



**UNIVERSIDADE FEDERAL DA FRONTEIRA SUL  
CAMPUS CHAPECÓ  
CURSO DE CIÊNCIA DA COMPUTAÇÃO**

**VLADIMIR BELINSKI**

**SITE DE UMA EMPRESA DE ENTRETENIMENTO**

**CHAPECÓ  
Julho de 2017**

**VLADIMIR BELINSKI**

**SITE DE UMA EMPRESA DE ENTRETENIMENTO**

Trabalho apresentado ao curso de Ciência da Computação da Universidade Federal da Fronteira Sul - UFFS - Campus Chapecó - como requisito parcial para aprovação no Componente Curricular Tópicos Especiais em Desenvolvimento Web.

Orientadora: Prof. Me. Andressa Sebben.

**CHAPECÓ**  
**Julho de 2017**

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO.....</b>	<b>4</b>
<b>2</b>	<b>CONSIDERAÇÕES GERAIS ACERCA DE DIFERENÇAS ENTRE O PROJETO</b>	
<b>1 E O PROJETO 2.....</b>		<b>5</b>
<b>3</b>	<b>CONFIGURAÇÃO DE AMBIENTE .....</b>	<b>5</b>
<b>4</b>	<b>COLEÇÕES, FORMULÁRIOS E INSERÇÕES/ALTERAÇÕES/REMOÇÕES DE</b>	
	<b>DADOS – CONSULTAS DAS COLEÇÕES NO MONGO.....</b>	<b>6</b>
<b>5</b>	<b>PACOTES UTILIZADOS .....</b>	<b>7</b>

## 1 INTRODUÇÃO

O presente trabalho consiste na descrição de uma solução desenvolvida para o Projeto 2. Inicialmente, serão realizadas considerações gerais acerca das diferenças entre o Projeto 1 e o Projeto 2. Também será descrito como foi configurado o ambiente para criação e testes do projeto.

Em sequência, serão destacadas as coleções do Mongo, formulários e componentes para inserção, alteração e exclusão de dados criados. Ademais, também será apresentado como visualizar os dados inseridos nas coleções através do terminal do Mongo.

Por fim, serão apresentados e comentados brevemente os pacotes utilizados.

## 2 CONSIDERAÇÕES GERAIS ACERCA DE DIFERENÇAS ENTRE O PROJETO 1 E O PROJETO 2

O Projeto 2 dá continuidade ao Projeto 1 e, assim, apresenta o mesmo tema e todas as seções que o anterior já apresentava. Complementarmente, adicionou-se uma nova página, “/fanarts”, que consiste em uma área surpresa para fãs, alcançada através de um clique em *link* revelado pelo componente *TapTarget* da página de notícias, o qual já existia no Projeto 1. Nessa nova área os usuários podem visualizar uma coleção de *fanarts* inseridos por todos os usuários que igualmente descobriram a área e filtrar *fanarts* por *uploader*. Ademais, estando nessa página o usuário pode criar uma conta no site ou realizar *login* via conta do Facebook, e, com isso, ser permitido a inserir *fanarts*, bem como avaliar e remover os *fanarts* inseridos por ele.

Ainda se tratando de novas funcionalidades, tem-se que no Projeto 2 os dados do formulário da página de audições são agora registrados em um banco de dados e ao serem submetidos com sucesso é apresentado um alerta de confirmação ao usuário, que faz uso de customizações do pacote *SweetAlert*.

Acerca do código, passou-se a utilizar o *framework* Meteor e alguns pacotes novos, não sendo mais utilizado Handlebars e PHP. Além disso, os arquivos de código (antigos e novos) e a organização de pastas foi alterada para a estrutura utilizada pelo Meteor.

