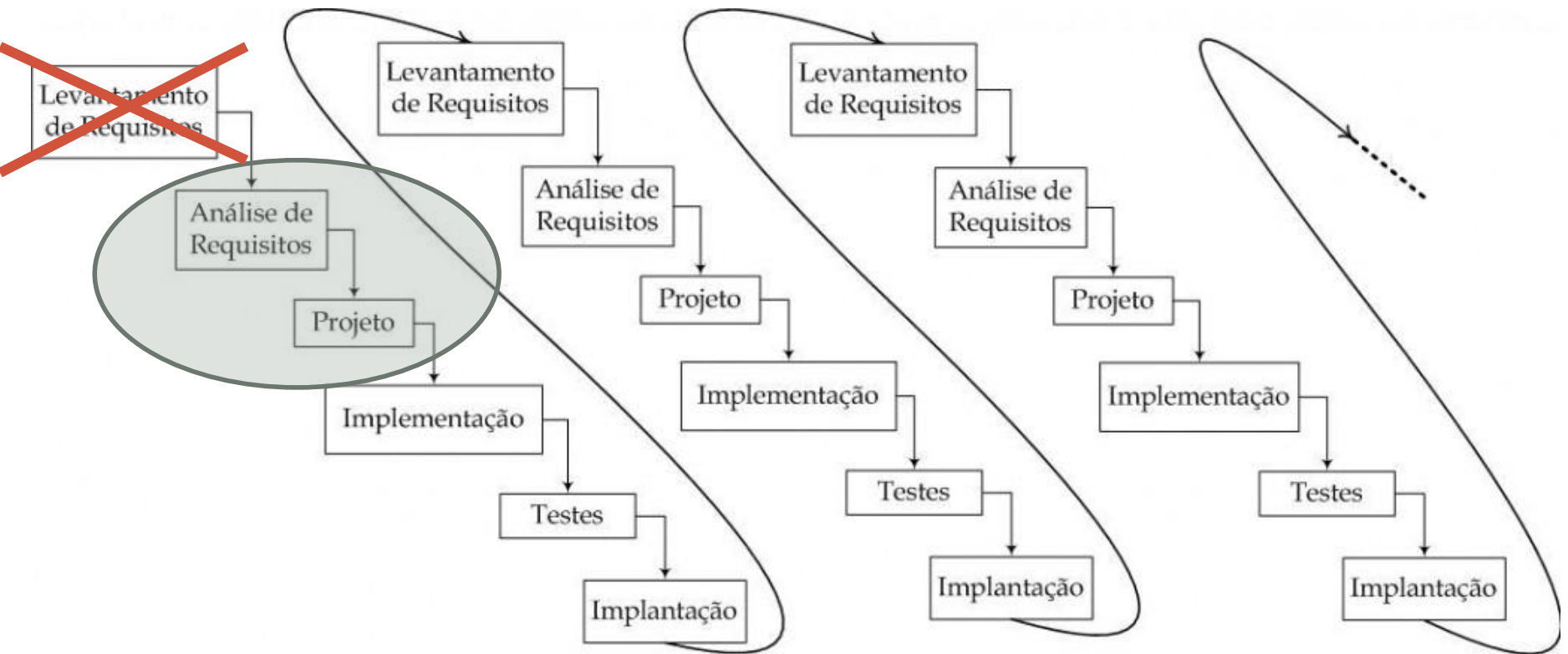


# PROJETO DE PROGRAMAS

---

Projeto de Programas – PPR0001

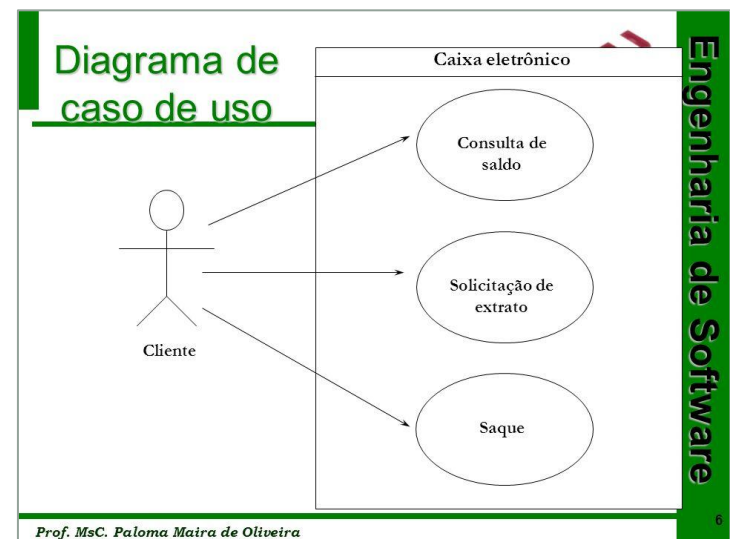
# Desenvolvimento de Software



# Desenvolvimento de Software

## Análise de Requisitos

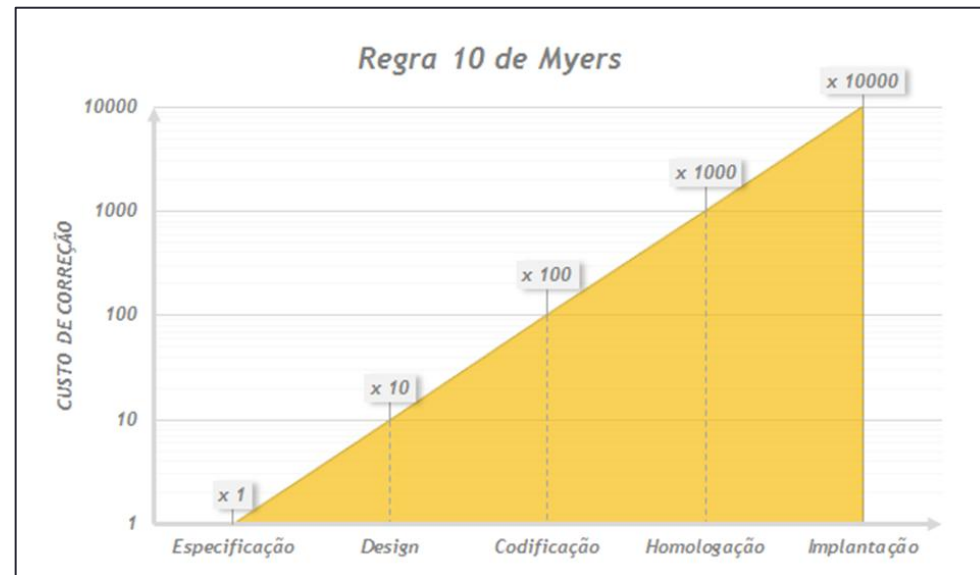
- Distinguir e dividir o sistema em componentes:
  - Analisar os componentes individualmente
  - Analisar como os componentes devem interagir uns com os outros
- São construídos modelos e/ou diagramas para representar o sistema a ser construído
- Não leva em conta o ambiente tecnológico a ser utilizado



# Desenvolvimento de Software

## Análise de Requisitos

- Fase de validação / verificação:
  - Analistas levam os modelos elaborados aos clientes e verificam se as necessidades dos clientes estão sendo atendidas pelo sistema
  - Se um modelo não é bem definido, usuários e programadores poderão ter interpretações diferentes
  - Se um erro nesta etapa for descoberto posteriormente haverá um impacto muito grande no sistema

