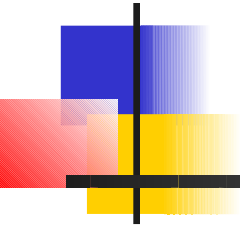


# Diagramas de Estado



Profa Dra Rosana T. V. Braga



# Eventos, estados e transições

---

- Um **evento** é uma ocorrência significativa ou digna de nota. Ex: um telefone é tirado do gancho.
- O evento normalmente é atômico, isto é, não consome tempo.
- Um **estado** é a condição de um objeto em um certo momento no tempo, o tempo entre dois eventos. Exemplo: o telefone fica no estado “ocioso” depois que foi colocado no gancho e até ser retirado do gancho novamente.



# Eventos, estados e transições

---

- Uma **transição** é um relacionamento entre dois estados, indicando que quando um evento ocorre um objeto passa do estado anterior para o subsequente.
- Ex. quando ocorre o evento “fora do gancho” o telefone passa do estado “ocioso” para o estado “ativo”

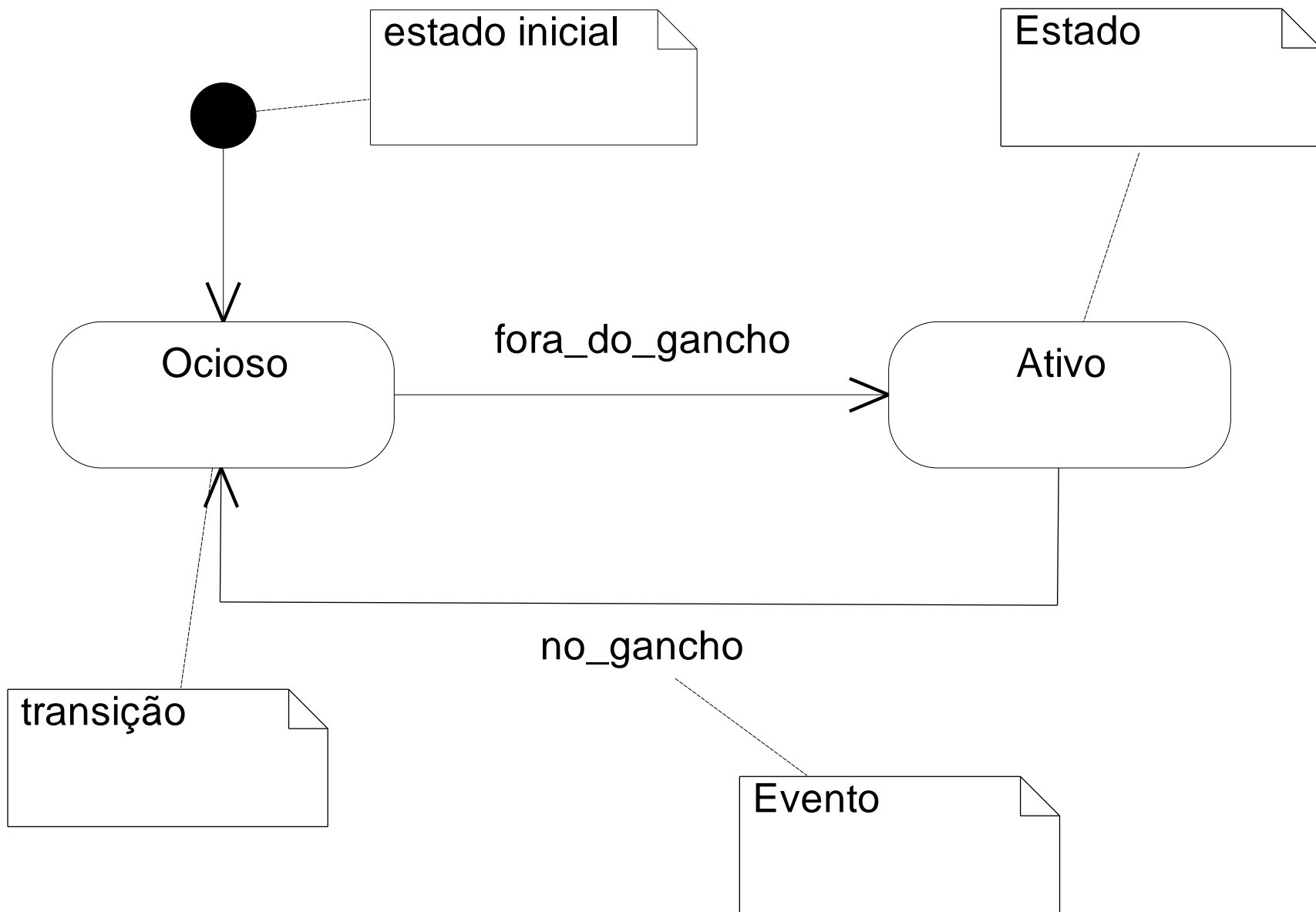


# Diagramas de Estado

---

- É um dos diagramas da UML: transições são mostradas como flechas e estados como retângulos com canto arredondado
- Mostram o ciclo de vida de um objeto
- Não é necessário mostrar todos os eventos possíveis. Se ocorrer um evento não mostrado, ele é ignorado. Isso permite criar diagramas com diferentes níveis de abstração.

# Telefone





# Assunto de um diagrama de Estado

---

- Um diagrama de estado pode ser aplicado a diversos elementos do modelo, entre os quais:
  - Classes de software
  - Tipos (conceitos)
  - Casos de uso
- O próprio sistema pode ter um diagrama de estado



# Utilidade dos diagramas de estado de Casos de Uso

---

