**Desenvolvimento de um sistema web para stream de filmes por assinatura**

**Erik Kened Lopes Ferreira¹**

### Prof. Hylson Vescovi Netto (orientador)¹

¹Instituto Federal Catarinense – Campus Blumenau

Blumenau, SC – Brasil

### {erikkened51@gmail.com, hylson.vescovi@ifc.edu.br}

***Resumo***​*. Este trabalho descreve as atividades envolvidas no planejamento e desenvolvimento de uma aplicação web voltada para o fornecimento de filmes por meio de streaming, esta aplicação disponibilizará vários títulos a seus usuários os quais realizaram pagamento de assinatura mensal para ter acesso ao conteúdo do site.*

# 1. Introdução

pendente.

## 1.1. Tema/Problema

pendente.

## 1.2. Objetivos Propostos/Solução dos Problemas

O objetivo deste trabalho é fornecer uma aplicação web para o fornecimento de filmes por meio de *stream*.

Os objetivos específicos são:

1. Pesquisar o referencial teórico para o desenvolvimento do sistema.
2. Levantar os requisitos funcionais e não funcionais do sistema.
3. Desenvolver o projeto do banco de dados relacional.
4. Implementar o sistema.
5. Realizar testes.

## 1.3. Escopo

Pendente.

## 1.4. Viabilidade do Projeto

Pendente.

## 1.5. Método de Trabalho

Pendente.

## 1.6. Ferramentas

### 1.6.1. WAMP

Pendente.

### 1.6.2. PHP MyAdmin

Pendente.

### 1.6.2. Microsoft Visual Studio Code

Pendente.

### 1.6.3. Astah Community 7.1

O Astah é uma ferramenta que possibilita a criação e modelagem dos diagramas UML - *Unified Modeling Language* ou Linguagem de Modelagem Unificada do projeto (ASTAH, 2017).

A ferramenta foi utilizada para o desenvolvimento do diagrama de Caso de Uso apresentado na seção 4.0 deste documento.

## 1.7. Front-end

### 1.7.1. HTML 5

*HyperText Markup Language* – Linguagem de Marcação de Texto ou ainda HTML, é uma linguagem utilizada na construção de páginas ​*web* ​para publicação de conteúdo, na qual os documentos criados na linguagem são interpretados pelos navegadores e apresentados ao usuário, no formato de letras diferenciadas em cores, em imagens e em vídeos.

Em sua versão 5 foram incluídas novas ​*tags* para realização de marcações responsáveis por identificar o conteúdo presente na página, novos elementos para vídeos e imagens, elementos para desenhos, controles para formulários e suporte ao CSS3 (W3SCHOOLS, 2021).

### 1.7.2. CSS3

A ferramenta denominada de ​*Cascading Style Sheets* ​(CSS) ou​Folhas de Estilo em Cascata é uma linguagem utilizada para definir a aparência de documentos, que adotam as linguagens de marcação como, por exemplo, o HTML, tendo em vista que o CSS define como serão exibidos os elementos existentes em uma página da ​*web*​.

Em sua versão 3 foram incluídas novas propriedades, de forma a facilitar a criação de estilos para os componentes, realizar avanços na utilização das cores e textos e, ainda, possibilitar selecionar atributos HTML com os seletores entre outros (​W3SCHOOLS​, 2021).

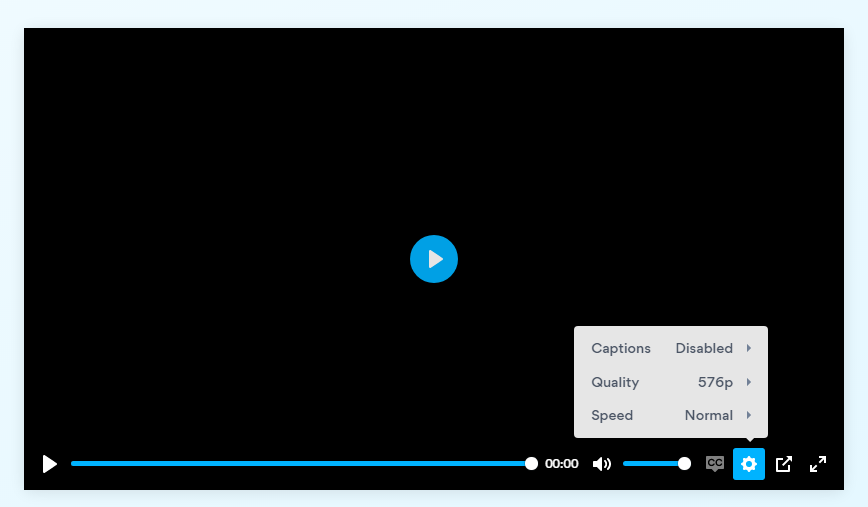
### 1.7.3. Angular cli 11.0.3

O Angular é um framework [*JavaScript*](#_1.7.6._JavaScript) código aberto, mantido pelo Google, que auxilia na execução de *single-page applications*. Seu objetivo é aumentar aplicativos que podem ser acessados por um navegador web, foi construído sob o padrão *model-view-view-model* (MVVM), em um esforço para facilitar tanto o desenvolvimento quanto o teste dos aplicativos. (Angular.io, 2021).

