

## Relatório individual - Desenvolvimento do código

Desenvolvi o código em conjunto com meu grupo e participei de toda a montagem do código. Mesmo assim eu acabei enfrentando dificuldades na montagem dos métodos *vender\_ingresso* da class *Festival*, mas depois de pesquisar e discutir com meus colegas consegui corrigir os erros do código.

### Conceitos Aplicados no Projeto:

O projeto simula o funcionamento de uma Empresa de eventos, com os festivais/eventos, palcos, funcionários, clientes, vendas de ingressos e com um auditor, foi utilizado python em conjunto com alguns conceitos da Programação Orientada a Objetos (POO).

**-Classes, Mixins e Interface:** O código foi estruturado com as classes *Pessoa*, *Ingresso*, *Cliente*, *Funcionário*, *Palco*, *Festival*, *EmpresaEventos* e *Auditor*, com os mixins *AuditavelMixin* e *IdentificavelMixin* e com a interface *Logavel*.

**-Encapsulamento:** Uso de atributos protegidos como o *self.\_cargo* e métodos públicos para ter acesso a esses atributos como o *get\_id()*.

**-Herança:** As classes *Funcionário* e *Cliente* herda da classe *Pessoa*, *Funcionário* e *Auditor* também herdam do mixin *IdentificavelMixin* e da interface *Logavel*.

**-Polimorfismo:** O método *logar\_entrada()* é definido na classe abstrata *Logavel* e implementada de maneira diferente nas classes *Funcionário* e *Auditor*.

**-interface:** A classe abstrata *Logavel* define o método *logar\_entrada()* obrigando as classes que herdam dela de implementarem esse método.

**-Mixins:** A classes *IdentificavelMixin* e *AuditavelMixin* apenas fornecem funcionalidades adicionais, como gera um ID ou login.

**-Tratamento de Erro:** Eu adicionei o *isinstance()* em métodos que utilizavam objetos instanciados por classes para garantir que não fosse colocado um objeto diferente. Também utilizei o *ValueError()* para apresentar o problema que o código está sofrendo.

**-Composição e Agregação:** A classe *Palco* faz parte da classe *Festival*, ou seja, a classe *Festival* compõe um um objeto *Palco* por meio do atributo *self.palco = palco*, sem o *Festival* esse objeto *Palco* não tem sentido.

Nome:Murilo Alves de Lima

12/08/2025

11:00