

# Projeto Final – Programação Orientada a Objetos

---

**Disciplina:** Programação Orientada a Objetos

**Professora:** Talita Vieira Ribeiro

**Data de Entrega no SAVA por Todos os Participantes:** 11/06/2025

**Data de Apresentação em Sala de Aula:** 12/06/2025

Atenção 1: A não entrega do trabalho no SAVA pelo aluno no dia determinado implicará numa penalização de 2 pontos na nota final do aluno.

Atenção 2: A falta no dia da apresentação do trabalho implicará em ZERO na avaliação de AP2, ainda que o aluno tenha feito o trabalho e publicado no SAVA.

## 1. Introdução

Este projeto tem como objetivo consolidar os conhecimentos adquiridos na disciplina de Programação Orientada a Objetos por meio da criação de uma aplicação prática dentro do tema crowdsourcing. Os alunos deverão aplicar os conceitos de orientação a objetos e realizar a persistência de dados em banco relacional.

## 2. Requisitos Técnicos

### 2.1 Pilares da Orientação a Objetos

- Abstração: modelagem adequada das entidades.
- Encapsulamento: uso apropriado de modificadores de acesso.
- Herança: hierarquia de classes com reutilização de código.
- Polimorfismo: sobrescrita e sobrecarga com invocação polimórfica.

### 2.2 Classes Abstratas e Interfaces

- Pelo menos **uma classe abstrata** com métodos abstratos e concretos.
- Pelo menos **uma interface** implementada por duas ou mais classes.

### 2.3 Relacionamentos entre Classes

- Diversidade de cardinalidades: 1:1, 1:N, N:N.
- Diversidade de direcionamento: unidirecional e bidirecional (ao menos um).
- Evidência de composição (todo-parte com criação de um objeto de uma classe em outra) **ou** agregação (todo-parte com recebimento de objeto de uma classe por outra para realização da referência).

### 2.4 Collections

- Uso de List, Map, Set etc. com operações de adição, busca e remoção.

### 2.5 Persistência de Dados

- Conexão com banco relacional via JDBC.
- Uso obrigatório do padrão DAO.
- Operações CRUD para as entidades principais.

CRUD é um acrônimo que representa as **quatro operações básicas** realizadas em sistemas que manipulam dados persistentes (como bancos de dados). Ele significa:

- **C – Create:** Criar um novo registro (ex: inserir um novo usuário no banco de dados).

- **R – Read:** Ler ou recuperar dados (ex: buscar os projetos cadastrados).
- **U – Update:** Atualizar informações existentes (ex: alterar o email de um usuário).
- **D – Delete:** Remover registros (ex: excluir um projeto).

### 3. Entrega

- Link do git com o código-fonte completo e instruções de execução (indicação da classe com o método main).
- Diagrama de classes.
- Script SQL de criação do banco de dados.
- Documento PDF com descrição, justificativas e evidências.

### 4. Critérios de Avaliação

A nota do trabalho em si será atribuída de acordo com os seguintes critérios:

- Aplicação dos pilares de OO – 25%
- Qualidade da modelagem das classes e do banco (entenda modelagem como sendo a concepção e implementação das entidades para o domínio de problema escolhido) – 20%
- Uso de classes abstratas e interfaces – 10%
- Uso adequado de collections – 10%
- Persistência com JDBC + DAO – 20%
- Organização do código e documentação – 15%

### 5. Observações Finais

- O trabalho é em grupo, mas a avaliação será individual. Para que o aluno individualmente obtenha a nota máxima dada ao trabalho apresentado, ele deverá comprovar a sua participação na concepção do trabalho realizando alterações no código a pedido da professora durante a apresentação do trabalho no dia estipulado.
- Plágio será penalizado com nota zero para todos os envolvidos.
- Dúvidas devem ser encaminhadas à professora durante as aulas ou pelo discord.