

UNIVERSIDADE DE CAXIAS DO SUL
Centro de Computação e Tecnologia da Informação
Curso de Bacharelado em Ciência da Computação

Rogério Bertolini

**ESTUDO DE CASO SOBRE VISUALIZAÇÃO DE DADOS NA ÁREA
DA SAÚDE**

Caxias do Sul

2009

Rogério Bertolini

**ESTUDO DE CASO SOBRE VISUALIZAÇÃO DE DADOS NA ÁREA DA
SAÚDE**

Trabalho de Conclusão de Curso
para obtenção do Grau de
Bacharel em Ciência da
Computação da Universidade de
Caxias do Sul.

**Prof.^a Dr.^a Carine Geltrudes Webber
Orientadora**

Caxias do Sul

2009

“Só existe dois dias no ano que nada pode ser feito. Um se chama ontem e o outro se chama amanhã, portanto hoje é o dia certo para amar, acreditar, fazer e principalmente viver”.

Dalai Lama

AGRADECIMENTOS

Primeiramente, quero agradecer a vocês, Agenor e Helena, meus pais, pela confiança e incentivo para concluir esta fase. Aos meus familiares e amigos que fizeram parte, direta ou indiretamente, desta jornada também quero levar meus agradecimentos.

A minha namorada, Caroline, pela compreensão de abdicar de muitos momentos de companheirismo em prol das responsabilidades curriculares, também manifesto minha gratidão.

Aos meus professores, pelo dom de disseminar conhecimentos e formar opiniões também quero demonstrar minha estima. Em especial a professora Carine Geltrudes Webber, que muito contribuiu para este trabalho, entendendo minhas dificuldades e permitindo contornar os problemas de forma eficaz.

E finalmente, mas com infinita importância, agradeço a Deus por ter proporcionado meios para chegar até este ponto.

SUMÁRIO

LISTA DE ABREVIATURAS E SIGLAS	7
LISTA DE FIGURAS	8
LISTA DE TABELAS	10
LISTA DE CÓDIGOS.....	11
RESUMO	12
ABSTRACT	13
1 INTRODUÇÃO	14
1.1 Motivação	15
1.2 Objetivos.....	16
1.3 Estrutura do trabalho	16
2 VISUALIZAÇÃO DE DADOS	18
2.1 Conceituação	18
2.2 Cognição.....	19
2.3 Processo de Visualização.....	20
2.4 Vantagens da Visualização de Dados.....	23
2.5 Formas de Visualização de Dados.....	28
2.6 <i>Softwares</i> para Visualização	34
2.7 Visualização de Dados na Área da Saúde	39
2.8 Considerações Finais do Capítulo	42
3 FERRAMENTA DE CONSULTA	44
3.1 Projeto OTICSSS.....	44
3.2 Ferramenta de Consulta	46
3.3 Arquitetura de <i>Software</i>	48
3.3.1 Linguagem de Programação Python.....	50
3.3.2 <i>Framework</i> Django.....	51
3.3.3 Banco de Dados PostgreSQL	51
3.3.4 Pacote Gráfico ReportLab Graphics.....	52
3.4 Considerações Finais do Capítulo	53
4 DESENVOLVIMENTO	54
4.1 Tecnologias Utilizadas	54

4.2	Caso de Uso	55
4.3	Descrição da Implementação.....	61
4.4	Considerações Finais do Capítulo	68
5	CONCLUSÃO	70
5.1	Síntese.....	70
5.2	Contribuições do Trabalho	71
5.3	Perspectivas de Trabalhos Futuros	71
	REFERÊNCIAS	73
	ANEXOS	77

LISTA DE ABREVIATURAS E SIGLAS

Sigla	Significado em Português	Significado em Inglês
BSD		Berkeley Software Distribution
CSS	Folhas de Estilo em Cascata	Cascading Style Sheets
CTI/PRC	Comitê Temático Interdisciplinar de Padronização de Registros Clínicos	
DATASUS	Departamento de Informática do Sistema Único de Saúde	
DICOM	Comunicação de Imagens Digitais em Medicina	Digital Imaging Communications in Medicine
DRY	Não se Repita	Don't Repeat Yourself
EUA	Estados Unidos da América	United States of America
GPL	Licença Pública Geral	General Public License
HTML	Linguagem de Marcação de Hipertexto	HyperText Markup Language
IHC	Interação Humano-Computador	
ISO	Organização Internacional para Padronização	International Standard Organization
OTICSSS	Observatório de Tecnologias de Informação e Comunicação em Sistemas e Serviços de Saúde	
PDF	Formato de Documento Portável	Portable Document Format
RIPSA	Rede Interagencial de Informações para a Saúde	
SAD	Sistemas de Apoio à Decisão	
SGBD	Sistema de Gerenciamento de Banco de Dados	
SI	Sistemas de Informações	
SIM	Sistema de Informações sobre Mortalidade	
SINASC	Sistema de Informações sobre Nascidos Vivos	
SIS	Sistemas de Informações em Saúde	
SUS	Sistema Único de Saúde	
URL	Localizador de Recursos Universal	Uniform Resource Locator

