

DADOS COLABORATIVOS COMO SOLUÇÃO PARA A CRIAÇÃO DE CADASTRO TÉCNICO MULTIFINALITÁRIO

Crowdsourced Data as a Solution to the Creation of a Multipurpose Technical Cadastrre

Elmo Neto

Universidade Federal de Santa Maria

Colégio Politécnico da UFSM

Avenida Roraima, 1000, 97105-900, Santa Maria, RS

elmo@inf.ufsm.br

Shirlei Wansing da Roza

Universidade Federal de Santa Maria

Colégio Politécnico da UFSM

Avenida Roraima, 1000, 97105-900, Santa Maria, RS

rozawansing@gmail.com

Marinêz da Silva

Universidade Federal de Santa Maria

Colégio Politécnico da UFSM

Avenida Roraima, 1000, 97105-900, Santa Maria, RS

marinez.silva@politecnico.ufsm.br

Angélica Cirolini

Universidade Federal de Pelotas

Centro de Engenharias

Praça Domingos Rodrigues, 2, 96010-440, Pelotas, RS

acirolini@gmail.com

Resumo:

O Cadastro Técnico Multifinalitário é usado pelo governo a nível municipal para fins de tributação e questões jurídicas, mas também pode servir como ferramenta importante para o desenvolvimento da sociedade e do espaço em que está inserida. Este artigo aborda a criação de um banco de dados espacial, a partir de dados colaborativos que seja capaz de responder questões em áreas extensas, como a de municípios, até questões sobre entidades pontuais como paradas de ônibus. A infraestrutura educacional do distrito Sede do município de São Gabriel (RS) é representada no banco de dados para posterior análise juntamente com rotas de ônibus, a fim de verificar se o transporte público municipal atende às necessidades de deslocamento dos alunos até as escolas.

Palavras-chave: Banco de Dados Espacial, OpenStreetMap, Cadastro Técnico Multifinalitário, Escolas, Rotas de Ônibus.

Abstract:

The multipurpose technical register is used by the municipal government for taxation purposes and juridical issues, but can also be used as an important tool to the development of the society and the space which is

inserted in. The work presented in this article discusses the creation of a spatial database from collaborative data that is capable of answering questions about large areas such as municipalities to questions about punctual entities such as bus stops. The educational infrastructure of the “Sede” district at the municipality of São Gabriel (RS) is represented in the database for later analysis along with bus routes in order to verify if the municipal public transportation meets the needs of students to move to the schools.

Keywords: Spatial Database, OpenStreetMap, Multipurpose Technical Cadastre, Schools, Bus Routes.

1. INTRODUÇÃO

O processo de Cadastro Técnico Multifinalitário (CTM) tem como base a aferição de dados acerca de parcelas de imóveis para posterior construção de uma base de dados que seja capaz de responder consultas importantes relacionadas a questões jurídicas, de tributação e de planejamento. Órgãos responsáveis por este trabalho comumente não possuem um planejamento de finanças dedicado à execução da tarefa de recolhimento de dados, ou ainda, não possuem profissionais capacitados para tal tarefa.

A alternativa que este artigo apresenta é o uso de dados coletados e cadastrados colaborativamente na base de dados geográficos OpenStreetMap, chamado de OSM doravante. Tal base é usada para a obtenção dos dados, sendo o armazenamento e consulta processos destinados aos Sistemas de Informação Geográfica (SIG) e Sistemas Gerenciadores de Banco de Dados (SGBD).

O uso de tecnologia e dados abertos é uma prática que se adotada por diversos setores do governo, principalmente prefeituras, pode reduzir drasticamente gastos com licenças de programas proprietários e problemas de interoperabilidade. No presente artigo, os experimentos e técnicas abordadas são realizados dentro um pequeno ambiente que faz parte do ecossistema de *software* livre.

Como a manutenção de tais dados é de suma importância para manter um cadastro atualizado e, por consequência mais eficiente, a base do OpenStreetMap é uma alternativa que deve ser considerada, pois é mantida por mapeadores voluntários dedicados à adição e correção de geometrias constantemente, seja por vetorização de imagens de satélite ou obtenção de posições com Sistemas Globais de Navegação por Satélites (GNSS) em campo. Além da geometria, também é possível obter dados atrelados às formas vetoriais por meio de um sistema de *tags* padronizado e amplamente difundido e discutido na comunidade.

A área de estudo para a qual o foco é direcionado neste artigo é o distrito Sede do município de São Gabriel, na região da Campanha do Estado do Rio Grande do Sul, mostrado na Figura 1.

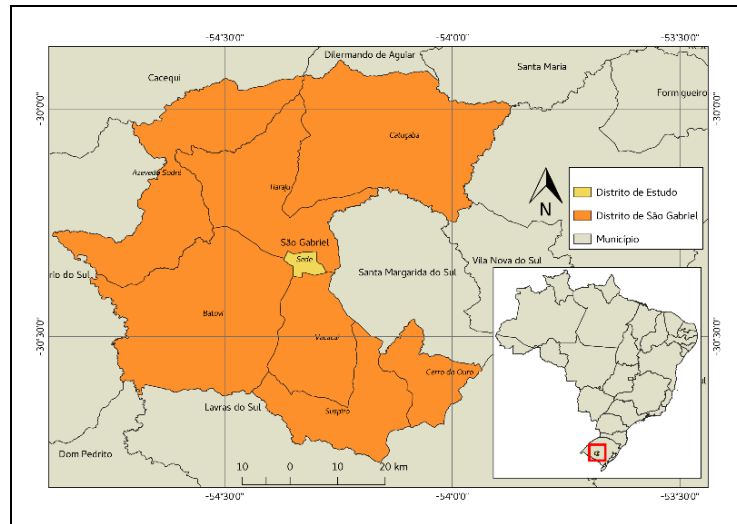


Figura 1 - Área de estudo do trabalho

2. REVISÃO BIBLIOGRÁFICA

Dentre as várias visões e definições de Sistemas de Informação Geográfica (SIG) presentes na literatura, é possível fazer a sobreposição delas buscando um panorama geral mais elucidativo sobre estes importantes sistemas na área de ciências geográficas.

Segundo Longley et. al (2011), os SIG são sistemas computacionais feitos para armazenar e processar informação geográfica, servindo como ferramentas que melhoram a eficiência e efetividade do tratamento da informação de aspectos e eventos geográficos.

