**UFU – UNIVERSIDADE FEDERAL DE UBERLÂNDIA**

**RELATÓRIO DO PROJETO - SISTEMA DE LOCADORA DE JOGOS**

**LOCADORA 2000**

**CURSO:** Sistema de Informação  
**DISCIPLINA:** Programação Orientada a Objetos

**ALUNOS:**

* Camila Imolesi Gomes – Matrícula: 12411BSI427
* Breno Enedino Carrijo Pereira - Matrícula:
* Victor Hugo Gabriel dos Reis - Matrícula:
* William Braga de Oliveira Junior - Matrícula:

**PROFESSOR:** Roberto Júnio Silva Caetano

**1. INTRODUÇÃO**

Este trabalho é um sistema para locadora de jogos que desenvolvemos em Java. A ideia foi criar um programa que ajuda uma locadora a controlar seus clientes, funcionários, jogos e aluguéis.

O sistema faz tudo que uma locadora precisa:

* Cadastra clientes e funcionários
* Controla os jogos disponíveis
* Faz o aluguel dos jogos
* Gerencia as devoluções
* Mostra tudo organizado na tela

Desenvolvemos usando os conceitos de POO que aprendemos nas aulas, com uma interface gráfica fácil de usar.

**2. MODELAGEM DO SISTEMA**

**2.1 Como as Classes se Organizam**

Pessoa (Classe principal)

|

|--- Cliente (Herda de Pessoa)

|

|--- Funcionario (Herda de Pessoa - tem cargo)

Jogo (Classe dos jogos)

Locacao (Junta Cliente + Jogo + tempo)

**2.2 O que cada Classe faz:**

* **Pessoa**: Classe pai com dados básicos (nome e telefone)
* **Cliente**: Pessoas que alugam jogos
* **Funcionario**: Pessoas que trabalham na locadora
* **Jogo**: Cada jogo com nome, console e preço
* **Locacao**: Registra cada aluguel feito

**3. CONCEITOS DE POO QUE USAMOS**

**3.1 Herança**

***Cliente é uma Pessoa:***

**Texto

O conteúdo gerado por IA pode estar incorreto.  *Herda nome e telefone da classe Pessoa***

***Funcionario é uma Pessoa:***

**Interface gráfica do usuário, Texto

O conteúdo gerado por IA pode estar incorreto.**

***Herda nome e telefone e acrescenta cargo***

**3.2 Polimorfismo (Comportamento Diferente)**

Cada classe filha se comporta de forma diferente:

*Na classe Pessoa:*

Texto

O conteúdo gerado por IA pode estar incorreto.

*Na classe Cliente:*

Interface gráfica do usuário, Texto

O conteúdo gerado por IA pode estar incorreto.

*Diz que é cliente*

*Na classe Funcionário:*

Texto

O conteúdo gerado por IA pode estar incorreto.

*Diz que é funcionário*

**3.3 Encapsulamento (Proteção dos Dados)**

*Atributos ficam privados (protegidos):*

Interface gráfica do usuário, Texto

O conteúdo gerado por IA pode estar incorreto.

*Só acessamos através dos métodos:*



**3.4 Composição**

Class Locacao:

Texto

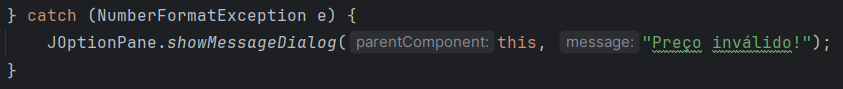
O conteúdo gerado por IA pode estar incorreto.

*Quando fazemos aluguel, juntamos cliente e jogo*

**3.5 Tratamento de Erros**

Interface gráfica do usuário, Texto, Aplicativo

O conteúdo gerado por IA pode estar incorreto.*Verifica se o telefone tem 11 números:*

Interface gráfica do usuário, Texto, Aplicativo

O conteúdo gerado por IA pode estar incorreto.*Verifica se o preço é válido:*

**4. COMO USAR O SISTEMA**

**4.1 Primeiros Passos**

1. Execute o programa: Clique duas vezes no arquivo SistemaLocadora.java
2. A tela principal aparece com 7 botões
   1. **Cadastrar Funcionário**
3. Clique em **"Add Funcionário"**
4. Digite o nome do funcionário
5. Digite o telefone com 11 números (ex: 11999998888)
6. Digite o cargo (ex: Atendente)
7. Clique em OK

**4.3 Cadastrar Cliente**

1. Clique em **"Add Cliente"**
2. Digite o nome do cliente
3. Digite o telefone com 11 números (ex: 11999998888)
4. Clique em OK