LISTA DE FIGURAS

Figura 1 – Processo de visualização segundo (HABER; MCNABB, 1990)	21
Figura 2 – Processo de visualização (STOREY, 2003).....	23
Figura 3 – Guerra do Vietnã Nick Ut	24
Figura 4 – Reconhecendo padrões	25
Figura 5 – Visualização das despesas através de um gráfico de pizza	26
Figura 6 – Visualização das despesas através de um gráfico de linhas	27
Figura 7 – Mapa da cidade de Konya na Turquia (FRIENDLY, 2004)	29
Figura 8 – Mapa do metrô de Londres de Beck (FRIENDLY, 2009)	30
Figura 9 – Exemplo de um grafo com muitos nodos (FRY, 2008)	31
Figura 10 - Treemaps (HOLDEN, 2007).....	32
Figura 11 - Popularidade das notícias (NEWSMAP, 2009)	33
Figura 12 – Densidade populacional dos EUA (TIME MAGAZINE, 2009).....	33
Figura 13 - Mapa de calor utilizado em um blog (CRAZYEGG, 2009)	34
Figura 14 – Relação entre as palavras (VISUAL THESAURUS, 2009)	35
Figura 15 – Gráfico de coluna gerado pelo ReportLab (REPORTLAB, 2009)	36
Figura 16 – Gráfico de coordenadas paralelas (GGOBI, 2009)	37
Figura 17 - Visualização da metabusca realizada pelo Kartoo (KARTOO, 2009).....	38
Figura 18 – Vasos sanguíneos (OPENDX, 2009)	39
Figura 19 - Informações do paciente geradas pelo Lifelines (IEPSEN, 2008).....	41
Figura 20 - Resultado da consulta (SANFELICE, 2008)	45
Figura 21 - Seleção dos filtros da consulta (SANFELICE, 2008).....	47
Figura 22 - Resultado da consulta em formato de gráfico (SANFELICE, 2008).....	47
Figura 23 - Arquitetura em camadas da ferramenta de consulta (SANFELICE, 2008).....	49
Figura 24 - Arquitetura em camadas da ferramenta de consulta com o ReportLab Graphics (SANFELICE, 2008).....	50
Figura 25 - Área de abrangência da consulta.....	55
Figura 26 - Filtros e campos de seleção para realizar a consulta.....	56
Figura 27 - Retorno da pesquisa em formato de tabela	57
Figura 28 - Taxa de mortalidade infantil em gráfico de pizza.....	58
Figura 29 - Taxa de mortalidade infantil em gráfico de linha	59
Figura 30 - Taxa de mortalidade infantil em gráfico de coluna	60
Figura 31 - Taxa de mortalidade infantil em gráfico de barra.....	60

Figura 32 - Estrutura de pastas que contém os códigos fontes	62
Figura 33 - Opção para o usuário selecionar a forma de visualização	63
Figura 34 - Formulário para o usuário preencher	67

LISTA DE TABELAS

Tabela 1 – Tabela de despesas	26
-------------------------------------	----

LISTA DE CÓDIGOS

Código 1 – Campo formas de visualização em código HTML	64
Código 2 – Trecho de código referente à URL elegante	64
Código 3 – Trecho do método que gera o gráfico em formato de pizza.....	65
Código 4 – Estrutura da tabela tperguntas	67
Código 5 - Método que grava as informações do questionário	68