### 1.7.4. Plyr

O Plyr é um ​*framework* ​utilizado, também, para facilitar o desenvolvimento do *front-end*​ da aplicação, contém recursos para o controle de exibição de vídeos em diversos formatos, inclui a possibilidade de adicionar legendas e outras informações pertinentes a exibição do vídeo. (Plyr.io, 2021).

A Figura 1 demonstra um exemplo de uso do ​*framework*​.



**Figura 1. Plyr. Fonte: Plyr.io.**

### 1.7.5. Material Design Icons Iconfont 6.1

Material Desing Icons Iconfont é um conjunto de arquivos pré-compilados, que possibilita a utilização de vários ícones e animações personalizados no projeto (Material.io, 2021).

Esta ferramenta foi utilizada para atender a todos os ícones do projeto.

### 1.7.6. JavaScript

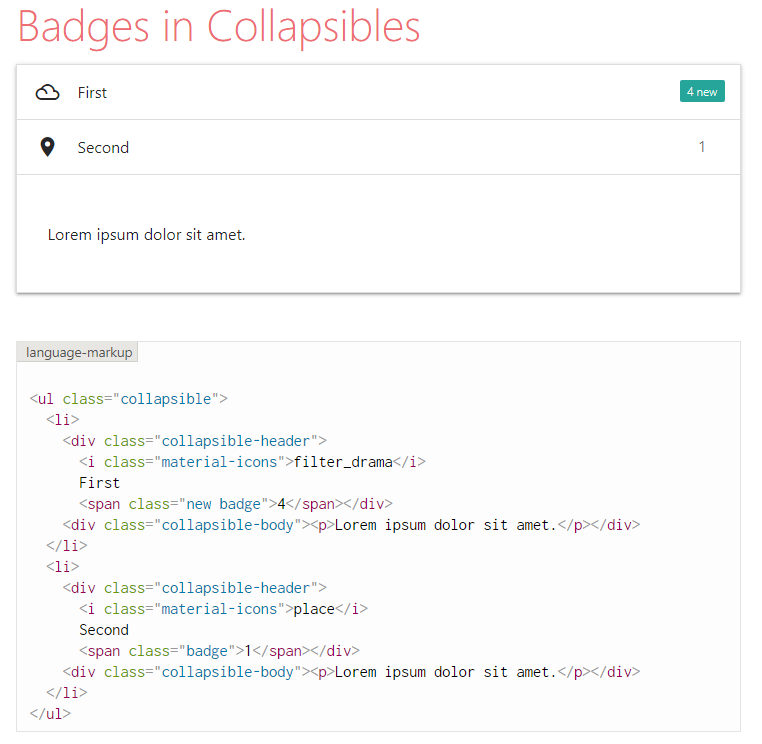
Considerada como uma linguagem de programação, o JavaScript executa no lado do usuário, ou seja, é processada pelo navegador, quando o usuário abre a página contendo a chamada do ​*script*​.

Com JavaScript se pode criar desde efeitos na página como, por exemplo, mostrar um ​*pop-up* ​em que se exibe uma mensagem ou conteúdo ou até mesmo coletar informações de itens de formulários e enviar para o ​*back-end* efetuar o processamento dessa informação e retornar com um determinado resultado (JAVASCRIPT, 2021).

### 1.7.7. Materialize CSS 1.0

O Materialize é um ​*framework,* ​cuja finalidade envolve facilitar o desenvolvimento do *front-end* da aplicação, possui vários componentes pré-compilados, responsáveis por implementar funcionalidades de forma fácil ao projeto, sendo a seguir exemplificado na figura que segue (MATERIALIZECSS, 2021).

A Figura 2 demonstra um exemplo de uso do Materialize.

