



UNIVERSIDADE ESTADUAL DE MATO GROSSO DO SUL – UEMS

HÁLEX VIOTTO GOMES

INTEGRAÇÃO DE FUNCIONALIDADE GEOGRÁFICA AO SIAMI
WEB

DOURADOS/MS

2017

HÁLEX VIOTTO GOMES

INTEGRAÇÃO DE FUNCIONALIDADE GEOGRÁFICA AO
SIAMI *WEB*

Este exemplar corresponde à redação final da monografia da disciplina Projeto Final de Curso devidamente corrigida e defendida por Hálex Viotto Gomes e aprovada pela Banca Examinadora, como parte dos requisitos para a obtenção do título de Bacharel em Ciência da Computação.

ORIENTADORA: Profa. Dra. GLAUCIA GABRIEL SASS

DOURADOS/MS
2017

HÁLEX VIOTTO GOMES

INTEGRAÇÃO DE FUNCIONALIDADE GEOGRÁFICA AO
SIAMI WEB

Banca Examinadora:

Profa. Dra. GLAUCIA GABRIEL SASS (Orientadora)
Área de Computação - UEMS

Prof. Me. ANDRÉ CHASTEL LIMA
Área de Computação - UEMS

Prof. Dr. RUBENS BARBOSA FILHO
Área de Computação - UEMS

DOURADOS/MS
2017

Este trabalho é dedicado aos meus pais Dinho e Silvia que me proporcionaram a oportunidade de poder ingressar no ensino superior. Além dos meus pais este trabalho é dedicado a minha família e amigos que sempre estiveram torcendo por mim. Muito obrigado!

AGRADECIMENTOS

Primeiramente os meus agradecimentos são direcionados para a minha Orientadora a Professora Glaucia Gabriel Sass e para o Professor Doutor Odival Faccenda. Ambos tiveram um papel importante na minha carreira acadêmica, principalmente pelo fato de me arrastarem para o vosso projeto de iniciação científica. Esta oportunidade oferecida por eles me proporcionou um diferencial acadêmico que irá gerar muitos bons resultados no futuro.

Professora Glaucia, te agradeço muito pelo que fez por mim, principalmente por não ter me deixado desistir deste trabalho sempre me dizendo que eu conseguiria concluí-lo. Sei que não sou o melhor dos estudantes, mas me esforcei ao máximo para não decepcioná-la.

Professor Faccenda, te agradeço do fundo do meu coração por sempre o senhor compartilhar comigo suas experiências de vida e dividir comigo sua amizade. Saiba que sempre me sinto feliz em sua companhia pois sei que estou ao lado de alguém iluminado.

Por fim, agradeço a todos os professores com quem tive oportunidade de ser aluno e agradeço aos meus colegas de curso pela força e torcida.

Se há algo que tenho que agradecer à Deus é pelas boas pessoas que Ele colocou em meu caminho.

Obrigado!

RESUMO

A utilização de tecnologias da informação para realizar monitoramentos não é novidade no mundo atual. A possibilidade de utilizar um sistema de informação que possui a capacidade de realizar monitoramentos e se expressar com seus utilizadores através de informações textuais e visuais é algo tentador. Uma área interessante para a aplicação de um sistema de informação como este é a área da saúde, onde os monitoramentos de indivíduos em relação aos seus estados de saúde são muito importantes. O presente trabalho aplicou conceitos de Sistema de Informação Geográfica (SIG) na forma de integração de funcionalidades geográficas para o Sistema de Informação para Avaliação e Monitoramento de Idosos (SIAMI) do Grupo de Pesquisa em Necessidades da Saúde do Idoso (GPENSI) da Universidade Estadual de Mato Grosso do Sul (UEMS), Dourados (MS). A integração de funcionalidades teve por objetivo oferecer uma alternativa de visualização de informações, a visualização geográfica de informações.

Palavras-chave: Sistema de Informação *Web*, Sistema de Informação Geográfica, Informática em saúde, SIAMI.

SUMÁRIO

1	INTRODUÇÃO	17
2	REVISÃO DA LITERATURA	19
2.1	Sistema de Informação	19
2.1.1	Sistemas de Apoio à Decisão	21
2.1.2	Sistemas de Informação com Inteligência Artificial	21
2.1.3	Sistemas de Informação <i>Web</i>	22
2.2	Sistema de Informação Geográfica	23
2.2.1	Banco de Dados Geográfico	24
2.2.2	Georreferenciamento e Sistema de Referência de Coordenadas	32
2.3	Sistema de Informação Geográfica <i>Web</i>	33
2.4	SIAMI	34
2.5	Trabalhos relacionados	35
3	METODOLOGIA	39
4	RESULTADOS	41
5	CONCLUSÃO	55
	REFERÊNCIAS	57
	APÊNDICE A – <i>SCRIPT</i> PARA ALTERAÇÃO DE PONTOS	59
	APÊNDICE B – CÓDIGO RESPONSÁVEL POR REALIZAR O PROCESSAMENTO PARA GERAR OS RELATÓRIOS GEOGRÁFICOS.	61
	APÊNDICE C – <i>SCRIPT</i> PARA REPLICAR DADOS NO BANCO DE DADOS	75
	APÊNDICE D – RELATÓRIO RENDA <i>PER CAPITA</i>	79
	APÊNDICE E – RELATÓRIO FAIXA ETÁRIA	83
	APÊNDICE F – RELATÓRIO ESCOLARIDADE	87
	APÊNDICE G – RELATÓRIO CONDIÇÃO DE MORADIA	91

LISTA DE ILUSTRAÇÕES

Figura 1 – Baseado no autor Longley et al. (2013)	26
Figura 2 – Fenômenos espaciais	28
Figura 3 – Formulário de teste de inserção de dado geográfico	41
Figura 4 – Interface do QuantumGIS	42
Figura 5 – Conjunto de símbolos	43
Figura 6 – Mapa mostrando uma pequena região de Dourados	45
Figura 7 – Mapa mostrando uma região maior de Dourados	46
Figura 8 – Mapa apresentando um nível menor de <i>zoom</i>	46
Figura 9 – Cidade de Dourados	47
Figura 10 – Aplicativos instalados	48
Figura 11 – Fragmento da classe Moradia	48
Figura 12 – Tela que capta os dados sobre moradia	49
Figura 13 – Trecho de código (<i>Python</i>) que apresenta o processamento	49
Figura 14 – Painel contendo os filtros	51

LISTA DE ABREVIATURAS E SIGLAS

APIs	<i>Application Programming Interfaces</i>
BDG	Banco de Dados Geográfico
CPF	Cadastro de Pessoas Físicas
DRG	<i>Digital Raster Graphic</i>
GPENSI	Grupo de Pesquisa em Necessidades da Saúde do Idoso
HTML	Hipertext Markup Language
IA	Inteligência Artificial
IDH	Índice de Desenvolvimento Humano
IG	Informação Geográfica
ISO	<i>International Standards Organization</i>
MS	Mato Grosso do Sul
OGC	<i>Open Geospatial Consortium</i>
QGIS	QuantumGIS
SGBD	Sistema Gerenciador de Banco de Dados
SIAMI	Sistema de Informação para Avaliação e Monitoramento de Idosos
SIG	Sistema de Informação Geográfica
SQL	<i>Structured/Standard Query Language</i>
UEMS	Universidade Estadual de Mato Grosso do Sul
UTM	Universal Transversa de Mercator
XML	<i>Extensible Markup Language</i>
WCS	<i>Web Coverage Service</i>
WFS	<i>Web Feature Service</i>

WGS84 *World Geodetic System* - Sistema Geodésico Mundial, de 1984

WKT *Well Known Text*

WMS *Web Map Service*

1 INTRODUÇÃO

O tema deste trabalho envolve a integração de funcionalidades geográficas que permitam ao Sistema de Informação para Avaliação e Monitoramento do Idoso (SIAMI) apresentar informações geográficas. De forma geral o tema proposto se relaciona com a teoria dos Sistemas de Informação Geográfica (SIG), os quais integram dados geográficos associados com dados não geográficos para produzir informações geográficas. O SIAMI é um projeto do Grupo de Pesquisa em Necessidades da Saúde do Idoso (GPENSI) da Universidade Estadual de Mato Grosso do Sul (UEMS), Dourados (MS), as informações geográficas serão apresentadas em formato de mapas. Os dados utilizados no SIAMI referem-se à pessoas do segmento etário idoso que residem em setores da cidade de Dourados em Mato Grosso do Sul. A forma de apresentação das informações em mapas se dará por um Sistema *Web*.

O objetivo geral deste trabalho é agregar ao SIAMI funcionalidades de um Sistema de Informação Geográfica. Para tanto é necessário a utilização de um modelo de Banco de Dados Geográfico (BDG) implementado por um Sistema Gerenciador de Banco de Dados (SGBD).

O problema abordado refere-se à apresentação de informações sobre saúde dos idosos do SIAMI em um mapa de Dourados na *Web*, com o objetivo de auxiliar na melhora dos métodos de acompanhamento e tratamento dos mesmos. Além do mais, a implementação do modelo de BDG com os respectivos dados inseridos, assistirá o sistema *Web* em sua função informativa.

A apresentação de informações de forma visual pode ajudar na interpretação dos possíveis problemas de saúde em determinada região. Um exemplo clássico de análise visual deriva de uma investigação realizada pelo médico inglês John Snow, no distrito de Soho na cidade de Londres em 1854. Nesta época estavam ocorrendo surtos de cólera. O médico em sua investigação elaborou um mapa contendo pontos que representavam os doentes e vítimas fatais da cólera, além dos poços de água. Através do mapa o médico inglês constatou através da análise visual que a quantidade de vítimas eram maior ao redor de um dos poços de água. Desta forma, o ajudando a resolver o motivo do surto de cólera (LONGLEY et al., 2013).

Consistindo-se de uma investigação pura e aplicada, o problema levantado procura verificar a aplicabilidade dos conceitos de Sistema de Informação Geográfica e Análise Visual de dados.

A análise visual é um método utilizado para auxiliar o processamento de dados e informações. Como atualmente a produção de dados e informações alcançou um nível de sobrecarga, a análise visual possui o objetivo de tornar essa sobrecarga em oportunidades através da ideia de construir nossas próprias formas de processar dados, com isto melho-

rando nossos conhecimentos e nossas decisões (KEIM et al., 2008; ANDRIENKO et al., 2010).

Este tipo de análise é um meio-termo que combina capacidades humanas juntamente com as capacidades das máquinas em forma de cooperação. Esta cooperação visa usar ambas as capacidades para encontrar resultados mais efetivos (KEIM et al., 2008; ANDRIENKO et al., 2010).

A área de pesquisa em Análise Visual é bastante interdisciplinar combinando diversas áreas de pesquisa, tais como visualização de dados, mineração de dados, gerenciamento de dados, fusão de dados, estatística e ciência cognitiva, entre outras (KEIM et al., 2008; ANDRIENKO et al., 2010).

Técnicas de análises automatizadas com visualizações interativas são combinadas pela Análise Visual para um efetivo entendimento, raciocínio e tomada de decisão sobre conjuntos de dados muito grandes e complexos. A Ciência da Informação Geográfica, por exemplo, possui fundamentos que envolvem processamento computacional de informação geográfica sendo reportada e dirigida através de representações gráficas de dados (KEIM et al., 2008; ANDRIENKO et al., 2010).

Portanto, a visão geográfica de informações é um tipo de visualização da informação que pode esclarecer questionamentos, os quais não poderiam ser respondidos por outras formas de apresentação de informações. Para a área da saúde, possui significado no monitoramento de diferentes regiões em certa localidade, como o município de Dourados por exemplo.

Este tipo de monitoramento resulta em estratégias de acompanhamento aos idosos diferentes das existentes, pois, a abordagem será baseada na distribuição geográfica das informações de saúde. As possíveis modificações nas estratégias de abordagem dos problemas de saúde dos idosos podem acarretar em um atendimento aprimorado e conseqüentemente, melhorar o serviço de saúde prestado. Desta forma, aumentando a qualidade de vida dos idosos, servindo de modelo para outras aplicações de monitoramento. Por estas razões justifica-se o presente estudo. Os dados necessários para a execução deste trabalho já foram coletados pelo GPENSI. A tarefa adicional em relação aos dados é de atribuir um significado geográfico a aquele dado, para que seja possível distribuí-lo geograficamente em um mapa.

O capítulo 2 apresenta a literatura consultada para embasar este trabalho, além de descrever de forma resumida os trabalhos correlacionados.

No capítulo 3, é apresentada a metodologia utilizada para desenvolver o trabalho e algumas estratégias elaboradas. O capítulo 4 mostra os resultados alcançados no desenvolvimento deste trabalho.

O trabalho se encerra no capítulo 5, onde as devidas conclusões são apresentadas.

2 REVISÃO DA LITERATURA

As definições de sistema de informação, sistema de informação *web*, sistema de informação geográfica, banco de dados geográfico, georreferenciamento e sistema de referência de coordenadas, sistema de informação geográfica *web* e SIAMI são importantes para compreender a dimensão deste trabalho. Portanto, serão apresentadas nesta seção as devidas definições.

2.1 Sistema de Informação

Um sistema é um conjunto de componentes que se inter-relacionam para o cumprimento de um objetivo em comum. O sistema é composto basicamente por três componentes: entrada, processamento e saída. A entrada é um componente do sistema responsável pela captação dos insumos ou dados brutos. O processamento possui a tarefa de aplicar transformações sobre os insumos captados pela entrada, estas transformações estão relacionadas com os objetivos que um sistema em particular quer atingir. A saída é o elemento do sistema que direciona os elementos produzidos pelo processamento até o seu respectivo destino final (O'BRIEN, 2004; LAUDON; LAUDON, 2010).

A existência de um sistema é englobada por um ambiente, pois o mesmo não consegue existir de forma isolada. O ambiente do sistema é composto pelos elementos que irão interagir com ele, sendo requisitando os produtos ou oferecendo insumos. Um ambiente de um sistema pode ser outro sistema, desta forma o sistema inserido neste ambiente é definido como um subsistema. No mesmo ambiente podem coexistir mais de um sistema que podem ou não interagir uns com os outros, caso ocorra a interação, esta é realizada através de interfaces de comunicação. Se houver necessidade, alguns sistemas podem transformar a si ou o ambiente no qual estão inseridos a fim de se adaptarem para sobreviver às mudanças (O'BRIEN, 2004; LAUDON; LAUDON, 2010).

O sistema de informação tem como meta a produção de informações. Estas informações possuem uma definição diferente dos dados, pois, dados representam um fato que foi observado, captado ou medido, enquanto que informação é um dado que sofreu transformações, formatações e a ele foi atribuído um contexto que faça sentido para os consumidores desta informação. As informações produzidas pelo sistema de informação se destinam a apoiar a organização nos seus processos de tomada de decisão, coordenação e controle, além de, apoiar os gerentes e trabalhadores da organização a analisarem e resolverem problemas e assuntos complexos, também auxiliando no desenvolvimento de novos produtos (O'BRIEN, 2004; LAUDON; LAUDON, 2010).

Atuando sobre esse sistema de informação temos também duas práticas fundamentais que ajudam à regulá-lo para que o mesmo sempre se mantenha funcionando e produzindo

informações corretas. Essas práticas são: O *feedback* e o controle.

O *feedback* é um processo avaliativo do desempenho do sistema. Através dos dados que o *feedback* coleta, é possível regular o sistema para que os procedimentos de entrada de dados, processamento e saída consigam produzir a informação correta (BARBOSA, 2003; O'BRIEN, 2004; LAUDON; LAUDON, 2010).

Complementando a função do *feedback* é necessário um processo de controle. O controle monitora e avalia o *feedback* para verificar se o sistema está cumprindo com as metas definidas. Caso seja necessário, a função de controle intervém e realiza ajustes nos componentes de entrada e processamento do sistema de informação para que ele cumpra com as metas (O'BRIEN, 2004).

Os Sistemas de Informação possuem recursos que complementam a sua funcionalidade. Segundo O'Brien (2004) esses recursos são divididos em cinco áreas básicas descritas a seguir:

- pessoas ou recursos humanos: são as pessoas que utilizarão o sistema, sendo essas pessoas usuários finais e as pessoas responsáveis por construir e regular o sistema.
- hardware: são os componentes físicos que o sistema utilizará, sendo computadores, discos de mídia, etc.
- redes: são os componentes responsáveis por fazer a comunicação entre vários sistemas e componentes de hardware.
- dados: aqueles armazenados em banco de dados e as bases de conhecimento do sistema.
- software: são os programas e os procedimentos para processamento da informação.

Para Laudon e Laudon (2010), esses recursos são divididos em três grandes áreas:

- organizações: trata em como a organização ao qual o sistema será inserido está definida.
- tecnologia: se refere às tecnologias utilizadas, hardwares, redes, dados e softwares, possuindo uma definição bem próxima daquela definida por O'Brien (2004) nos itens acima.
- pessoas: basicamente a mesma ideia definida por O'Brien (2004).

Os Sistemas de Informações podem ser classificados de diferentes formas dependendo do tipo de função que o sistema assume em certo ambiente. Conforme O'Brien (2004) essa classificação é dividida em dois grandes grupos: Sistemas de Apoio às Operações e Sistemas de Apoio Gerencial.

Os Sistemas de Apoio às Operações têm como função a produção de informações que apoiem o processo operacional de um negócio. Essas informações podem servir a outros Sistemas de Informações, mas não possuem utilidade para o processo de tomada de decisão.

Os Sistemas de Apoio Gerencial têm como objetivo a produção de informações que sejam úteis para o processo de tomada de decisão.

2.1.1 Sistemas de Apoio à Decisão

Os Sistemas de Apoio à Decisão fazem parte do conjunto maior dos Sistemas de Apoio Gerencial. O que o diferencia dos demais sistemas classificados neste grupo é sua capacidade de auxiliar a tomada de decisão para assuntos no qual não existe um procedimento totalmente definido. Geralmente esses assuntos se referem a problemas que se alteram muito rápido, o que acaba exigindo procedimentos que gerem informações no mesmo ritmo (BARBOSA, 2003; LAUDON; LAUDON, 2010).

De acordo com O'Brien (2004), "os sistemas de apoio à decisão são projetados para serem sistemas de resposta rápida que são iniciados e controlados por usuários finais." Esses sistemas utilizam como fonte de recursos bancos de dados e modelos de referências. (O'BRIEN, 2004; LAUDON; LAUDON, 2010).

Os sistemas de apoio à decisão têm como produto informações que expressam indicativos, esses indicativos é que auxiliam os gerentes na tomada de decisão. Os indicativos são produzidos através de uma série de relações definidas no sistema que classifica os dados processados e os gera.

O conceito de Sistema de Apoio à Decisão se baseia no fato de que além de gerar indicativos, o gerente tem total autonomia de inserir informações hipotéticas a fim de analisar os resultados gerados para a sua tomada de decisão.

2.1.2 Sistemas de Informação com Inteligência Artificial

A Inteligência Artificial (IA) tenta sistematizar em computador o comportamento humano e seus padrões de pensamento. Ela auxilia os Sistemas de Informação com técnicas inteligentes para aprimorar a tomada de decisão. Essas técnicas incluem Sistemas Especialistas, raciocínio baseado em casos, lógica difusa, redes neurais, algoritmos genéticos e agentes inteligentes (LAUDON; LAUDON, 2010).

Dentre as várias técnicas inteligentes citadas no parágrafo anterior, o SIAMI incorpora um Sistema Especialista.