RESUMO

Este trabalho apresenta um estudo sobre a Visualização de Dados na área da saúde, destacando a complexidade de se trabalhar com representações visuais em formato de gráficos com grandes quantidades de dados. A Visualização de Dados pode ajudar intuitivamente os usuários a compreender melhor as informações, principalmente quando se tratar de usuários especialistas. No Brasil, diversos sistemas de informação capturam dados da área da saúde. Entretanto, esses sistemas foram projetados de forma heterogênea, sem um padrão de armazenamento e representação dos dados. Isso implica na dificuldade de se ter uma base integrada, onde se possa buscar as informações atualizadas sobre a saúde, que irão apoiar processos de construção de conhecimento e tomada de decisão. Diante desse cenário, a Universidade de Caxias do Sul, juntamente com outras instituições, construíram uma ferramenta de consulta a bases de dados da área da saúde para gerar indicadores (segundo padrões da RIPSAs) a partir das bases de dados do DATASUS. Essa ferramenta permite realizar consultas e gera os resultados em formato de tabela. Sanfelice (2008), em um trabalho prévio incrementou a ferramenta para gerar o resultado também em formato de gráfico, fixando esse formato a um determinado tipo de informação. O presente trabalho de conclusão de curso visou desenvolver um sistema que possibilite ao usuário gerar vários tipos de gráficos para as informações disponíveis, com o objetivo de ajudar o usuário a compreender melhor as informações obtidas, construindo intuitivamente conhecimento para a tomada de decisão.

Palavras-chave: Visualização de Dados, Construção do Conhecimento, Informática na Saúde.

ABSTRACT

This work presents a study on Health Data Visualization, highlighting the complexity of working with graphical representations of large databases. It also stands out as Data Visualization can help users and experts to intuitively understand information. In Brazil, there are several information systems capturing Health data. These systems were designed in a heterogeneous way, without following any storage and data representation standards. This fact implies in the difficulty of having an integrated database, where one can seek for updated health information for knowledge acquisition and decision making processes. Given this scenario, the University of Caxias do Sul, working with other institutions, are conceiving a tool to consult health data, through RIPSAs indicators, using from DATASUS databases. This tool presents its results in table format. Sanfelice (2008), in a previous work, have developed a tool that generates results in chart specific format. In order to advance this project, aim at developing a system that allows the user to visualize graphical data during data consulting. Our intention is to help the user to better understand health information, and intuitively acquire knowledge for decision making.

Keywords: Data Visualization, Construction of Knowledge, Health Informatics.

1 INTRODUÇÃO

A Visualização de Dados procura representar graficamente um conjunto de dados de modo que a visualização gerada auxilie no processo de compreensão das informações, explorando a capacidade de cognição do ser humano. Geralmente ela envolve técnicas de computação gráfica que permitem ao usuário interagir com a representação visual, possibilitando com isso melhorar o entendimento do conjunto de dados e gerando conhecimento (WARE, 2004).

A problemática da Visualização de Dados torna-se evidente em aplicações voltadas para o apoio à tomada de decisão, onde as informações devem ser apresentadas ao usuário de forma clara e objetiva.

Problemas comuns encontrados na Visualização de Dados:

- Em formato de tabela: se a tabela tiver diversas colunas ou linhas, logo não será possível visualizar todos os dados na tela. E não é preciso ter muitos dados numa tabela para que não seja possível visualizá-los em toda tela. Uma ferramenta bastante conhecida nos *softwares* é a barra de rolagem, que possibilita o usuário visualizar o restante dos dados, mas mesmo assim não será possível visualizar todas as informações em um mesmo momento. O usuário, ou o proprietário dos dados terá grande dificuldade para analisá-los. Na maioria dos casos esse problema está relacionado com o fator quantidade. Numa massa de dados o usuário terá um grande esforço para extrair informação, conseqüentemente dificultando a geração de conhecimento.
- Em formato de gráfico: ao gerar determinados gráficos poderá haver uma grande compressão dos dados, ocasionando a falta de informações, dificultando o cruzamento das informações disponíveis, com isso prejudicando a compressão por parte do usuário. Ou seja, estão esquecendo de analisar se a nova forma de visualização realmente irá ajudar o usuário a obter conhecimento de forma mais rápida. Ou determinar se existe outra forma de visualização (outro tipo de gráfico) mais apropriada para a compreensão dos dados.

Em aplicações de análise de dados e aplicações voltadas à geração de conhecimento, o objetivo da Visualização de Dados é promover *insights*, ou seja, permitir que o usuário

associe e interprete informações, construindo conhecimento para auxiliá-lo no processo de tomada de decisão. Com o objetivo de auxiliar o decisor, os Sistemas de Apoio a Decisão (SAD) facilitam o acesso às informações e delas obtêm subsídios para a tomada de decisão.

Barbosa (2003) comenta que os SAD, quando usados de forma adequada, proporcionam uma melhor compreensão do negócio, aumentam o número de alternativas examinadas para a solução de um problema, possibilitam novos conhecimentos e aprendizagens e uma melhoria na comunicação e controle.

Será utilizada neste trabalho uma base de dados oriunda dos Sistemas de Informações (SI) do Departamento de Informática do Sistema Único de Saúde (DATASUS).