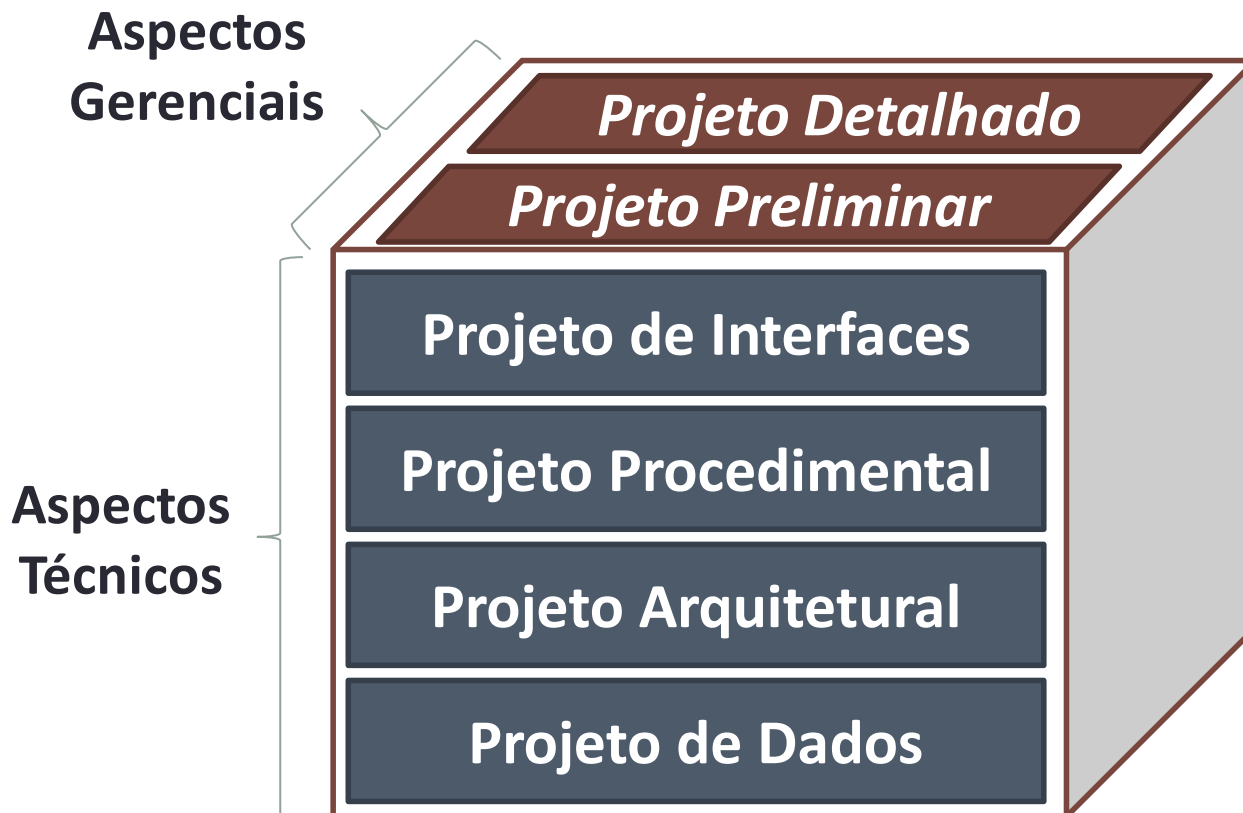


# Desenvolvimento de Software

## Projeto

- Determina-se “como” o sistema atenderá aos requisitos
  - Descrição computacional do que o software deve fazer
  - Considera aspectos físicos e tecnológicos
  - Arquitetura do sistema, padrão de interface gráfica, linguagem de programação, gerenciador de banco de dados, etc.