Sistema Especialista tenta atuar como um consultor especializado que tem a tarefa de substituir um consultor real sem perder a função de auxiliar nas tomadas de decisões (O'BRIEN, 2004).

O sistema transforma o conhecimento de uma determinada área em regras e fatos que serão utilizados para resolver problemas de um determinado domínio específico (MENDES, 1997; LAUDON; LAUDON, 2010).

Para a construção de um Sistema Especialista é necessário a presença de um engenheiro do conhecimento, que tem como função obter essas regras e fatos específicos para compor a base de conhecimento do Sistema Especialista.

De acordo com O'Brien (2004) a vantagem de um Sistema Especialista é que ele substitui um especialista humano em muitas situações que são consideradas problemáticas. O sistema se comporta de uma forma mais consistente, é rápido, não é influenciado por estresse e pode ainda ser composto pelo conhecimento de mais de um especialista.

Os Sistemas Especialistas possuem como desvantagem um foco mais limitado a certas áreas do conhecimento. São incapazes de aprender e possuem um alto custo de desenvolvimento.

Portanto, os Sistemas Especialistas são uma forma de ajuda na tomada de decisão que tem como característica a presença de um especialista em sua base de conhecimento, sendo que, sua aplicação prática deve estar limitada a uma área do conhecimento de forma específica.

O Sistema Especialista presente no SIAMI aborda a área da saúde e possui como foco principal o idoso. Este Sistema Especialista tem como função adquirir conhecimento necessário em sua base de conhecimento para gerar mecanismos (indicadores) que ajudem na tomada de decisão.

2.1.3 Sistemas de Informação *Web*

Os Sistemas de Informação *Web* são Sistemas de Informação que atuam sobre o ambiente *web*. Ele funciona no esquema cliente/servidor, sendo que o cliente são os dispositivos que têm acesso a internet e interagem com o servidor através de um navegador ou aplicativo em dispositivos móveis. (ZANETI JUNIOR; ROCHA VIDAL, 2006).

O servidor é onde está instalado o Sistema de Informação que recebe dos clientes requisições de informações. Todas as definições vistas na seção 2.1 e subseções 2.1.1 e 2.1.2 até agora para Sistemas de Informação, se aplicam ao ambiente *web*.

A vantagem dos Sistemas de Informação *Web* é que eles podem ser acessados de qualquer lugar do planeta, propiciando uma maior mobilidade. O trabalho de manutenção do sistema é outro fator vantajoso propiciado pelo ambiente *web*. Como o sistema está instalado em apenas uma máquina, no caso o servidor, qualquer problema que haja com o sistema, basta arrumá-lo no servidor e após essa manutenção todos os dispositivos terão acesso imediato ao sistema. (CONALLEN, 2003 apud PERIZZOLO, 2005).

Segundo Conallen (2003 apud PERIZZOLO, 2005) os Sistemas de Informação *Web* possuem certas desvantagens:

- A falta de padronização entre os navegadores pode resultar em uma exibição da aplicação de forma equivocada.
- O tempo para processar as tarefas depende da velocidade de conexão entre o cliente e servidor.
- Como os sistemas baseados na *web* são dependentes dos recursos dos navegadores *web* para interagir com os usuários, certos recursos podem estar presentes ou não nesses navegadores. Desta forma dificultando o processo de previsão de como a aplicação irá se comportar.
- A falta de uma rede de internet impossibilita o acesso ao sistema *web*, resultando em um difícil gerenciamento do estado de conexão de um cliente no servidor quando a queda de rede acontece.

Da mesma forma que um Sistema de Informação pode estar trabalhando em paralelo com outros sistemas, o Sistema de Informação *Web* também possui essa funcionalidade, trabalhando com outros Sistemas de Informação *Web* em servidores diferentes, sendo a ligação entre esses servidores feita pela internet. (ZANETI JUNIOR; ROCHA VIDAL, 2006).

2.2 Sistema de Informação Geográfica

Definir o que é um Sistema de Informação Geográfica (SIG) não parece ser uma tarefa simples. Segundo Silva (2003) e Miranda (2010) existem diversos autores que definem SIG de diversas maneiras diferentes, algumas possuindo uma certa convergência e outras divergências. Entre essas definições existem aquelas que tratam o SIG como apenas uma ferramenta tecnológica, outras se referem a SIG como um sistema que abrange os recursos de dados, software, redes, hardware e pessoas (definições de sistema de informação apresentados na seção 2.1) (SILVA, 2003; MIRANDA, 2010). Há ainda a definição de que SIG é um sistema e também uma ciência, a ciência da informação geográfica (LONGLEY et al., 2013).

Os SIG são uma derivação de Sistema de Informação, possuindo as mesmas funcionalidades mas com a adição de poder informar em que local tal fato ou informação está ocorrendo (LONGLEY et al., 2013).

A importância dos SIG é a possibilidade de se resolver problemas de ordenamento territorial que envolvem aspectos de localização geográfica. De acordo com Longley et al. (2013), são apresentados alguns problemas que estão relacionados a fatores geográficos:

- planejamento de onde localizar clínicas e hospitais.

- decisão de rotas e horários levados em consideração por parte das empresas de entregas expressas.
- escolha de quais rodovias poderiam ser transformadas em autoestradas.
- visualizar performance de vendas de uma loja em uma determinada região ou verificar quais pontos de uma cidade ou região seriam melhores para expandir ou construir novas lojas.
- manejo de florestas, incluindo quais árvores cortar, definir locais de estrada e locais para novo plantio.
- obter informações a respeito de uma cidade, como hotéis disponíveis e mapa.
- em uma lavoura decidir locais para fertilização e ou aplicação de pesticidas.

De acordo com Longley et al. (2013) os Sistemas de Informação Geográfica:

- São sistemas computacionais construídos para armazenar e processar informações geográficas.
- São ferramentas que melhoram a eficiência e efetividade no tratamento da informação geográfica.
- Podem ser utilizados como repositórios de informações geográficas em bancos de dados.
- Automatizam o processo de confecção de mapas e diminuem muito o tempo relativo para fazer análises sobre dados ou informações geográficas.
- Possuem a capacidade para lidarem com o processamento de informações, possuindo limitantes (quando o SIG construído e focado em lidar apenas com informações geográficas) em relação a transformação de dados em informação.

2.2.1 Banco de Dados Geográfico

O Banco de Dados Geográfico (BDG) ocupa uma posição importante na composição de um SIG. Como essa base de dados geralmente está relacionada com os elementos da superfície da Terra, ela acaba sendo composta por dados geográficos que descrevem forma e posição, e atributos que são divididos em atributos qualitativos e quantitativos. Os dados geográficos e atributos são utilizados para representar os elementos da superfície da Terra em uma base de dados. Em certos SIGs podem existir BDGs que separam os dados geográficos dos atributos, enquanto que em outros SIGs os dados geográficos e atributos são mantidos em uma mesma entidade (ou tabela de um BDG) (MIRANDA, 2010).

Segundo Miranda (2010), os atributos quantitativos são divididos em intervalo e razão enquanto que atributos qualitativos são divididos em nominal e ordinal. A tabela 1 apresenta um esquema para melhor compreensão.

Tabela 1 – Dados quantitativos e qualitativos

Atributos	Quantitativos	Intervalo	Dados numéricos considerados discretos ou agrupados de forma discreta. Exemplo: quantidade de pessoas, classificação baseada no nível de renda.
		Razão	Dados numéricos considerados contínuos. Exemplo: temperatura, distância, volume.
	Qualitativos	Nominal	Nomes ou códigos em que não exista relação de ordem. Exemplo: Carro, Fábrica, Concessionária.
		Ordinal	Nomes ou códigos onde há relação de ordem. Exemplo: Rodovia, Avenida, Rua.

Atributos do tipo razão representam medidas precisas, como temperatura, comprimento ou área. A razão é definida em relação a uma origem de uma escala contínua de uma unidade de medida. Intervalo corresponde ao agrupamento de dados numéricos, como idade ou nível de renda. Os atributos nominais categorizam os elementos por nomes ou códigos, por outro lado os atributos ordinais representam a ordem dos elementos (rodovias, avenidas, ruas).

Mas para que seja possível trabalhar com os elementos que compõe a superfície da Terra são necessários a utilização de modelos, especificamente modelos relacionados aos elementos citados neste parágrafo. A vantagem de se utilizar modelos é que eles estão relacionados a conceitos derivados da observação humana da realidade, desconsiderando tecnologias e restrições de implementações (MIRANDA, 2010).

Uma outra vantagem de se utilizar modelos é que eles são uma abstração da realidade, representando-a com um certo nível de informações. Essa representação considera uma quantidade de informações e detalhes menores, desta forma facilitando o estudo sobre uma área de interesse (MIRANDA, 2010).

O modelo de dados adotado neste trabalho é o modelo de dados geográficos. Este modelo de dados tem a função de definir como o espaço geográfico se torna discreto para efeitos de análise e comunicação (MIRANDA, 2010). Segundo Miranda (2010), a existência de modelos se dá pelas suposições criadas. Desta forma o modelo de dados geográficos descrito em (MIRANDA, 2010) é baseado em três suposições:

- quais fenômenos geográficos podem ser identificados unicamente.
- quais atributos podem ser medidos ou especificados.
- quais coordenadas geográficas podem ser registradas.

Na visão de Miranda (2010), como o modelo de dados geográfico está relacionado a trabalhos que envolvem relações espaciais entre objetos geográficos, este modelo acaba sendo

retratado como dado espacial. Martin (1996) *apud* Miranda (2010), define o termo dado espacial como sendo medidas que estão relacionadas a objetos geográficos em qualquer escala.

O dado espacial engloba fenômenos espaciais e geográficos, desta forma um analista SIG consegue produzir resultados que são considerados informações geográficas. Estes fenômenos, para melhor entendimento, podem ser uma área de um parque, um fenômeno climático como uma frente fria, um fenômeno global como alterações climáticas, ocupação histórica de uma região por seres humanos, entre outros (MIRANDA, 2010).

Conforme Miranda (2010), um modelo de dados que é frequentemente utilizado e simples (e que será aplicado a este estudo), envolve entidades que contém apenas dado espacial e atributos. O dado espacial representa a localização geográfica e geometria, enquanto que, os atributos representam dados não espaciais como descrições, nomes, identificadores, características do objeto geográfico sendo analisado, etc.

Para efeito de uma melhor compreensão sobre a representação de fenômenos espaciais e geográficos em computadores, serão descritas algumas informações relacionadas a dados matriciais e dados vetoriais. Estas formas de representação de dados foram definidas pois os computadores são limitados em termos de processamento e capacidade de armazenamento para lidar com dados extremamente detalhados ou infinitos relacionados aos fenômenos espaciais e geográficos reais (LONGLEY et al., 2013).

Os dados matriciais são representados dentro de uma malha retangular ou quadrada. Possuem a função de representar variações geográficas. Cada variação é representada dentro da malha por uma célula popularmente conhecida como *pixel* (contração de *picture elements*). Cada célula armazenará propriedades ou atributos referentes ao elemento geográfico sendo representado (LONGLEY et al., 2013). A figura 1 apresenta um exemplo de malha.

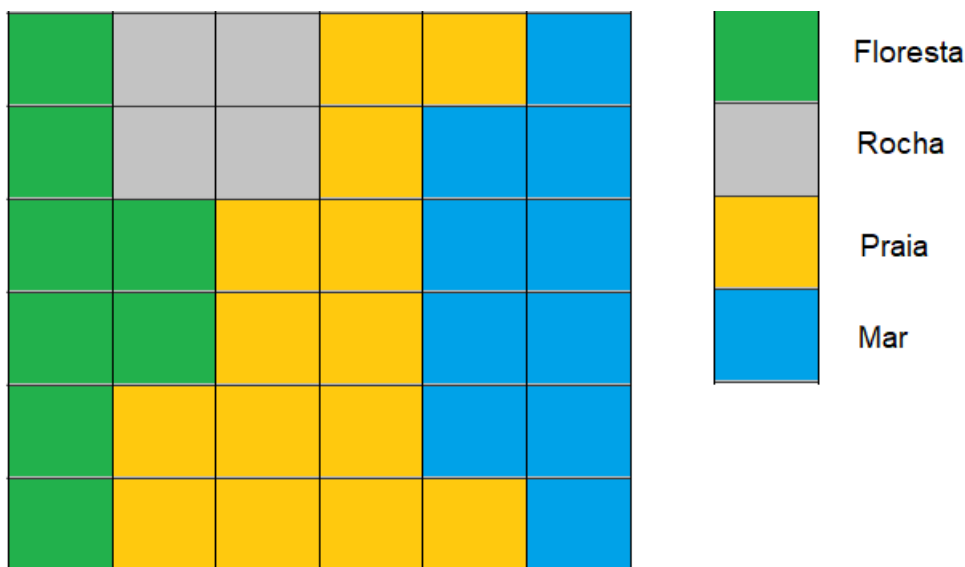


Figura 1 – Baseado no autor Longley et al. (2013)

Nesta figura 1, a malha está representando uma área que envolve um litoral. As células na cor verde possuem atributos relacionados a uma área de floresta. As células de cor cinza estão representando uma ambiente rochoso. Células com a cor amarela representam uma faixa de areia (praia) e células com a cor azul representam o mar.

Os dados matriciais podem ser gerados por diversas fontes, uma das mais comuns são dados matriciais gerados por satélites. Esses dados matriciais são capturados por satélites e enviados à Terra para que possam ser analisados. Vale ressaltar que toda variação de informação dentro de uma célula é perdida, pois um único valor ou atributo deve ser atribuído a célula em questão (LONGLEY et al., 2013).

Dados matriciais se acomodam muito bem em superfícies planas, desta forma exigindo-se uma superfície plana ou projetada para que seja possível sua representação (LONGLEY et al., 2013).

Além da matriz ou malha, são necessários também alguns dados que descrevem informações a respeito de uma malha em particular. Estes dados denominados Metadados (dados sobre dados), armazenam informações a respeito da coordenada geográfica do canto superior esquerdo da matriz (informação útil para georreferenciar a malha), o tamanho da célula, o número de elementos por linha e coluna e a projeção que deve ser utilizada (LONGLEY et al., 2013).

Os dados matriciais são utilizados muitas vezes como pano de fundo em visualização de mapas por se assemelharem com mapas convencionais. Para se referir a dados matriciais são utilizados termos como *digital raster graphic* (DRG) ou apenas *raster* (LONGLEY et al., 2013).

No caso dos dados vetoriais, estes são capturados utilizando-se pontos. Para compor uma linha são utilizados um conjunto de dois pontos que são atravessados por uma linha perfeitamente reta. Para compor uma área são utilizados um conjunto de pontos ou vértices conectados por linhas retas. Diferentemente dos dados matriciais, para se compor uma área por exemplo, são necessários apenas dados sobre a localização geográfica de cada ponto desta área (LONGLEY et al., 2013).

Estes dados acabam sendo utilizados para representar de forma discreta os fenômenos geográficos (LONGLEY et al., 2013).

Segundo Miranda (2010), aceita-se universalmente a abstração em três ou mais classes do fenômeno espacial ou geográfico para dados vetoriais. A abstração corresponde a:

- ponto, possuindo dimensão zero.
- linha, possuindo uma dimensão.
- área ou polígono, possuindo duas dimensões.
- superfície, possuindo três dimensões.

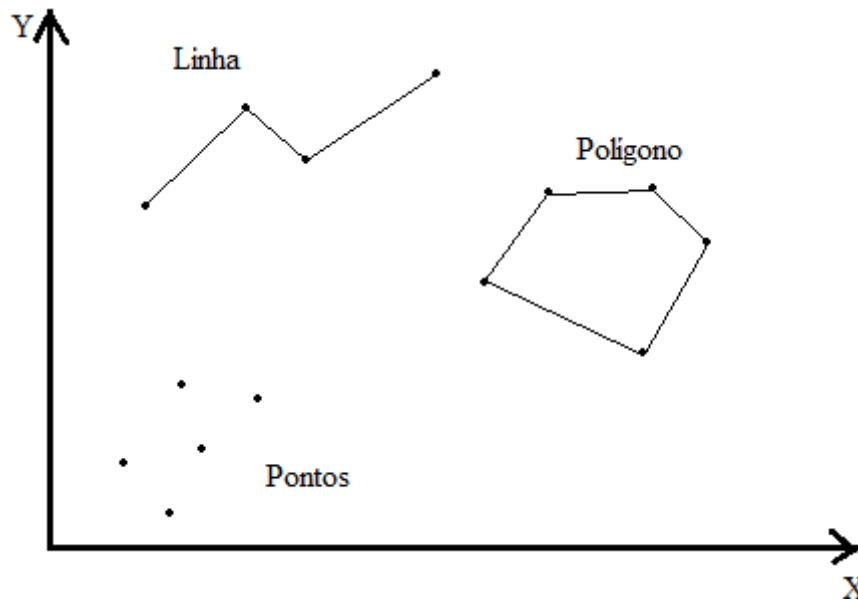


Figura 2 – Fenômenos espaciais

Há ainda mais uma abstração que possui quatro dimensões, considerando a dimensão tempo. Para uma compreensão melhor, a figura 2 apresenta os três primeiros tipos de abstrações de fenômenos espaciais.

Para um objeto geográfico o ponto é a representação mais simples. Quando se utiliza um ponto para representar um fenômeno espacial, quer dizer que este fenômeno ocorre em apenas um local do espaço. Além disso o ponto é um elemento discreto ocupando apenas uma posição no espaço e tempo. O ponto não possui dimensão, comprimento ou largura, mas pode ser apresentado em mapa utilizando símbolo. Por fim, como o ponto é um elemento discreto, ele não está sujeito a sobreposições (MIRANDA, 2010).

Com os modelos definidos o próximo passo é apresentar como estes modelos podem ser aplicados a um projeto de banco de dados. Um banco de dados nada mais é do que uma coleção de dados estruturalmente armazenados sobre um determinado tipo de assunto. Um BDG é um banco de dados que suporta o armazenamento de dados relacionados a objetos geográficos. Estes dados podem ser armazenados em arquivos ou podem ser armazenados utilizando-se um Sistema Gerenciamento de Banco de Dados (SGBD) (LONGLEY et al., 2013).

Segundo Longley et al. (2013), a abordagem de banco de dados para armazenamento de dados geográficos apresenta algumas vantagens:

- redução de redundância causado pelo armazenamento de todos os dados em único local.
- por conta da organização dos arquivos e diminuição de redundâncias, os custos de manutenção são diminuídos.

- aplicações se tornam independentes dos dados, ou seja, como os dados obedecem uma estrutura definida de gerenciamento e organização dos dados, as aplicações devem apenas conhecer os comandos necessários para realizar operações sobre estes dados. Desta forma diversas aplicações diferentes podem utilizar um mesmo BDG.
- facilidade em compartilhar os dados.
- padrões de segurança e dados podem ser utilizados.
- SGBDs são construídos para possibilitar a uma grande quantidade de usuários os acessos aos dados quase que simultaneamente.

Porém algumas desvantagens surgem:

- a aquisição e manutenção de softwares SGBDs podem ter custos elevados, principalmente se tratando de SGBDs de licença comercial.
- em projetos que sejam pequenos, um SGBD pode aumentar a complexidade do gerenciamento de dados.
- para usuários individuais, uma solução baseada em arquivos pode ser a melhor escolha, pois, tipos de dados mais complexos e estruturas onde índices e algoritmos possam ser aplicados podem ter um desempenho melhor.