- Garantir que não ocorram eventos fora de seqüência
- Assegurar que nada foi esquecido
- Soluções possíveis
  - Testes condicionais rigidamente codificados
  - Uso do padrão State (Estado- a ser visto adiante)
  - Desabilitar elementos da interface
  - Um interpretador de máquinas de estado que executa uma tabela de estados



# Diagrama de Estados do Sistema

---

- Ilustra todas as transições para os eventos do sistema para todos os casos de uso.
- É a união de todos os diagramas de estado de casos de uso
- É útil até o limite que o número total de eventos do sistema é pequeno o suficiente para que o diagrama seja compreensível.



# Tipos que necessitam de Diagramas de Estado

- Se um objeto responde a um evento sempre da mesma forma, então ele é considerado **independente de estado** (ou não modal) com relação àquele evento.
- Se para todos os eventos de interesse um tipo sempre reage da mesma maneira, então ele é um tipo independente de estado.
- Ao contrário, tipos **dependentes de estado** reagem de maneira diferente a eventos, dependendo de seu estado.

# Tipos e classes comumente dependentes de estado



---

- Casos de Uso
- Sistemas
- Janelas (Ex. copiar-colar só é válida se existir algo na área de transferência).
- Coordenadores de Aplicação.
- Controladores
- Transações
- Dispositivos (TPV, modem etc)
- Mutantes (tipos que mudam de papel)

# Notação UML para diagramas de estado

- Estado

- Inicial



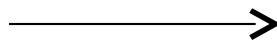
- Final



- Comum



- Transição

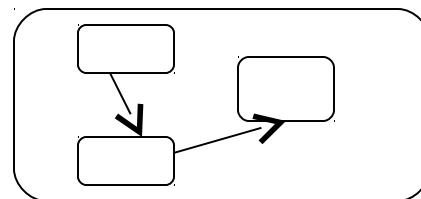


- Evento (evento)