# Aspectos Técnicos de Projeto



# Aspectos Técnicos de Projeto

## Projeto

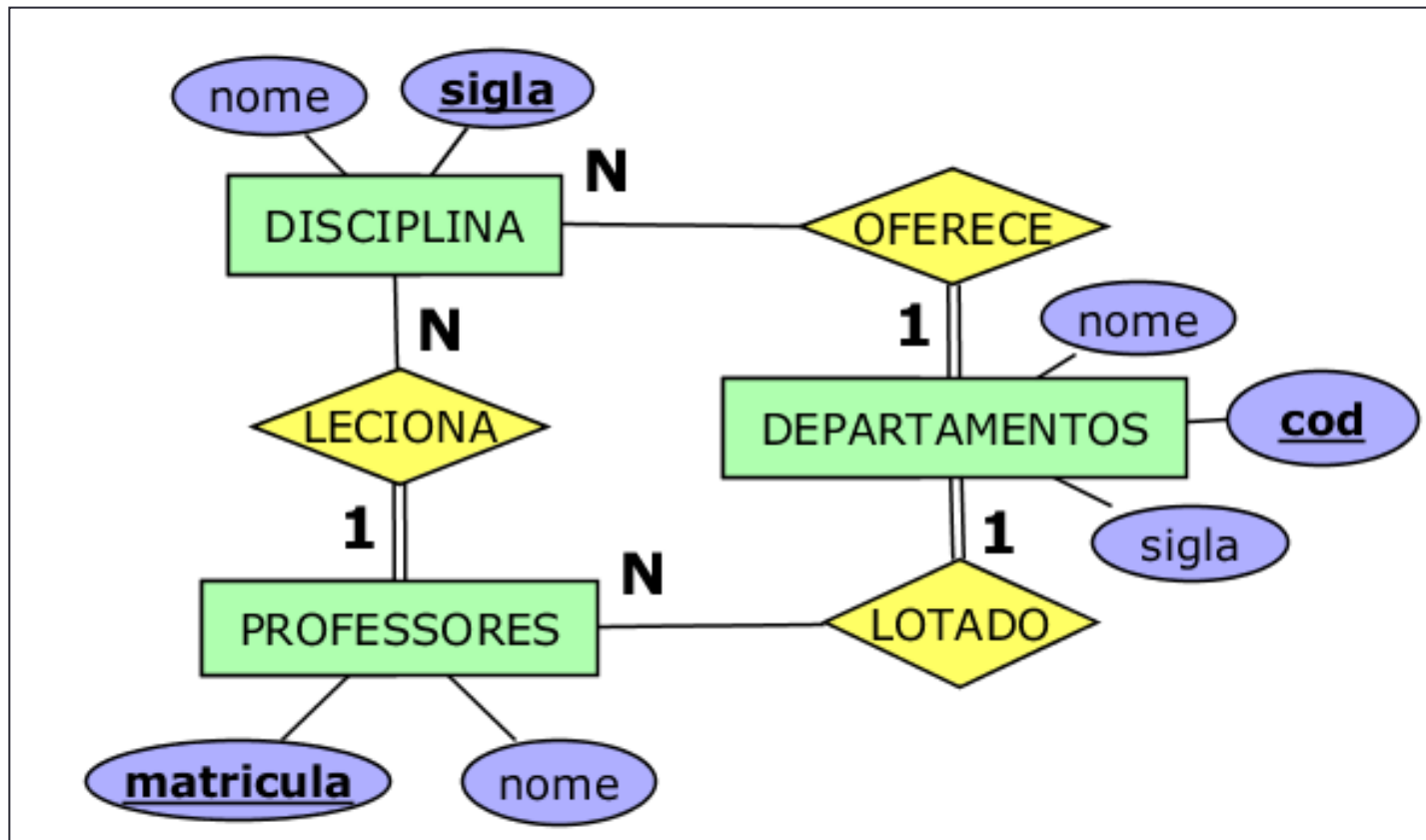
- Possui duas etapas principais:
  - ❖ Projeto preliminar (projeto de alto nível)
    - Distribuir classes de objetos e seus subcomponentes
  - ❖ Projeto detalhado (projeto de baixo nível)
    - Descrição da colaboração entre os objetos de cada módulo
    - Projeto de interface com o usuário
    - Projeto do banco de dados
    - Diagramas: de classe, interação, estados e atividades

# Projeto de Dados

- Durante a análise se gera o modelo de informação, e com este modelo serão definidas as estruturas de dados necessárias para a implementação do software
- A etapa final é influenciada pela tecnologia de persistência escolhida para a implementação
  - ❖ Banco de dados, arquivos (xml, csv), etc...
- A base do projeto de dados é um **modelo entidade-relacionamento** (MER), que armazena os objetos de dados e as relações entre eles.



