



Universidade de São Paulo

Instituto de Ciências Matemáticas e Computação

Departamento de Ciências de Computação

Disciplina de Programação Orientada a Objeto (SSC0103)

**Documentação do Trabalho de Final de Programação
Orientada a Objeto**

Exibição de Filmes em Cartaz

Prof. Dr. Márcio Delamaro
Estagiário PAE: Lucas Diniz Dallilo

Alunos:

Alcino Salviano Cavalcanti, nUSP: 11892963
Gustavo Wadas Lopes , nUSP: 12745640
José Pedro Cioni do Carmo, nUSP: 12623988
Victor Lucas de Almeida Fernandes, nUSP: 12675399

São Carlos

Data de Entrega: 22 de julho de 2022

1. Introdução

1.1 Sobre o Projeto

O Projeto Final criado para a disciplina de Programação Orientada a Objeto (SSC0103) consiste de um programa que exibe os filmes em cartaz nos cinemas para o usuário através de uma interface gráfica. Por meio dele o usuário tem à disposição os filmes e as suas informações básicas, que por sua vez foram coletadas da rede por meio de uma extração de dados.

O programa foi criado utilizando a linguagem de programação Python bem como as bibliotecas disponíveis na linguagem. Os testes foram realizados no sistema operacional Windows e a interface gráfica foi implementada com a biblioteca *PySimpleGUI*.

2. Objetivos

O objetivo principal do projeto foi a elaboração de um programa que permitisse ao usuário ter acesso a um número de filmes, em exibição no momento da utilização do programa, e também às suas informações básicas, como avaliação e sinopse. Além disso, foi buscado aplicar os conhecimentos de programação orientada a objetos e da linguagem Python para criar as principais ferramentas do programa: a extração de dados feita de um site na internet e o desenvolvimento de uma interface gráfica.

3. Descrição do Software

3.1 Funcionamento do Programa e Implementação do Scrapper

O programa utiliza a biblioteca scrapper para coletar os dados em html do site ["https://www.adorocinema.com/filmes/numero-cinemas/"](https://www.adorocinema.com/filmes/numero-cinemas/) . obtendo o título, descrição, nota da primeira avaliação disponível (normalmente dos usuários) e os links das imagens dos pôsteres de cada filme utilizando o seletor css. As notas tem que passar por uma etapa de verificação pois nem todos os filmes têm uma disponível, com a nota default sendo 0,0. Similarmente, é necessário fazer a checagem dos links das imagens uma vez que eles podem não ser válidos. Caso esse seja o caso, as imagens são adicionadas a partir de links data-src. Os dados são retornados como listas de cada respectivo tipo e são passados para outras bibliotecas através do vetor de info.

Na main, os filmes obtidos com o scrapper são armazenados em um vetor de filmes o qual é usado para chamar a interface do programa. A interface apresenta cada filme como uma pequena página que permite ao usuário navegar entre os filmes como desejar.

Ao final do programa, ele apaga os arquivos de imagem do diretório.

3.2 Classes Implementadas

O programa é composto por quatro classes:

- **Main:** Chama a interface a partir dos filmes obtidos com o scrapper.
- **Interface:** Chama a interface gráfica para os filmes
 - **Construtor:** recebe uma lista com os dados de todos os filmes e implementa o padrão da interface gráfica. Cria uma lista com as janelas, uma para cada filme, e desenvolve um layout padrão em colunas que insere as informações dos filmes que serão exibidas para o usuário. Implementa os botões de auxílio na navegação entre as páginas. O processo é repetido para cada janela, gerando, assim, uma página de exibição padrão para cada filme.
 - **Execute:** método que realiza a leitura das janelas implementadas pelo construtor. Permite a execução do programa ao administrar a navegação entre as diferentes telas por meio da gestão dos eventos. O *looping* que mantém a execução da interface é quebrado quando o evento de fechamento de janela é detectado.
- **Movie:** Contém os dados de um filme.
 - **Construtor:** Cria 4 variáveis, 3 strings que recebem o título, a sinopse e o link da imagem do filme e 1 inteiro que recebe a avaliação do filme;
 - **getters:** Retornam um determinado dado do filme;
 - **setters:** Alteram um determinado dado do filme;
 - **str String:** Escreve os dados do filme como uma única string utilizada para teste.
- **Scraper:** Chama o scrapper para a leitura dos dados de filmes no site utilizados;
 - **CineSpider:** Spider que obtém as informações dos filmes a partir do url '<https://www.adorocinema.com/filmes/numero-cinemas/>' com o seletor css e armazena a lista das informações no vetor global **info**. Como alguns filmes não possuem avaliação, é necessário fazer uma verificação antes da montagem da lista de notas. O mesmo vale para links de imagens dos filmes.

- **vector[[]][[]][[]] get_filmes:** chama o scrapper e retorna o vetor das listas de dados do filmes;
- **clean_images:** Libera a lista de link das imagens do vetor de informações e fecha todos os arquivos de imagem no diretório.

4. Interface Gráfica

A interface gráfica foi implementada fazendo o uso da biblioteca *PySimpleGUI*, apresentada em sala de aula. Por meio dela foi possível criar um layout que permitisse a exibição do *poster* (cartaz) do filme junto com o seu título, avaliação e breve descrição do enredo. Foi escolhido trabalhar com a exibição de apenas um filme por página, dessa forma uma lista com as diversas janelas foi utilizada, sendo a exibição delas para o usuário organizadas de acordo com a interação com os botões da interface. Foi utilizada, ainda, a biblioteca *Pillow* para o tratamento das imagens dos cartazes, uma vez que o *PySimpleGUI* só trabalha nativamente com o formato .png.



5. Classes Python Utilizadas

- *scrappy*: usada na classe scrappy para coletar dados a partir da url utilizada;
- *urllib.request*: utilizada para acessar o site na internet;
- *validators*: usada para verificar os links das imagens no CineSpider();
- *os*: usado para acessar o diretório e apagar os arquivos de imagem nele;
- *io*: utilizada para salvar e atualizar os arquivos de imagem;
- *PySimpleGUI*: utilizada para a criação da interface gráfica;
- *PIL*: utilizada para o tratamento das imagens na interface.

6. Conclusão e Contribuição por Aluno

No geral, a realização do trabalho se deu como o esperado e acreditamos que conseguimos colocar em prática os diversos conceitos vistos sobre programação orientada a objetos e a linguagem Python. Tivemos a oportunidade de aprender a usar novas funcionalidades, como as da biblioteca *scrappy*, e relacioná-las com a nossa proposta. Por outro lado, houve uma certa dificuldade em se lidar com a parte da interface gráfica, fazendo com que desenvolvêssemos algo mais simples do que havíamos planejado na proposta inicial. Acreditamos que a exploração de mais cenários envolvendo interfaces gráficas nas aulas poderiam ter auxiliado o projeto a ficar mais próximo do imaginado.

Todo o trabalho, desde o código até a documentação, foi bem dividido entre o grupo e todos contribuíram igualmente para a realização desse projeto.

7. Link

Link para o vídeo com uma demonstração do programa sendo executado:

<https://www.youtube.com/watch?v=WeQb2aWQI-g>