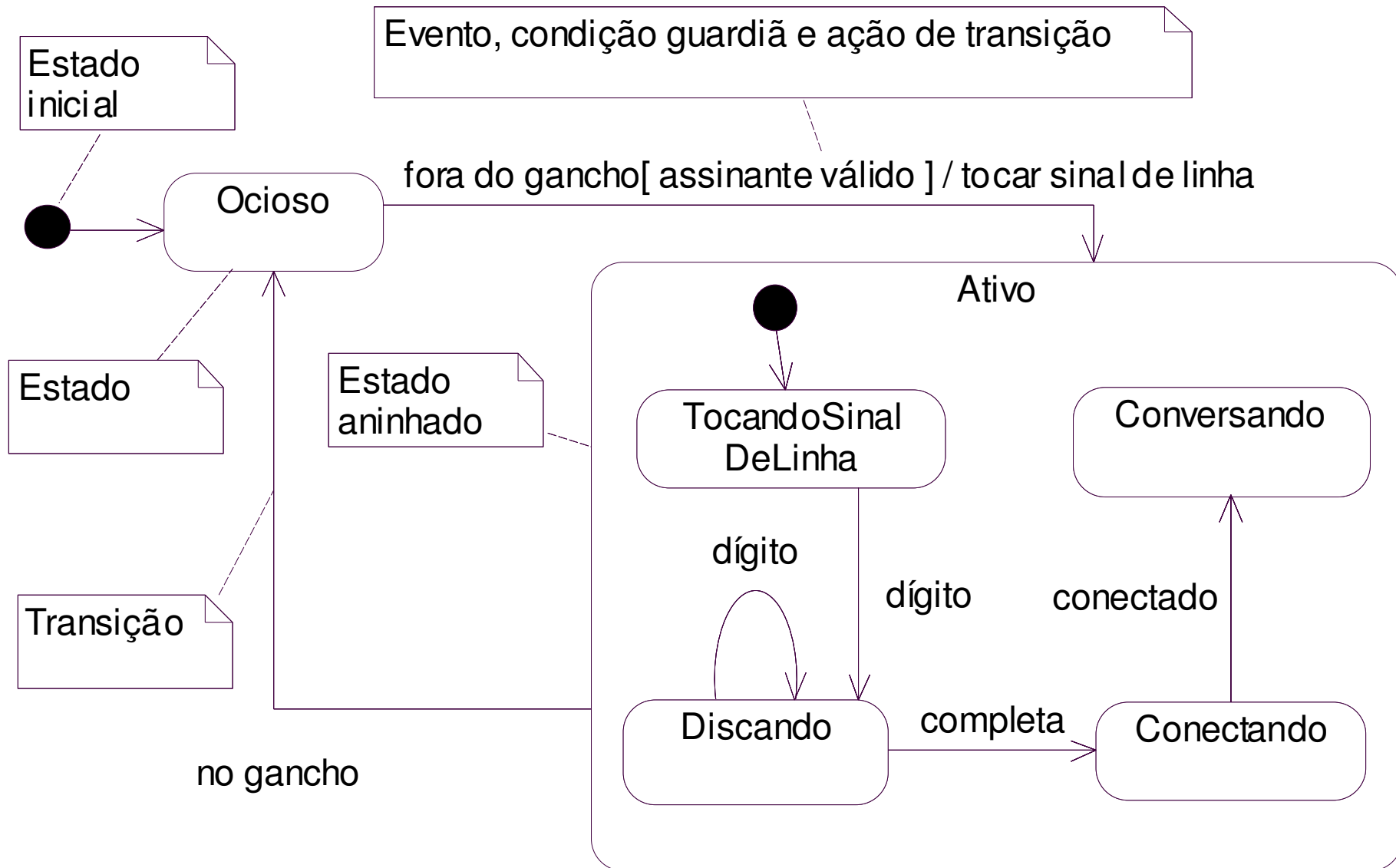
- Ação de transição (evento / ação)

- Condições guardiãs de transição [ cond ]