Segundo Câmara e Queiroz, são sistemas que realizam o tratamento computacional de dados geográficos e recuperam informações não apenas com base em suas características alfanuméricas, mas também através de sua localização espacial. Ainda, segundo os autores, a principal diferença entre os SIGs é a forma como os dados geográficos são gerenciados, onde há três diferentes arquiteturas de SIGs que utilizam os recursos de um SGBD: dual, integrada baseada em SGBDs relacionais e integrada baseada em extensões espaciais sobre SGBDs objeto-relacionais (SGBDOR).

O SGBD objeto-relacional escolhido para a realização do estudo foi o PostgreSQL pelo fato de ser desenvolvido sobre as premissas de software livre, sendo, segundo Uchoa e Ferreira (2004), o primeiro SGBD de código aberto a trabalhar com um módulo específico para o tratamento dos dados geográficos vetoriais.

O mecanismo de extensibilidade do PostgreSQL permite incorporar capacidades adicionais ao sistema de forma a torná-lo mais flexível para o gerenciamento de dados para cada classe de aplicação. No caso dos SIG, isso significa a possibilidade do desenvolvimento de uma extensão geográfica capaz de armazenar, recuperar e analisar dados espaciais. (QUEIROZ; FERREIRA, 2005)

Muitos municípios de médio e pequeno porte têm dificuldades em promover a justiça tributária através de seus impostos municipais e é possível notar também a presença de deficiências em outras áreas do CTM, como no planejamento e manutenção do sistema viário, na administração da forma de uso e ocupação das parcelas territoriais e das secretarias e

setores responsáveis principalmente pela educação, saúde e segurança públicas etc. (GONÇALVES; LISBOA FILHO; RIBEIRO, 2008)

Para o estudo das rotas de ônibus, é necessário o armazenamento de uma camada subjacente fundamental, a das ruas. Segundo Loch e Erba (2007), o cadastro de ruas é formado por bases alfanuméricas e cartográficas, que descrevem o arruamento da cidade com nomes oficiais, tipo de calçamento e todos os detalhes úteis para o planejamento em geral e de tráfego e transporte.

O uso do OSM é uma alternativa que pode demandar mais trabalho dependendo da localidade onde o cadastro será considerado, em decorrência da discrepância na densidade de elementos cadastrados entre regiões do território brasileiro. Segundo Medeiros (2017), ferramentas como o OSM contribuem para um índice elevado de heterogeneidade em relação aos dados inseridos, uma vez que o trabalho de contribuição colaborativa na maioria das vezes não ocorre de maneira uniforme.

3. FERRAMENTAS E MÉTODOS

3.1. Ferramentas Utilizadas

O SIG utilizado para a manipulação das camadas vetoriais, com o objetivo de prepará-las para posterior armazenamento no banco de dados, é o Quantum GIS 3.0 (QGIS). Quanto à obtenção dos dados de geometria, duas ferramentas são utilizadas: o *site overpass turbo* (<https://overpass-turbo.eu>), onde é possível realizar consultas espaciais sobre a base de dados do OSM e o *plug-in QuickOSM* que permite executar consultas e fazer *download* de resultados do OSM através do *software* QGIS.

Demais informações sobre instituições de ensino foram obtidas a partir do acesso ao *site* do IBGE. O SGBD utilizado para o armazenamento e consulta de tabelas contendo dados geográficos foi o PostgreSQL 10, em conjunto com o pgAdmin 4 e PostGIS 2.4.

3.2. Modelagem do Banco de Dados

O banco de dados espacial construído para fins de estudo neste artigo é composto por relações que possuem um atributo destinado ao armazenamento de informações geográficas em WKB (*Well Known Binary*). Este atributo em todas as relações em que está presente foi nomeado com a *string* “geometria”, com o objetivo de padronizar consultas realizadas posteriormente.

Antes da criação, foi realizada a modelagem do banco de dados, mas a sua estrutura não permaneceu inalterada ao longo do desenvolvimento do projeto. Durante a obtenção e importação dos dados, novos requisitos foram identificados e a estrutura de tabelas do banco de dados foi ajustada para melhor atender às necessidades de armazenamento e consulta.

A ferramenta utilizada para modelagem do banco de dados foi o DBDesigner (<https://dbdesigner.net>), onde é possível criar e exportar gratuitamente um modelo lógico de banco de dados composto por até 30 tabelas.

Na Figura 2, é apresentada a estrutura final do banco de dados criado. Como, aproximadamente, 15 campos obtidos do Micro Censo Escolar 2017 do INEP foram inseridos na tabela “escolas”, com informações sobre número de funcionários, acesso à internet banda

larga, dentre outros, no modelo ela foi reduzida para não tornar o diagrama muito extenso. O *dbdesigner* não permitiu a configuração do tipo de atributo *geometry*, que é fornecido pela extensão PostGIS dentro do PostgreSQL. Então, o tipo dos atributos que encapsulam geometrias foi mantido com o tipo *string*, apenas para fins de representação.

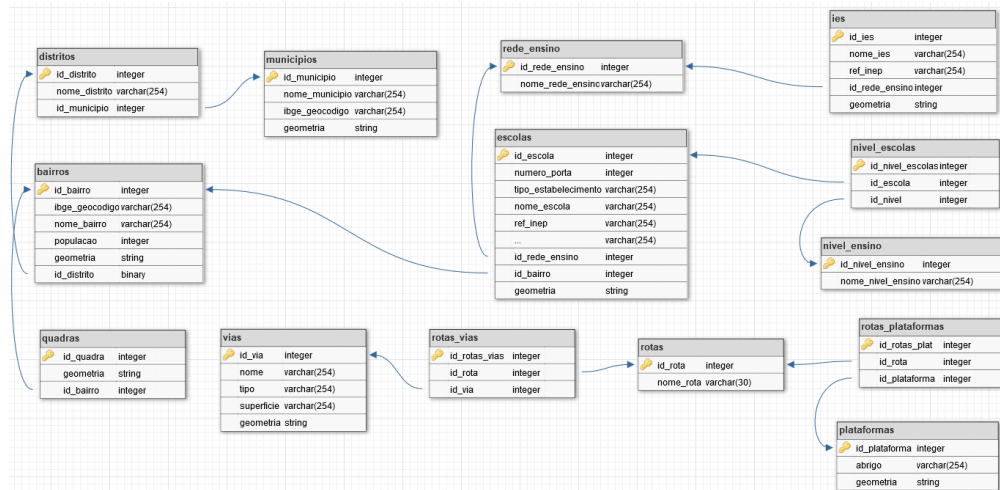


Figura 2 - Modelo lógico do banco de dados

3.3. Obtenção de Dados

As tabelas que envolvem atributos de geometria foram obtidas por dois métodos: consulta e *download* de dados por meio do *site overpass turbo* e do *plug-in QuickOSM*. Demais tabelas foram obtidas de censos do IBGE e do INEP e construídas a partir de operações de geoprocessamento no *software QGIS*. O processo de obtenção de dados para cada tabela será detalhado a seguir.

