

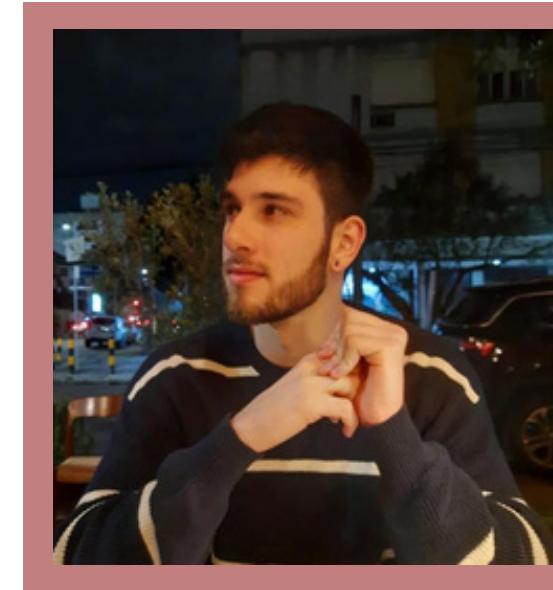


Técnicas de Construção de Programas

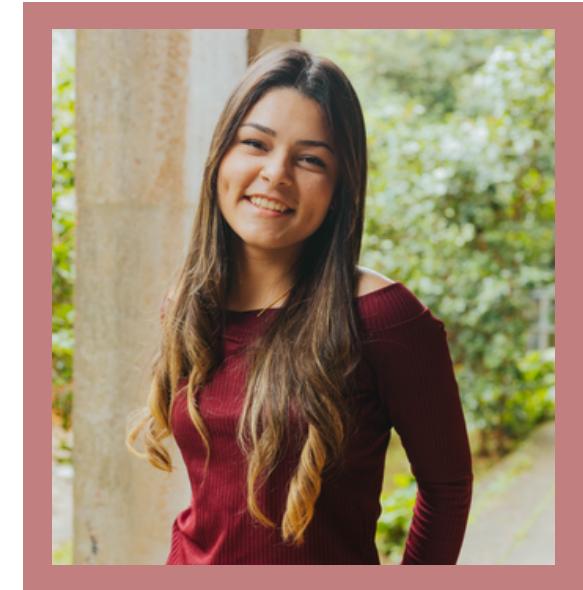
ARIGÓFLIX

Artur Webber – Bruna Rosa – Elano Tavares – João Dorneles

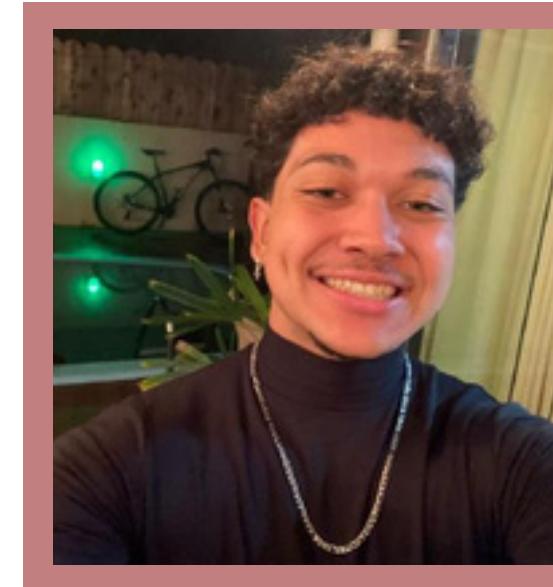
NOSO TIME



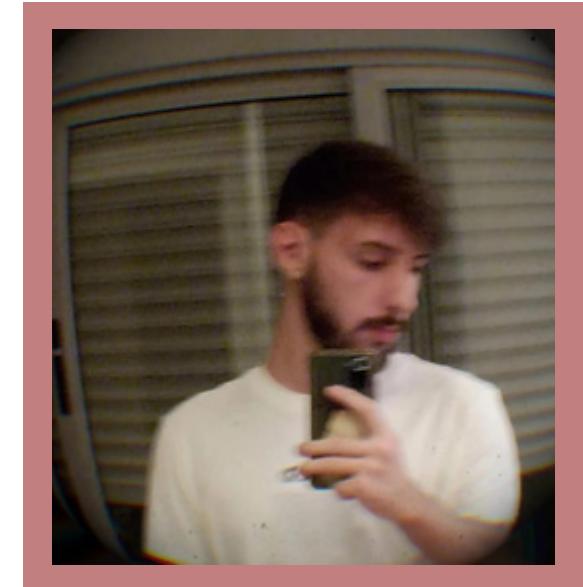
ARTUR WEBBER



BRUNA ROSA



ELANO TAVARES