# Projeto de Dados



Exemplo de um Diagrama Entidade-Relacionamento (DER)

# Projeto Arquitetural

- Define quais tecnologias serão utilizadas:
  - Arquitetura de computador
  - Sistema Operacional
  - SGBD (se for o caso)
  - Linguagem de programação
  - Considerações dos requisitos não-funcionais
- Define o **estilo arquitetural** adotados:
  - Hierarquia / relação dos componentes e das estrutura de dados
  - Padrões arquiteturais: centrado nos dados, orientado a objetos, em camadas, etc...

# Projeto Arquitetural

- Define as interações internas (consigo mesmo):

- Assinaturas de métodos

- Define como interage com outros sistemas:

- E.g.: Troca de Informações entre sistemas bancários

Como outro sistema bancário deve acessar meus dados:

```
string clientData (int conta, int agencia, string codSeg);
```

Como meu sistema deve acessar o sistema bancário X:

```
string getClient (int conta, string codSeg);
```

- Diagrama utilizado:

- Diagrama de classes

# Projeto Procedimental

- Especifica detalhes de processamento de cada módulo
  - Transforma os elementos estruturais da arquitetura em uma descrição procedimental dos componentes
  - Sequência de eventos realizados: Funcionalidades, Algoritmos
- Diagramas utilizados:
  - Diagrama de Fluxo de Dados (DFD)
  - Diagrama de Estados (UML)
  - Diagrama de Sequência (UML)

# Projeto de Interfaces

- Define aspectos da interface do sistema:
  - Qual tecnologia será utilizada?
  - Como será a interface gráfica com o usuário?
  - O que deve ser feito para usar a funcionalidade X?
  - Questões envolvendo a Psicologia Cognitiva:
    - Percepção visual
    - Psicologia Cognitiva de leitura;
    - Memória Humana;
    - Raciocínio Indutivo e Dedutivo
    - Comunicação Textual ou Pictórica (ícones)
    - Nível Intelectual e/ou Habilidades do Usuário

# PROJETO DE DADOS

---

# Atividades Envolvidas

Realizar a abstração de dados (visando o escopo do problema)

- Uma abstração é uma representação das características e do comportamento relevantes de um conceito do mundo real para um determinado problema.
  - O que faz parte do escopo do projeto?
  - Quais os dados necessários para representar as entidades do sistema

# Atividades Envolvidas

A abstração dos dados pode ser realizada em duas etapas:

➤ **Etapla conceitual:**

- Não se preocupa com tecnologia
- Identificar os objetos com seus dados e ações significativos

➤ **Etapla lógica e física:**

- Estudar e escolher as estruturas de dados que permitam a implementação mais adequada;
- Caracterizar a abrangência dos dados:
  - ☐ componente (local) ou parte do software (global);
  - ☐ persistência de dados: uso de Banco de Dados , arquivos ou memória;