3.3.1. Municípios e distritos

Os dados para popular a tabela “municípios” foram obtidos realizando a consulta no *overpass*, conforme os procedimentos que seguem: a) como os polígonos desejados são os que representam cidades, o atributo “admin_level” deve ser igual a 8, já que este valor associado a tal *tag* é destinado a municípios no OSM. b) Para filtrar os resultados de todos os municípios retornados, mostrando apenas os municípios gaúchos, mais uma condição foi estabelecida: o código do IBGE atrelado deve começar com o valor 43, não importando o valor subsequente. Os municípios do Rio Grande do Sul carregam em seu código o número 43, que é o código identificador unívoco atribuído ao Estado pelo IBGE. Dados sobre distritos foram obtidos com consulta semelhante, alterando o valor da *tag* “admin_level” para “9”.

Código para execução da consulta:

```
[out:json][timeout:25];
(relation["admin_level"~"8"]["IBGE:GEOCODIGO"~"^43"]({{bbox}}));
out meta;
>;
out meta qt;
```

3.3.2. Vias

Para o estudo aqui apresentado, somente as vias da zona urbana do distrito Sede de São Gabriel, foram consideradas. A classificação de vias no OSM segue alguns preceitos básicos que não fazem parte do escopo do estudo, mas algumas considerações devem ser feitas sobre os valores assumidos pela tag “highway”, que tipifica o nível hierárquico de vias.

- *primary* - primordialmente rodovias estaduais e federais;
- *secondary* - avenidas principais de grande tráfego na cidade;
- *tertiary* - vias coletoras de tráfego;
- *residential* - vias de tráfego de local, principalmente de moradores da região;
- *unclassified* - sem classificação.

Além destes valores, existem também os valores *_link que fazem ligações entre vias de alta hierarquia. Na consulta (Figura 3) para a obtenção de linhas referentes às vias do distrito Sede, a única condição é de que a tag “highway” assumira algum dos valores citados acima. A *bounding box* (área de consulta) foi centralizada no distrito Sede para recuperar todos os resultados dentro desta área, eventualmente até ultrapassando-a. Para obter somente as vias dentro da área desejada, uma nova consulta foi executada para obtenção do polígono que representa o distrito Sede de São Gabriel e posteriormente as duas camadas foram importadas para o QGIS, a fim de obter uma nova camada resultado do recorte.

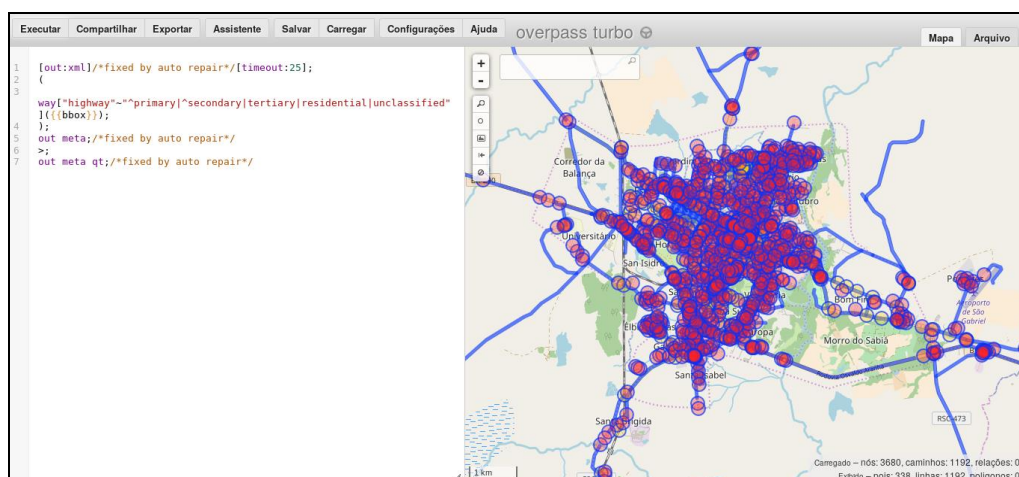


Figura 3 - Consulta para obtenção de vias da zona urbana de São Gabriel

3.3.3. Bairros

Os polígonos de bairros foram obtidos através do *plug-in QuickOSM*, onde os critérios de pesquisa foram configurados de acordo com os seguintes critérios:

- key = admin_level; value = 10*, pois o valor numérico é o nível mais alto/específico de limite administrativo representável no OSM, o bairro.
- In: São Gabriel, RS*, pois somente os bairros de São Gabriel são os objetos de estudo neste trabalho.

3.3.4. Quadras

Como a geometria de quadras ainda não está presente no município de São Gabriel no OSM, ela foi desenhada manualmente no QGIS com auxílio do *plug-in OpenLayers*. Para o estudo, apenas as quadras do bairro Centro foram vetorizadas.

3.3.5. Rede de ensino e nível de ensino

A tabela *rede_ensino* foi criada com valores identificadores condizentes com as tabelas do Micro Censo Escolar obtido no *site* do INEP. Na tabela foram inseridas quatro entradas: “Municipal”, “Estadual”, “Federal” e “Privada”. Na tabela *nível_ensino*, foram criados identificadores diferentes dos presentes na tabela do Micro Censo Escolar, pois para o propósito do estudo serão necessários apenas oito níveis, não todos os disponíveis no censo. A tabela então foi criada com os níveis: “Creche”, “Pré-escola”, “Anos Iniciais Fundamental”, “Anos Finais Fundamental”, “Ensino Médio”, “EJA”, “Ensino Profissionalizante” e “Superior”.

3.3.6. Escolas

A relação “escolas” engloba três categorias: creches, escolas de ensino básico e de nível pós-médio. A consulta para obter as geometrias foi executada no *overpass turbo*. A camada de instituições foi recortada no QGIS pelo polígono do distrito Sede de São Gabriel.

Demais informações foram obtidas do Micro Censo 2017 do INEP. Apenas alguns dos atributos foram usados no banco criado. A junção de informações de escolas com a camada vetorial obtida pelo *overpass* foi feita no QGIS, por meio do campo identificador atribuído à escola pelo INEP, pois este atributo está associado a cada feição da camada vetorial.

```
[out:json][timeout:25]
(way["amenity"~"kindergarten | school | college"]({{bbox}});
relation["amenity"~"kindergarten | school | college"]({{bbox}}));
out meta;
>;
out meta qt;
```

3.3.7. Universidades

A tabela “universidades” foi populada com dados obtidos a partir de consulta semelhante à consulta para obtenção de dados para “escolas”. A única alteração foi o campo “amenity” que recebeu o valor “university”.