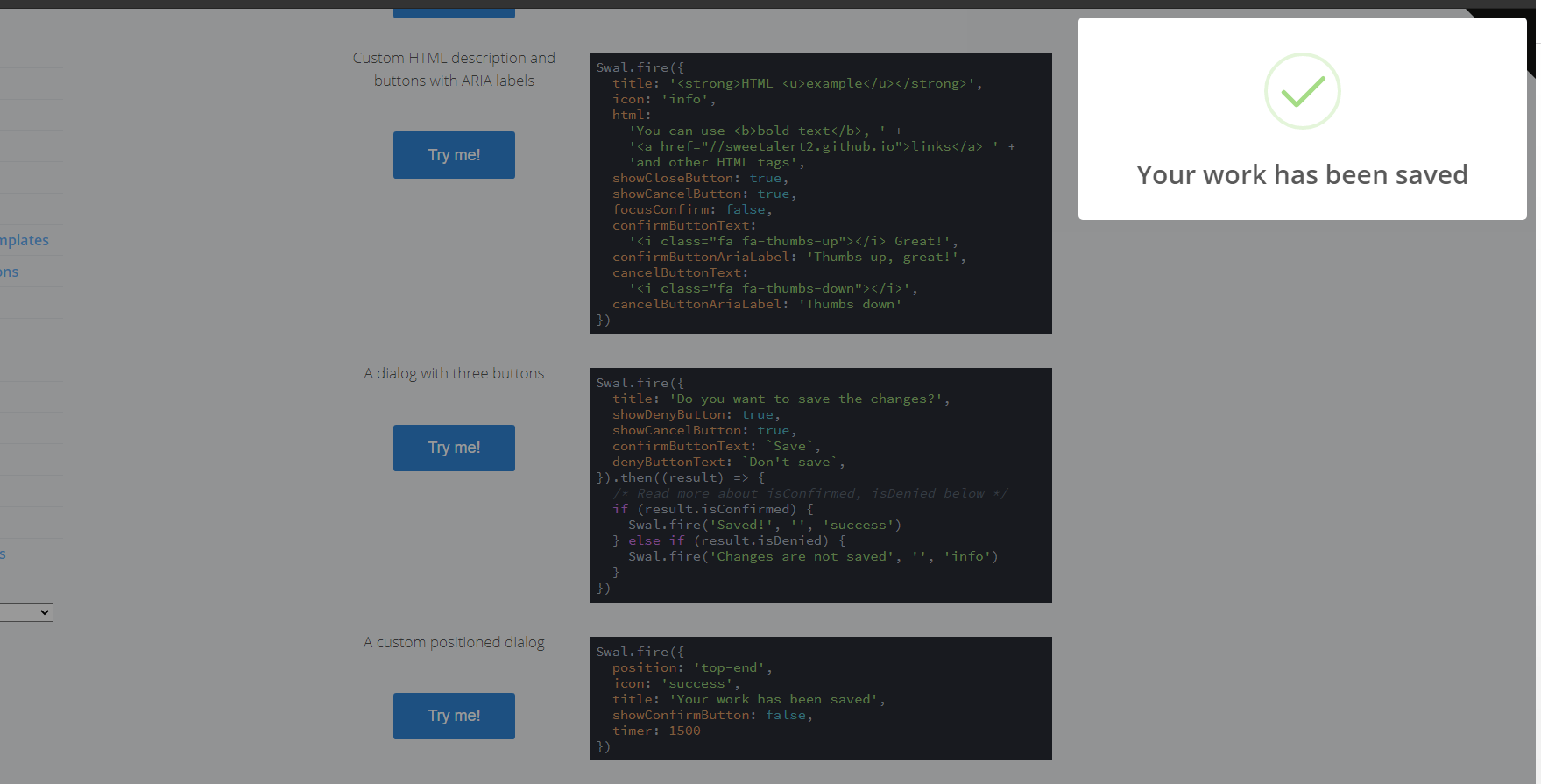


**Figura 2. Materialize. Fonte: MATERIALIZE.**

### 1.7.8. SweetAlert2 10.12.5

O SweetAlert2 é um ​*framework javascript* de código aberto*,* ​sua finalidade envolve otimizar a emissão de alertas ao usuário durante a utilização de aplicações web, a figura a seguir exemplifica a sua utilização. (SWEETALERT2, 2021).

A Figura 2 demonstra um exemplo de uso do SweetAlert2.



**Figura 2. Materialize. Fonte:** **SWEETALERT2.GITHUB.IO.**

## 1.8. Back-end

### 1.8.1. Node.js 14.15.1

Pendente.

### 1.8.2. AWS-SDK 2.792.0

Pendente.

### 1.8.3. Express 4.17.1

Pendente.

#### 1.8.3.1. DotEnv 8.2.0

Pendente.