Neste cenário, o presente trabalho se propõe a analisar os sistemas de Visualização de Dados e como eles podem auxiliar a área de gestão de políticas públicas em saúde no Brasil, agregando técnicas que facilitem o entendimento de informações a partir de representações visuais de dados. A área de Sistemas de Informação em Saúde (SIS) apresenta uma série de necessidades a serem supridas, tais como: os dados estão separados em diversos sistemas desenvolvidos de forma heterogênea e em diversos bancos, que armazenam uma grande quantidade de dados; sem padrão para representação; e ainda há uma demanda crescente por informações estruturadas, de fácil visualização e em tempo real, ao qual, sem essas visualizações, exigiria maior esforço para ser compreendido.

1.1 Motivação

Este trabalho visa contribuir com o projeto de pesquisa OTICSSS (Observatório de Tecnologias da Informação e Comunicação em Sistemas e Serviços em Saúde) que se encontra em desenvolvimento pelo Núcleo de Pesquisa em Gestão das Tecnologias de Informação e Comunicação. O objetivo geral do projeto OTICSSS é o de desenvolver e implementar um conjunto de soluções tecnológicas e operacionais, visando qualificar o monitoramento e avaliação de indicadores de saúde. Também possui como objetivo criar meios que facilitem o acesso e a troca de informações entre os atores envolvidos com a informação em saúde, estabelecendo a base para um processo permanente e contínuo de gestão no sistema de saúde.

Os dados manipulados pelo sistema do projeto OTICSSS são em formato de tabelas. Para melhorar a compreensão por parte do usuário Sanfelice (2008) desenvolveu uma

ferramenta de visualização que através da qual o usuário pode construir conhecimento. Essa ferramenta complementa a Visualização dos Dados, originalmente em formato de tabelas, apresentando-os através de gráficos.

O que motiva a realização do trabalho é poder disponibilizar ferramentas que sejam úteis, que fortalecem o projeto OTICSSS, que ofereçam aos usuários uma opção diferente de Visualização de Dados que é a representação de gráficos e que os gráficos possam contribuir significativamente para a construção de conhecimento. A possibilidade de integrar o sistema ao projeto OTICSSS e avaliar o trabalho em um contexto real, podendo ter um impacto importante na organização dos dados, construção de conhecimento e conseqüentemente na tomada de decisões.

1.2 Objetivos

Os objetivos do trabalho são os seguintes:

1. O objetivo geral deste trabalho é desenvolver um sistema de Visualização de Dados para auxiliar no entendimento e na compreensão de dados na área da saúde, possibilitando ao usuário a construção de conhecimento.
2. Estudar o problema da Visualização de Dados e desenvolver uma implementação que permita complementar o sistema desenvolvido em um trabalho prévio (SANFELICE, 2008), criando mais formas e tipos de gráficos para Visualização de Dados utilizando a linguagem Python. Esse sistema de visualização será integrado ao projeto OTICSSS.
3. Propor uma interface a partir da qual usuários especialistas possam informar sua percepção sobre o gráfico gerado.

1.3 Estrutura do trabalho

Este trabalho está organizado da seguinte forma:

O capítulo 2 apresenta os conceitos pertinentes aos sistemas de Visualização de Dados, seu processo, suas vantagens, algumas formas e ferramentas de visualização.

Destacando a importância de um projeto adequado de visualização de informações ao se trabalhar com grandes bases de dados na área da saúde.

O capítulo 3 apresenta o projeto OTICSSS, a ferramenta de consulta às bases de dados da área da saúde e a ferramenta de visualização gráfica desenvolvida por Sanfelice (2008). Também apresenta as tecnologias utilizadas para o desenvolvimento dessas ferramentas.

O capítulo 4 apresenta o desenvolvimento do sistema de Visualização de Dados, explicando a implementação realizada, bem como, demonstrando um cenário de uso do sistema desenvolvido com os resultados obtidos.

O capítulo 5 descreve conclusões baseadas no desenvolvimento deste trabalho e nos resultados obtidos, bem como contribuições desse trabalho para área Visualização de Dados e sugestões de trabalhos futuros.

2 VISUALIZAÇÃO DE DADOS

Este capítulo aborda conceitos e métodos fundamentais da área de Visualização de Dados, além de aplicações e vantagens do uso de sistemas de representação visual dos dados. Finaliza-se o capítulo tratando da problemática particular ligada a Visualização de Dados na área da saúde, tema central deste trabalho.