3.3.8. Rotas

O conceito de relação no OSM é utilizado para representar entidades formadas por diferentes elementos materializados ou linhas imaginárias, de diversos tipos geométricos e atributos. Uma relação que representa uma rota de ônibus tem o campo “type” com o atributo “route” associado. A consulta (Figura 4) sobre rotas retorna um conjunto de linhas (ruas) que fazem parte da rota e outro conjunto de pontos (e/ou polígonos), representando cada plataforma de embarque ou desembarque.

A fim de obter o nome de todas as rotas que cruzam o distrito Sede, a consulta no *overpass* foi executada com a *bounding box* envolvendo todo o distrito. A seguinte consulta foi realizada e os resultados retornados foram adicionados à tabela “rotas” em conformidade com a tupla (“id_rote”, “nome_rote”), onde o id_rote é incrementado automaticamente.



```

Executar  Compartilhar  Exportar  Assistente  Salvar  Carregar  Configurações  Ajuda  overpass turbo
1  [out:csv(name)][timeout:25];
2  (
3  relation["route"~"bus"]
4  ["network"~"BR:RS:SGA:coletivo"]{{{bbox}}});
5  );
6  out meta;
7  >;
8  out meta qt;
9
10
1  name
2  Bom Fim - Cidade Nova
3  Independência - Vila Maria
4  Gabrielense - Menino Jesus
5  Unipampa - Centro
6
7
8
9
10
  
```

Figura 4 - Consulta para obtenção de rotas municipais em São Gabriel

3.3.9. Paradas

A obtenção de dados de paradas foi realizada com consulta no *overpass*, buscando os nós e polígonos que possuem o valor “platform” atribuído ao campo “public_transport”.

Consulta para obtenção de dados sobre plataformas:

```

[out:json][timeout:25];
(way["public_transport"~"platform"]({{bbox}});
node["public_transport"~"platform"]({{bbox}}));
out meta;
>;
out meta qt;
  
```

Paradas de ônibus no OSM são representadas com a *tag* “public_transport” que possuem o valor “platform”. Polígonos representam paradas que tem uma estrutura física dedicada à espera dos passageiros e pontos representam paradas onde há o sinal indicativo de parada, mas não a estrutura física. Na Figura 5, há exemplos de paradas e valores de *tags*

associados. Para armazenamento no banco de dados, todas as paradas de ônibus representadas por polígonos foram convertidas em seus centróides e mescladas com a camada de paradas representadas por pontos. Esta operação foi executada no QGIS.



geometria: polígono
public_transport=platform



geometria: ponto
public_transport=platform

Figura 5 - Comparação entre tipos de plataformas

Fonte das imagens: <http://www.onibusdaparaiba.com/2016/04/dos-1960-pontos-de-onibus-de-joao.html>;
<https://oglobo.globo.com/rio/bairros/novos-abrigos-de-onibus-de-niteroi-sao-reprovados-por-moradores-13054988>

3.4. Manipulação e Armazenamento de Dados

3.4.1. Tabelas com atributos geométricos

As consultas para “municípios”, “bairros”, “vias”, “escolas”, “ies” e “paradas” baixadas do *overpass* com extensão *.geojson* foram manipuladas no QGIS e salvas como arquivos no formato *shapefile* (.shp), para posterior importação no banco de dados por meio do utilitário de linha de comando *shp2pgsql*. Os arquivos tiveram suas tabelas de atributos modificadas a fim de armazenar somente os atributos de interesse para o estudo. O arquivo *.shp* resultado da manipulação foi transportado para a tabela correspondente com o comando abaixo, no ambiente Linux:

Comando para importar arquivo *shapefile* para tabela do banco de dados:

```
shp2pgsql -d -g geometria -s 4326 -W "UTF-8" /<caminho>/<nome_arquivo>.shp  
tabela_destino | psql --host=localhost --port=5432 --dbname=nome_banco --  
username=postgres --password
```

A “tabela_destino” no banco de dados é sobrescrita devido à opção “-d” do comando, onde um atributo identificador “gid” é criado automaticamente para cada feição e todos os atributos associados são copiados, cada um para uma coluna diferente, com o mesmo nome do atributo. A coluna “geometria” é criada e populada usando o sistema de referência WGS84, que é especificado pela opção “-s 4326”, pois 4326 é o código do sistema. A codificação é configurada pela opção “-W UTF-8”. Com auxílio da conexão a banco de dados fornecida pelo QGIS, as tabelas com atributos geométricos foram importadas como camadas vetoriais

para o ambiente do programa. Desta forma, é possível obter um panorama geral do trabalho, como mostrado na Figura 6.