#### 1.8.3.2. Cors 2.8.5

Pendente.

#### 1.8.3.3. Mime-Types 2.1.27

Pendente.

### 1.8.4. Sequelize 6.3.5

Pendente.

#### 1.8.4.1. MySQL 2.2.5

Pendente.

#### 1.8.4.2. Criação de tabelas com Sequelize

Pendente.

#### 1.8.4.3. Relacionamento de tabelas com Sequelize

Pendente.

#### 1.8.4.3. Construção de queries com Sequelize

Pendente.

### 1.8.5. Passport

Pendente.

#### 1.8.5.1 Json Web Token 8.5.1

Pendente.

#### 1.8.5.2 Passport JWT 4.0.0

Pendente.

### 1.8.6. NodeMailer 6.4.16

Pendente.

### 1.8.7. Multer 1.4.2

Pendente.

#### 1.8.7.1. MulterS3 2.9.0

Pendente.

## 1.9. Banco de dados

### 1.9.1 MySQL

Pendente.

# 2. Trabalhos Correlatos Existentes

Pendente.

# 3. Requisitos

Esta seção apresenta os requisitos funcionais e não funcionais utilizados para o desenvolvimento do projeto.

## 3.1. Requisitos Funcionais

Pendente.

## 3.2. Requisitos Não Funcionais

Pendente.

# 4. Diagramas UML

Pendente.

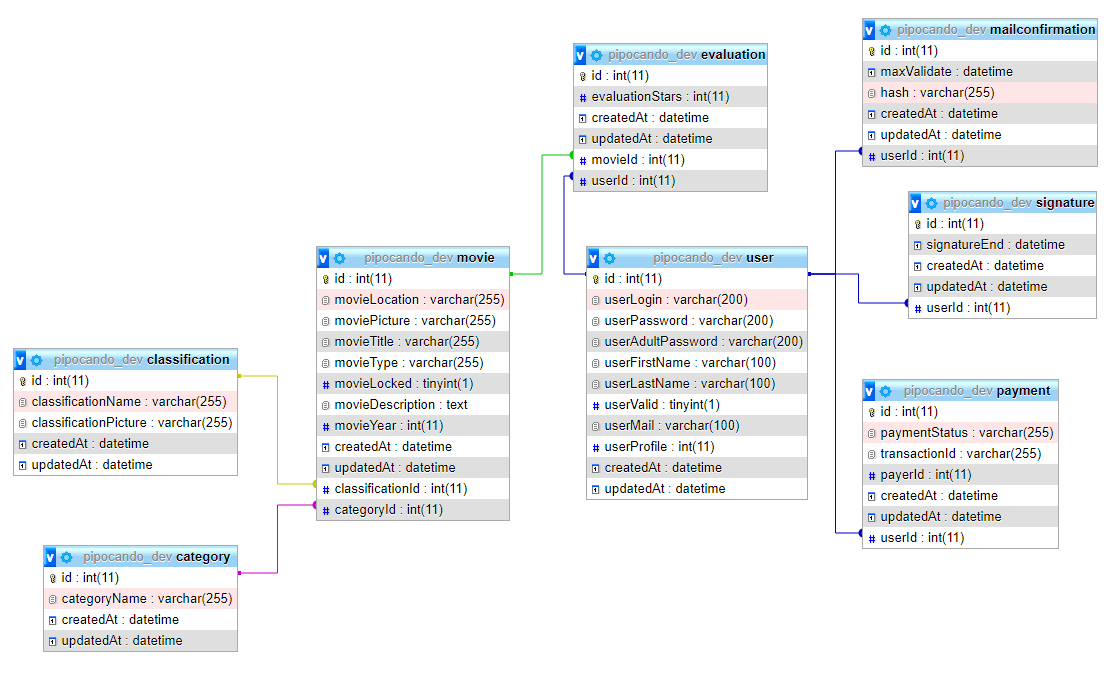
# 5. Modelagem de dados

Esta seção apresenta o modelo do banco de dados relacional e seu dicionário de dados.

## 5.1. Modelo Entidade Relacionamento

O Modelo Entidade Relacionamento (MER) ou somente Entidade Relacionamento (ER) é utilizado para representar o banco de dados. Neste modelo são definidas entidades, relacionamentos, atributos, chaves primárias e chaves estrangeiras responsáveis por manter a integridade das informações presentes no banco de dados (​SMARTDRAW​, 2021).