JOÃO DORNELES

O que é o ARIGÓFLIX?

- **Proposta:** Plataforma centralizada para descoberta e avaliação de entretenimento (Filmes, Séries, Jogos e Livros).

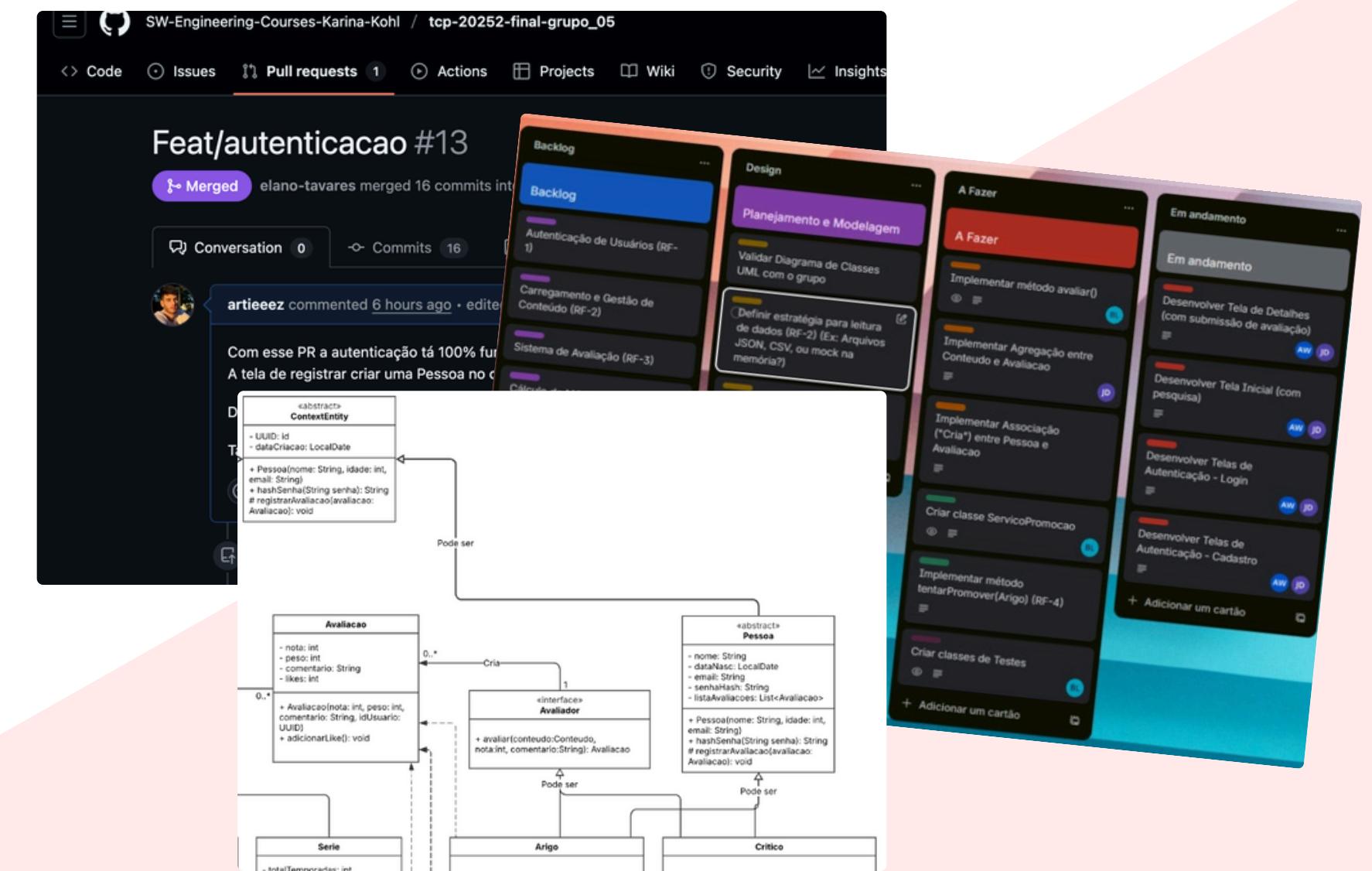
- **O Problema:** Agregadores comuns tratam todas as opiniões com o mesmo peso, misturando reviews casuais com análises técnicas.