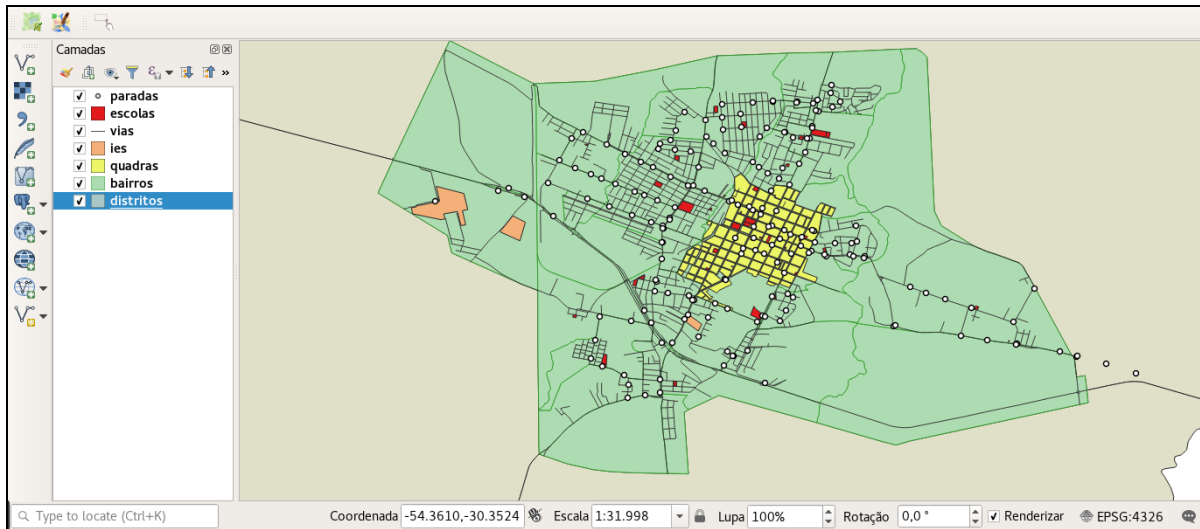


Figura 6 - Tabelas com geometria importadas como camadas para o QGIS

3.4.2. Tabela nivel_escolas

A tabela nivel_escolas foi construída no QGIS a partir das geometrias de escolas que já estavam presentes no banco. Para isso realizou-se a junção entre os dados vetoriais e o arquivo *csv* do micro censo escolar, feita pelo código identificador do INEP, presente nas duas camadas. O arquivo *csv* foi filtrado e somente os campos de identificação única e tipos de matrículas ofertadas foram mantidos. Após a junção, foi feita a análise da tabela de atributos e, para cada escola, foi criada uma entrada no arquivo *csv*, com o seu campo identificador no banco de dados, seguido pelo campo identificador do nível de ensino ofertado. O arquivo *csv* contendo todas as entradas após a análise da tabela de atributos foi importado para a tabela nivel_ensino.

3.4.3. Tabelas rotas_vias e rotas_plataformas

As tabelas que relacionam as tabelas de paradas e vias com a tabela de rotas capturam relacionamentos *n* para *n*, ou seja, são transformadas em uma tabela onde existem duas chaves estrangeiras referenciando duas tabelas diferentes e também existe uma chave primária da própria tabela para identificar o relacionamento. Na tabela “rotas_vias”, deve existir um campo que referencie a chave primária da tabela rota e uma que referencie a chave primária da tabela “vias”. Analogamente, na tabela “rotas_plataformas” também há uma chave estrangeira que referencia a chave primária de “rotas” e outra que referencia a chave primária de “plataformas”.

A relação correspondente a cada rota cadastrada no item 3.3.7 foi obtida através do *overpass* e cada uma delas foi manipulada separadamente. Conforme a Figura 7, com o auxílio da conexão de banco de dados disponível no QGIS, as feições já inseridas no banco de

dados serviram de base para a junção de informações com a camada vetorial referente à rota.

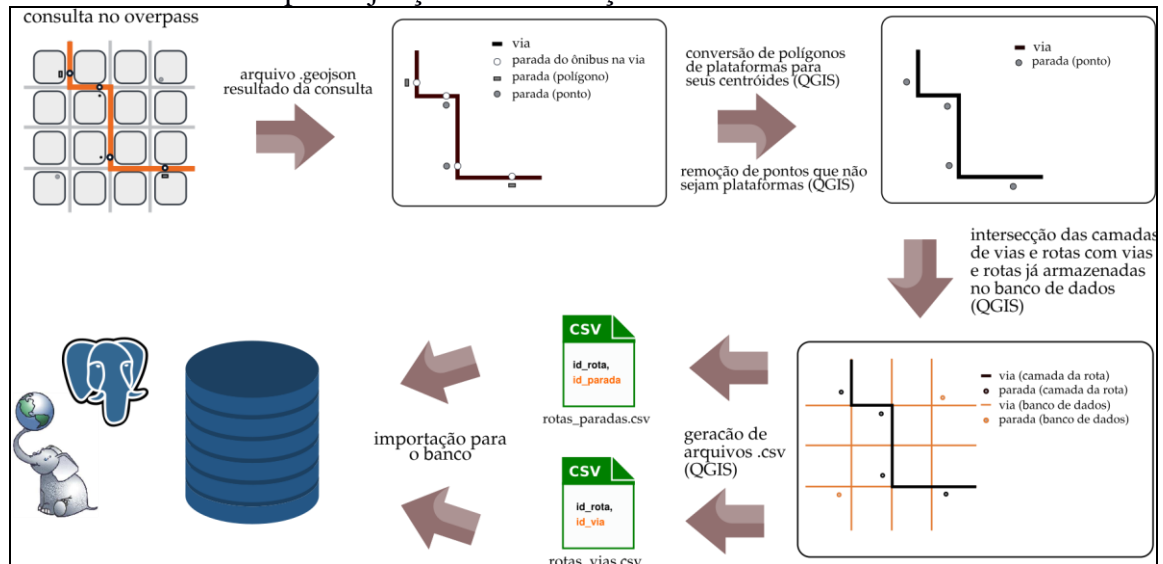


Figura 7 - Sequência desde obtenção de dados até armazenamento no banco de dados

4. CONSULTAS E OBTENÇÃO DE INFORMAÇÃO

4.1. Informações sobre o município e sua vizinhança

a. Área total do município: A função *ST_Transform* foi usada para projetar a geometria de cada um dos registros da tabela para o sistema de referência adequado em metros. O distrito sede de São Gabriel está localizado no fuso 21, hemisfério sul. O sistema de referência usado é SIRGAS 2000 UTM zone 21S e código correspondente é 31981. A função *ST_Area* retorna a área de polígonos em metros quadrados. O valor de retorno foi dividido por 10000 para obtenção do valor da área em hectares (Figura 8).

```
SELECT nome_municipio, ROUND(ST_Area(ST_Transform(geometria,31981))/10000)
as hectares
FROM municipios
```

	nome_municipio	hectares
1	São Gabriel	502311

Figura 8 - Resultado da Consulta 4.1.a.

b. Municípios vizinhos: A função *ST_Touches* retorna um valor booleano que serve como verificação para a vizinhança de geometrias. É usada nesta consulta para buscar os municípios vizinhos do município de São Gabriel (Figura 9).

```
SELECT m2.nome_municipio as "Municípios Vizinhos"
FROM municipios m1, municipios m2
WHERE ST_Touches(m1.geometria,m2.geometria)
AND m1.nome_municipio = 'São Gabriel'
AND m2.nome_municipio <> 'São Gabriel'
```

	Municípios Vizinhos
	character varying (254)
1	Rosário do Sul
2	Dom Pedrito
3	Dilermando de Aguiar
4	Santa Maria
5	Santa Margarida do Sul
6	São Sepé
7	Vila Nova do Sul
8	Lavras do Sul
9	Cacequi

Figura 9 - Resultado da Consulta 4.1.b.