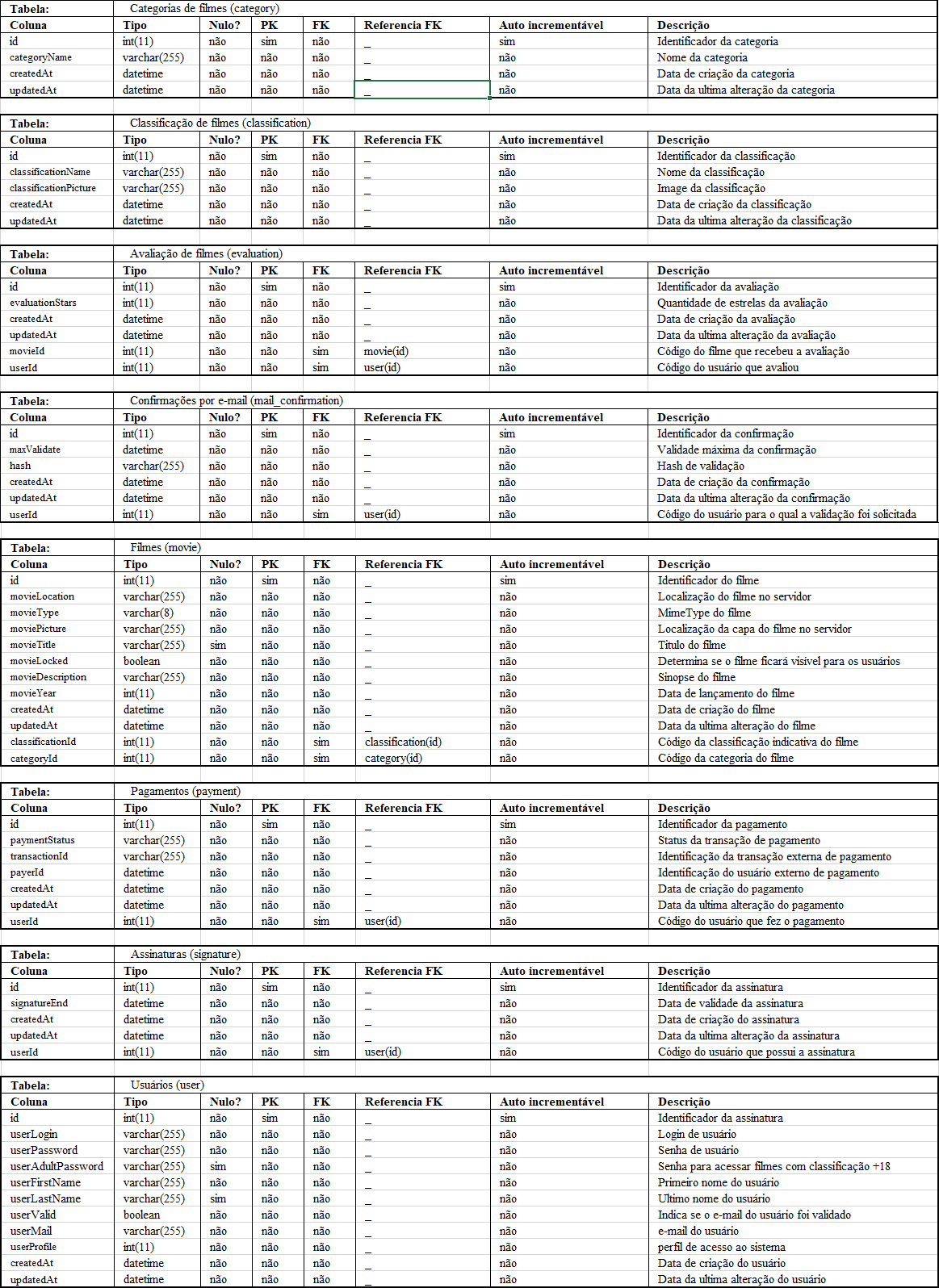
A Figura 21 demonstra o MER da aplicação, apresentando as tabelas e os relacionamentos do banco de dados. As informações das tabelas são: colunas, chaves e índices.



**Figura 21. MER. Fonte: Elaborada pelo autor.**

## 5.2. Dicionário de dados

Esta seção apresenta o dicionário de dados do banco de dados da aplicação, utilizado para descrever cada atributo, valor suportado, tamanho limite, se o campo oferece auto incremento, restrições de chave e nulidade.



**Figura 22. Dicionário de dados. Fonte: Elaborada pelo autor.**

# 6. Projeto da Interface

Pendente.

# 7. Resultados Obtidos

Pendente.

# 8. Considerações Finais e Trabalhos Futuros

Pendente.

# Referências Bibliográficas

Pendente.