2.1 Conceituação

Segundo o dicionário Michaelis (2009) pode-se definir visualização como: transformação de conceitos abstratos em imagens reais ou mentalmente visíveis; conversão de números ou dados para um formato gráfico, que pode ser mais facilmente entendido. Já Storey (2003) define visualização como uma representação visual dos dados. Com o auxílio dos computadores e *softwares* específicos para gerar visualizações, a representação visual tem como objetivo ampliar a cognição, facilitando o processo de construção de conhecimento (STOREY, 2003). Friendly (2009) complementa a definição de visualização como ciência da representação visual dos dados.

Visualização de Dados é uma área de aplicação de técnicas de computação gráfica, geralmente interativas, visando auxiliar o processo de análise e compreensão de um conjunto de dados, através de representações gráficas manipuláveis (WARE, 2004). Para Nascimento et al (2005) a Visualização de Dados é uma área que estuda formas de transformar dados abstratos em imagens reais ou mentalmente visíveis, de forma a facilitar o seu entendimento e/ou ajudar na descoberta de novas informações contidas nestes dados.

O principal objetivo da Visualização de Dados para Storey (2003) é promover *insights*, ou seja, ativar o desenvolvimento da percepção interna das pessoas. Ao estimular naturalmente a percepção e a inteligência humana, a visualização aumenta a capacidade de entendimento, revelando novos padrões e relações entre os dados.

Segundo Fayyad et al (2002), uma visualização pode facilitar uma visão geral qualitativa de conjuntos complexos de dados, como pode também auxiliar a sintetizar esses dados facilitando assim, identificar regiões proveitosas para uma análise quantitativa.

Se o volume de dados que o usuário está trabalhando for pequeno, é muito provável que a representação visual dos dados não ganhe tanta importância. Porém, quando se trabalha com uma grande quantidade de dados, a representação visual dos dados torna-se essencial. A Visualização de Dados tem a função de tornar os dados mais atrativos e compreensíveis para os usuários (TURBAN; ARONSON, 2000). Sanfelice (2008) reforça a importância da visualização dos dados quando se tem grandes quantidades de dados, mas também cita sobre a complexidade que envolvem os sistemas que geram representações visuais.

“De acordo com Fry (2008), a maioria dos usuários de sistemas de informação está acostumada a pensar em “dados” como valores fixos a serem analisados. Entretanto, com o aumento da capacidade de processamento e armazenamento dos computadores e do avanço das tecnologias de informação, esta idéia de dados “estáticos” é errônea. Os dados são modificados de forma dinâmica e em tempo real, criando uma necessidade de visualização que se ajustam a estes novos valores a cada momento. Além disso, está cada vez mais fácil de desassociar os dados da razão pela qual foram coletados, permitindo que os dados e as informações sejam utilizados de múltiplas formas e em diferentes contextos. Neste sentido, Fry (2008) levanta uma pergunta pertinente: Como é possível visualizar tantos dados?”.

Resumindo, uma representação visual adequada dos dados é como uma narrativa, fornecendo uma resposta clara e precisa para determinada questão, eliminando detalhes extras (FRY, 2008).

2.2 Cognição

É muito importante explicar a cognição, pois as imagens geradas pelos sistemas de Visualização de Dados exploram bastante a percepção humana, devido a uma série de vantagens naturalmente apresentadas pela capacidade visual humana.

Segundo o dicionário Michaelis (2009) pode-se definir cognição como ato de adquirir um conhecimento; faculdade do conhecimento. Já o dicionário Aurélio (2009) define cognição como faculdade, ato ou ação de conhecer; aquisição de um conhecimento.

Geralmente os seres humanos percebem as coisas através de três sentidos: visual, auditivo e tato. Segundo Alexandre (2006), esses sentidos habilitam o usuário de um sistema interativo a perceber a informação, armazená-la (em sua memória) e processar a informação usando o raciocínio dedutivo ou indutivo. Esse autor deixa claro na citação seguinte a importância da visão humana na exploração da visualização.

“A visualização da informação explora principalmente o sentido humano que possui maior aptidão para captação de informação temporal: a visão. Além de ser o primeiro componente do sistema sensorial, a visão é o sentido mais rapidamente captado pelo cérebro e possui ainda capacidade de paralelismo; isto é, mesmo tendo a atenção focada num ponto de uma cena, o que lhe circunvizinha num raio bastante largo, também é alvo de nossa visão.”

Geralmente quando o homem interage com o computador é através do sentido da visão, como por exemplo: relatórios, gráficos e outros. Neste caso, o olho e o cérebro trabalham juntos a fim de receber e interpretar a informação visual em alta velocidade, baseada no tamanho, forma, cor, orientação e movimento (WARE, 2004).