- Estados aninhados



# Exemplo: Telefone



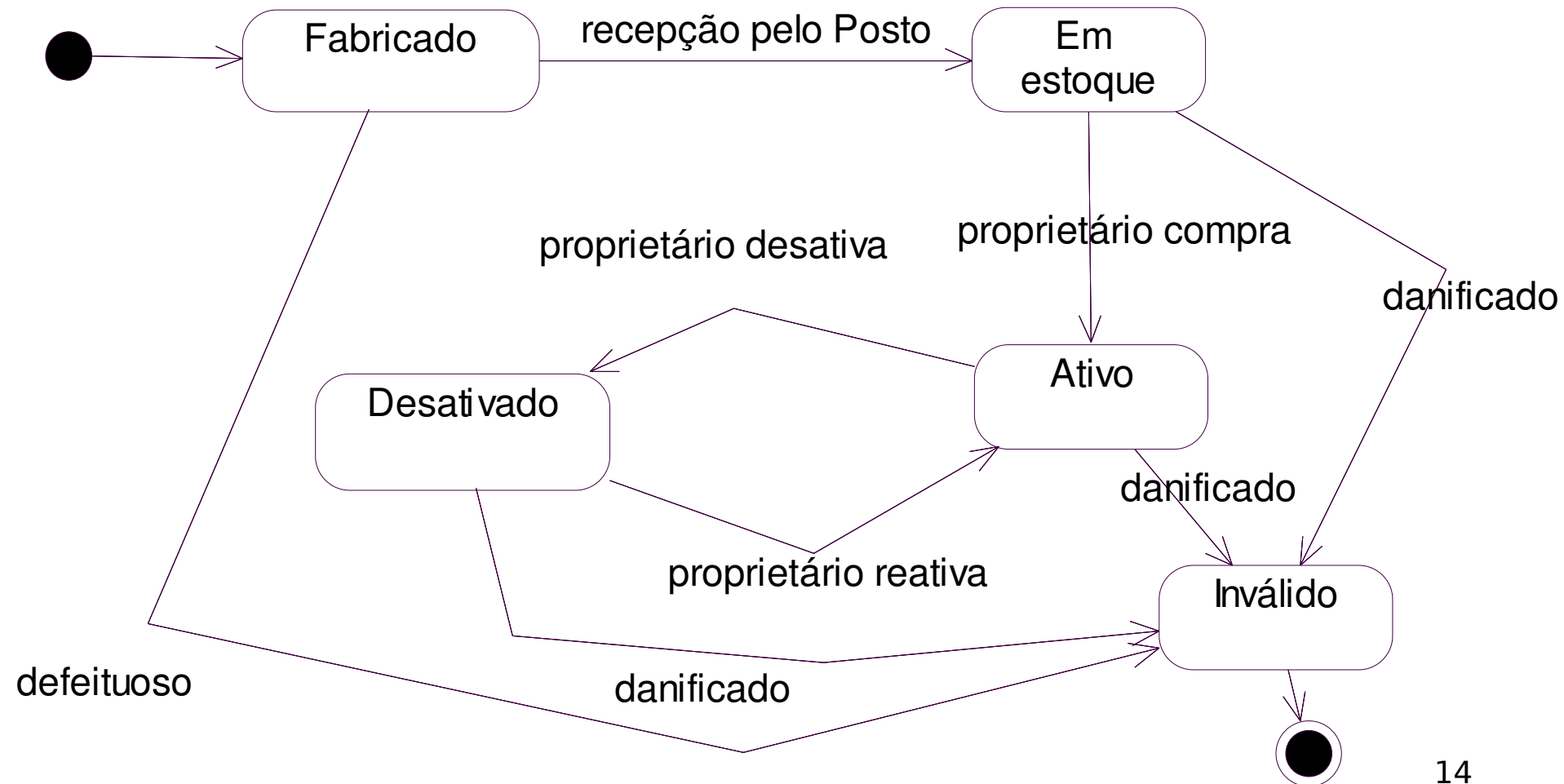


# Exercício em sala

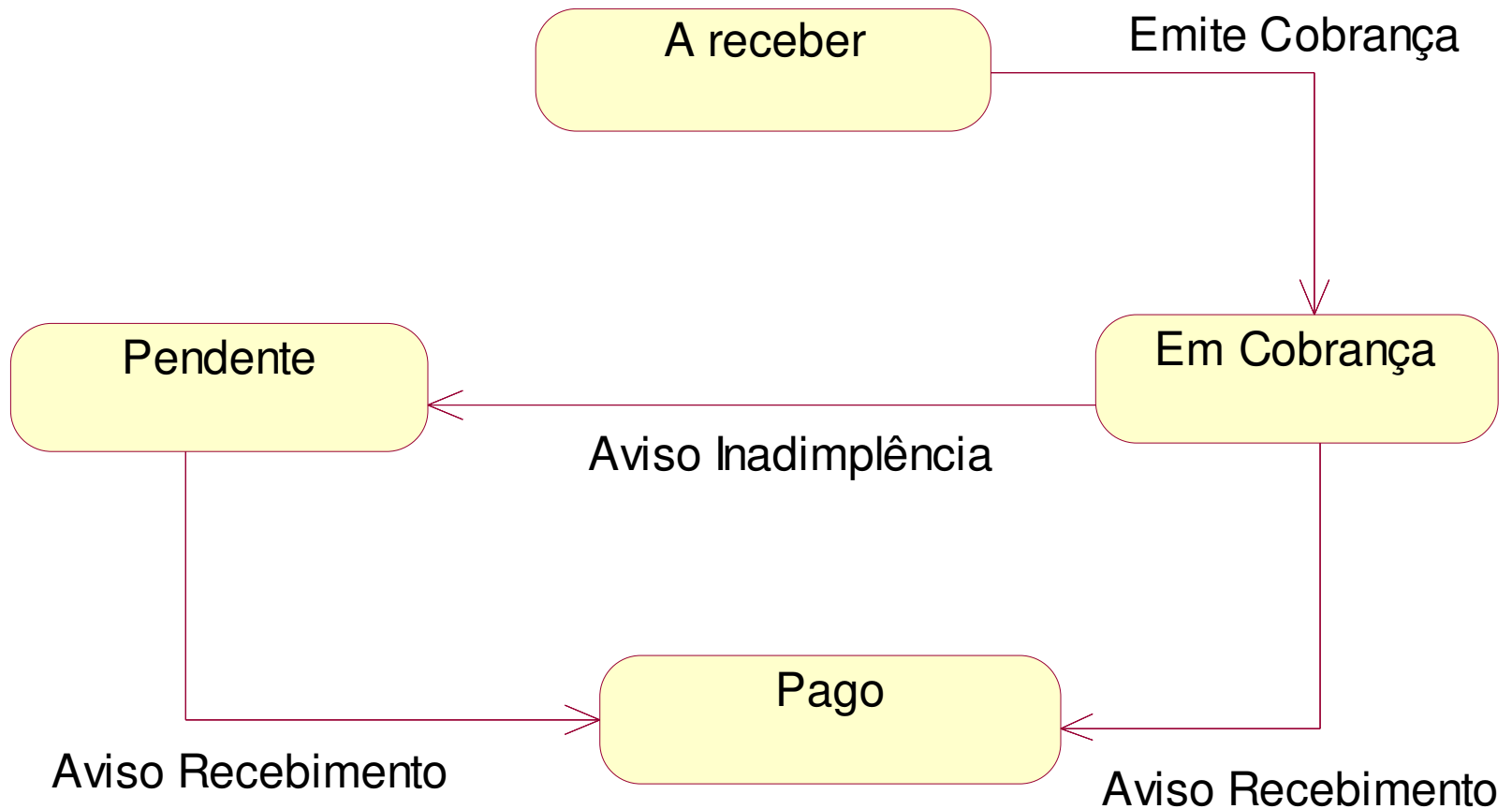
---

- Faça um diagrama de estados para os seguintes exemplos:
  - Um gizmo do sistema Passe Livre
  - Caso de uso comprar gizmo
  - Um registro de uso do sistema Passe livre
  - Sub-sistema de pedágio (um veículo transitando pela autopista)