4.2. Informações sobre bairros

a. População de bairros: Uma única tabela é usada para esta consulta, onde os campos nome e população são retornados, ordenando do bairro mais populoso para o menos populoso. Bairros foram omitidos do resultado da consulta 4.2.a. para não tornar a imagem extensa (Figura 10).

```
SELECT nome_bairro, populacao
FROM bairros
ORDER BY populacao DESC
```

	nome_bairro	populacao
	character varying (254)	integer
1	Centro	8615
2	Independência	6010
3	Cidade Nova	3004
4	Baltar	2719
5	Mariana	2650
6	Santo Antônio	2354
7	Menino Jesus	2171
8	Gabrielense	2116

Figura 10 - Resultado da consulta 4.2.a.

b. Rotas passantes por bairros: Para auxílio nas consultas e para torná-las menos complexas, *views* auxiliares foram criadas. A *view* “bairros_paradas” une o campo “nome_bairro”, que contém o nome do bairro, e o campo “id_parada”, que contém o atributo identificador da parada. A *view* rotas_passam_parada une o campo identificador da parada “id_parada” com o campo que contém o nome da ou das paradas associadas.

```
CREATE VIEW bairros_paradas AS
(
SELECT bairros.nome_bairro AS nome_bairro, paradas.id_parada AS id_parada
FROM bairros, paradas
WHERE ST_Contains(bairros geometria, paradas geometria)
GROUP BY nome_bairro, id_parada
ORDER BY paradas.id_parada
)

CREATE VIEW rotas_passam_parada AS
(
SELECT paradas.id_parada AS id_parada, rotas.nome AS nome_rota
FROM paradas NATURAL JOIN rotas_paradas NATURAL JOIN rotas
ORDER BY paradas.id_parada
)
```

O bairro utilizado como exemplo para consulta foi “Baltar”. Código da consulta depois da criação das *views* e o seu resultado (Figura 11).

```
SELECT DISTINCT(rotas_passam_parada.nome_rota)
FROM bairros_paradas NATURAL JOIN rotas_passam_parada
WHERE nome_bairro = 'Baltar'
```

Data Output	Explain	Messages
	nome_rota	
	character varying (30)	
1	Bom Fim - Cidade Nova	
2	Unipampa - Centro	

Figura 11 - Resultado da consulta 4.2.b.

4.3. Informações sobre quadras do bairro Centro

a. Área total de quadras: A função de agregação *SUM* foi usada para obter o total da área de todas as quadras cadastradas (Figura 12). Outras informações a respeito da quadras podem ser executadas como por exemplo: ao retirarmos a função *SUM* obtêm-se a área de cada quadra do bairro em estudo. Substituindo-se a função *ST_AREA* por *ST_Perimeter* teremos o retorno

do perímetro de geometrias de polígono (ou multipolígono).

```
SELECT SUM(ST_Area(ST_Transform(geometria,31981)))/10000 as hectares
FROM quadras
```

Data Output		Explain	Messages
	hectares		
	double precision		
1	150.743497582056		

Figura 12 - Resultado da Consulta 4.3.a.

4.4. Informações sobre escolas

a. Número de escolas que oferecem determinado nível de ensino: Esta consulta executa uma junção de tabelas escolas, nivel_escolas e nivel_ensino para retornar o nível de ensino e o número de instituições que o ofertam. A ordenação é feita do nível de ensino mais ofertado para o nível de ensino menos ofertado (Figura 13).

```
SELECT n_ens.nome_nivel_ensino as "Nível", COUNT(id_escola) as "Número de
instituições que ofertam"
FROM escolas as e, nivel_escolas as n_esc, nivel_ensino as n_ens
WHERE e.id_escola = n_esc.id_escolas
AND n_esc.id_nivel_ensino = n_ens.id_nivel_ensino
GROUP BY "Nível"
ORDER BY "Número de instituições que ofertam" DESC
```

Data Output		Explain	Messages	Query History
	Nível			
	character varying (30)			
	Número de instituições que ofertam			
	bigint			
1	Pré-Escola			26
2	Anos Iniciais Fundamental			21
3	Anos Finais Fundamental			19
4	Creche			11
5	Médio			5
6	EJA			3
7	Profissionalizante			3

Figura 13 - Resultado da consulta 4.4.a.

b. Densidade de funcionários por escola, agrupamento por bairro: O nome do bairro, o número de escolas no bairro, a quantidade total de funcionários trabalhando em escolas do bairro e a densidade de funcionários por escolas de cada bairro. O resultado da consulta,

apenas com as primeiras linhas retornadas, é mostrado na Figura 14.

```
SELECT bairros.nome_bairro as "Nome do Bairro", COUNT(id_escola) as "Quantidade de
escolas",
SUM(escolas.qtde_funcionarios) as "Quantidade de funcionários em escolas",
SUM(escolas.qtde_funcionarios)/COUNT(id_escola) as "Densidade de
funcionários por escola"
FROM bairros, escolas
WHERE ST_Contains(bairros.geometria,escolas.geometria) AND qtde_funcionarios IS
NOT NULL
GROUP BY "Nome do Bairro"
ORDER BY "Densidade de funcionários por escola" DESC
```

	Nome do Bairro character varying (254)	Quantidade de escolas bigint	Quantidade de funcionários em escolas bigint	Densidade de funcionários por escola bigint
1	Cidade Nova	1	69	69
2	Três de Outubro	2	101	50
3	Mariana	1	43	43
4	Vargas	2	87	43
5	Baltar	1	42	42
6	Centro	15	632	42
7	Bela Vista	1	39	39
8	Santo Antônio	2	65	32
9	Vila Maria	1	30	30
10	Élbio Vargas	2	59	29

Figura 14 - Resultado da consulta 4.4.b.

c. Escolas que não ficam a menos de 100 m de distância em linha reta de nenhuma parada de ônibus cadastrada: A função *ST_Buffer* retorna uma geometria que envolve a geometria de entrada com um *buffer* de raio especificado. É usada em conjunto com a função *ST_Intersects* para verificar se o *buffer* retornado entra em contato com a geometria de alguma escola e o seu resultado pode ser verificado na Figura 15. A representação geográfica da consulta é mostrada na Figura 16.

```
SELECT escolas.nome_escola
FROM escolas
WHERE escolas.nome_escola NOT IN(
SELECT DISTINCT(e.nome_escola)
FROM escolas as e, paradas as p
WHERE ST_Intersects (ST_Transform(e.geometria,31981),
ST_Buffer(ST_Transform(p.geometria,31981),100) )
)
```

Data Output		Explain	Messages	Query History
	nome_escola character varying (254)			7 EMEF Senador Tarso Dutra
1	EEEEF Doutor Pery da Cunha ...			8 EMEI Vó Edy
2	EEEEF Brandina Mello			9 EEEF Profª Sueni Goulart Sa...
3	EMEI Menino Jesus Caic			10 Colégio Nossa Senhora do P...
4	EMEF Doutor João Manoel S...			11 EEEB Doutor Celestino Lope...
5	EMEI Suely B. de Abreu			12 EMEF Carlota Vieira da Cunha
6	EEEM Doutor José Sampaio ...			13 EMEF José Ferreira da Fons...