- **Nossa Solução:** Um sistema de Média Ponderada que distingue dois perfis de usuários:
 1. Arigó: Usuário comum (peso padrão).
 2. Crítico: Usuário especializado (peso maior na nota final).

FERRAMENTAS

Foram usadas três ferramentas:

1. Github para colaborar e revisar pull requests;
2. Um quadro kanban no Trello para organizar as tarefas;
3. LucidChart para produção do UML;



Organização

Branches:

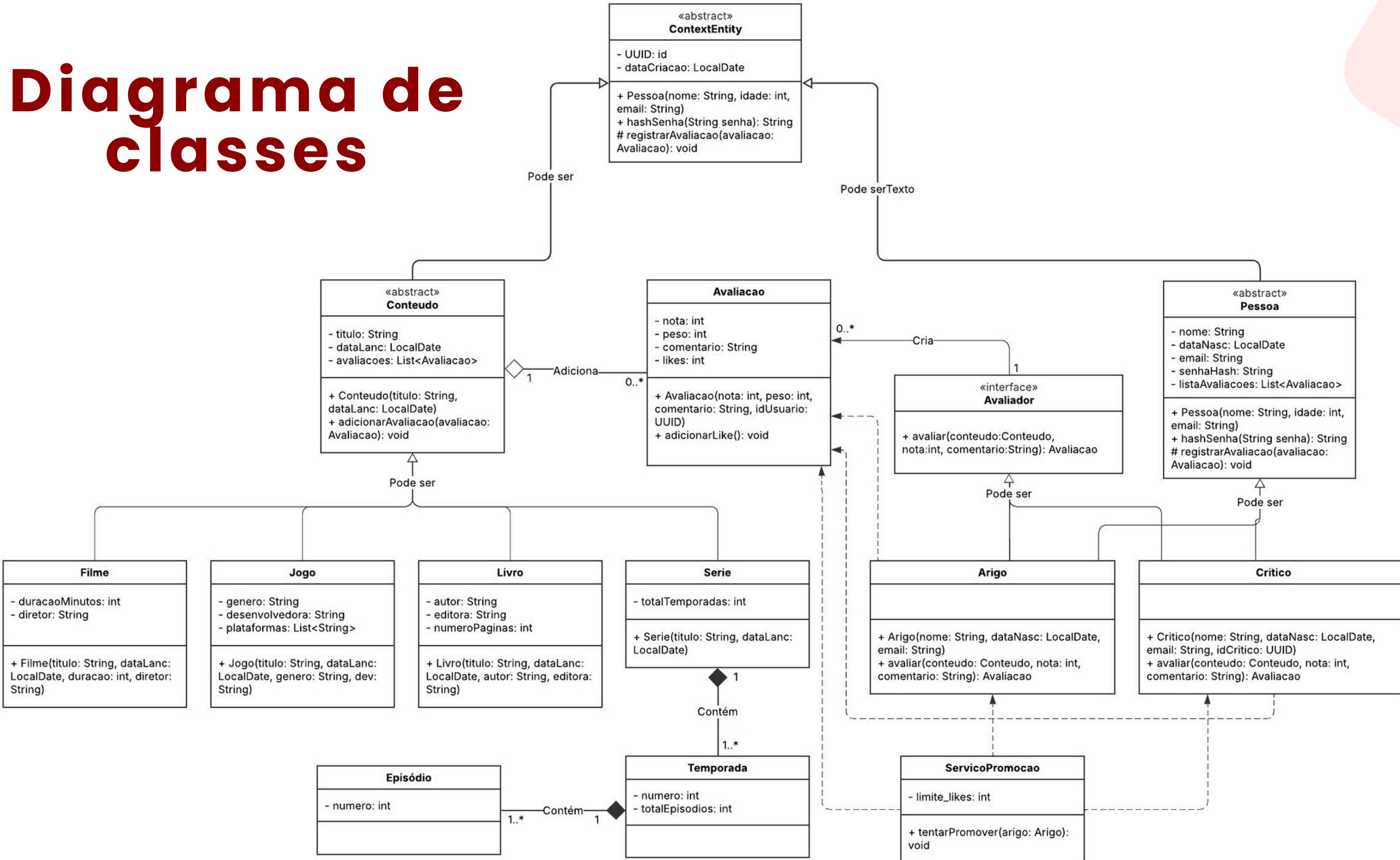
- main
- develop
- feat/feature-name

Fluxo de trabalho:

- Criar branch feat
- Manter atualizada com develop
- PR para develop ao terminar a implementação
- Revisão por par e aceitação do PR



Diagrama de classes



Estrutura do Projeto

```
TCP-20252-final-grupo-05
|
+-- config           <-- Arquivos de configuração (tinylog.properties)
+-- lib              <-- Dependências externas (Tinylog, JUnit)
+-- scripts          <-- Scripts de automação (build/test para Windows e Linux/macOS)
+
+-- src
|   |
|   +-- main    <-- Pacote principal
|   |   |
|   |   +-- Main.java      <-- Classe principal (Ponto de entrada)
|   |   |
|   |   +-- models          <-- (Entidades do domínio)
|   |   |   ContextEntity.java (abstrata base)
|   |   |   Pessoa.java      (abstrata)
|   |   |   Arigo.java
|   |   |   Critico.java
|   |   |   Avaliador.java   (interface)