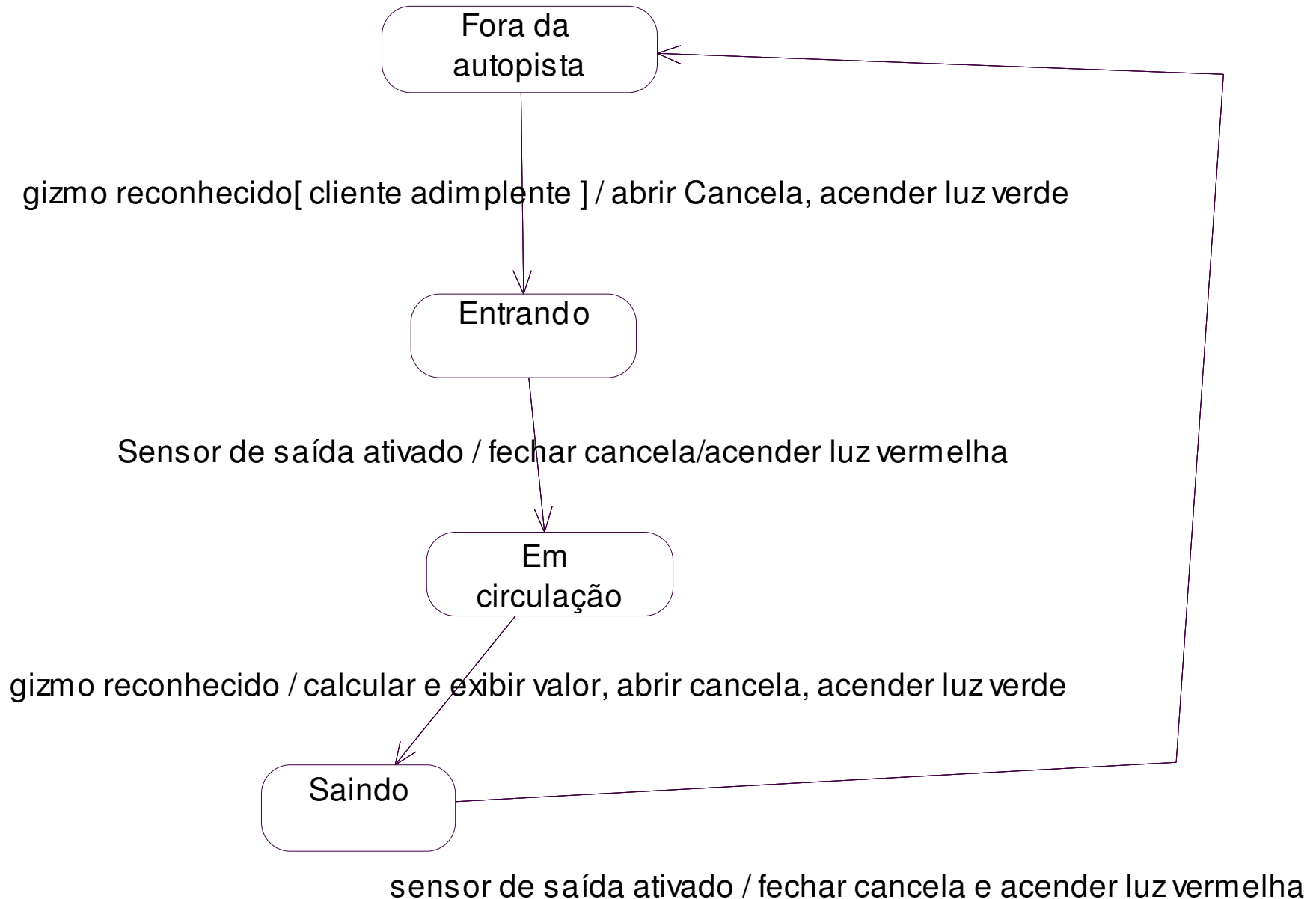
# Gizmo



# Registro de Uso



# Sub-sistema Pedágio





# Gizmo