Quando um pequeno grupo de pessoas utiliza um pequeno banco de dados simples, este banco de dados pode ser armazenado em disco rígido do computador em arquivos normais, mas quando o banco de dados é grande e acessado por um grande grupo de pessoas, se faz necessário o uso SGBDs. Os SGBDs são softwares construídos para gerenciar de forma efetiva os dados e acessos por parte dos usuários (LONGLEY et al., 2013).

Na visão de Longley et al. (2013), os SGBDs são bem sucedidos por oferecer as seguintes características:

- possuem nativamente modelos de dados para representar elementos comuns da realidade. Por exemplo, representar números inteiros, datas, textos e até os elementos geográficos como pontos, linhas, polígonos.
- disponibilização de ferramentas para carregar dados em banco de dados.
- utilização de índices para aceleração de pesquisa de dados.
- suporte a uma linguagem padrão de pesquisa e manipulação de dados, a linguagem SQL (*Structured/Standard Query Language*)
- controles de segurança aplicados aos acessos aos dados.
- gerenciamento automático de atualização do banco de dados.

- *backup* e recuperação do banco de dados.
- ferramentas de administração de banco de dados.
- oferecimento de APIs (*Application Programming Interfaces*).

Existem três principais tipos de SGBDs utilizados em SIG: SGBD relacional, SGBD objeto e SGBD objeto-relacional (LONGLEY et al., 2013).

Um banco de dados relacional é composto por tabelas que são estruturas bidimensionais possuindo linhas e colunas. São utilizadas para conter os atributos de um objeto.

SGBD objeto foi projetado para possibilitar o armazenamento de objetos e prover ferramentas de consulta orientadas a objetos. Este tipo de SGBD tem a capacidade de lidar com objetos ricos em dados como objetos geográficos, áudios e vídeos.

SGBD objeto-relacional é um SGBD relacional que possui certas funcionalidades de um SGBD objeto. Este tipo de SGBD tem a capacidade de armazenar tanto atributos quanto as funcionalidades de um objeto.

Segundo Longley et al. (2013), através da adição de sete capacidades fundamentais é que atualmente diversos SGBDs objeto-relacional se tornaram capazes de lidar com objetos geográficos. As sete capacidades são:

- consulta de análise sintática.
- otimizadores de consulta melhorados para lidar com consultas geográficas.
- linguagem de consulta atualizada para trabalhar com dados geográficos.
- serviços de indexação expandidos.
- gestão de armazenamento aprimorado para gerenciar o armazenamento de objetos geográficos.
- serviços de transação estendidos para lidar com transações de longa duração que estão relacionadas aos objetos geográficos.
- replicação de tipos geográficos e tratamento de problemas na conciliação de alterações feitas por usuários.

Conforme Longley et al. (2013), alguns dos principais fornecedores comerciais de SGBDs disponibilizaram versões de seus produtos que suportam objetos geográficos. A IBM oferece o DB2 *Spatial Extender* e o *Informix Spatial Data-blade*; a Microsoft ofereceu capacidades em seu SQLServer e a Oracle já possui esta versão em seu SGBD Oracle. O SGBD de código aberto PostgreSQL possui uma extensão chamada PostGIS que oferece suporte a manipulação de objetos geográficos.

Apesar dos SGBDs disponíveis que suportam objetos geográficos possuírem algumas diferenças entre eles, todos eles conseguem oferecer recursos básicos de armazenamento,

gerenciamento e consulta de objetos geográficos graças as sete capacidades fundamentais que foram definidas anteriormente (LONGLEY et al., 2013).

Segundo Longley et al. (2013), o objetivo destas extensões é armazenar, recuperar e gerenciar dados geográficos. Para operações mais complexas que envolvam estes dados é necessário a utilização do SGBD em conjunto com um SIG.

Baseados na ISO (*Internacional Standards Organization*) e nos padrões do OGC (*Open Geospatial Consortium*), a comunidade SIG definiu alguns tipos de objetos geográficos e operações que podem ser realizadas pelos bancos de dados geográficos (LONGLEY et al., 2013).

Segundo Longley et al. (2013) foram definidos operações que geram resultados booleanos e operações de análise espacial. As operações booleanas necessitam de duas geometrias para realizar sua ação. Essas operações podem ser do tipo:

- verificação de igualdade.
- verificação de disjunção.
- verificação de interceptação.
- se as geometrias se tocam em suas fronteiras.
- cruzamento de geometrias.
- se uma geometria está contida na outra.
- se uma geometria contém completamente a outra.
- sobreposição de geometria.
- se há interseções que envolvam o interior, limite ou exterior de uma geometria.

De acordo com Longley et al. (2013), existem também as operações de análise espacial que envolvem as geometrias e são do tipo:

- distância.
- buffer.
- envoltório convexo.
- interseção.
- união.
- diferença.
- diferença simétrica.

2.2.2 Georreferenciamento e Sistema de Referência de Coordenadas

Apenas o armazenamento da geometria e de atributos não geográficos não é o suficiente para se trabalhar com o estudo dos fenômenos geográficos. Para que o dado geográfico tenha seu valor é necessário *georreferenciá-lo* (atribuir uma localização). Esta localização deve ser única e seu significado deve ser compartilhado e compreendido por qualquer pessoa no mundo. Entretanto, para que seja possível tal feito é necessário a utilização de algum sistema específico de referência de coordenadas (LONGLEY et al., 2013).

Esta subseção tem a função de esclarecer alguns termos que são utilizados neste trabalho. O aprofundamento destes temas não serão apresentados, para maiores detalhes recorra a (SILVA, 2003; MIRANDA, 2010; LONGLEY et al., 2013).

Datum ou elipsoide de rotação, é um modelo matemático utilizado para definir posições geográficas denominadas latitudes e longitudes. Este modelo tenta aproximar uma superfície teórica com a superfície real da Terra. Isto faz com que seja possível obter posições geográficas precisas e acuradas (LONGLEY et al., 2013).

O WGS84 (*World Geodetic System* - Sistema Geodésico Mundial, de 1984) é um elipsoide amplamente aceito para utilização global, pois cada país pode definir um elipsoide diferente e específico para georreferenciamento. Um elipsoide definido em um país pode não funcionar muito bem se utilizado em outro. Portanto existem elipsoides globais como o WGS84 que podem servir como um meio de comunicação entre os diferentes *Datum* (LONGLEY et al., 2013).

No Brasil especificamente, foram adotados no passado os *Datum* Córrego Alegre e SAD69, atualmente o SIRGAS2000 é o sistema de referência oficial. (LONGLEY et al., 2013).

Apesar de latitudes e longitudes serem utilizadas para georreferenciamento e possuírem importância em diversos tipos de análises como cálculos de distâncias entre pontos sobre a superfície curva da Terra, a maioria das tecnologias para se trabalhar com georreferenciamento utilizam apenas superfícies planas (LONGLEY et al., 2013).

Para tanto, são necessários usos de projeções aliadas a um sistema de coordenadas para tornar a superfície curva da Terra em uma superfície projetada em um plano. O sistema de coordenadas adotado possui como base o sistema de coordenadas cartesianas (LONGLEY et al., 2013).

A projeção mencionada neste trabalho é a projeção universal transversa de Mercator (UTM). Este sistema de projeção é encontrado muitas vezes em aplicações militares ou em aplicações que envolvem cobertura global ou nacional (LONGLEY et al., 2013).

O sistema UTM é baseado na projeção de Mercator mas na orientação *transversal*. Neste sistema o plano que representa a Terra é subdividido em 60 zonas. Cada zona possui 6 graus de largura. Além do mais, o norte e sul são considerados para facilitar a localização e são diferenciados a partir da linha do equador. O norte se situa na parte

superior da linha do equador enquanto que o sul se situa na parte inferior (LONGLEY et al., 2013).

As coordenadas de uma zona UTM são definidas em metros, o que facilita cálculos de distâncias precisas entre pontos (LONGLEY et al., 2013).

2.3 Sistema de Informação Geográfica Web

A evolução dos computadores e a queda de seus preços, fizeram com que o acesso aos mesmos mudasse de acesso apenas em universidades para acesso no conforto de seu lar. Neste mesmo período de evolução que compreende os anos de 1960 até 1970, foi quando começaram a surgir trabalhos que envolviam temas relacionados ao que hoje se conhece como internet (LONGLEY et al., 2013).

Toda esta evolução de computadores e internet fez com que o termo SIG distribuído surgisse. A visão de um SIG que possa ser utilizado através da rede está começando a se tornar realidade (LONGLEY et al., 2013).

Do ponto de vista de Longley et al. (2013), alguns conceitos relacionados a SIG devem ser esclarecidos para melhor compreensão do que seja SIG distribuído:

- U, se refere a localização do usuário e da interface pelo qual o usuário acessa as funcionalidades de um SIG.
- D, são dados que o usuário obtém através de um SIG.
- P, local onde os dados são processados.
- S, está relacionado ao sujeito, área de estudo ou foco de um projeto SIG.

Utilizando as definições dos itens acima é possível apresentar a diferença entre um SIG tradicional e um SIG distribuído. Em um SIG tradicional os itens U, D e P estão concentrados em um mesmo local, por exemplo, em um computador que possua um software de SIG instalado. Em relação a um SIG distribuído os itens U, D e P podem estar em locais completamente diferentes uns dos outros, por exemplo, U pode ser uma interface SIG instalada no computador presente na casa de um usuário, D pode ser um banco de dados armazenado em um servidor da empresa onde o usuário trabalha e P pode ser servidores na nuvem contratados pela empresa para o processamento dos dados captados de D. Em relação ao item S, utilizando SIG ou SIG distribuído é possível estar em contato direto com o sujeito ou área de estudo (LONGLEY et al., 2013).

Para os SIGs distribuídos os padrões de comunicações e especificações são importantes para manter uma interoperabilidade entre esses sistemas. Alguns padrões como ASCII (codificação de caracteres), XML (*extensible markup language*), HTML (*hypertext markup language*) são padrões universais envolvidos na arquitetura *web*. Outros são especificações que definem algumas características dos SIG, como o *Open Geospatial Consortium*

(OGC) que define padrões abertos e interoperabilidade dos SIGs. Sem deixar de lado especificações para os serviços *web*, como o *Web Map Service* (WMS), *Web Feature Service* (WFS) e *Web Coverage Service* (WCS) para solicitação automática de dados por parte dos programas SIGs dos usuários a servidores remotos (LONGLEY et al., 2013).

Serviço de Informação Geográfica ou serviço de IG é o termo utilizado para descrever um programa que executa uma atividade específica de um SIG em um site remoto. Este serviço é acessado remotamente pelo usuário e pode compreender funcionalidades que dependam do fornecimento de dados por parte do usuário, por parte do próprio serviço ou ambos. Para que um serviço de IG possa ser disponibilizado como serviço é necessário que se possua certas características. Uma dessas características é a necessidade de um banco de dados que precise ser constantemente atualizado, fazendo com que seja caro para um simples usuário manter um banco desses. E por fim a característica de dependência de operações de SIG que são muito complexas, fazendo com que um serviço de IG se especialize em oferecer esta operação que um SIG comum poderia não fornecer (LONGLEY et al., 2013).

Os SIGs distribuídos possuem suas vantagens e desvantagens. Como vantagens estes sistemas oferecem benefícios, como:

- redução na duplicação de esforços.
- acesso de funcionalidades de SIG através de dispositivos simples como navegadores *web*.
- custo financeiro menor para acesso a funcionalidades de SIG na *web*.

Por contra partida existem as desvantagens:

- limitações impostas por largura de banda e confiabilidade de banda.
- dependência do funcionamento da rede de internet para acesso ao SIG distribuído.

2.4 SIAMI

O SIAMI é um Sistema de Informação que tem como função monitorar a classe idosa da sociedade. Esse monitoramento permite que as equipes de saúde se preparem melhor para atender a diversos casos específicos de problemas de saúde que podem ocorrer com os idosos. O SIAMI auxiliará os gestores a decidir quando, onde e por que serão realizadas visitas ou campanhas de saúde. Ele possui diversas características englobadas pela definição de Sistemas de Informação (SASS et al., 2012).

A informação produzida pelo SIAMI será utilizada pelas equipes de saúde. Ele é um sistema de apoio à decisão, pois, gera indicativos que podem auxiliar os tomadores de

decisão. Também é um sistema especialista, possuindo em sua base de conhecimento as definições e regras de um ou mais especialistas na área que compõe o GPENSI.

2.5 Trabalhos relacionados

Nesta seção são apresentados alguns trabalhos realizados que abordam de forma geral os SIGs, além de relacionar SIG e saúde, SIG e *Web* e exemplos de SIG com outras aplicações.

Primeiramente, seguem os trabalhos que relacionam os SIGs e a saúde. Santana, Costa e Loureiro (2014), realizaram um estudo de caso no município de Amadora em Portugal, que envolveu a aplicabilidade de um SIG em escala local para monitorar aspectos de saúde da população. Neste trabalho os autores buscaram referências acadêmicas que relacionavam o processo de urbanização com a saúde, considerando qualidade de vida urbana e saúde associada ao planejamento urbano.

Discutiram os contextos e seus ambientes que envolviam o processo de associar saúde ao planejamento urbano. Os ambientes destacados foram: ambiente físico, social, econômico e cultural. Fatores como poluição, qualidade da água, capital social ou as oportunidades locais (bens e serviços) são alguns desdobramentos que podiam ser obtidos de cada um dos ambientes considerados. O planejamento de lugares considerados saudáveis deviam envolver todos os fatores e ambientes apresentados. Os autores levantaram conteúdos acadêmicos sobre as capacidades do SIG em conseguir correlacionar os fatores e ambientes para medição, avaliação e monitorização que possuíam impacto no bem-estar e na saúde individual e comunitária. O objetivo do trabalho foi utilizar um SIG para correlacionar os ambientes urbanos com a saúde e apresentaram três indicadores que foram analisados utilizando SIG.

Os três indicadores:

- Os SIG na avaliação do Ambiente Físico: criado para avaliar as condições do meio envolvente (natural e construído) ao qual a população está exposta. Este indicador refletia condições do tipo: declividade do terreno, proximidade a linhas de alta tensão, proximidade a linhas de transporte, entre outros.
- Os SIG na avaliação das condições sociodemográficas e estilos de vida: elaborado para avaliar a constituição sociodemográfica e preponderância de grupos de indivíduos. Um indicador apresentado neste trabalho e considerado pelos autores como grande impacto foi o indicador relacionado ao sentimento de insegurança.
- Os SIG na avaliação das condições de vida: analisou-se condições básicas de saúde, incorporando aspectos que envolviam a identificação de áreas e situações que apresentavam desenvolvimento de doenças e acidentes. Um exemplo de indicador apresentado por Santana, Costa e Loureiro (2014) relacionava a acessibilidade geográfica aos equipamentos e serviços de saúde.

A escolha do município de Amadora como estudo de caso de acordo com Santana, Costa e Loureiro (2014), ocorreu por causa das altas disparidades observadas no município.

Outro trabalho relacionando SIG e saúde foi apresentado na dissertação de mestrado de Lopes (2005), mostrando um exemplo de utilização de um SIG que relaciona causas geográficas a saúde e bem-estar do ser humano. Em sua dissertação Lopes (2005), realizou um estudo que teve por objetivo investigar se a queima de palha de cana-de-açúcar estava correlacionada aos problemas de saúde enfrentados pelas populações no período de queima, além de, explicitar os benefícios da utilização de um SIG para realizar a investigação.

Neste próximo exemplo de aplicação do SIG e saúde, Hino (2007) em sua tese de doutorado apresentou um estudo que relacionou o Indicador Adaptado de Condição de Vida e a distribuição espacial e temporal da ocorrência de Tuberculose compreendida nos anos 2000 a 2006, no município de Ribeirão Preto, São Paulo. Destaca-se deste material um tema importante para justificar este trabalho de conclusão de curso, a utilização de conceitos e ferramentas que envolvem SIG para realizar análises espaciais sobre dados que foram georreferenciados. Especificamente em (HINO, 2007), após o georreferenciamento dos dados (utilizando critérios do autor), foram obtidos padrões de densidade espaciais por meio de análise estatística espacial. Vale ressaltar que nesta tese de doutorado a análise espacial serviu de apoio ao estudo, não sendo necessariamente o foco.

Mais um exemplo é o estudo apresentado por Cianci (2006), o qual utilizou um SIG que correlacionava dados do Índice de Desenvolvimento Humano (IDH), Aids e ocorrências de apreensões de tráfico de entorpecentes pela Polícia, no estado de São Paulo, entre 1997 e 2004. Os resultados obtidos foram expressos em mapas temáticos gerados pelo SIG. Os mapas gerados apresentavam temas como: a distribuição do IDH no estado, a distribuição regional da incidência de Aids pelo compartilhamento de agulhas/seringas contaminadas e os municípios em que houveram apreensões de entorpecentes e mapas que continham a sobreposição de todos estes mapas.

O último trabalho relacionando SIG e saúde foi apresentado em um artigo publicado por Almeida et al. (2009) na Revista Brasileira de Cartografia, mostrou um estudo realizado no município de Vitória no estado do Espírito Santo. Este estudo abordou casos de hepatite A e E ocorridos no período de 1999 a 2005. Os autores deste artigo utilizaram técnicas de SIG para gerar mapas que mostravam a distribuição espacial da hepatite, com o objetivo de analisar, discutir e apontar possíveis causas. Destaca-se deste artigo um interessante tópico abordado pelos autores, a geografia da saúde. Este tópico apresentou o levantamento teórico a respeito da aplicação da geografia na área de saúde.

Os próximos trabalhos relacionados apresentam o desenvolvimento de SIG para a *Web*. O primeiro de Medeiros et al. (2013), mostra a construção de um SIG *Web* utilizando softwares livres para disponibilizar dados georreferenciados para o governo e população do estado do Ceará, fornecendo dados socioeconômicos e cartográficos para a sociedade e oferecendo informações aos gestores públicos no planejamento de políticas públicas e

na tomada de decisão. Este sistema disponibilizado na internet possuía certas funcionalidades, como: consulta e análise de dados georreferenciados, manipulação interativa dos elementos que compõem um mapa (manipulação de temas e legendas) e visualização do mapa.

Outro trabalho envolvendo SIG *Web* foi desenvolvido por Camargo (2005), um SIG *Web* foi criado para gerenciar e visualizar projetos de Agricultura de Precisão por parte de usuários especializados e não especializados. Para os usuários especializados Camargo (2005) concluiu as seguintes funcionalidades:

- criação e edição de novos projetos;
- gerenciamento de mapas de talhões;
- gerenciamento de mapas vetoriais ou raster;
- gerenciamento de usuários para acesso aos projetos;

Enquanto que para os usuários não especializados, o autor concluiu as seguintes ações:

- visualização de mapas e imagens;
- funcionalidades de visualização e inclusão de dados relacionados aos talhões;

A seguir são apresentados dois trabalhos relacionados a SIGs em outras áreas de aplicação. O primeiro, em sua dissertação de mestrado, Gonçalves (2005), apresenta um estudo realizado utilizando um SIG para apoiar o processo de decisão. Neste estudo o objetivo do SIG foi de informatizar o sistema de localização de pontos de abastecimento de água do corpo de bombeiros. O ponto a ser destacado deste trabalho é a capacidade do SIG auxiliar no processo de decisão, neste caso, melhorando o tempo para a localização de pontos de abastecimento e auxílio ao bombeiro em decidir qual o melhor local para abastecimento da viatura, quando a mesma estiver atendendo uma ocorrência de incêndio.