Muito da informação visual ainda é apresentada no formato de textos e tabelas. Os seres humanos precisam interpretar os padrões visuais e recuperar o significado dos dados dispostos gerando informação. Neste caso, a velocidade do processo de extração da informação é controlada pelo movimento do olho ao escanear os dados. Em grandes volumes de dados esse processo pode ser bastante demorado (WARE, 2004). Por isso a tendência de se utilizar sistemas que geram informações em imagens, permitindo facilidade e agilidade no processo de leitura, ainda mais quando estamos trabalhando com uma grande quantidade de dados. Ware (2004) apud Alexandre (2006) cita sobre a importância dos sentidos humanos na percepção.

“Assim todos os sentidos da percepção humana tem um papel importante na área de Visualização e podem melhorar significativamente tanto a qualidade como a quantidade de informação que é apresentada através de imagens”.

2.3 Processo de Visualização

De acordo com Malhotra (2001), é necessário compreender o processo de construção de conhecimento das pessoas a fim de entender se as informações processadas por sistemas de informações são apropriadas para os usuários, ou seja, se os usuários conseguem atribuir um significado às informações para, então, transformar estes significados em ações. Barbosa (2003) complementa que informação não se limita a dados coletados, é necessário que eles sejam organizados de forma a se tornarem úteis e assim se transformarem em informação.

Fry (2008) cita alguns passos para facilitar a definição de um projeto de Visualização de Dados. Os passos de um processo de visualização são os seguintes:

1. Aquisição: ocorre a aquisição dos dados de qualquer fonte, a captura dos dados propriamente dita, por exemplo: obter os dados através de um arquivo;

2. Análise: projeta uma estrutura para armazenar os dados. Esses dados geralmente são ordenados em categorias para tornarem mais úteis quando forem manipulados;
3. Filtro: ocorre a remoção de todos os dados desnecessários. Nesta fase, algoritmos matemáticos são frequentemente aplicados para normalizar formatos numéricos;
4. Mineração: nesta fase são aplicados métodos estatísticos, métodos de mineração de dados. Tendo objetivo descobrir padrões ou posicionar dados dentro de um contexto matemático;
5. Representação: nesta fase um modelo básico de representação visual é escolhido para representar os dados, como: gráficos de barra, listas e árvores;
6. Refinamento: aqui o objetivo é melhorar a representação visual básica, torná-la mais clara e visualmente atraente;
7. Interação: são adicionados métodos para manipular e interagir os dados, permitindo ao usuário controlar e explorar os dados.

Dependendo da quantidade de dados, das estruturas dos dados, do modelo de visualização que se pretende obter e outros fatores, não é necessário seguir todos os passos. Por exemplo, Campo et al (1997) cita três fases no processo de visualização: preparação dos dados e mapeamento e a representação.

A seguir é citado o esforço que é desenvolver uma ferramenta de visualização quando envolve diversas disciplinas ou áreas.

“O desenvolvimento de uma ferramenta personalizada de visualização demanda o envolvimento de diferentes áreas para resolver partes de um único processo. Fry (2008) aponta que o envolvimento de diversas pessoas como um possível problema a ser enfrentado pelo informata. Assim, é necessário um esforço maior para coordenar os trabalhos e evitar que informações importantes ao projeto não sejam perdidas no caminho (SANFELICE, 2008).”

Um processo de visualização permite a identificação dos componentes essenciais a serem considerados na utilização de uma determinada técnica ou no desenvolvimento de uma nova. Haber e McNabb (1990) propuseram um *pipeline* simples (Figura 1), onde dados sofrem filtragem e mapeamento para alguma representação geométrica, a qual, finalmente, passa por um processo de geração de imagem (*rendering*).

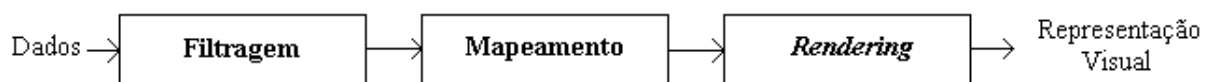


Figura 1 – Processo de visualização segundo (HABER; MCNABB, 1990)

O processo de visualização de Storey (2003), representado pela Figura 2, começa organizando os dados em tabelas para facilitar as consultas. A seguir as estruturas visuais são definidas, a fim de representar as informações como: gráficos de barra, diagramas, mapas e esquemas. Para tal, é realizada uma transformação dos atributos de entidade (tabela de dados) para formas gráficas espaciais representativas. Por fim, as visões permitem a interação com os dados, adicionando funções de manipulação de consulta. Para cada avanço nas etapas é aplicado uma transformação. A geração de uma imagem a partir de um conjunto de dados prevê três etapas descritas a seguir:

Etapa 1 - Transformações de dados: a tarefa aqui é processar e organizar um conjunto de dados brutos em uma representação lógica mais estruturada, geralmente na forma de uma ou mais tabelas. Esse processo pode remover dados redundantes, errados ou incompletos. Como pode também fazer a filtragem agrupando os dados importantes, permitindo incluir novas informações, como, por exemplo, de resultados de análises estatísticas (média, soma total, desvio padrão, etc.) realizadas sobre os dados brutos;

Etapa 2 - Mapeamento Visual: nesta etapa os dados da tabela são representados em uma estrutura visual (imagem). A estrutura visual é composta por três partes: substrato espacial, marcas visuais e propriedades gráficas das marcas;

- O substrato visual: é caracterizado pelo espaço da visualização, sendo geralmente representado por eixos, tais como os eixos X e Y do plano cartesiano;
- As marcas visuais: são símbolos utilizados para representar os itens de dados. Alguns tipos de marcas consistem figuras geométricas, linhas, figuras simples também chamadas de ícones, entre outras;
- Propriedades gráficas das marcas: são os atributos visuais que caracterizam as mesmas. Exemplo tamanho, volume, ângulo, inclinação, orientação, textura, entre outras.

Etapa 3 - Transformação Visual: nesta etapa são criados mecanismos para interagir com as estruturas visuais modificando-as através de operações básicas como:

- Testes de localização: é possível obter informações adicionais sobre um item da tabela de dados;

- Controles de ponto de vista: são funções que permitem ampliar, reduzir e deslocar a imagem com o objetivo de proporcionar visões alternativas;
- Distorções da imagem: visa criar distorções e ampliações de uma região específica da imagem.

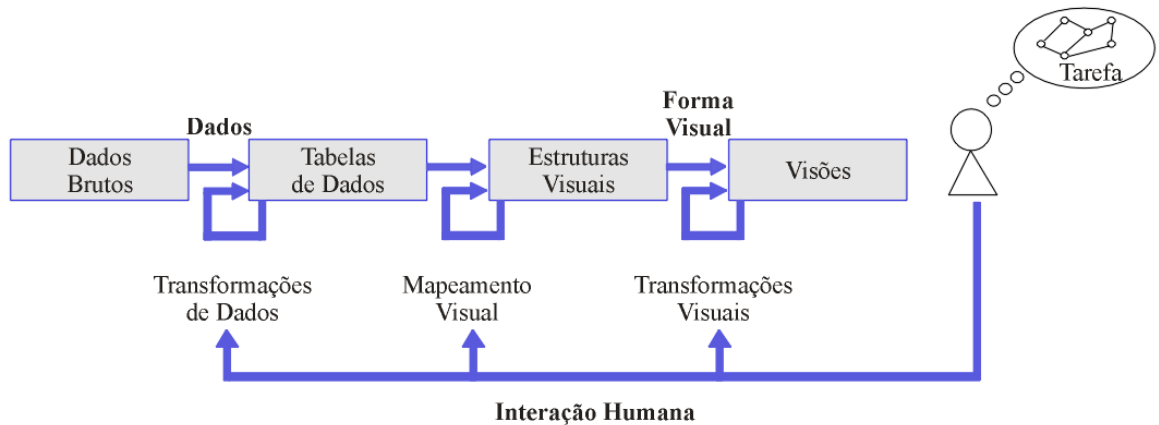


Figura 2 – Processo de visualização (STOREY, 2003)

Os mecanismos implementados na fase de transformação visual, permitem que o usuário explore diferentes cenários com o intuito de auxiliar na compreensão dos dados visualizados. Além disso, o esforço da geração desses novos cenários ou modificações nas imagens fica a cargo do *software*, deixando para o usuário a tarefa de analisar as alterações nos gráficos buscando novas informações.

2.4 Vantagens da Visualização de Dados

O ditado popular “uma imagem vale mais do que mil palavras” resume bem essa ideia de uma simples visualização poder expressar uma grande quantidade de dados. A foto tirada por Nick Ut em 1972 (Figura 3) que mostra uma menina vietnamita de nove anos correndo nua juntamente com outras pessoas, e que ganhou o prêmio Pulitzer, é um exemplo típico do poder expressivo das imagens. Não é preciso muitas palavras para entender o que a foto transmite e sabe-se hoje que ela teve um grande impacto nas decisões tomadas pelos Estados Unidos sobre a sua participação na guerra do Vietnã.