|   |   |   Conteudo.java    (abstrata)
|   |   |   Filme.java
|   |   |   Serie.java
|   |   |   Livro.java
|   |   |   Jogo.java
|   |   |   Temporada.java
|   |   |   Episodio.java
|   |   |   Avaliacao.java
|   |   |
|   |   +-- service         <-- (Lógica de negócios e persistência em memória)
|   |   |   Context.java     (Container dos repositórios e estado global)
|   |   |   BaseRepository.java
|   |   |   ConteudoRepository.java
|   |   |   CarregadorDeDados.java
|   |   |   ServicoPromocao.java
|   |   |
|   |   +-- autenticacao    <-- (Subpacote de segurança)
|   |   |   Autenticacao.java (interface)
|   |   |   ServicoAutenticacao.java
|   |   |   Exceptions... (CredenciaisInvalidas, EmailJaCadastrado)
|   |   |
|   |   +-- ui (ou view)     <-- (Interface Gráfica Swing)
|   |   |   GerenciadorTelas.java (Controlador de navegação)
|   |   |   TelaLogin.java
|   |   |   TelaCadastro.java
|   |   |   TelaInicial.java
|   |   |   TelaDetalhes.java
|   |   |   TelaFilmeLista.java
|   |   |   TelaSerieLista.java
|   |   |   TelaLivroLista.java
|   |   |   TelaJogoLista.java
|   |   |
|   +-- resources
|       +-- data           <-- (Arquivos CSV: filmes, series, usuarios, etc.)
+
+-- test             <-- (Testes Unitários JUnit para models e services)
```



**OBRIGADO PELA
ATENÇÃO!**