## 3 CONFIGURAÇÃO DE AMBIENTE

Para a configuração do ambiente de desenvolvimento e testes, bem como instalação dos pacotes utilizados no projeto, já tendo sido realizada a instalação do Meteor (<https://www.meteor.com/>) e estando no local onde se desejava criar a pasta do projeto, foram executados os comandos:

- a) `meteor create cleverEntertainment`
- b) `cd cleverEntertainment`
- c) `meteor`
- d) `meteor npm install --save bcrypt`
- e) `meteor add materialize:materialize`
- f) `meteor add barbatus:stars-rating`
- g) `meteor add fortawesome:fontawesome`
- h) `meteor add accounts-ui accounts-password`
- i) `meteor add accounts-facebook`

- j) *meteor add facebook-config-ui*
- k) *meteor add session*
- l) *meteor add iron:router*
- m) *meteor add kevhogan:sweetalert*
- n) *meteor remove insecure*

Após isso, foram criados e/ou editados os arquivos de código presentes nas pastas “client” e “server” deste repositório e adicionadas as pastas “lib” e “public” e seu conteúdo. Os testes foram realizados diretamente na aplicação ou no *console* do navegador Chrome (aba *console* a partir de F12) e através do terminal do Mongo. Os pacotes inseridos através dos comandos acima serão comentados na Seção 5.

#### 4 COLEÇÕES, FORMULÁRIOS E INSERÇÕES/ALTERAÇÕES/REMOÇÕES DE DADOS – CONSULTAS DAS COLEÇÕES NO MONGO

O Projeto 2 apresenta duas coleções de dados. Na primeira, nomeada *Auditions*, são armazenados os dados provindos do formulário presente na página de audições: nome, sobrenome, data de nascimento, telefone, sexo, e-mail, categoria para audição, país da audição e descrição. Nesse formulário, todos os campos são validados e, ao serem inseridos dados válidos em todos os campos e pressionado o botão de envio, é apresentado um alerta informando que o formulário foi enviado com sucesso. Cabe destacar que foram realizados tratamentos para que o envio do formulário não gerasse um recarregamento de página e o alerta foi customizado utilizando-se o pacote *SweetAlert*.

A segunda coleção, denominada *Images*, armazena os dados provindos do formulário presente na página de *fanarts*: URL do *fanart*, título e autor. Também armazena quem realizou o *upload* e a data do envio.

Sobre os *fanarts* é possível realizar as operações de um CRUD: inserção (ao estar logado), visualização/leitura, atualização (da avaliação do *fanart*, ao estar logado e ser o *uploader* do *fanart*) e remoção (ao estar logado e ser o *uploader* do *fanart*). Também é possível filtrar a visualização dos *fanarts* por usuário. Cabe destacar que também foram tomados cuidados com segurança para que tais operações e restrições não sejam violadas por comandos a partir do *console* do navegador (feito uso do comando *meteor remove insecure* e realizados tratamentos no arquivo *lib/banco.js*).

A visualização dos dados das coleções pode ser realizada como segue. Pelo terminal, estando na pasta do projeto inserir o comando *meteor mongo* e:

- a) para visualizar os dados da coleção *Images*: *db.Images.find({})*;
- b) para visualizar os dados da coleção *Auditions*: *db.Auditions.find({})*;

## 5 PACOTES UTILIZADOS

Os pacotes utilizados no trabalho foram:

- a) *meteor add materialize:materialize*: pacote que adiciona as funcionalidades do Materialize ao projeto;
- b) *meteor add barbatous:stars-rating*: pacote que adiciona um controle de avaliação com estrelas. Utilizado na avaliação dos *fanarts*;
- c) *meteor add fontawesome:fontawesome*: pacote de fontes. Utilizado para alterar a fontes das estrelas da avaliação dos *fanarts*;
- d) *meteor add accounts-ui accounts-password*: pacote para autenticação de usuários, criação de contas...
- e) *meteor add accounts-facebook*: pacote para autenticação de usuário a partir de conta do Facebook;
- f) *meteor add facebook-config-ui*: pacote para configuração da autenticação de usuário a partir de conta do Facebook;
- g) *meteor add session*: pacote para utilização de sessões/ armazenamento de variáveis temporárias. Utilizado para a filtragem de *fanarts* por usuário;
- h) *meteor add iron:router*: pacote para adição de mecanismo de roteamento;
- i) *meteor add kevhogan:sweetalert*: pacote utilizado para customização de mensagens de alerta. Utilizado no alerta emitido ao se submeter com sucesso dados pelo formulário da página de audições.