No segundo, os autores (FILHO; SILVA, 2010), realizaram um trabalho onde utilizaram um SIG para realizar a gestão da arborização urbana com o objetivo de avaliar as possibilidades de uso desta ferramenta no município de Guarapuava, Paraná. Criando a partir do SIG um inventário (informações históricas) e diagnósticos das árvores plantadas naquele município, facilitando o manejo. Este trabalho apresentado pelos autores está relacionado a este trabalho de conclusão de curso, pelo fato de utilizarem um SIG para a gestão de indivíduos (em Filho e Silva (2010), os indivíduos são as árvores enquanto que neste trabalho de conclusão de curso os indivíduos são os idosos).

O objetivo dessa seção foi apresentar alguns trabalhos relacionados ao tema geral do presente trabalho. Destacando a importância que a utilização dos SIGs na gestão dos dados, que antes eram puramente descritivos. Os SIGs têm sido utilizados nas mais diversas áreas da ciência, seu uso na área da saúde vem surgindo aos poucos, mas já

tem demonstrado sua importância na gestão pública da saúde. Acrescentando as suas características ao desenvolvimento e a popularização da *Web* fornecem mais recursos aos gestores.

3 METODOLOGIA

O presente trabalho é do tipo de pesquisa tecnológica, subdividido em duas principais atividades: pesquisa dos fundamentos teóricos e pesquisa de desenvolvimento tecnológico.

Na atividade de pesquisa dos fundamentos teóricos foram consultados livros e artigos sobre a área de SIG para obter uma compreensão dos conceitos que estão envolvidos com este trabalho e que foram redigidos no capítulo 2. Estes conceitos são importantes, pois, guiaram o desenvolvimento do trabalho de uma forma que respeitasse ao máximo os conceitos que envolvem SIG, desta forma garantindo que o resultado a ser buscado estivesse correto. Com o entendimento dos conceitos partiu-se para a pesquisa de quais tecnologias seriam utilizadas para a implementação do trabalho. Em um primeiro momento, como o presente trabalho está relacionado a um outro trabalho em desenvolvimento (neste caso o SIAMI *Web*), houve a necessidade de verificar se as tecnologias que já estavam sendo utilizadas ofereciam suporte ou funcionalidades que abrangiam a ideia deste trabalho. Portanto, pesquisas sobre estas tecnologias foram realizadas com o intuito de descobrir se estas ofereciam o suporte necessário.

O SIAMI *Web* está sendo desenvolvido utilizando o *framework* de desenvolvimento *web* Django com o banco de dados PostgreSQL. Através das pesquisas descobriu-se que o Django possui uma extensão denominada GeoDjango, a qual oferece possibilidade de se trabalhar com dados geográficos. Outro componente importante como o banco de dados deveria fornecer estrutura para se trabalhar com dados geográficos. Desta forma, pesquisas relacionadas ao banco de dados PostgreSQL foram realizadas e obteve-se a constatação de que esta tecnologia oferece uma extensão com o nome de PostGIS. PostGIS é uma extensão do PostgreSQL a qual implementa as definições da OGC para se trabalhar com os dados geográficos.

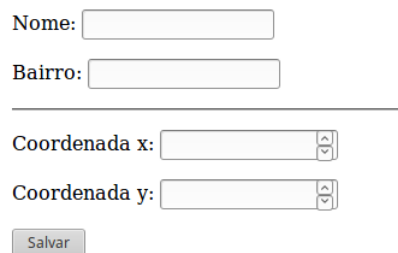
Em relação a pesquisa de desenvolvimento tecnológico, esta foi uma atividade que envolveu os estudos de como tornar real e funcional, através dos conceitos adquiridos, o presente trabalho. Para este fim foi utilizado um ambiente de testes que teve o objetivo de servir como um experimento para as tentativas de implementação e compreensão das tecnologias. Neste ambiente foram instalados e configurados o GeoDjango e o PostgreSQL com o PostGIS além do desenvolvimento de uma aplicação bem simples.

Nesta pequena aplicação foram feitos e testados a inserção de dados não geográficos associados com dados geográficos, além da apresentação em mapa em uma página *web* das informações geográficas. Para a utilização de representações em mapas interativos em uma página *web*, foi utilizada uma biblioteca JavaScript denominada Leaflet (LeafletJs). Antes de adotar o Leaflet como a biblioteca a ser utilizada, outras bibliotecas foram pesquisadas, como por exemplo, a biblioteca OpenLayers. Adotou-se o Leaflet por questões de simplicidade em atender as devidas necessidades atuais.

A partir de tudo isso foi elaborado um modelo para a apresentação dos dados sobre a vulnerabilidade do idoso no mapa.

4 RESULTADOS

Os primeiros resultados do desenvolvimento tecnológico foi uma pequena aplicação desenvolvida com o intuito de realização de testes, consistia em um primeiro momento, de apresentar um formulário com dois campos de valores não geográficos e dois campos de valores geográficos, os quais consistiam nas coordenadas x e y do ponto geográfico, apresentado pela figura 3.



Nome:

Bairro:

Coordenada x:

Coordenada y:

Figura 3 – Formulário de teste de inserção de dado geográfico

Este formulário serviu para testar a função de armazenar as coordenadas geográficas como um ponto geográfico no banco de dados. Os pontos informados no formulário são armazenados no banco de dados através de uma função que transforma o sistema de referência de coordenadas do ponto, no sistema com *datum* SIRGAS 2000, com o sistema projetado de coordenadas UTM Zone 21 Sul. Para realizar os testes com este formulário foi utilizado um aparelho GPS configurado com o *datum* WGS84 com o sistema projetado de coordenadas UTM Zone 21 Sul. Com o GPS, pontos de exemplos foram coletados para simular a coleta de dados. Com o intuito de verificar se os pontos coletados correspondem a posição correta no mapa ou globo foi utilizado o software QuantumGIS (QGIS) enquanto a função de plotagem em um mapa na *web* ainda não estava pronta. Para a utilização deste software, o autor, realizou um curso *on-line* para utilização básica do QGIS. Com um projeto do QGIS devidamente configurado, a verificação se deu através de sua interface que apresenta as camadas vetoriais e *raster* mostradas na figura 4. A camada *raster* do mapa da cidade de Dourados foi gerado pelo OpenStreetMap. O QGIS possui a funcionalidade de buscar camadas vetoriais através de uma conexão com o banco de dados. Desta forma a visualização apresentada são pontos buscados no banco de dados de exemplo no PostgreSQL.

Após esta etapa concluída, os testes se voltaram em como apresentar um mapa em uma página *web* com os pontos plotados derivados do banco de dados. Neste teste foram consultadas as documentações referentes ao Leaflet e GeoDjango para compreender como seria realizada a transferência de dados entre a aplicação GeoDjango (*Backend*) e a página HTML que irá conter o mapa gerado pelo Leaflet (*Frontend*). Consultando a documenta-

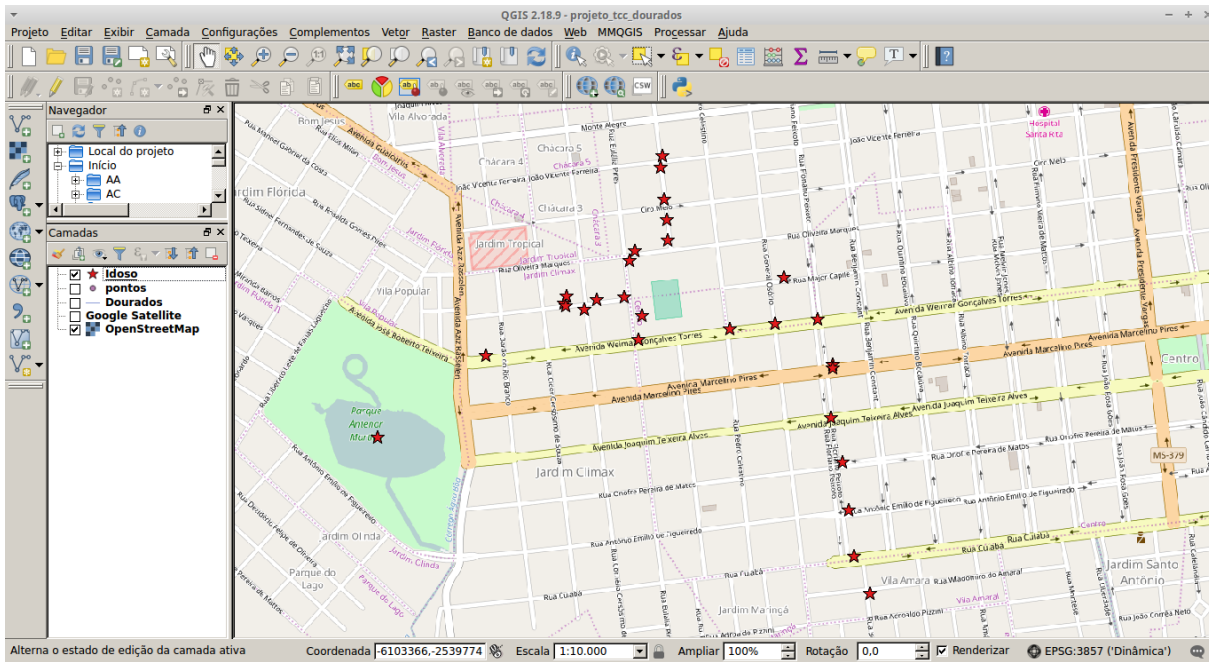


Figura 4 – Interface do QuantumGIS

ção do Leaflet descobriu-se que esta biblioteca possui funcionalidades que trabalham com um formato de estruturação de dados denominado GeoJSON. GeoJSON é um formato de estruturação de dados semelhante ao JSON com o acréscimo de oferecer funcionalidades para estruturar dados geográficos. Outra descoberta importante está relacionado ao GeoDjango, o qual disponibiliza uma funcionalidade para estruturar as instâncias recuperadas do banco de dados no formato GeoJSON, possibilitando a comunicação entre o *Backend* e o *Frontend*. Com esta questão resolvida, consultas a documentação do Leaflet foram realizadas para primeiramente gerar o mapa interativo com a camada raster do OpenStreetMap. Após isto, seguindo ambas as documentações, os pontos foram plotados no mapa.

Uma grande dúvida surgiu a mente do autor deste trabalho. Qual o sistema de referência de coordenadas o Leaflet estava utilizando? Consultando a documentação observou-se que o Leaflet opera com poucos sistemas de referência de coordenadas. O padrão utilizado (*default*) é o EPSG:3857 que é o código que se refere ao *datum* WGS84 com o sistema projetado de coordenadas Pseudo Mercator. A função do GeoDjango que estrutura os dados possui a capacidade de antes de estruturar os dados realizar transformações de um sistema de referência de coordenadas para outro. Quando se usa esta função sem definir um código de referência de coordenadas, esta assume o padrão (*default*), o EPSG:4326, que é apenas o *datum* WGS84. Com isto, os dados armazenados no banco de dados com o sistema de referência SIRGAS 2000/UTM Zone 21 Sul eram transformados para o sistema WGS84. Quando o Leaflet plotava os pontos no mapa, ele antes, realizava transformações para o sistema de referência que ele utilizava. Desta forma fazendo com que os pontos aparecessem na posição correta. Responder a este questionamento foi de vital importân-

cia, pois, desta forma compreendeu-se o que deveria ser feito antes de gerar os pontos no mapa.

Outra dúvida levantada estava relacionada a utilização de símbolos customizáveis para representar os pontos no mapa. O estudo realizado na documentação do Leaflet foi o suficiente para conseguir utilizar estes tipos de símbolos.

Com todo este conhecimento adquirido, foi proposto um tipo de relatório para o SIAMI *web* que deveria ser gerado contendo um mapa apresentando pontos. Este mapa deveria apresentar os idosos como pontos, para cada ponto deveria ser utilizado uma cor que representa o indicador de saúde do idoso. As cores foram definidas da seguinte forma:

- Verde: condição boa.
- Amarelo: condição razoável.
- Vermelho: condição ruim.

Além do mais, este relatório deveria incluir a dimensão tempo, ou seja, foi determinado que seriam gerados no máximo três pontos para cada idoso (derivados das três últimas entrevistas realizadas com o idoso). Cada ponto além de possuir uma cor, possuiria uma das seguintes formas:

- Círculo: última entrevista realizada com o idoso (mais atual).
- Triângulo: penúltima entrevista realizada com o idoso.
- Quadrado: antepenúltima entrevista realizada com o idoso (menos atual).

Sendo assim, a figura 5, apresenta todas as possibilidades de ícones para serem utilizados na representação.

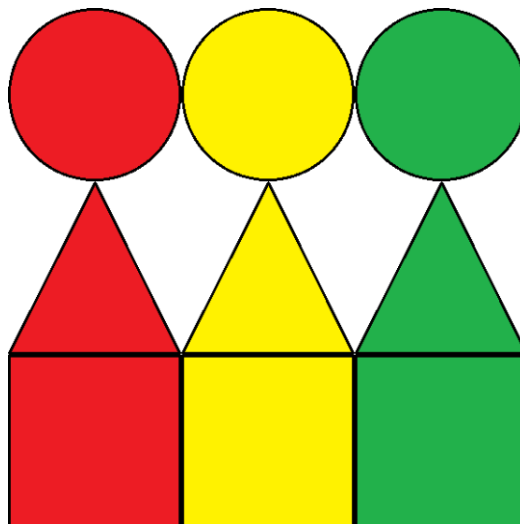


Figura 5 – Conjunto de símbolos

A ideia de utilizar ícones com formas diferentes é para que quando as três últimas entrevistas com o idoso se referirem a uma mesma posição geográfica (o idoso não se mudou no intervalo de tempo entre essas entrevistas), seja possível identificar as três representações além de suas respectivas cores. Isto é necessário porque os símbolos se sobrepõem. A informação a respeito de sobreposição é muito importante, pois isto afeta na estratégia de processamento que será apresentada a seguir.

Como existe o problema de sobreposição de ícones no mapa, a estratégia construída parte do princípio de plotar primeiramente os pontos referentes as antepenúltimas entrevistas dos idosos (quadrados), depois, plotar os pontos das penúltimas entrevistas (triângulos) e por fim plotar os pontos das últimas entrevistas (círculos). As cores que cada forma irá assumir serão decididas no *script* que gera o mapa. Para definir a cor, um valor associado aos dados geográficos do idoso será comparado com os seguintes números:

- 1, se for igual, o ícone verde daquela forma será plotada.
- 2, se for igual, o ícone amarelo daquela forma será plotada.
- 3, se for igual, o ícone vermelho daquela forma será plotada.

Para que isto seja possível os dados buscados no banco de dados devem ser organizados e processados para que o processamento no navegador seja linear. Neste caso, haverá três pares de listas que estão associadas. Cada par corresponde a uma das seguintes organizações:

- Par antepenúltimo. Contém apenas informações das antepenúltimas entrevistas.
- Par Penúltimo. Contém apenas informações das penúltimas entrevistas.
- Par Último. Contém apenas informações das últimas entrevistas.

Cada par consiste em duas listas, uma armazena um valor que varia de um a três que é resultado do processamento da entrevista e a outra armazena os dados da entrevista e do ponto geográfico.

Estes pares de listas são geradas no *Backend* da seguinte forma:

1. um conjunto maior de entrevistas de idosos é gerado por uma busca no banco de dados e é retornado ordenado pelo número que identifica o idoso e pela data da entrevista de forma decrescente.
2. utiliza-se um procedimento que pega apenas as três últimas entrevistas do idoso deste conjunto maior e as divide em no máximo em outras três listas.

3. cada vez que uma instância de entrevista é adicionada, é realizado o cálculo que gera o valor do indicador e este valor é automaticamente adicionado em uma lista paralela que vai aumentando de tamanho juntamente e na mesma ordem que a lista que armazena as informações da entrevista.
4. por fim, a função que gera GeoJSON, é aplicada sobre as listas que contém as entrevistas e as listas pares são mandadas para o navegador que realiza o processamento final.
5. o processamento final no navegador se resume a percorrer as listas utilizando o ponto geográfico de cada posição para saber onde plotar e o valor do indicador para selecionar a cor do ícone.

Estes conceitos e estratégias foram desenvolvidos no ambiente de testes que gerou os mapas apresentados pelas figuras 6, 7, 8, 9, e que foram aplicados no SIAMI *web*.

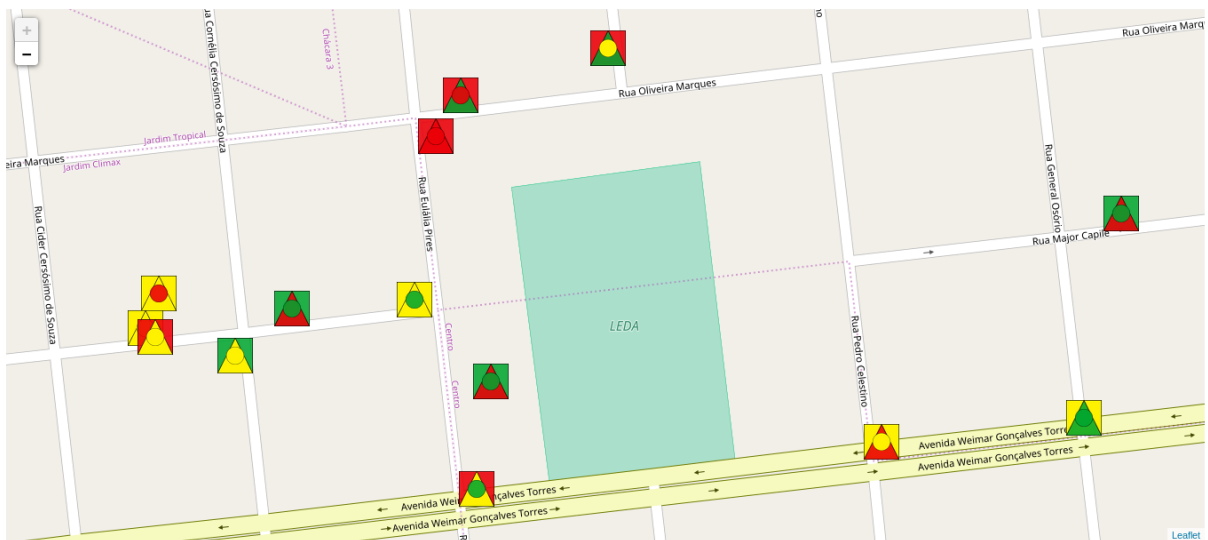


Figura 6 – Mapa mostrando uma pequena região de Dourados

O mapa da figura 6, mostra uma pequena região da cidade de Dourados. O nível de *zoom* utilizado para focar a região foi o máximo. Os pontos plotados apresentam as três formas (três últimas entrevistas). A cor de cada forma foi gerada aleatoriamente para fins de testes.