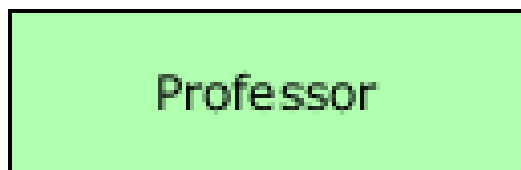
# Atividades Envolvidas

## **Etapas conceituais:**

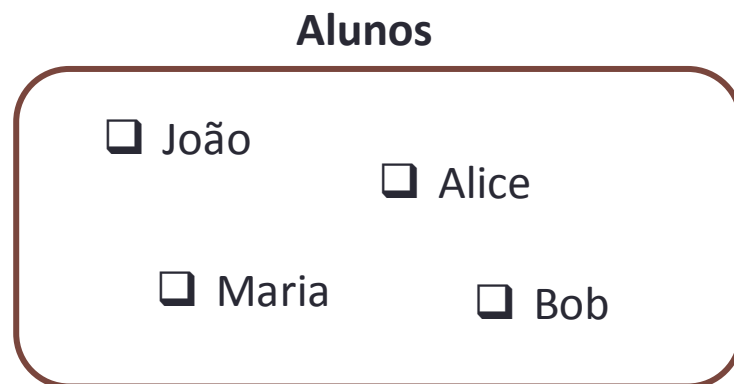
- Utilização do Modelo Entidade-Relacionamento (MER)
  - Representação gráfica das entidades (objetos), seus atributos e relações
  - Criada em 1976 por Peter Chen
  - Muito utilizada em Banco de Dados
  - Ferramentas: BR-Modelo (binário) e Terra-ER (java)

# Modelo Entidade-Relacionamento

**Entidade:** representa um conjunto de objetos do mundo real  
significante ao problema



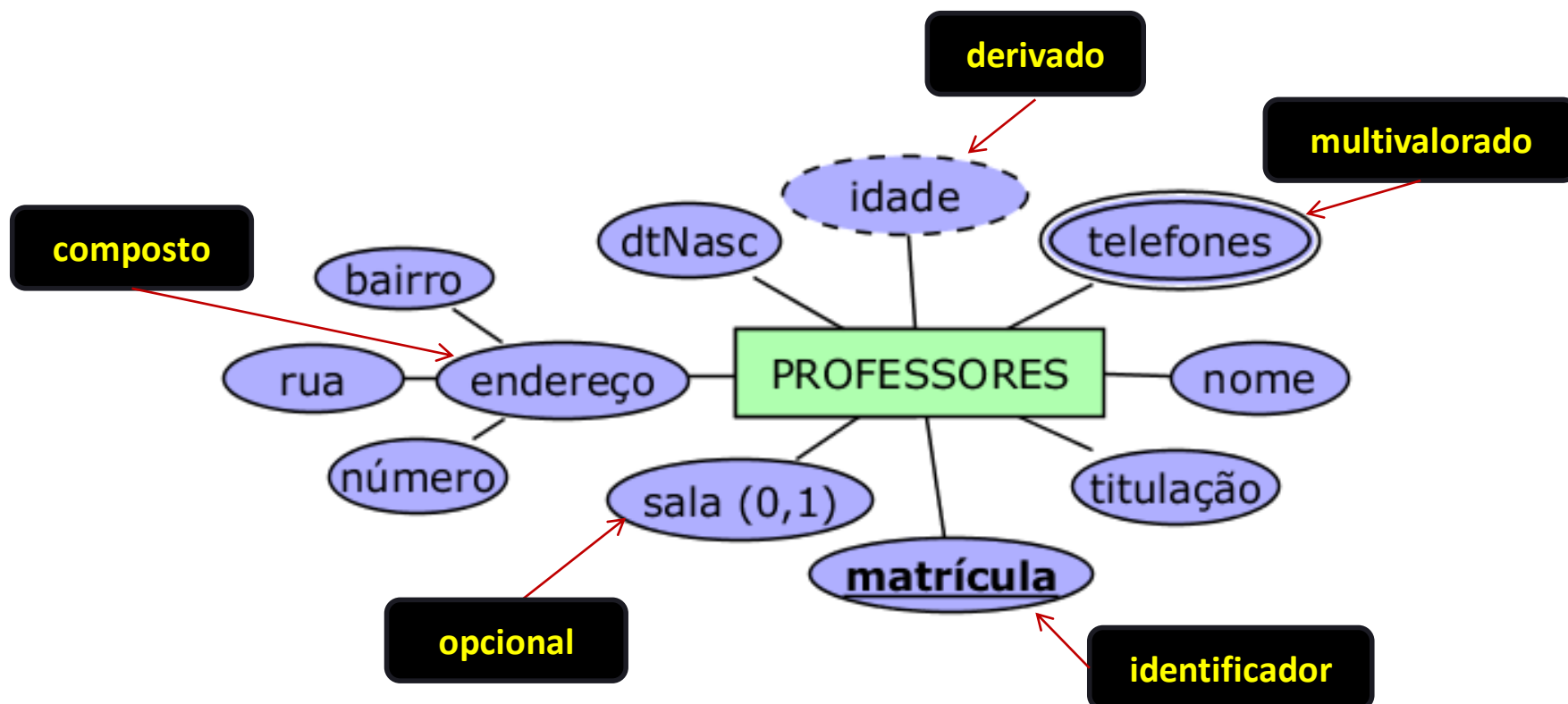
(representação gráfica)