Figura 15 - Resultado da consulta 4.4.c.

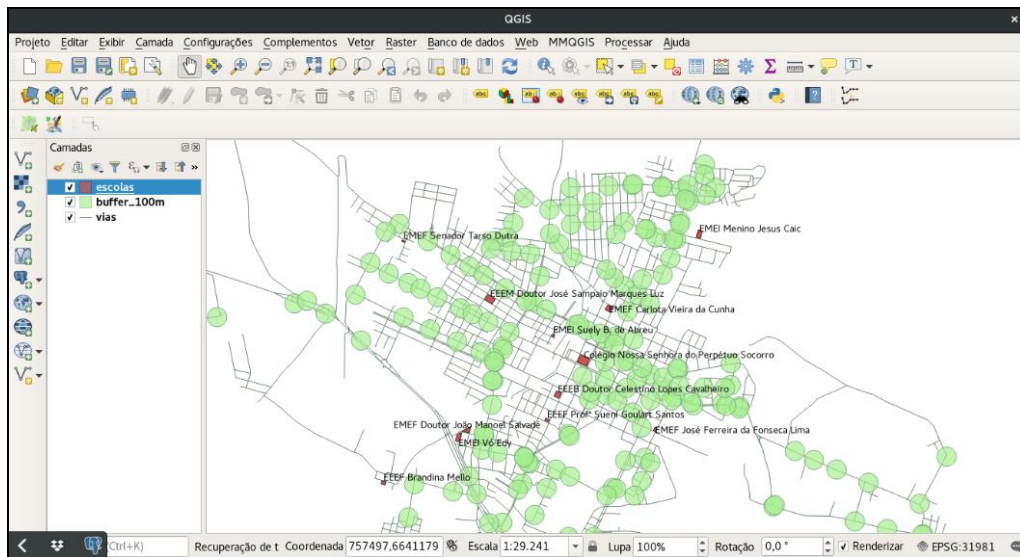


Figura 16 - Representação geográfica da consulta 4.4.c.

4.5. Informações sobre vias

a. Comprimento total de vias: a função *ST_Length* retorna o comprimento de geometrias de linha. As geometrias foram reprojctadas para o *datum* adequado retornando assim o resultado na Figura 17.

```
SELECT SUM(ST_Length(ST_Transform(geometria,31981)))/1000 as km
FROM vias
```

Data Output		Explain	Messages
	km double precision		
1			229

Figura 17 - Resultado da consulta 4.5.a.

b. Comprimento total de cada via: As vias foram agrupadas pelo campo nome, pois existem vários registros (segmentos da via) que compartilham o mesmo nome (Figura 18). A geometria de cada registro foi projetada para o *datum* correto para obtenção de seu comprimento. As vias foram ordenadas em função da via com maior comprimento para o menor comprimento.

```
SELECT nome_via, SUM(ST_Length(ST_Transform(geometria,31981)))/1000 as km
FROM vias
WHERE nome_via IS NOT NULL
GROUP BY nome_via
ORDER BY km DESC
```

	nome_via	km
	character varying (254)	double precision
1	Rodovia Osvaldo Aranha	12.0470782272845
2	Avenida Francisco Chagas	5.33722157204312
3	Avenida Francisco Hermene...	4.8709032969034
4	Avenida Antônio Trilha	4.07686172764221
5	Avenida das Acácias	2.76070243712154
6	Rua Francisco da Silva	2.66081792830097
7	Entrada Lateral Secundária ...	2.62967383892362
8	Corredor da Casa Branca	2.5090671915793
9	Rua Presidente Juscelino Ku...	2.46906831451469
10	Rua Maurício Cardoso	2.37865309205544
11	Avenida Antônio Mercado	2.22578914101852

Figura 18 - Resultado da consulta 4.5.b.

4.6. Informações sobre rotas

a. Representação visual das rotas: Uma *view* para cada rota é criada selecionando o campo “id_rotas_vias” e o campo “geometria” das tabelas “rotas_vias” e “vias”. As *views* são criadas pela janela de SQL do gerenciador de banco de dados do QGIS e posteriormente importadas como camadas vetoriais. O resultado da importação é mostrado na Figura 19.

```
SELECT rotas_vias.id_rotas_vias, vias.geometria
FROM rotas_vias NATURAL JOIN vias
WHERE rotas_vias.id_rota = 1
```

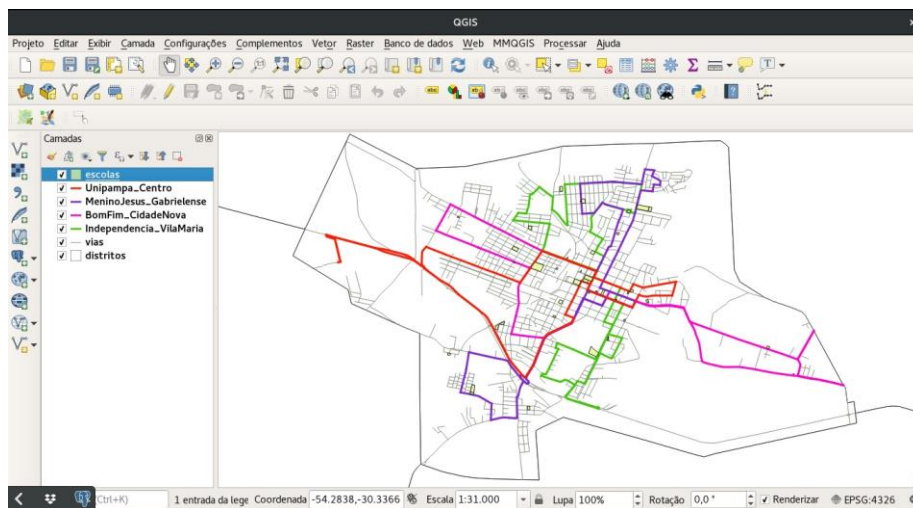


Figura 19 - Representação visual das rotas de ônibus cadastradas

b. Comprimento de cada rota: Junção entre tabelas rotas, rotas_vias e vias para retornar o nome da rota e a extensão total do trajeto. Os dados geométricos de vias foram reprojatados para o *datum* apropriado e os valores retornados (Figura 20) pela função *ST_Length* foram agrupados pelo campo com o nome da rota.

```
SELECT rotas.nome_rota as "Nome da Rota",
SUM(ST_Length(ST_Transform(vias.geometria,31981)))/1000 as "Extensão da rota em
km"
FROM rotas, rotas_vias, vias
WHERE rotas.id_rota = rotas_vias.id_rota
AND rotas_vias.id_via = vias.id_via
GROUP BY "Nome da Rota"
```

