



Universidade Federal da Bahia

Departamento de Ciência da Computação

MATC82 – Sistemas Web

Professor Leandro Andrade

Nilton Vasques

Vinicius Lins Gesteira

**RELATÓRIO DE IMPLEMENTAÇÃO DE UMA
APLICAÇÃO WEB VOLTADA PARA SUGESTÃO DE
FILMES**

Salvador,

2014

O sistema Sugestão de Filmes foi desenvolvido em linguagem HTML 5, utilizando também recursos de HTML 4, CSS, Java Script bem como a sua biblioteca JQuery. A aplicação tem como finalidade coletar alguns dados a partir do usuário que identifiquem filmes em um banco de dados para retornar o perfil de um filme com imagem de capa, sinopse do enredo, nomes do elenco de atores e vídeo do seu trailer.

Primeiramente, um pop-up “Carregando banco de dados”, juntamente com uma animação de carregamento, exibido na tela pela função `show_popup(popup_id)` do arquivo `filmes.js`, onde constam as principais funções JavaScript do sistema, deve alertar o usuário para o carregamento do banco de dados, escurecendo a página por detrás e travando todas as suas outras funcionalidades, para, em seguida, quando for concluído o carregamento, apagar o pop-up e liberar as funcionalidades da tela a partir da função `hide_popup(popup_id)`.

Os dados de entrada são obtidos a partir dos campos ano (preenchimento de texto), categoria (elemento de uma lista definida previamente), atores (preenchimento de texto) e informações. Ao clicar no botão “Indicamos”, o sistema realiza a consulta ao banco de dados seguindo uma disjunção simples não-exclusiva, ou seja, o resultado será obtido de a partir de qualquer um dos campos preenchidos.

Em outras palavras, o algoritmo de busca utilizado nessa implementação pode ser definido nos seguintes termos: com base nos parâmetros fornecidos pelo usuário, é realizada uma busca comparativa na base de filmes, visando a encontrar os filmes que coincidirem com todos os parâmetros fornecidos, entretanto para os parâmetros que podem ser compostos, como atores e categorias, a busca verifica se o filme tem pelo menos um dos parâmetros fornecidos.

Estilizamos a tela principal no arquivo `index.html` com esses campos definidos sobre um arquivo de estilo `style.css` que definiu o design dos campos em lista para forma de tabela, o formato do botão azul centralizado que realiza o processamento da consulta selecionada pelo usuário, a formatação dos textos retornados, entre outras estilizações voltadas para dar uma aparência leve ao nosso site.

Em seguida, projetamos a aplicação para, ao usuário submeter o formulário clicando no botão “Indicamos”, novamente avisar o usuário para o processamento do sistema com um outro pop-up “Aguarde” exibido pela função `hide_popup(popup_id)` e retirado da tela pela

função `hide_popup(popup_id)`, de forma semelhante ao explicado para o início da execução do sistema, para logo após realizar uma consulta no banco de dados a partir da função JavaScript `listaFilmes()` do arquivo.

Para complementar as buscas, elaboramos as funções `filter_actors(actors, movieActors)`, que conta a quantidade de ocorrências dos atores inseridos pelo usuário, em relação ao filme, e a função `filter_categories(categoriaField, categorias)`, que conta a quantidade de ocorrências das categorias.

Adiante, entra em ação a função `sort_movie()`, pela qual todos os filmes encontrados com base nos filtros são então armazenados no array `movies_matched`, da função `listaFilmes()`, tendo por finalidade sortear de maneira aleatória um dos filmes presentes naquela listagem, utilizando a função `randomFromInterval(max, min)` para sortear números, e em seguida retirá-lo da listagem, garantindo, assim, que durante uma nova indicação, o filme que já foi indicado não possa ser sorteado novamente.

Logo após o sorteio, os dados do filme indicado são exibidos na tela, enquanto os campos e outros elementos da consulta de sugestão são ocultados na tela pela função `show_after_search_panel()` e só são novamente exibidos quando o usuário seleciona a opção do botão “Iniciar nova busca”, que ativa a função `hide_after_search_panel()`.

O ponto mais importante do sistema, a exibição dos dados do filme indicado sobre a tela, está implementado pela função `create_result_row(movie)`, que cria em tempo de execução o código html responsável pela estilização da indicação do filme na tela, assim como os botões “Indica outro nas mesmas condições” e “Iniciar nova busca”.

Caso nenhum filme possa ser indicado ou por que não conste nenhum dos parâmetros no banco ou porque todos desses parâmetros já foram indicados, um pop-up é exibido na tela para comunicar o usuário sobre essa impossibilidade, retornando para a exibição normal dos campos de busca iniciais.

Para complementar as buscas, elaboramos as funções `filter_actors(actors, movieActors)`, que conta a quantidade de ocorrências dos atores inseridos pelo usuário, em relação ao filme, e a função `filter_categories(categoriaField, categorias)`, que conta a quantidade de ocorrências das categorias.

Os dados são: o título e ano do filme, descrição, categorias, atores, imagem de capa e trailer do filme. Enquanto todos os dados com exceção ao último são buscados no banco de dados do sistema, o último está relacionado com a função `search_youtube_trailer(titulo, callback)`, que é responsável por fazer a requisição ajax do trailer do filme a API do Youtube.

Implementamos a ainda a função `watch_trailer(titulo)`, mas ela deixou de ser utilizada. Seu papel era exibir o trailer do filme em um popup na tela do usuário. Na versão mais recente, o filme é exibido juntamente com os outros dados do filme, atendendo, assim, um padrão de desenvolvimento de software mais próximo das especificações do sistema.

Executando testes de aceitação do sistema, conseguimos observar que as funcionalidades apresentam êxito em todas as rotinas designadas, tanto no navegador Google Chrome, quanto no navegador Mozilla Firefox. O sistema de sugestão de filmes proposto por nós encontra-se disponível para todos os cinéfilos indecisos que agora podem contar com uma pequena ajuda dos computadores.