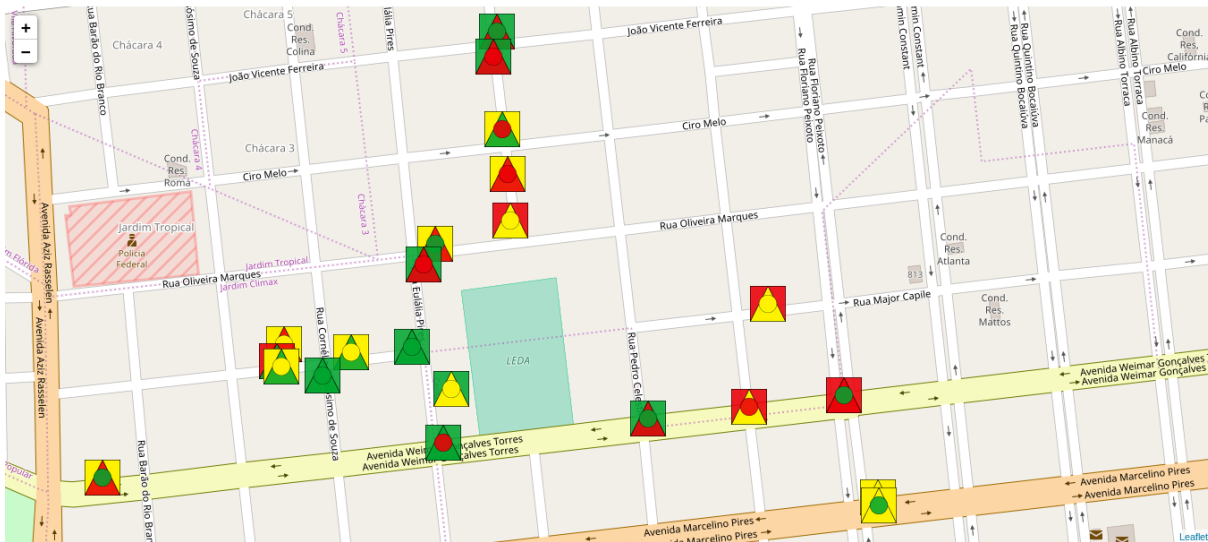


Figura 7 – Mapa mostrando uma região maior de Dourados

O mapa da figura 7, apresenta um nível de visualização maior, consequentemente mostrando mais pontos em uma região um pouco maior.

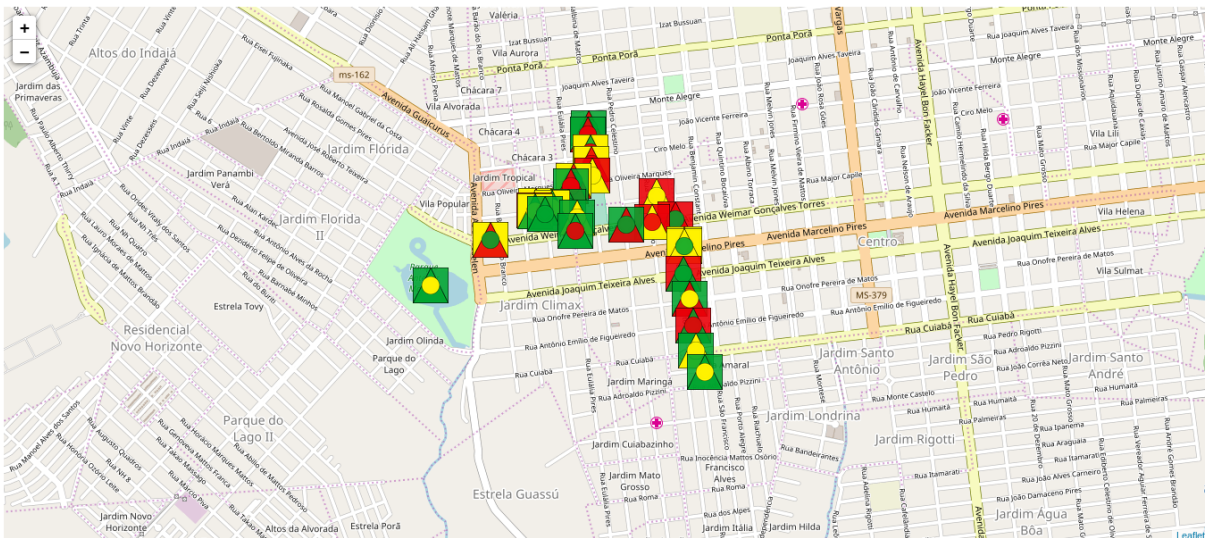


Figura 8 – Mapa apresentando um nível menor de *zoom*

A figura 8, mostra uma região maior da cidade de Dourados. Praticamente todos os pontos de testes conseguem ser visualizados.

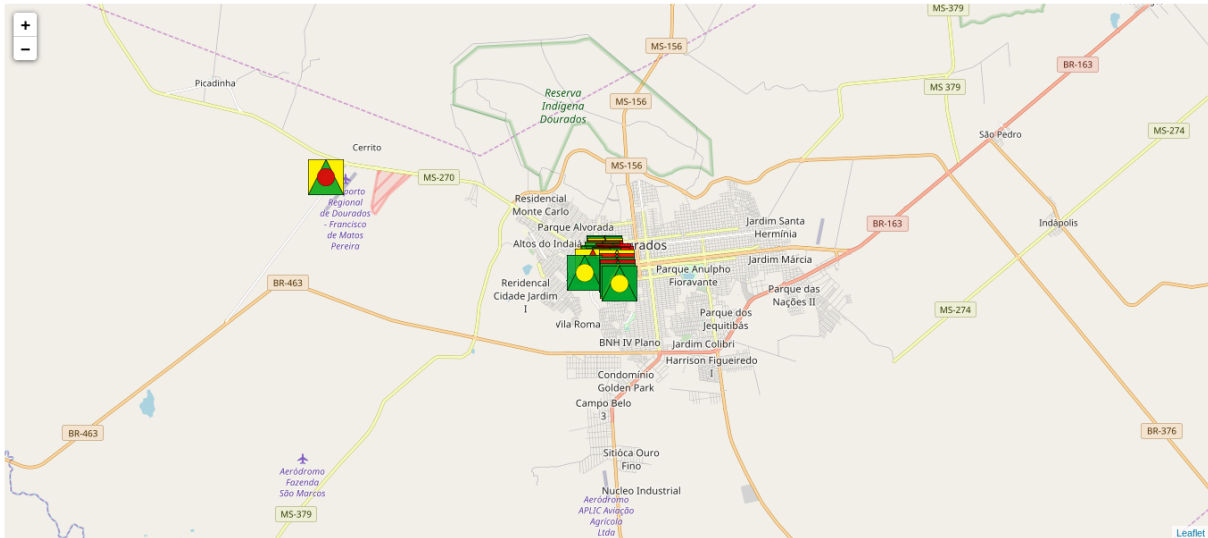


Figura 9 – Cidade de Dourados

Este mapa, figura 9, apresenta a visualização total da cidade de Dourados. Este tipo de visualização permite observar a cidade como um todo.

Como estes conceitos e estratégias foram aplicados ao SIAMI *web*, se faz necessário a descrição dos passos que possibilitaram tal feito.

Em virtude do SIAMI *web* estar sendo desenvolvido com o *framework* Django, o primeiro passo foi configurar o projeto e instalar bibliotecas para que o GeoDjango pudesse ser ativado e conseqüentemente funcionasse de forma correta. Para tanto foi seguida a documentação disponível no site do Django (FOUNDATION, 2017a; FOUNDATION, 2017c).

Antes que as bibliotecas pudessem ser instaladas foi recomendado a instalação das seguintes dependências: binutils, libproj-dev e gdal-bin.

Foram instaladas as seguintes bibliotecas em ordem:

- GEOS: biblioteca para operações geométricas.
- PROJ.4: biblioteca para converter dados geoespaciais para diferentes sistemas de referência de coordenadas.
- GDAL: biblioteca para ler dados vetoriais e *raster*, a grande maioria dos tipos.

Após a instalação das dependências e bibliotecas o próximo passo a ser seguido foi a configuração do banco de dados. Como o SIAMI *web* está utilizando o SGBD PostgreSQL e o *framework* Django possui uma maior compatibilidade com este banco de dados então a configuração a ser feita foi adicionar o PostGIS ao PostgreSQL e ativar a extensão na base de dados que se queira trabalhar. Nesta ocasião o PostGIS foi instalado ao PostgreSQL e a base de dados que estava sendo utilizada teve a extensão adicionada utilizando o seguinte comando SQL: *create extension postgis;*

Em relação a aplicação foram realizadas as seguintes modificações:

No arquivo de configurações do projeto (*settings.py*) foi adicionado a linha que contém a instrução 'django.contrib.gis' no *installed_apps*, apresentado pela figura 10.

```
INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'siamiant',
    'siami_relatorios',
    'siami_telas_entrada',
    'siami_site',
    'django_extensions',
    'siami_fatores',
    'django.contrib.gis',
    'relatorios_geograficos',
]
```

Figura 10 – Aplicativos instalados

Esta instrução faz com que GeoDjango seja inserido no projeto. Outra alteração importante nas configurações se refere a *engine* utilizada para realizar a comunicação com o banco de dados geográfico. A *engine* utilizada foi a 'django.contrib.gis.db.backends.postgis' responsável pela comunicação.

Após estas etapas de configurações foi necessário decidir em qual tabela do banco de dados o dado geográfico seria armazenado. Nesta situação a tabela Moradia foi selecionada, pois, fazia muito mais sentido armazenar a posição geográfica em uma tabela que está armazenando dados relacionados a moradia do idoso. A alteração foi realizada na classe Moradia no arquivo de modelos (*models.py*) que o Django utiliza para mapear o banco de dados, sendo destacado na figura 11.

```
idmoradia = models.AutoField(primary_key=True)
idinstrumentado = models.ForeignKey('Instrumentado', m
rua = models.CharField(max_length=45, blank=True, nu
numero = models.IntegerField(blank=True, null=True,v
bairro = models.CharField(max_length=45, blank=True,
possemoradia = models.IntegerField(blank=True, null=
condicaomoradia = models.IntegerField(blank=True, nu
abastecimentoagua = models.IntegerField(blank=True,
dejetos = models.IntegerField(blank=True, null=True,
lixo = models.IntegerField(blank=True, null=True, chc
data = models.DateField(blank=True, null=True)

ponto = models.gis.PointField(srid=31981, null=True)

class Meta:
    db_table = 'moradia'
```

Figura 11 – Fragmento da classe Moradia

Concluído a modificação no banco de dados o próximo objetivo era alterar a tela do sistema responsável por captar, através de um formulário, os dados referentes a moradia. A figura 12, apresenta com um destaque na cor vermelha, os dois campos responsáveis por receber os valores de coordenadas x e y do ponto geográfico.

Moradia

Rua: Número: Bairro:

Posse residência:

Condições moradia:

Abastecimento de água:

Destino dejetos:

Destino lixos:

Coordenada x:

Coordenada y:

Figura 12 – Tela que capta os dados sobre moradia

Os campos destacados recebem valores reais. Para que estas duas coordenadas gerem um ponto que será salvo no banco de dados, foi necessário processar estes dados colocando os em um formato específico apresentado pela figura 13.

```
obj_moradia.ponto = 'POINT('+str(ponto.cleaned_data['coordenada_x'])+' '+str(ponto.cleaned_data['coordenada_y'])+')'
```

Figura 13 – Trecho de código (*Python*) que apresenta o processamento

Este trecho de código produz uma *string* (cadeia de caracteres) que se assemelha a esta *string*: 'POINT(-95.3385 29.7245)'. Dentro dos parênteses, o número a esquerda corresponde a coordenada x e o número a direita corresponde a coordenada y. Toda esta cadeia de caracteres é um formato conhecido como *Well Known Text*(WKT), um padrão proposto pela OGC para representar uma geometria. Neste caso a geometria a ser representada é um ponto (FOUNDATION, 2017b). Após, a *string* foi atribuída ao campo ponto do objeto `obj_moradia` que foi salvo no banco de dados.

A utilização da tela para cadastro dos dados referentes a moradia serve apenas para novas entrevistas que serão realizadas com os idosos. Um problema surgiu ao adicionar um campo ponto geográfico ao banco de dados, pois, na tabela moradia já existiam instâncias que eram resultados de cadastros feitos anteriormente a aplicação da funcionalidade geográfica ao SIAMI *web*. Desta forma estas instâncias ficavam com o campo ponto geográfico vazio e não haviam telas para alteração destes dados. Neste caso o autor deste

trabalho desenvolveu um pequeno *script* em *python* para adicionar ou até mesmo alterar o ponto geográfico. O *script* completo está disponível no apêndice A.

Utilizando o *shell* do *python* no projeto Django através do comando, *python manage.py shell*, são necessários os seguintes argumentos: o CPF em formato *string* e a data da entrevista no formato 'aaaa-mm-dd' (sendo 'a' o ano, 'm' o mês e 'd' o dia). Com estes argumentos o *script* tentará acessar a instância da tabela Moradia que corresponda ao idoso e sua entrevista. Se tudo estiver correto deverá ser informado a coordenada x e após a coordenada y. Desta forma o ponto geográfico pode ser alterado ou corrigido ou até mesmo inserido caso a instância não possua este dado.

Encerrado o assunto sobre a inserção ou alteração dos pontos geográficos, foi sugerido inicialmente a produção de três relatórios geográficos. Estes relatórios deveriam apresentar as seguintes características:

- apresentar informações sobre as três últimas entrevistas de cada idoso.
- possuir um painel de filtros que afetará as entrevistas selecionadas.
- apresentar uma tabela que correlacione os indicadores verde, amarelo e vermelho a temporariedade das entrevistas.
- apresentar um gráfico que faça a mesma correlação da tabela.
- apresentar o mapa interativo da cidade de Dourados contendo os pontos com suas respectivas formas e cores, obviamente relacionados aos idosos considerados no relatório.

No fim das contas foram produzidos cinco relatórios que serão apresentados mais abaixo no texto. Estes cinco relatórios são:

- relatório renda per capita
- relatório faixa etária
- relatório escolaridade
- relatório condição de moradia
- relatório diagnóstico médico

Dentre os quatorze indicadores do SIAMI foram selecionados cinco. Cada indicador corresponde a um dos relatórios listados logo acima. Como esses relatórios deveriam apresentar suas informações de uma mesma forma (definida logo acima no texto) então foi utilizado uma estratégia que aproveita uma mesma tela que pode apresentar as informações para cada relatório sem a necessidade de programar a mesma tela para cada um.

Figura 14 – Painel contendo os filtros

A figura 14 mostra o painel de filtros utilizado pelos relatórios.

Neste painel de filtros é possível definir restrições sobre o conjunto de entrevistas que serão selecionadas para compor os relatórios. Se nenhum filtro for selecionado ou preenchido então todas as entrevistas presentes no banco de dados serão selecionadas. O filtro Equipe de Saúde da Família corresponde aos idosos que são monitorados por uma equipe de saúde. Nesta ocasião pode se informar o número da equipe e então as entrevistas serão selecionadas caso corresponda a um idoso sendo monitorado por essa equipe. Outro filtro que pode ser utilizado corresponde ao período de tempo. É possível escolher entrevistas considerando apenas um mês, apenas um ano ou um intervalo de tempo. Este intervalo de tempo corresponde a um ano e mês iniciais e um ano e mês final. No filtro de período as opções são exclusivas, mas é possível utilizar o filtro de equipe juntamente com um filtro de período.

Os códigos em *python* utilizados no processamento para gerar os relatórios estão disponíveis de forma completa no apêndice B. Cada relatório utiliza uma função diferente responsável por gerar o valor do indicador. Os relatórios possuem suas próprias funções que utilizam a função geral *relatorio_geografico_temporal* passando como argumentos um objeto requisição (contém informações a respeito da requisição do navegador) e uma função que será utilizada para gerar os valores do indicador para o relatório em questão. Esta função passada como argumento nada mais é do que a função de indicador utilizada para gerar o valor de indicador. Então mudando a função sendo passada é possível obter um reaproveitamento de código. Pelo apêndice B:

- relatório renda *per capita*, apêndice D: este relatório utiliza a função *relatorioGeograficoRendaPerCapita* e passa para a função *relatorio_geografico_temporal* sua função de indicador *i3_Renda*. Para que este relatório possa ser gerado são necessárias as tabelas Social e Salariominimo utilizadas pela função indicador. Este relatório necessita da tabela Salariominimo pois o indicador de renda é baseado no salário mínimo corrente no ano em que houve a entrevista com o idoso.
- relatório faixa etária, apêndice E: este relatório utiliza a função *relatorioGeograficoFaixaEtaria* e passa para a função *relatorio_geografico_temporal* sua função de

indicador *i1_Faixaetaria*. Para que este relatório possa ser gerado é necessário a tabela Idoso que contém a data de nascimento.

- relatório escolaridade, apêndice F: este relatório utiliza a função *relatorioGeograficoEscolaridade* e passa para a função *relatorio_geografico_temporal* sua função de indicador *i2_Escolaridade*. Para que este relatório possa ser gerado é necessário a tabela Social que contém o dado relacionado ao tempo de estudo do idoso.
- relatório condição de moradia, apêndice G: este relatório utiliza a função *relatorioGeograficoCondicaoMoradia* e passa para a função *relatorio_geografico_temporal* sua função de indicador *i4_Condicaomoradia*. Para que este relatório possa ser gerado é necessário a tabela Moradia.
- relatório diagnóstico médico, apêndice H: este relatório utiliza a função *relatorioGeograficoDiagnosticoMedico* e passa para a função *relatorio_geografico_temporal* sua função de indicador *i8_Diagnostico*. Para que este relatório possa ser gerado é necessário a tabela Diagnostico.

Relacionados aos relatórios, os mesmos possuem algumas restrições:

- se a entrevista não possui uma relação com uma instância de moradia (indicando que a entrevista feita com o idoso a tela de moradia não foi respondida) então esta entrevista é desconsiderada para o relatório.
- a quantidade de idosos georreferenciados é contabilizado.
- se a entrevista não possui os dados para o indicador realizar o cálculo então a mesma é desconsiderada.
- a quantidade de idosos envolvidos no relatório é contabilizado.

Por conta do banco de dados do SIAMI *web* que não foi povoado totalmente pela equipe de coleta de dados, foi então necessário criar um conjunto de dados de teste para apresentar os resultados do trabalho. As ações a seguir foram necessárias:

- selecionar um conjunto de cinco de idosos que possuem dados relacionados a endereço para que seja possível obter o ponto geográfico através de um mapa interativo.
- para cada idoso selecionado seus dados foram duplicados através da utilização de um *script* em *python* (apêndice C), sendo que alguns campos foram alterados para simular mudanças entre uma entrevista e outra. O resultado deste processo fez com que cada idoso tivesse em teoria três entrevistas realizadas.

- para obter o ponto geográfico foi necessário a utilização do GoogleMaps o qual fornecia as coordenadas geográficas do endereço. Com estas coordenadas foi preciso realizar uma calculadora de coordenadas para converter as coordenadas geográficas para o sistema SIRGAS 2000/UTM Zone 21 Sul.

Apesar dos dados utilizados serem fictícios a estratégia pensada e desenvolvida para a geração dos relatórios geográficos considera dados reais. A necessidade de gerar dados fictícios possui a justificativa para demonstrar o funcionamento do trabalho desenvolvido.

5 CONCLUSÃO

Os mapas interativos que foram gerados possibilitam uma visão alternativa das informações que o SIAMI *web* pode produzir. Para que pudesse ser possível alcançar este objetivo, gastou-se uma parcela de tempo considerável para compreender a dimensão da área acadêmica em que este trabalho está situado. Os resultados obtidos são os artefatos que foram visualizados como uma forma de aplicação da teoria do SIG. No processo de conquistar os objetivos definidos neste trabalho, houveram diversas descobertas intelectuais no decorrer da pesquisa até o momento em que era possível transpor para a realidade a ideia. As descobertas e novos aprendizados gerados por esta atividade, agregam valor ao profissional.