(interpretação)

# Modelo Entidade-Relacionamento

**Atributos:** representam dados (características) significativos que estão associados a uma entidade ou relacionamento

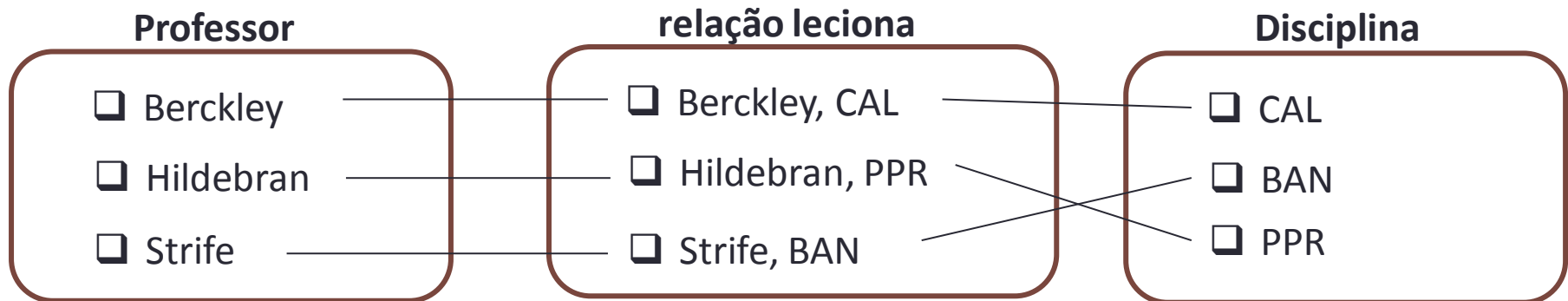


# Modelo Entidade-Relacionamento

- **Relacionamento:** representa associação entre os itens de duas entidades
  - Utilizar nomes representativos ou verbos
  - Também podem ter atributos



(interpretação)



## Exemplo

Realizar o projeto de dados conceitual para um sistema acadêmico que permite o controle e gerenciamento de matrícula, frequência e desempenho dos discentes e a organização das disciplinas ofertadas. O sistema acadêmico deverá permitir que os acadêmicos realizem suas matrículas nas turmas de disciplinas disponíveis, considerando restrições de pré-requisitos, número máximo de créditos (9) e limite de alunos por turma. Deverá permitir que chefes de departamento incluam novas disciplinas e novos professores, abram novas turmas para as disciplinas existentes com sala, horário, lotação máxima e professor definidos. As disciplinas só poderão ser ofertadas entre 7:30 e 12:00, e, 13:30 e 21:40, em blocos de 50 minutos por aula (hora-aula). Também deverá ser possível que professores acessem suas turmas e registrem frequência e notas para seus alunos.

## Exemplo

O sistema deverá ter uma opção para finalizar o semestre, possibilitando a inclusão das notas de exame. Um aluno deverá ter frequência superior a 75% e deverá ter uma média superior a 3 para realizar exame. Caso sua nota seja maior ou igual a 7 está aprovado (desde que tenha a frequência necessária). Após a digitação das notas de exame o professor deverá finalizar a turma e o sistema mostrará o resultado final. O sistema deverá funcionar nos sistemas operacionais Windows e Linux e deverá ter seu acesso controlado por login e senha.

(adicionar atributos que considerar relevantes ao problema)

# Atividade

Agora é a sua vez!

Construa o projeto de dados conceitual para o sistema descrito no documento que está disponível na página da disciplina.