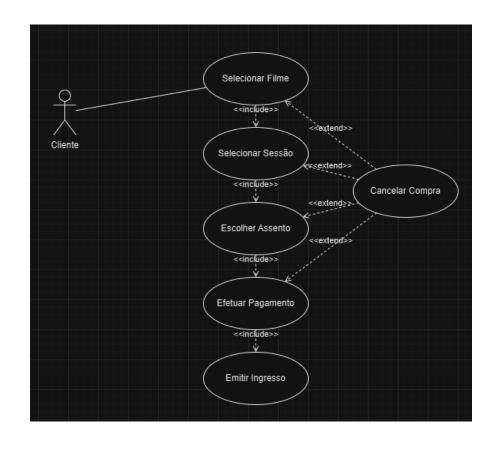


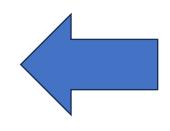




 Ainda há um erro. Vamos ajudá-la novamente. Dessa vez precisei especificar 2 prompts, sendo um somente apontando o padrão de um campo corretamente.









perceba que o padrao em value segue value=<&lt;extend&gt;&gt;, o qual voce nao esta seguindo nem para o include e nem para o extend

Atividades de Fixação

Opa! Supresa!!!

Dúvidas?





Apresentador

Thales Levi Azevedo Valente

E-mail:

thales.l.a.valente@gmail.com