Este trabalho de um modo geral se propôs a abrir um novo horizonte a ser explorado pelo projeto SIAMI *web*. O objetivo de agregar funcionalidades geográficas ao projeto foi alcançado. Atingir este objetivo não foi uma tarefa simples e exigiu muito trabalho e diversos cuidados para que o resultado gerado possa ser muito bem aproveitado futuramente.

No decorrer do desenvolvimento deste trabalho diversos desafios apareceram. Compreender a teoria que envolve SIG sendo um tema que está fora do alcance proposto pelo curso de Ciência da Computação, foi um destes desafios. Os outros desafios estavam relacionados a utilização das tecnologias para a executar o trabalho. Em relação as tecnologias empregadas, a facilidade em fazer com que estas tecnologias operassem juntas foi um dos grandes marcos que fizeram com que o trabalho pudesse seguir um fluxo um pouco mais tranquilo no desenvolvimento.

A produção dos relatórios apresentados nos apêndices e que foram resultados deste trabalho, somam para as ciências envolvidas. As áreas de análise visual, georreferenciamento e análise espacial encontram neste projeto mais um exemplo de aplicação prática, sendo este feito um grande ponto positivo.

Para projetos futuros o grande objetivo a ser atingido agora é conseguir utilizar o que foi proposto e construído neste trabalho para compor trabalhos mais complexos, e que consigam explorar mais afundo os recursos disponibilizados pelas tecnologias empregadas. A utilização dos recursos para trabalhos que envolvam análises e estatísticas é uma recomendação proposta por mim, o autor.

Enfim, este trabalho proporcionou diversos momentos de altos e baixos, que no fim das contas, somaram para o enriquecimento intelectual, teórico e profissional.

REFERÊNCIAS

- ALMEIDA, R. D. de et al. Uso de técnicas de sistemas de informações geográficas (sigs) no mapeamento da hepatite para o município de vitória, es. *Revista Brasileira de Cartografia*, n. 61/04, 2009.
- ANDRIENKO, G. et al. Space, time and visual analytics. *Internacional Journal of Geographical Information Science*, 2010.
- BARBOSA, G. R. *Sistemas de apoio a decisão sob o enfoque de profissionais de TI e decisores*. Dissertação (Mestrado) — Universidade Federal de Pernambuco, 2003.
- CAMARGO, W. P. *Desenvolvimento de um ambiente Web para a interação entre participantes de projetos de agricultura de precisão*. Dissertação (Mestrado) — Escola Superior de Agricultura Luiz de Queiroz, Universidade de São Paulo, Piracicaba, SP, Março 2005.
- CIANCI, F. G. *Uso de Sistemas de Informações Geográficas no Estudo da Relação entre a Incidência de Aids em Usuários de Drogas Injetáveis, o Desenvolvimento Humano e as Apreensões de Entorpecentes pela Polícia no Estado de São Paulo no Período de 1997 à 2004*. Monografia (Trabalho de Conclusão de Curso) — Faculdade de Medicina de Ribeirão Preto da Universidade de São Paulo, Ribeirão Preto, SP, 2006.
- FILHO, P. C. de O.; SILVA, S. V. K. da. Um sistema de informações para suporte espacial e de decisões à gestão da arborização urbana no município de guarapuava, paraná. *REVESBAU*, v. 5, n. 3, p. 82 – 96, 2010. Piracicaba - SP.
- FOUNDATION, D. S. *GeoDjango Installation*. 2017. Acesso em: 04/11/2017. Disponível em: <<https://docs.djangoproject.com/en/1.11/ref/contrib/gis/install/>>.
- FOUNDATION, D. S. *GeoDjango Tutorial*. 2017. Acesso em: 04/11/2017. Disponível em: <<https://docs.djangoproject.com/en/1.11/ref/contrib/gis/tutorial/>>.
- FOUNDATION, D. S. *Installing Geospatial libraries*. 2017. Acesso em: 04/11/2017. Disponível em: <<https://docs.djangoproject.com/en/1.11/ref/contrib/gis/install/geolibs/>>.
- GONÇALVES, P. R. *Sistema de Informação Geográfica para apoio a decisão ao combate a incêndio*. Dissertação (Mestrado) — Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, SP, 2005.
- HINO, P. *Padrões espaciais da Tuberculose associados ao Indicador Adaptado de Condição de Vida no município de Ribeirão Preto*. Tese (Doutorado) — Escola de Enfermagem de Ribeirão Preto da Universidade de São Paulo, Ribeirão Preto, SP, 2007.
- KEIM, D. et al. Visual analytics: Definition, process and challenges. *Information Visualization - Human-Centered Issues and Perspectives*, 2008.
- LAUDON, K.; LAUDON, J. *Sistemas de informação gerenciais*. 9. ed. [S.l.]: Pearson Prentice Hall, 2010. Tradução Luciana do Amaral Teixeira; revisão técnica Belmiro Nascimento João.

- LONGLEY, P. A. et al. *Sistemas e ciência da informação geográfica*. 3. ed. Porto Alegre: Bookman, 2013. Tradução: André Schneider e outros; revisão técnica: Heinrich Hasenack, Eliseu José Weber.
- LOPES, F. S. *A Utilização de Sistemas de Informação Geográfica no Estudo da Exposição Humana aos Produtos da Queima da Palha de Cana-de-Açúcar (Saccharum sp) no Estado de São Paulo*. Dissertação (Mestrado) — Faculdade de Saúde Pública da Universidade de São Paulo, São Paulo, SP, 2005.
- MEDEIROS, C. N. de et al. Utilização de software livre para disponibilização de dados georreferenciados na internet: Caso do sistema ceará em mapas interativo. *Revista Brasileira de Cartografia*, n. 65/1, p. 25 – 34, 2013.
- MENDES, R. D. Inteligência artificial: Sistemas especialistas no gerenciamento da informação. *Ciência da Informação*, scielo, v. 26, n. 1, 1997. ISSN 0100-1965. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0100-19651997000100006&nrm=iso>.
- MIRANDA, J. I. *Fundamentos de Sistemas de Informações Geográficas*. 2. ed. Brasília, DF: Embrapa Informação Tecnológica, 2010. Revista e atualizada.
- O'BRIEN, J. A. *Sistemas de informação e as decisões gerenciais na era da Internet*. Tradução Célio Knipel Moreira e Cid Knipel Moreira. 2. ed. São Paulo: Editora Saraiva, 2004.
- PERIZZOLO, D. M. Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação), *Desenvolvimento de um Sistema de Informação Web para empresas de Factoring*. 2005.
- SANTANA, P.; COSTA, C.; LOUREIRO, A. Os sistemas de informação geográfica e o planejamento urbano saudável na amadora. *Revista do Departamento de Geografia - USP, Volume Especial Cartogeo*, p. 368 – 369, 2014.
- SASS, G. G. et al. Sistema de informação para monitoramento da saúde de idosos. *Journal of Health Informatics*, v. 4, p. 209–215, 2012.
- SILVA, A. de B. *Sistemas de Informações Geo-referenciadas: conceitos e fundamentos*. Campinas, SP: Editora da Unicamp, 2003. 2ª reimpressão, 2012.
- ZANETI JUNIOR, L. A.; ROCHA VIDAL, A. G. da. Construção de sistemas de informação baseados na tecnologia web. *Revista de Administração - RAUSP*, v. 41, n. 3, p. 232–244, Julho–Setembro 2006.

APÊNDICE A — SCRIPT PARA ALTERAÇÃO DE PONTOS

```

1 # -*- coding: utf-8 -*-
2 from siamiant.models import Idoso, Instrumidoso, Moradia
3
4
5 def alteraPonto(cpf, data_entrevista):
6     try:
7         idoso = Idoso.objects.get(cpf=cpf)
8     except:
9         print "CPF errado ou algum problema."
10
11     try:
12         entrevista = Instrumidoso.objects.get(ididoso=idoso,
13         dataentrevista=data_entrevista)
14     except:
15         print "Entrevista do idoso não existe ou algum erro
16         ocorreu."
17
18     print "Idoso: "+idoso.nome
19     print "Data da Entrevista: "+str(entrevista.dataentrevista)
20     print "SUS: "+str(entrevista.sus)
21     print "ID Família: "+str(entrevista.idfamilia)
22
23     try:
24         moradia = Moradia.objects.get(idinstrumidoso=entrevista)
25     except:
26         print "Para esta data o idoso não possui informações
27         sobre moradia."
28
29     coordenada_x = raw_input("Informe a coordenada x: ")
30     coordenada_y = raw_input("Informe a coordenada y: ")
31
32     moradia.ponto = 'POINT(' +str(coordenada_x)+' '+str(

```

```
        coordenada_y)+'')'  
32 moradia.save()  
33  
34 print "Concluído!"
```

APÊNDICE B — CÓDIGO RESPONSÁVEL POR REALIZAR O PROCESSAMENTO PARA GERAR OS RELATÓRIOS GEOGRÁFICOS.

```

1  # -*- coding: utf-8 -*-
2  from __future__ import unicode_literals
3
4  from django.shortcuts import render
5
6  from django.core.serializers import serialize
7  #from world.models import WorldIdoso
8
9  from random import randint
10
11 from siamiant.models import Instrumidoso, Moradia
12
13 #from .forms import PontoForm
14
15 from forms import RelatorioGestaoForm
16
17 from siami_fatores.views import i3_Renda, i1_Faixaetaria,
    i2_Escolaridade, i4_Condicaomoradia, i8_Diagnostico
18
19
20 def index_relatorios_geograficos(request):
21     return render(request, 'index_relatorios_geograficos.html')
22
23 def relatorio_geografico_temporal(request, funcao):
24
25     context={}
26     context['relatorio_gerado']=False
27     context['msg_erro']=" "
28     context['idosos_selecionados']=0
29
30
31
32     if request.method=='POST':

```

```

33
34     # Listas que armazenarão as entrevistas do idoso e que
35     foram geradas pela query, podendo ter filtros ou não.
36     antepenultima = [] # Guarda a antepenultima entrevista
37     do idoso
38
39     penultima = [] # Guarda a penultima entrevista do idoso
40     ultima = [] # Guarda a ultima entrevista do idoso
41
42
43     fator_resultante_antepenultima = []
44     fator_resultante_penultima = []
45     fator_resultante_ultima = []
46
47     resultado_verde_ultima=0
48     resultado_verde_penultima=0
49     resultado_verde_antepenultima=0
50
51     resultado_amarelo_ultima=0
52     resultado_amarelo_penultima=0
53     resultado_amarelo_antepenultima=0
54
55     resultado_vermelho_ultima=0
56     resultado_vermelho_penultima=0
57     resultado_vermelho_antepenultima=0
58
59     filtros=RelatorioGestaoForm(request.POST)
60
61     if filtros.is_valid():
62         print(filtros.cleaned_data['equipe'])
63         print(filtros.cleaned_data['mes'])
64         print(filtros.cleaned_data['ano'])
65
66         print(filtros.cleaned_data['mes_inicial'])
67         print(filtros.cleaned_data['ano_inicial'])
68
69         print(filtros.cleaned_data['mes_final'])
70         print(filtros.cleaned_data['ano_final'])

```



```
70     equipe=filtros.cleaned_data['equipe']
71     mes=filtros.cleaned_data['mes']
72     ano=filtros.cleaned_data['ano']
73
74     try:
75         mes_inicial=int(filtros.cleaned_data['
76             mes_inicial'])
77         ano_inicial=int(filtros.cleaned_data['
78             ano_inicial'])
79         mes_final=int(filtros.cleaned_data['mes_final'])
80         ano_final=int(filtros.cleaned_data['ano_final'])
81     except:
82         pass
83
84     opcao_filtro=request.POST.get('opcao-filtro',None)
85
86     print(request.POST.get('opcao-filtro',None))
87
88     # Condições para obter as entrevistas do idoso.
89     # É necessário obter uma entrevista por idoso. Esta
90     # entrevista deve ser atual.
91     if equipe != None: #Foi informado um valor de equipe
92         .
93
94         if opcao_filtro!=None:
95
96             opcao_filtro=int(opcao_filtro)
97
98             if opcao_filtro==1:
99                 print "Equipe selecionada com filtro 1"
100
101                 entrevistas=Instrumidoso.objects.
102                     order_by('ididoso','-dataentrevista')
103                     .filter(idesf=int(equipe),
104                             dataentrevista__month=int(mes))
105
106             elif opcao_filtro==2:
107                 print "Equipe selecionada com filtro 2"
```

```
102     entrevistas=Instrumidoso.objects.  
        order_by('ididoso','-dataentrevista')  
        .filter(idesf=int(equipe),  
        dataentrevista__year=int(ano))  
  
103  
104  
105     elif opcao_filtro==3:  
106         print "Equipe selecionada com filtro 3"  
107  
108         if ano_final>=ano_inicial:  
109             if ano_final==ano_inicial:  
110                 if mes_final>=mes_inicial:  
111                     print "Anos iguais ,  
                            Mes_final>=Mes_inicial"  
112                     entrevistas=Instrumidoso.  
                            objects.order_by('ididoso  
                            ','-dataentrevista').  
                            filter(idesf=int(equipe),  
                            dataentrevista__month__gte  
                            =mes_inicial ,  
                            dataentrevista__year__gte  
                            =ano_inicial ,  
                            dataentrevista__month__lte  
                            =mes_final ,  
                            dataentrevista__year__lte  
                            =ano_final)  
  
113                 else:  
114                     print "O mes final deve ser  
                            maior ou igual ao mes  
                            inicial"  
115                     context['msg_erro']="O mês  
                            final deve ser posterior  
                            ou igual ao mês de início  
                            !"  
  
116  
117                     context['form']=  
                            RelatorioGestaoForm(  
                            request.POST)  
118                 return request , context
```

```
119
120         else :
121             print "ano final > ano inicial"
122             entrevistas=Instrumidoso.objects
                .order_by('ididoso', '-
                dataentrevista').filter(idesf
                =int(equipe),
                dataentrevista__month__gte=
                mes_inicial,
                dataentrevista__year__gte=
                ano_inicial,
                dataentrevista__month__lte=
                mes_final,
                dataentrevista__year__lte=
                ano_final)
123
124         else :
125             print "O ano final deve ser maior ou
                igual que o ano inicial."
126             context['msg_erro']="O ano final
                deve ser posterior ou igual ao
                ano inicial!"
127
128             context['form']=RelatorioGestaoForm(
                request.POST)
129             return request, context
130
131         else :
132             print "Equipe selecionada sem filtros"
133
134             entrevistas=Instrumidoso.objects.order_by('
                ididoso', '-dataentrevista').filter(idesf=
                int(equipe))
135
136         else :
137             if opcao_filtro!=None:
138
139                 opcao_filtro=int(opcao_filtro)
140
```

```
141         if opcao_filtro==1:
142             print "Equipe não selecionada com filtro
143                 1"
144
145             entrevistas=Instrumidoso.objects.
146                 order_by('ididoso','-dataentrevista')
147                 .filter(dataentrevista__month=int(mes
148                     ))
149
150         elif opcao_filtro==2:
151             print "Equipe não selecionada com filtro
152                 2"
153
154             entrevistas=Instrumidoso.objects.
155                 order_by('ididoso','-dataentrevista')
156                 .filter(dataentrevista__year=int(ano)
157                     )
158
159         elif opcao_filtro==3:
160             print "Equipe não selecionada com filtro
161                 3"
162
163             if ano_final>=ano_inicial:
164                 if ano_final==ano_inicial:
165                     if mes_final>=mes_inicial:
166                         print "Anos iguais ,
167                             Mes_final>=Mes_inicial"
168                         entrevistas=Instrumidoso.
169                             objects.order_by('ididoso
170                                 ','-dataentrevista').
171                             filter(
172                                 dataentrevista__month__gte
173                                 =mes_inicial ,
174                                 dataentrevista__year__gte
175                                 =ano_inicial ,
176                                 dataentrevista__month__lte
177                                 =mes_final ,
```

```
dataentrevista__year__lte
=ano_final)
161     else:
162         print "O mes final deve ser
maior ou igual ao mes
inicial"
163         context['msg_erro']="O mês
final deve ser posterior
ou igual ao mês de início
!"
164
165         context['form']=
RelatorioGestaoForm(
request.POST)
166         return request, context
167
168     else:
169         print "ano final > ano inicial"
170         entrevistas=Instrumidoso.objects
.order_by('ididoso', '-
dataentrevista').filter(
dataentrevista__month__gte=
mes_inicial,
dataentrevista__year__gte=
ano_inicial,
dataentrevista__month__lte=
mes_final,
dataentrevista__year__lte=
ano_final)
171
172     else:
173         print "O ano final deve ser maior ou
igual que o ano inicial."
174         context['msg_erro']="O ano final
deve ser posterior ou igual ao
ano inicial!"
175
176         context['form']=RelatorioGestaoForm(
request.POST)
```

```

177             return request , context
178         else :
179             print 'Faz a query selecionando todos os
                instrumentos do idoso .'
180             # Esta query requer atenção. Verificar se
                ela sempre funciona .
181             # A ideia desta query é selecionar as
                entrevistas do idoso .
182             # A entrevista selecionada corresponde a
                última entrevista do idoso .
183             entrevistas=Instrumidoso.objects.order_by('
                ididoso ', '-dataentrevista ')
184
185         if len(entrevistas)==0:
186             context ['msg_erro']="Suas configurações de busca
                não geraram resultados!"
187
188             context ['form']=RelatorioGestaoForm(request.POST
                )
189             return request , context
190
191         print entrevistas
192
193         for entrevista in entrevistas:
194             print entrevista.dataentrevista
195             print entrevista.idinstrumidoso
196
197         #print "Número de idosos selecionados: "+str(len(
                entrevistas))
198
199         #context ['idosos_selecionados']=len(entrevistas)
200
201         context ['form']=RelatorioGestaoForm(request.POST)
202
203         idIdoso = -1 # Usado para verificar se as
                entrevistas ainda são do mesmo idoso .
204             # Para pegar apenas as 3 últimas
                entrevistas .
205

```

```
206     quantidadeEntrevistaIdoso = 0
207
208     context [ 'idosos_selecionados ']=0
209     context [ 'idosos_georreferenciados ']=0
210
211     for entrevista in entrevistas:
212
213         #print entrevista.idIdoso
214         #print i3_Renda(entrevista)
215
216         try:
217             moradia = Moradia.objects.get(idinstrumidoso
218                                     =entrevista)
219
220             if entrevista.ididoso != idIdoso:
221
222                 # Verificar depois quando a primeira
223                 entrevista não tem ponto geográfico
224                 # mas as outras entrevistas do idoso
225                 possuem.
226                 if moradia.ponto != None:
227                     context [ 'idosos_georreferenciados '
228                             ]+=1
229
230                 flag=1
231
232                 idIdoso = entrevista.ididoso
233                 quantidadeEntrevistaIdoso = 0
234
235             if quantidadeEntrevistaIdoso != 3:
236
237                 resultado = funcao(entrevista)
238
239                 if resultado != -1:
240
241                     if flag:
242                         context [ 'idosos_selecionados '
243                                 ]+=1
```

```

240         flag=0
241
242         if quantidadeEntrevistaIdoso == 0:
243             ultima.append(moradia)
244             fator_resultante_ultima.append(
                resultado)
245
246             if resultado == 0:
247                 resultado_verde_ultima+=1
248             elif resultado == 1:
249                 resultado_amarelo_ultima+=1
250             else:
251                 resultado_vermelho_ultima+=1
252
253         elif quantidadeEntrevistaIdoso == 1:
254             penultima.append(moradia)
255             fator_resultante_penultima.
                append(resultado)
256
257             if resultado == 0:
258                 resultado_verde_penultima+=1
259             elif resultado == 1:
260                 resultado_amarelo_penultima
                +=1
261             else:
262                 resultado_vermelho_penultima
                +=1
263
264         else:
265             antepenultima.append(moradia)
266             fator_resultante_antepenultima.
                append(resultado)
267         print 'entrou'
268
269         if resultado == 0:
270             resultado_verde_antepenultima
                +=1
271         elif resultado == 1:
272             resultado_amarelo_antepenultima