**4.3 Cadastrar Jogo**

1. Clique em **"Add Jogo"**
2. Digite o nome do jogo (ex: FIFA 2024)
3. Digite o console (ex: PS5)
4. Digite o preço por dia (ex: 15.00)
5. Clique em OK

**4.4 Fazer Aluguel**

1. Clique em **"Alugar"**
2. Escolha o cliente na lista
3. Escolha o jogo disponível
4. Digite quantos dias de aluguel
5. Clique em OK

**4.5 Devolver Jogo**

1. Clique em **"Devolver"**
2. Escolha o aluguel para devolver
3. O sistema calcula o valor total
4. Clique em OK

**4.6 Ver Tudo**

1. Clique em **"Listar"**
2. A tela mostra todos os clientes, jogos e aluguéis
3. Atualiza automaticamente

**4.7 Screenshots**

* Tela inicial:

Tabela

O conteúdo gerado por IA pode estar incorreto.

* Interface gráfica do usuário, Aplicativo

  O conteúdo gerado por IA pode estar incorreto.Cadastro de Funcionário (nome):
* Interface gráfica do usuário, Aplicativo

  O conteúdo gerado por IA pode estar incorreto.Cadastro de Funcionário (telefone):
* Interface gráfica do usuário, Aplicativo

  O conteúdo gerado por IA pode estar incorreto. Cadastro de Funcionário (cargo):
* Interface gráfica do usuário, Aplicativo

  O conteúdo gerado por IA pode estar incorreto.Cadastro de Funcionário efetuado com sucesso:
* Interface gráfica do usuário, Aplicativo

  O conteúdo gerado por IA pode estar incorreto.Cadastro de cliente (Nome):
* Interface gráfica do usuário, Aplicativo

  O conteúdo gerado por IA pode estar incorreto.Cadastro de cliente (Telefone):
* Interface gráfica do usuário, Aplicativo

  O conteúdo gerado por IA pode estar incorreto.Cadastro de cliente efetuado com sucesso:
* Interface gráfica do usuário, Aplicativo

  O conteúdo gerado por IA pode estar incorreto.Cadastro de jogos (Nome):
* Interface gráfica do usuário, Aplicativo

  O conteúdo gerado por IA pode estar incorreto.Cadastro de jogos (Console)
* Interface gráfica do usuário, Aplicativo

  O conteúdo gerado por IA pode estar incorreto.Cadastro de jogos (Valor da diária):
* Cadastro de jogo efetuado com sucesso:

Interface gráfica do usuário, Aplicativo

O conteúdo gerado por IA pode estar incorreto.

* Interface gráfica do usuário, Aplicativo

  O conteúdo gerado por IA pode estar incorreto.Tela de aluguel (escolha de cliente):
* Interface gráfica do usuário, Aplicativo

  O conteúdo gerado por IA pode estar incorreto.Tela de aluguel (escolha de jogo):
* Tela de aluguel (quantidade de dias):

Interface gráfica do usuário, Aplicativo

O conteúdo gerado por IA pode estar incorreto.

* Tela de aluguel, locação efetuada com sucesso:

Interface gráfica do usuário, Aplicativo

O conteúdo gerado por IA pode estar incorreto.

* Interface gráfica do usuário, Aplicativo

  O conteúdo gerado por IA pode estar incorreto.Tela de devolução
* Interface gráfica do usuário, Aplicativo

  O conteúdo gerado por IA pode estar incorreto.Tela de devolução, devolução efetuada com sucesso:
* Interface gráfica do usuário, Aplicativo, Tabela

  O conteúdo gerado por IA pode estar incorreto.Listagem
* Interface gráfica do usuário, Aplicativo

  O conteúdo gerado por IA pode estar incorreto.Erro (Telefone inválido):

**5. CONCLUSÃO**

Este projeto nos mostrou como usar na prática os conceitos de POO. Aprendemos muito:

* Como fazer herança entre classes
* Como usar polimorfismo para comportamentos diferentes
* Como proteger os dados com encapsulamento
* Como juntar objetos com composição
* Como fazer uma interface gráfica que funciona

Foi desafiador fazer o sistema completo, mas conseguimos implementar todas as funcionalidades que uma locadora precisa, incluindo o cadastro de funcionários que havia ficado faltando inicialmente.

O sistema está pronto para uso e pode ser melhorado no futuro com mais recursos como persistência em banco de dados e relatórios financeiros.

**BIBLIOGRAFIA**

1. DEITEL, P. J.; DEITEL, H. M. Java: como programar. 10ª ed. Pearson, 2017.
2. SIERRA, K.; BATES, B. Use a Cabeça! Java. 2ª ed. Alta Books, 2010.