	Nome da Rota character varying (30)	Extensão da rota em km double precision
1	Independência - Vila Maria	17.611368801752
2	Menino Jesus - Gabrielense	17.3518311001063
3	Bom Fim - Cidade Nova	19.2512208269004
4	Unipampa - Centro	17.1814819896605

Figura 20 - Resultado da consulta 4.6.b.

c. Superfície de cada rota: Junção entre tabelas rotas, rotas_vias e vias para retornar o nome da rota e a extensão de cada tipo de superfície. O valor “asphalt” representa via “asfalto”, “cobblestone:flattened” representa via com pavimentação com pedras irregulares, “dirt” representa via sem pavimentação e “sett” via pavimentada com pedras de formato regular. A rota utilizada como exemplo é “Bom Fim – Cidade Nova” e o resultado pode ser observado

na Figura 21.

```
SELECT rotas.nome_rota as "Nome da Rota", vias.superficie as "Superfície",
SUM(ST_Length(ST_Transform(vias.geometria,31981)))/1000 as "Extensão em km"
FROM rotas, rotas_vias, vias
WHERE rotas.id_rota = rotas_vias.id_rota
AND rotas_vias.id_via = vias.id_via
AND rotas.nome_rota = 'Bom Fim - Cidade Nova'
GROUP BY "Nome da Rota", "Superfície"
ORDER BY "Superfície"
```

	Nome da Rota character varying (30)	Superfície character varying (254)	Extensão em km double precision
1	Bom Fim - Cidade Nova	asphalt	11.9040854433171
2	Bom Fim - Cidade Nova	cobblestone:flattened	4.96757607013355
3	Bom Fim - Cidade Nova	dirt	2.25239755544037
4	Bom Fim - Cidade Nova	sett	0.127161758009371

Figura 21 - Resultado da consulta 4.6.c.

5. CONCLUSÕES E TRABALHOS FUTUROS

A gestão responsável de recursos e a identificação de áreas carentes é essencial para a melhoria do sistema educacional como um todo. O cadastro criado neste trabalho é de grande valia para a orientação de decisões, a fim de suprir necessidades de áreas que mais necessitam de investimentos. As consultas de caráter geográfico são um norte para adaptações necessárias nos planos das secretarias de educação e desenvolvimento urbano.

A iniciativa pode partir não só do setor público, mas do setor privado, com o objetivo de melhorar a infraestrutura de apoio às escolas. As rotas de ônibus poderiam ser adaptadas para atingir o maior número de escolas possível e para percorrer a superfície menos desconfortável ao passageiro, no caso o aluno. Além disso, a otimização de percurso das superfícies acarreta em menor custo de manutenção na frota de ônibus. Estabelecimentos como bibliotecas e papelarias poderiam utilizar o cadastro para localizar o ponto de menor distância entre determinado número de escolas.

O cadastro multifinalitário como suporte para decisões responsáveis de governantes pode ser construído com dados colaborativos, que são de grande importância, pois são atualizados frequentemente. A tendência é de que, quanto mais os dados sejam usados, mais os colaboradores trabalhem para tornar essa informação a mais precisa possível, o que cria um sistema que beneficia a sociedade como um todo.

O desafio futuro é de discutir formas de automatizar eficientemente e de forma segura o banco de dados que é alimentado a partir de dados colaborativos. Assim, como a exploração de possibilidades em questão de roteamento de transporte público para melhor atender a população.

Referências Bibliográficas

CÂMARA, G.; QUEIROZ, G. R.; Arquitetura de Sistemas de Informação Geográfica. In: CÂMARA, G.; DAVIS, C.; MONTEIRO, A.M.V. (Org). **Introdução à Ciência da Geoinformação**. INPE. cap. 3. Disponível em: <<http://dpi.inpe.br/gilberto/livro/introd/>>. Acesso em: 19 set. 2018.

GONÇALVES, R. P.; LISBOA FILHO, J.; VIEIRA, C. A. O. Proposta de coleta eficiente de dados para o Cadastro Técnico Multifinalitário. In: II SIMGEO (Simpósio Brasileiro de Ciências Geodésicas e Tecnologias da Geoinformação), 2., 2008, Recife - PE. **Anais do II Simpósio Brasileiro de Ciências Geodésicas e Tecnologias da Geoinformação – II SIMGEO**. Disponível em: <https://www3.ufpe.br/cgtg/SIMGEOII_CD/Organizado/cad/068.pdf>. Acessado em: 19 set. 2018.

LOCH, C.; ERBA, D. A. **Cadastro Técnico Multifinalitário Rural e Urbano**. Cambridge, MA: Lincoln Institute of Land Policy, 2007.

LONGLEY, P. A. et al. **Sistemas e Ciência da Informação Geográfica**. 3. ed. Porto Alegre, RS: Bookman, 2011.

MEDEIROS, G. F. B. **OpenStreetMap: Uma Análise Sobre a Evolução de Dados Geográficos Colaborativos no Brasil**. Brasília, DF: Universidade de Brasília, 2017. Apresentado como Trabalho de Conclusão de Curso. Disponível em: <http://bdm.unb.br/bitstream/10483/19524/1/2017_GabrielFranklinBrazdeMedeiros.pdf>. Acesso em: 19 set. 2018.

QUEIROZ, G. R.; FERREIRA, K. R. SGBD com extensões espaciais. In: CASANOVA, M. et. al. **Banco de Dados Geográficos**. Curitiba, PR: MundoGEO, 2005. cap. 8. Disponível em: <<http://www.dpi.inpe.br/livros/bdados/>>. Acesso em: 19 set. 2018.

UCHOA, E. N.; FERREIRA, P. R. **Geoprocessamento com Software Livre**. Campina Grande, PB: UFCG, 2004. Disponível em: <http://www.hidro.ufcg.edu.br/twiki/pub/Disciplinas/GeotecnologiaAplicada/geoprocessamento_software_livre_uchoa-roberto-v1.0.pdf>. Acesso em: 19 set. 2018.