```

```

                +=1
273         else:
274             resultado_vermelho_antepenultima
                +=1
275
276             quantidadeEntrevistaIdoso+=1
277     except:
278         print "Idoso sem dados de moradia"
279
280
281     #print WorldIdoso.objects.all()
282
283     pontos_ultimo=serialize('geojson', ultima,
        geometry_field='ponto')
284     pontos_penultimo=serialize('geojson', penultima,
        geometry_field='ponto')
285     pontos_antepenultimo=serialize('geojson',
        antepenultima, geometry_field='ponto')
286
287     #print pontos
288
289     print 'lista ultima'
290
291     for item in ultima:
292         print item.idinstrumidoso
293         print item.data
294
295     print 'lista penultima'
296
297     for item in penultima:
298         print item.idinstrumidoso
299         print item.data
300
301     print 'lista antepenultima'
302
303     for item in antepenultima:
304         print item.idinstrumidoso
305         print item.data
306
```

```
307     print fator_resultante_antepenultima
308     print fator_resultante_ultima
309
310     context['pontos_ultimo'] = pontos_ultimo
311     context['pontos_penultimo'] = pontos_penultimo
312     context['pontos_antepenultimo'] =
313         pontos_antepenultimo
314     context['fator_resultante_ultima'] =
315         fator_resultante_ultima
316     context['fator_resultante_penultima'] =
317         fator_resultante_penultima
318     context['fator_resultante_antepenultima'] =
319         fator_resultante_antepenultima
320
321     context['resultado_verde_ultima']=
322         resultado_verde_ultima
323     context['resultado_verde_penultima']=
324         resultado_verde_penultima
325     context['resultado_verde_antepenultima']=
326         resultado_verde_antepenultima
327
328     context['resultado_amarelo_ultima']=
329         resultado_amarelo_ultima
330     context['resultado_amarelo_penultima']=
331         resultado_amarelo_penultima
332     context['resultado_amarelo_antepenultima']=
333         resultado_amarelo_antepenultima
334
335     context['resultado_vermelho_ultima']=
336         resultado_vermelho_ultima
337     context['resultado_vermelho_penultima']=
338         resultado_vermelho_penultima
339     context['resultado_vermelho_antepenultima']=
340         resultado_vermelho_antepenultima
341
342     context['relatorio_gerado']=True
```

```
333
334     else :
335         context [ 'form ']=RelatorioGestaoForm()
336
337     return request , context
338
339
340 def relatorioGeograficoRendaPerCapita (request) :
341
342     retorno = relatorio_geografico_temporal (request , i3_Renda)
343
344     retorno [1] [ 'relatorio ']= "Relatório Renda per Capita"
345
346     return render (retorno [0] , '
347         relatorio_geografico_renda_per_capita.html' , retorno [1])
348
349 def relatorioGeograficoFaixaEtaria (request) :
350
351     retorno = relatorio_geografico_temporal (request ,
352         i1_Faixaetaria)
353
354     retorno [1] [ 'relatorio ']= "Relatório Faixa Etária"
355
356     return render (retorno [0] , 'relatorio_geografico_faixa_etaria .
357         html' , retorno [1])
358
359 def relatorioGeograficoEscolaridade (request) :
360
361     retorno = relatorio_geografico_temporal (request ,
362         i2_Escolaridade)
363
364     retorno [1] [ 'relatorio ']= "Relatório Escolaridade"
365
366     return render (retorno [0] , 'relatorio_geografico_escolaridade .
367         html' , retorno [1])
368
369 def relatorioGeograficoCondicaoMoradia (request) :
370
371     retorno = relatorio_geografico_temporal (request ,
```

```
        i4_Condicaomoradia)
367
368     retorno [1][ 'relatorio ']="Relatório Condição de Moradia"
369
370     return render(retorno [0] , '
        relatorio_geografico_condicao_moradia.html' ,retorno [1])
371
372 def relatorioGeograficoDiagnosticoMedico(request):
373
374     retorno = relatorio_geografico_temporal(request ,
        i8_Diagnostico)
375
376     retorno [1][ 'relatorio ']="Relatório Diagnóstico Médico"
377
378     return render(retorno [0] , '
        relatorio_geografico_diagnostico_medico.html' ,retorno [1])
```

APÊNDICE C — SCRIPT PARA REPLICAR DADOS NO BANCO DE DADOS

```

1 # -*- coding: utf-8 -*-
2 from siamiant.models import (
3     Idoso ,
4     Instrumidoso ,
5     Diagnostico ,
6     Social ,
7     Moradia ,
8     Tipodiagnostico
9 )
10
11 from datetime import datetime
12
13
14 def get_idoso(id_idoso):
15
16     a=Idoso.objects.get(ididoso = id_idoso)
17
18     return a
19
20 def get_entrevista(idoso):
21
22     return Instrumidoso.objects.filter(ididoso=idoso).order_by('
23         dataentrevista').first()
24
25
26 def new_entrevista(entrevista_antiga):
27
28     nova_entrevista = Instrumidoso()
29
30     data = datetime.strptime(raw_input('Informe a data da nova
31         entrevista: '), '%d/%m/%Y').date()
32
33     nova_entrevista.dataentrevista = data
34
35     #nova_entrevista.dataentrevista = datetime.today()

```

```
33
34     print nova_entrevista.dataentrevista
35
36     nova_entrevista.sus = entrevista_antiga.sus
37     nova_entrevista.idfamilia = entrevista_antiga.idfamilia
38     nova_entrevista.rg = entrevista_antiga.rg
39     nova_entrevista.idesf = entrevista_antiga.idesf
40     nova_entrevista.ididoso = entrevista_antiga.ididoso
41     nova_entrevista.idmicroarea = entrevista_antiga.idmicroarea
42
43     nova_entrevista.save()
44
45     print nova_entrevista.dataentrevista
46
47     return nova_entrevista
48
49 def new_social(entrevista_antiga , entrevista_nova ):
50
51     novo_social = Social()
52
53     social_antigo = Social.objects.get(idinstrumidoso =
54         entrevista_antiga)
55
56     novo_social.escolaridade = social_antigo.escolaridade
57     novo_social.data = entrevista_nova.dataentrevista
58     novo_social.estadocivil = social_antigo.estadocivil
59     novo_social.tempoviuvez = social_antigo.tempoviuvez
60     novo_social.arranjofamiliar = social_antigo.arranjofamiliar
61     novo_social.religiao = social_antigo.religiao
62     novo_social.praticantereligiao = social_antigo.
63         praticantereligiao
64
65     novo_social.ocupacaoatual = social_antigo.ocupacaoatual
66     novo_social.tempocupacaoatual = social_antigo.
67         tempocupacaoatual
68
69     novo_social.rendafamiliar = int(raw_input( 'Informe a renda
70         familiar: '))
71
72     novo_social.numpessoaresidente = int(raw_input( 'Informe a
73         quantidade de pessoas que residem: '))
```

```
67
68     novo_social.idinstrumidoso = entrevista_nova
69
70     novo_social.save()
71
72 def new_moradia(entrevista_antiga , entrevista_nova):
73
74     nova_moradia = Moradia()
75
76     moradia_antiga = Moradia.objects.get(idinstrumidoso =
77         entrevista_antiga)
78
79     nova_moradia.rua = moradia_antiga.rua
80     nova_moradia.numero = moradia_antiga.numero
81     nova_moradia.bairro = moradia_antiga.bairro
82     nova_moradia.data = entrevista_nova.dataentrevista
83
84     nova_moradia.possemoradia = int(raw_input('Posse da moradia:
85         '))
86     nova_moradia.condicaomoradia = int(raw_input('Condicao
87         moradia: '))
88     nova_moradia.abastecimentoagua = int(raw_input('
89         Abastecimento de água: '))
90     nova_moradia.dejetos = int(raw_input('Dejetos: '))
91     nova_moradia.lixo = int(raw_input('Lixo: '))
92
93     nova_moradia.idinstrumidoso = entrevista_nova
94
95     nova_moradia.save()
96
97
98 def new_diagnostico(entrevista_nova):
99
100     resposta = 1
101
102     while resposta != 0:
103         novo_diagnostico = Diagnostico()
104
105         novo_diagnostico.idinstrumidoso = entrevista_nova
```

```
102     novo_diagnostico.idtipodiagnostico = Tipodiagnostico.
        objects.get(idtipodiagnostico = int(raw_input('
        Informe o diagnostico: ')))
103
104     novo_diagnostico.save()
105
106     resposta = int(raw_input('Continuar? - (0) pra finalizar
        : '))
107
108 def main():
109
110     resposta = 1
111
112     while resposta != 0:
113
114         anciao=get_idoso(int(raw_input('Informe o idoso: ')))
115
116         print anciao.nome
117
118         entrevista_antiga = get_entrevista(anciao)
119
120         nova_entrevista = new_entrevista(entrevista_antiga)
121
122         new_social(entrevista_antiga , nova_entrevista)
123         new_moradia(entrevista_antiga , nova_entrevista)
124         new_diagnostico(nova_entrevista)
125
126         resposta = int(raw_input('Continuar? - (0) Sair: '))
```


APÊNDICE D — RELATÓRIO RENDA *PER CAPITA*

Relatório Renda per Capita

Equipe de Saúde da Família: < >

Período:

Mês

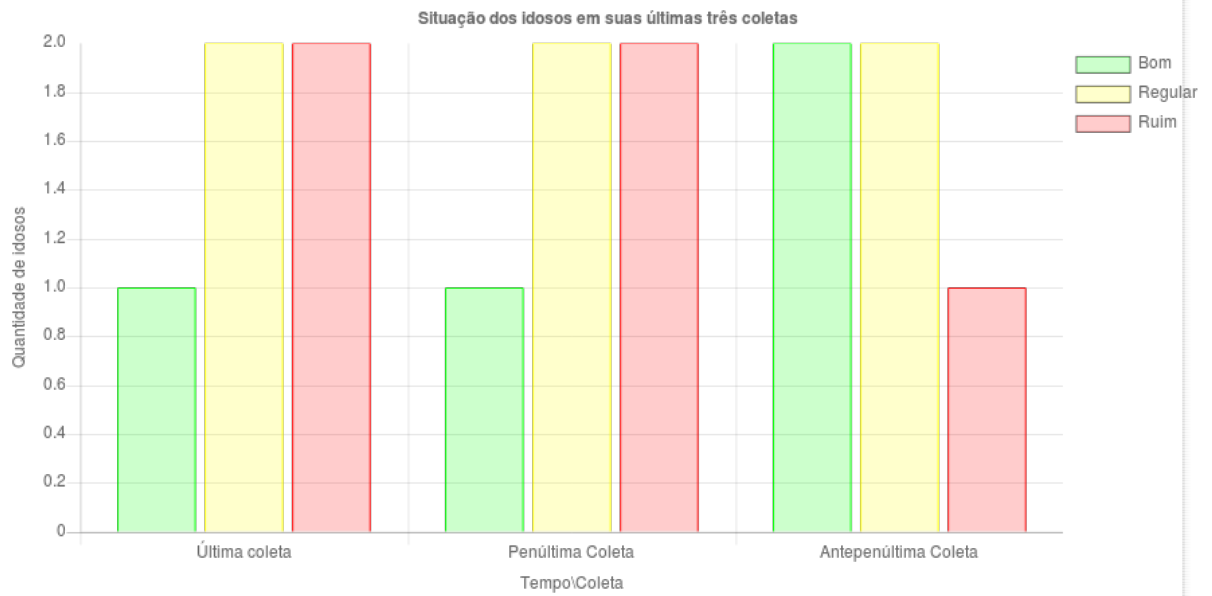
Ano

Intervalo

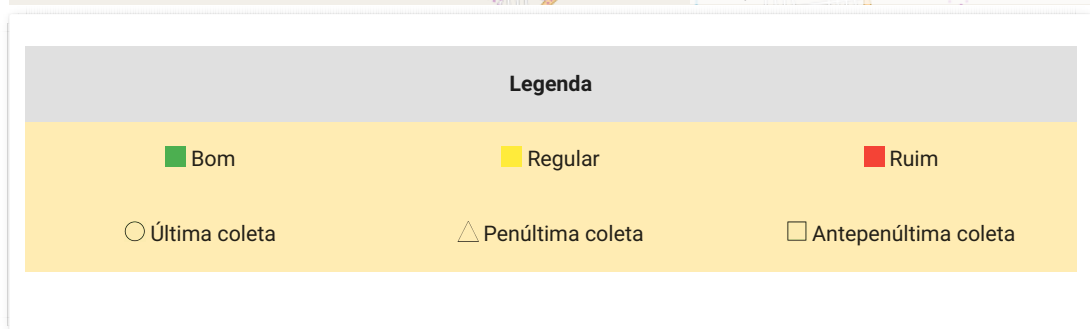
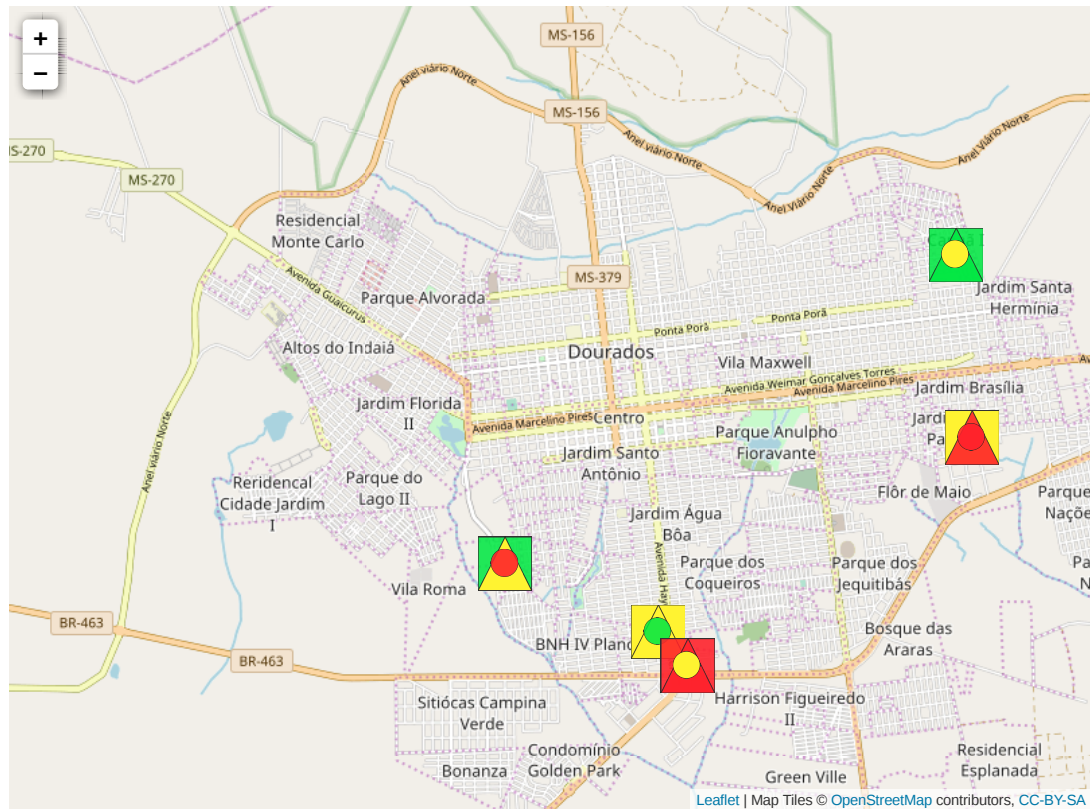
GERAR RELATÓRIO RESETAR

Idosos considerados: 5

Classificação\Entrevista	Última	Penúltima	Antepenúltima
Bom	1	1	2
Regular	2	2	2
Ruim	2	2	1



Idosos Georreferenciados: 5



APÊNDICE E — RELATÓRIO FAIXA ETÁRIA

Relatório Faixa Etária

Equipe de Saúde da Família: < >

Período:

Mês

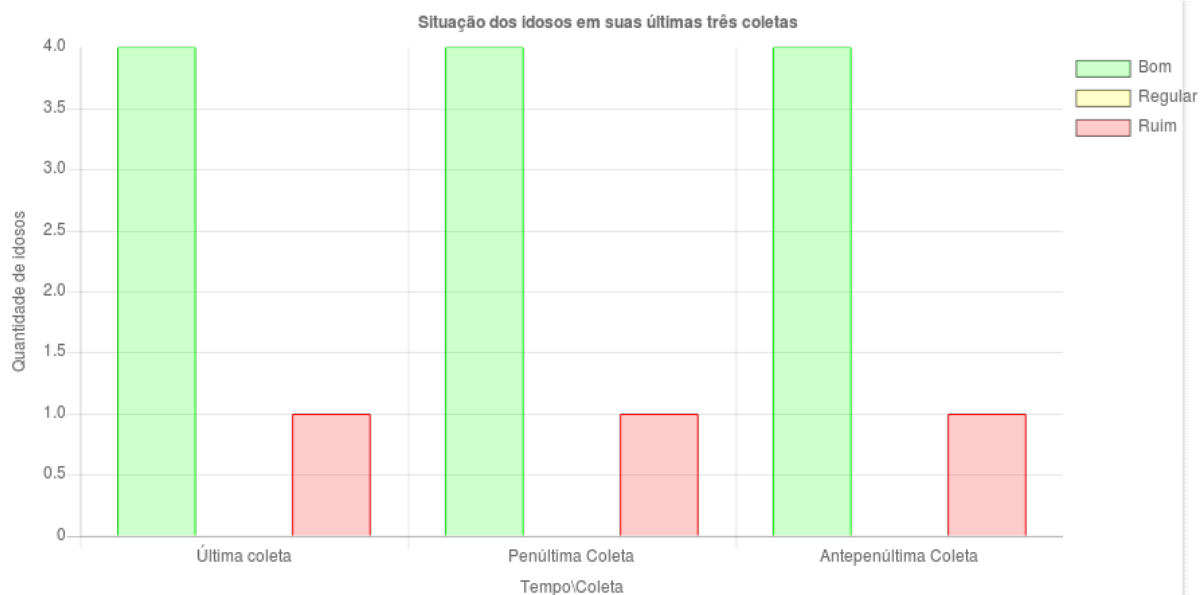
Ano

Intervalo

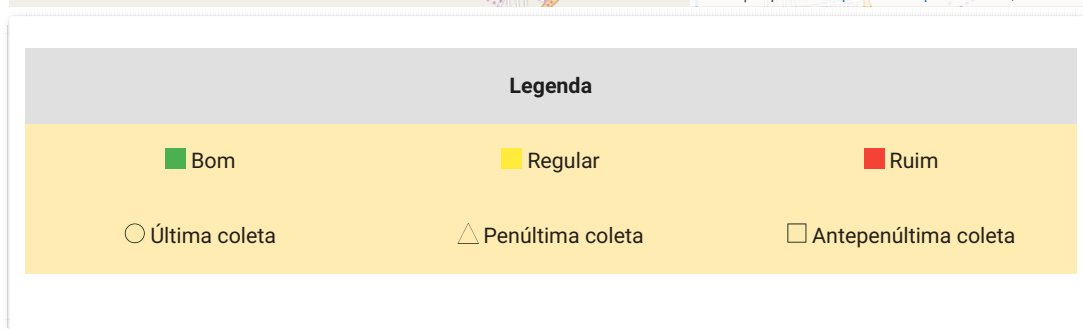
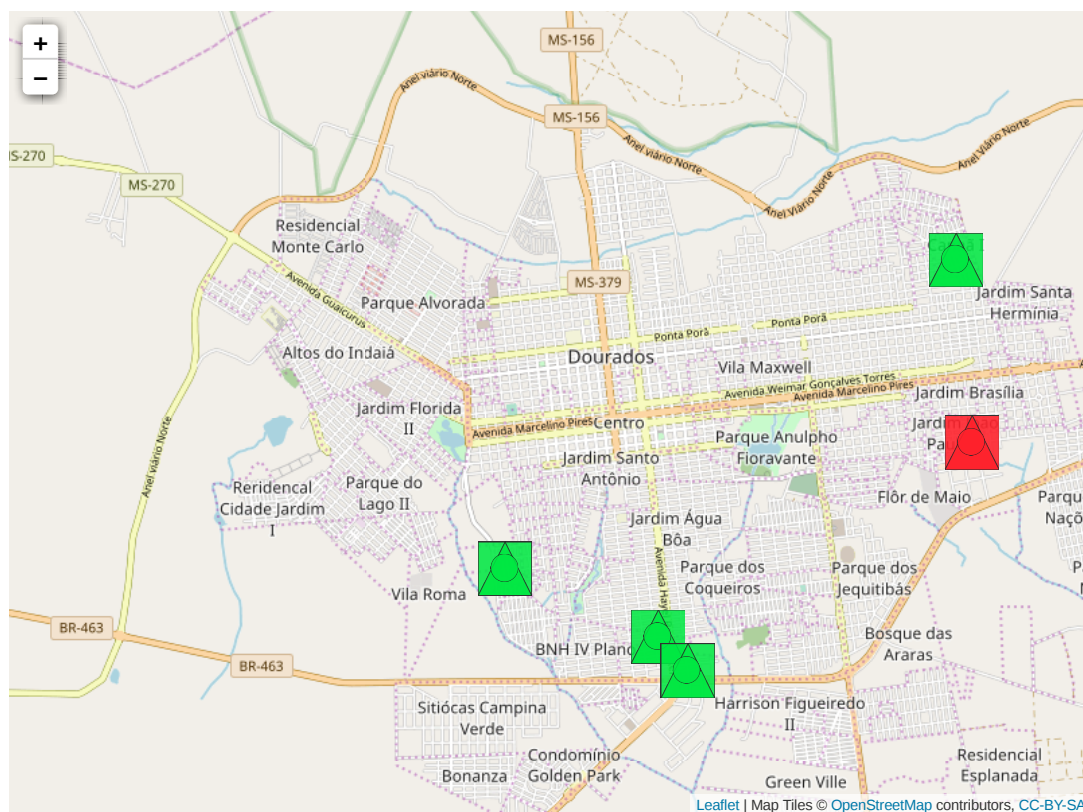
GERAR RELATÓRIO RESETAR

Idosos considerados: 5

Classificação\Entrevista	Última	Penúltima	Antepenúltima
Bom	4	4	4
Regular	0	0	0
Ruim	1	1	1



Idosos Georreferenciados: 5



APÊNDICE F — RELATÓRIO ESCOLARIDADE

Relatório Escolaridade

Equipe de Saúde da Família: < >

Período:

Mês

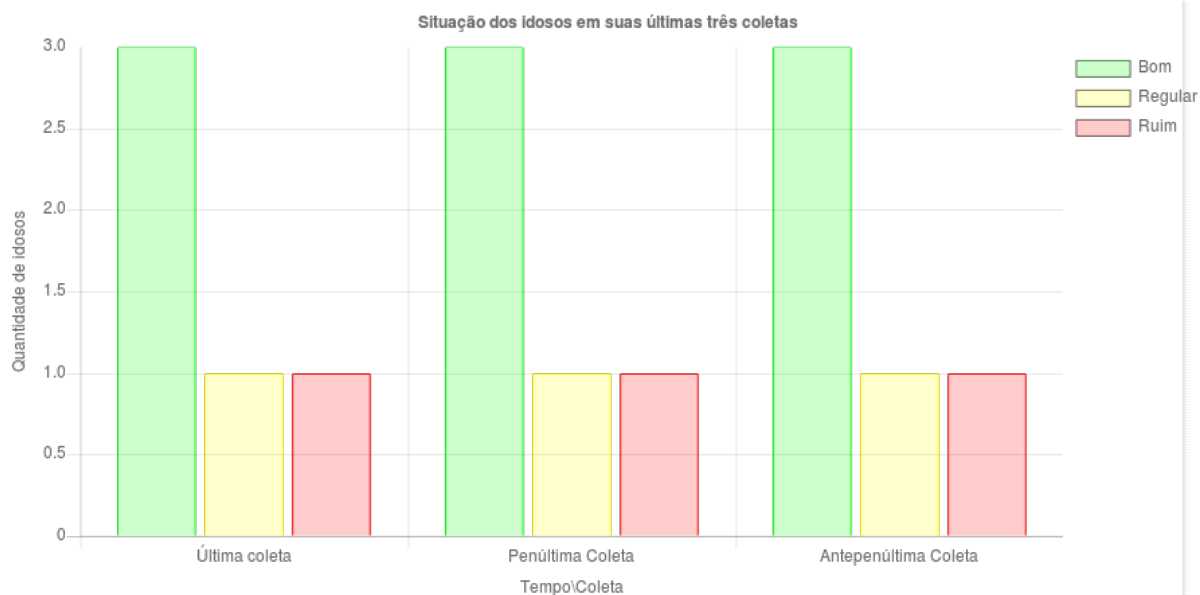
Ano

Intervalo

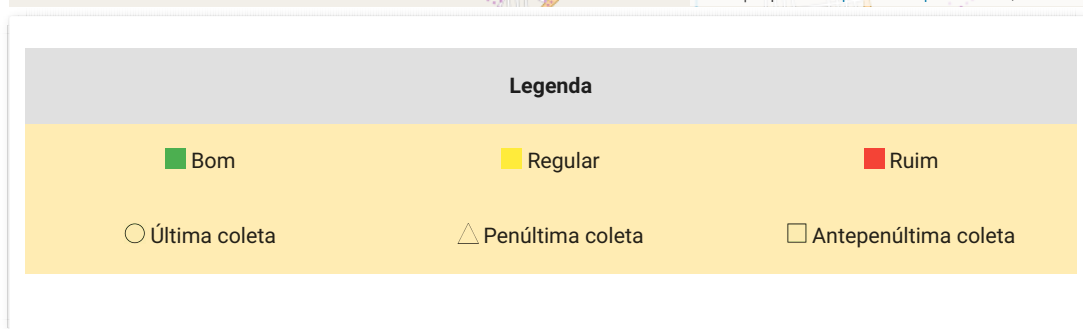
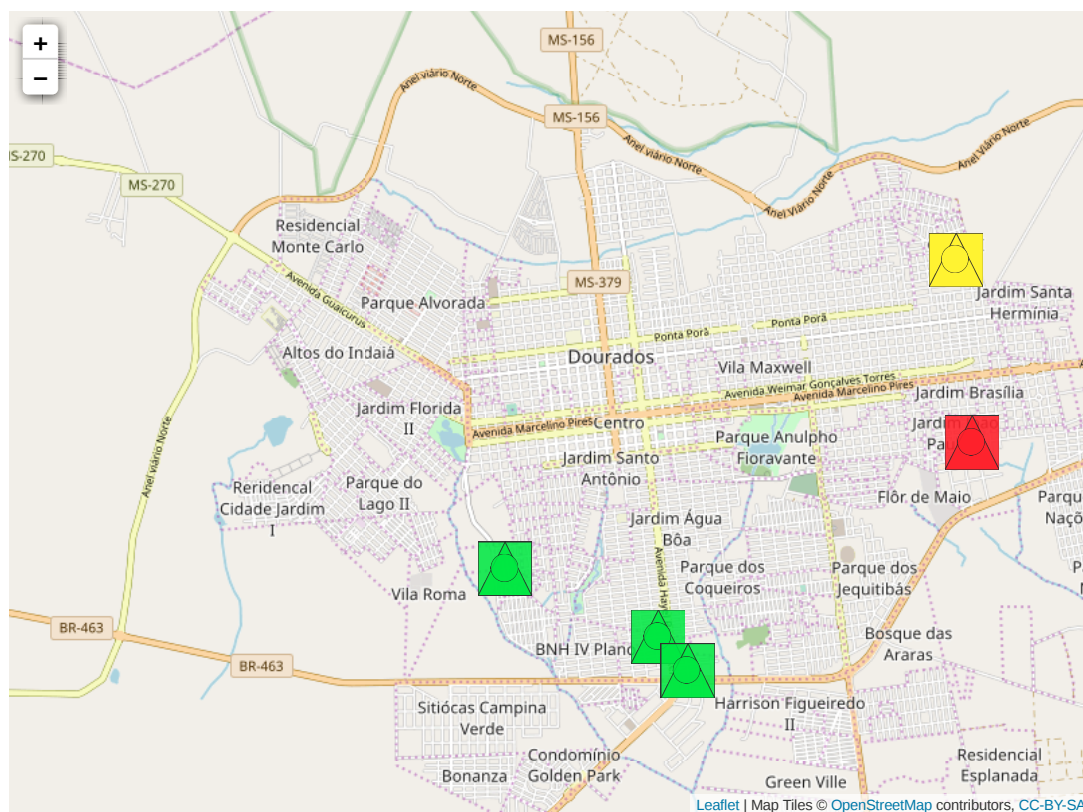
GERAR RELATÓRIO **RESETAR**

Idosos considerados: 5

Classificação\Entrevista	Última	Penúltima	Antepenúltima
Bom	3	3	3
Regular	1	1	1
Ruim	1	1	1



Idosos Georreferenciados: 5



APÊNDICE G — RELATÓRIO CONDIÇÃO DE MORADIA

Relatório Condição de Moradia

Equipe de Saúde da Família: < >

Período:

Mês

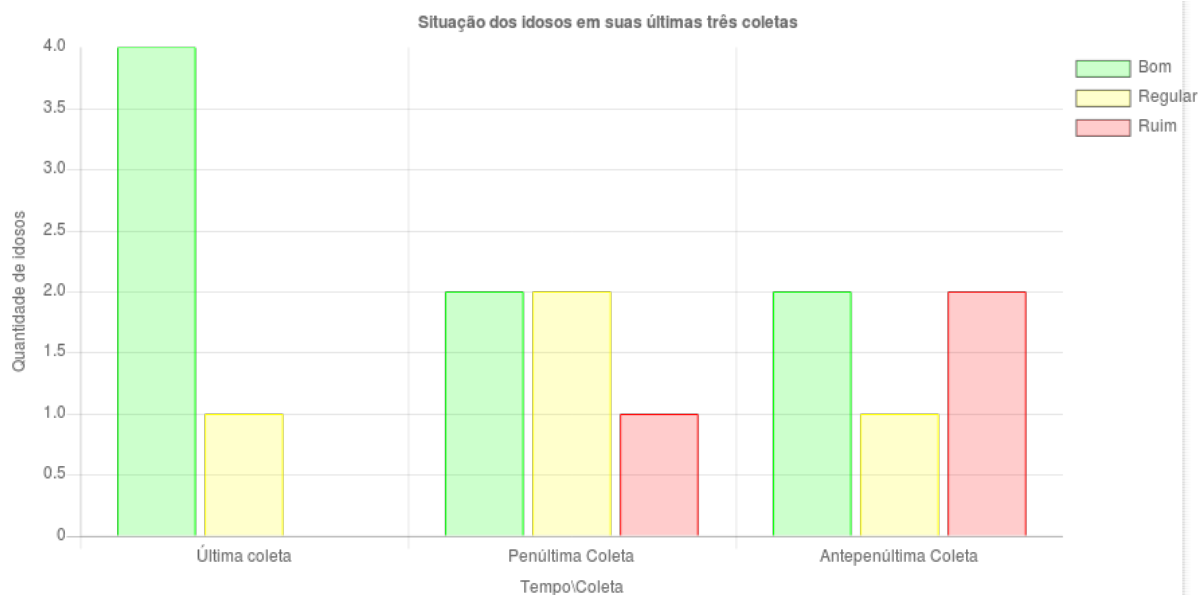
Ano

Intervalo

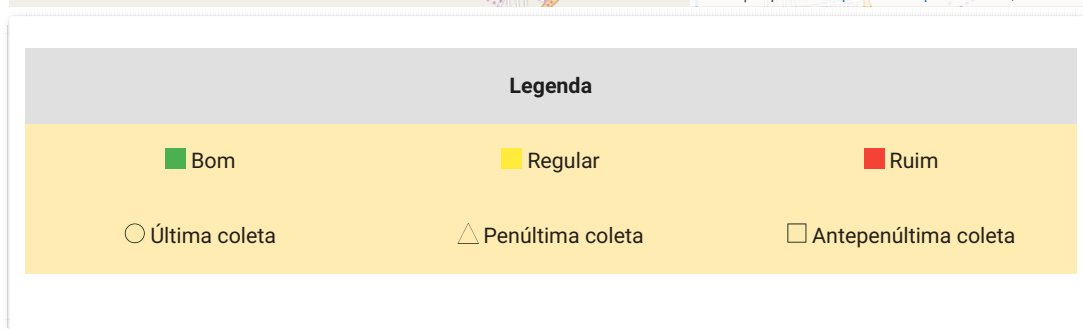
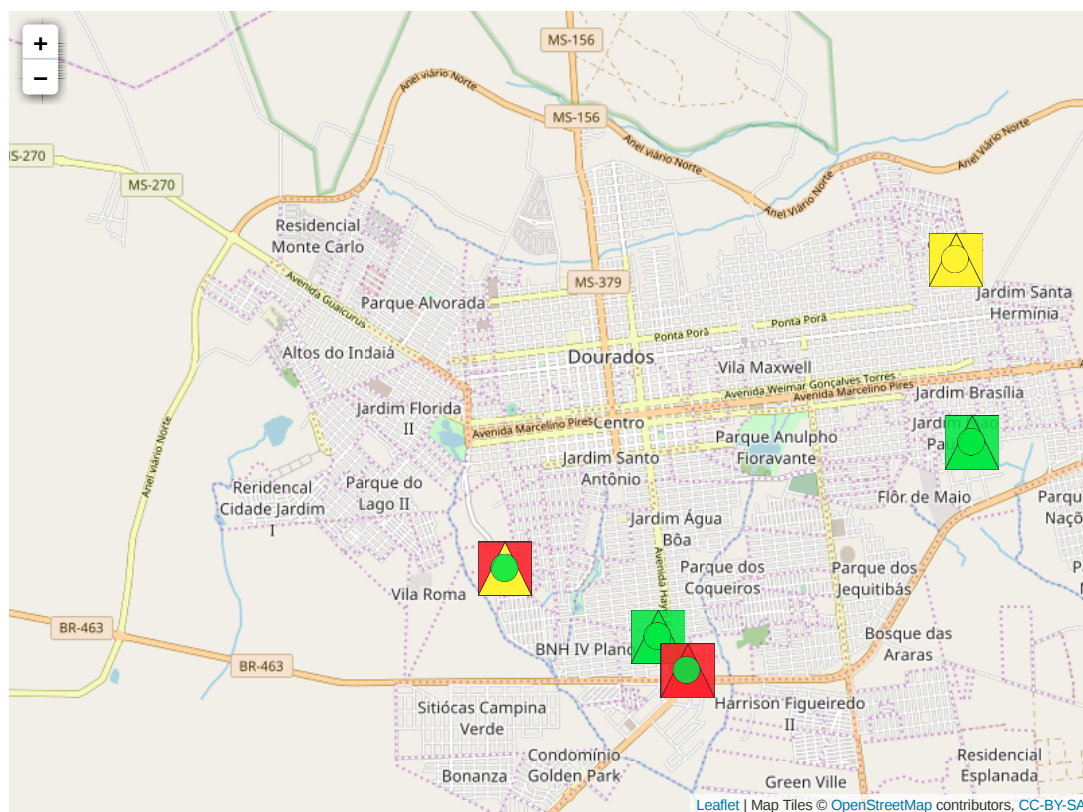
GERAR RELATÓRIO **RESETAR**

Idosos considerados: 5

Classificação\Entrevista	Última	Penúltima	Antepenúltima
Bom	4	2	2
Regular	1	2	1
Ruim	0	1	2



Idosos Georreferenciados: 5



APÊNDICE H — RELATÓRIO DIAGNÓSTICO MÉDICO

Relatório Diagnóstico Médico

Equipe de Saúde da Família: < >

Período:

Mês

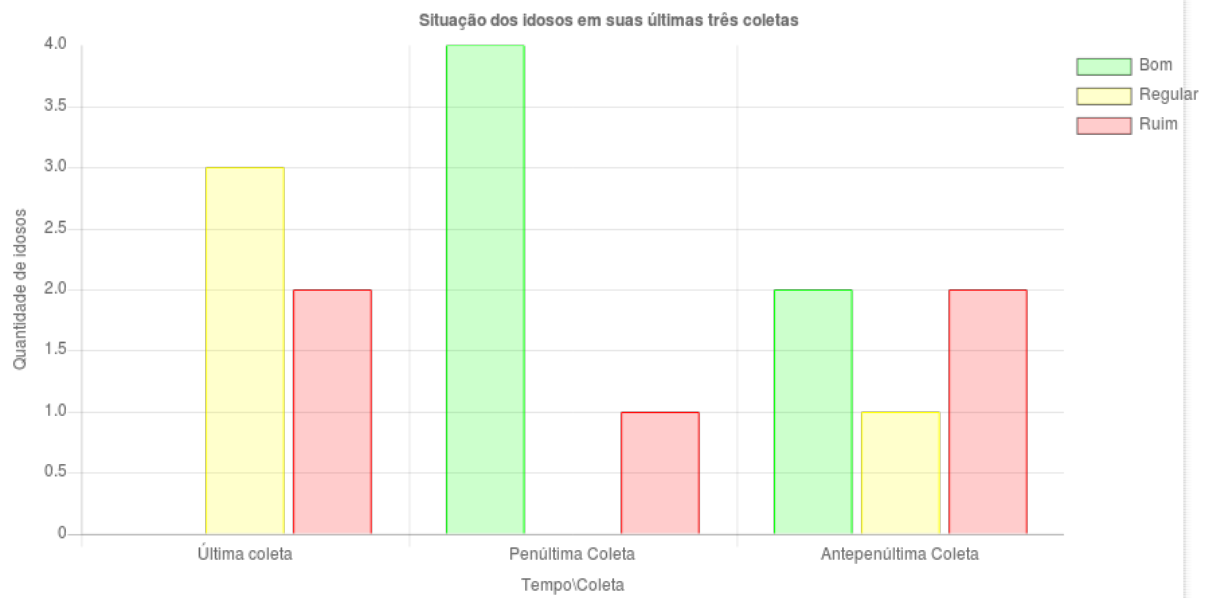
Ano

Intervalo

GERAR RELATÓRIO **RESETAR**

Idosos considerados: 5

Classificação\Entrevista	Última	Penúltima	Antepenúltima
Bom	0	4	2
Regular	3	0	1
Ruim	2	1	2



Idosos Georreferenciados: 5

