



**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
DE SÃO PAULO**

CAMPUS GUARULHOS

BEATRIZ MAZZUCATTO - GU3046745

EXERCÍCIO DE USO DE CLASSES ABSTRATAS

**GUARULHOS - SP
2025**

EXERCÍCIO DE USO DE CLASSES ABSTRATAS

Atividade apresentada à disciplina de API e Microsserviços, do Instituto Federal de Educação, Ciência e Tecnologia Campus Guarulhos como nota parcial para aprovação na disciplina do curso Engenharia de Computação, sexto semestre.

Professor Mestre: Giovani Fonseca Ravagnani Disperati

1. Objetivo

Este documento apresenta propostas de melhorias para o sistema, com foco em aumentar a confiabilidade, extensibilidade, usabilidade e persistência de dados, seguindo boas práticas de engenharia de software.

2. Escopo das Melhorias

As melhorias propostas abrangem as áreas de:

- Validação de dados
- Organização e uso de polimorfismo
- Extensibilidade da hierarquia de classes
- Persistência de informações
- Interface de usuário

3. Melhorias Propostas

3.1 Validação de Dados

Situação Atual:

O sistema não realiza validações consistentes nos dados de entrada.

Proposta:

- Implementar validações para:
 - o **Data de Nascimento:** Verificar se a data informada é válida.
 - o **Salário:** Garantir que não sejam aceitos valores negativos.
- Modificar a classe Data para lançar exceções (IllegalArgumentException) em casos de entrada inválida.

Benefícios:

- Garantia de integridade dos dados.

- Prevenção de inconsistências no processamento.

3.2 Polimorfismo Aprimorado

Situação Atual:

A classe Funcionario é abstrata e instanciada no teste através de classe anônima.

Proposta:

- Criar a classe concreta FuncionarioRegular para substituir a implementação anônima.

Benefícios:

- Código mais organizado e reutilizável.
- Facilidade na manutenção e extensão da lógica de negócio.

3.3 Extensão do Sistema

Situação Atual:

O sistema trata apenas de clientes do tipo PessoaFisica e PessoaJuridica.

Proposta:

- Adicionar novas subclasses de Pessoa, como:
 - o Fornecedor: Inclui CNPJ e condições comerciais.
 - o Estagiario: Inclui período de estágio e bolsa-auxílio.

Benefícios:

- Maior flexibilidade para representar diferentes perfis.
- Aproveitamento da arquitetura atual baseada em herança e classes abstratas.

3.4 Persistência de Dados

Situação Atual:

Todas as informações são armazenadas em memória volátil, sendo perdidas após o encerramento da aplicação.

Proposta:

- Implementar persistência utilizando:
 - o Arquivos (.txt, .csv, .json)
 - o Banco de dados relacional (ex.: MySQL, PostgreSQL)
 - o Banco de dados embarcado (ex.: SQLite)

Benefícios:

- Manutenção dos dados entre execuções.
- Possibilidade de integração com outros sistemas e relatórios.

3.5 Interface de Usuário

Situação Atual:

A interação é feita apenas via console simples.

Proposta:

- Desenvolver:
 - o Interface gráfica (JavaFX, Swing)
 - o Ou menu de console aprimorado, com opções claras e mensagens padronizadas.

Benefícios:

- Melhor experiência do usuário.
- Aumento da acessibilidade para operadores não técnicos.

4. Conclusão

As melhorias propostas alinham-se às boas práticas de design e arquitetura, seguindo princípios **SOLID** e padrões de projeto. A implementação trará benefícios diretos à qualidade do sistema, permitindo crescimento seguro e sustentável.