Figura 3 – Guerra do Vietnã Nick Ut

Algumas vantagens de Ware (2004):

- Fornece a capacidade de compreender grandes quantidades de dados;
- Permite perceber características que não são encontrados apenas analisando os dados;
- A visualização permite que problemas presentes nos dados sejam rapidamente identificados;
- Facilita o entendimento de padrões, tanto em pequenas como em grandes quantidades de dados;
- Simplifica na elaboração de hipóteses.

Outra vantagem para explorarmos a visualização é que ele envolve o sistema visual humano, tendo maior capacidade de captar informações por unidade de tempo. O sentido da visão é rápido, sendo possível também, prestar atenção em um objeto, sem perder de vista (obviamente, com menos detalhes) o que está acontecendo ao redor.

Além disso, o sistema visual humano tem um enorme poder para reconhecer padrões. Podemos verificar essa capacidade através da Figura 4. Tente encontrar a elipse presente na imagem. Note que é possível localizá-la rapidamente no meio dos quadrados. O reconhecimento da elipse seria ainda mais rápido se ela estivesse pintada de outra cor.

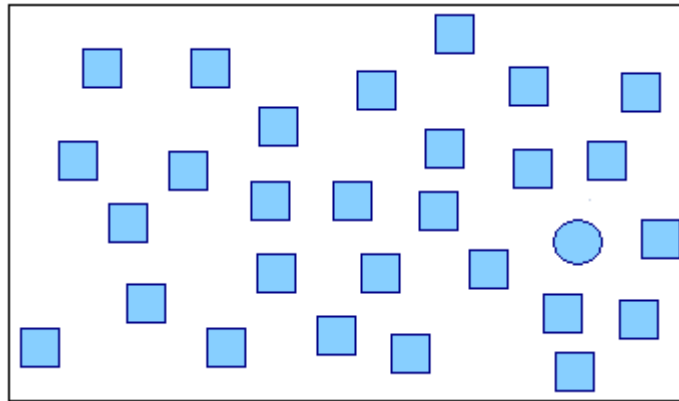


Figura 4 – Reconhecendo padrões

A seguir Nascimento et al (2005), complementa em sua visão sobre os benefícios que a visualização trás.

“As visualizações por si só trazem benefícios, uma vez que podem funcionar como uma extensão da memória humana e como um auxílio para o processo cognitivo. Por exemplo, fazemos anotações em uma agenda ou em um calendário para nos lembrarmos de assuntos a serem discutidos ou de eventos que ocorreram ou que irão ocorrer. Também desenhamos diagramas e organizamos informações espacialmente em uma folha de papel quando estamos estudando um problema que envolve diversas partes. As imagens nos ajudam a entender o problema e/ou a encontrar uma solução para o mesmo. Elas, inclusive, facilitam a memorização do objeto em estudo.”

A Tabela 1 mostra os gastos mensais de uma pessoa com despesas domésticas durante o período compreendido, entre os meses de janeiro à dezembro. Podemos levantar várias questões interessantes sobre os dados da tabela, tais como: “qual é a menor despesa durante o período?”, “qual é a segunda menor despesa?”, “qual despesa variou continuamente com o tempo?”, “qual despesa apresentou uma possível redução?”.

Com base apenas na Tabela 1, as duas primeiras questões podem ser respondidas sem dificuldade. Entretanto, para responder as últimas duas perguntas é necessário uma análise mais detalhada dos dados. Contudo, se fosse construída uma representação visual baseada em um gráfico de pizza com os dados dessa tabela, como ilustrado na Figura 5, então não apenas as duas primeiras questões podem ser facilmente respondidas, mas também várias outras questões relacionadas com a totalidade que cada despesa representa no período.

Tabela 1 – Tabela de despesas

Meses	Internet	Comida	Celular	Passagens	Total
Jan	75,00	250,00	40,00	12,50	377,50
Fev	75,00	260,00	52,00	115,00	502,00
Mar	75,00	255,00	62,00	115,00	507,00
Abr	75,00	251,00	61,00	115,00	502,00
Mai	75,00	250,00	58,00	115,00	498,00
Jun	75,00	250,00	56,00	115,00	496,00
Jul	75,00	240,00	55,00	115,00	485,00
Ago	75,00	265,00	55,00	115,00	510,00
Set	75,00	275,00	55,00	115,00	520,00
Out	75,00	252,00	54,00	115,00	496,00
Nov	75,00	249,00	60,00	110,00	494,00
Dez	75,00	280,00	200,00	105,00	660,00

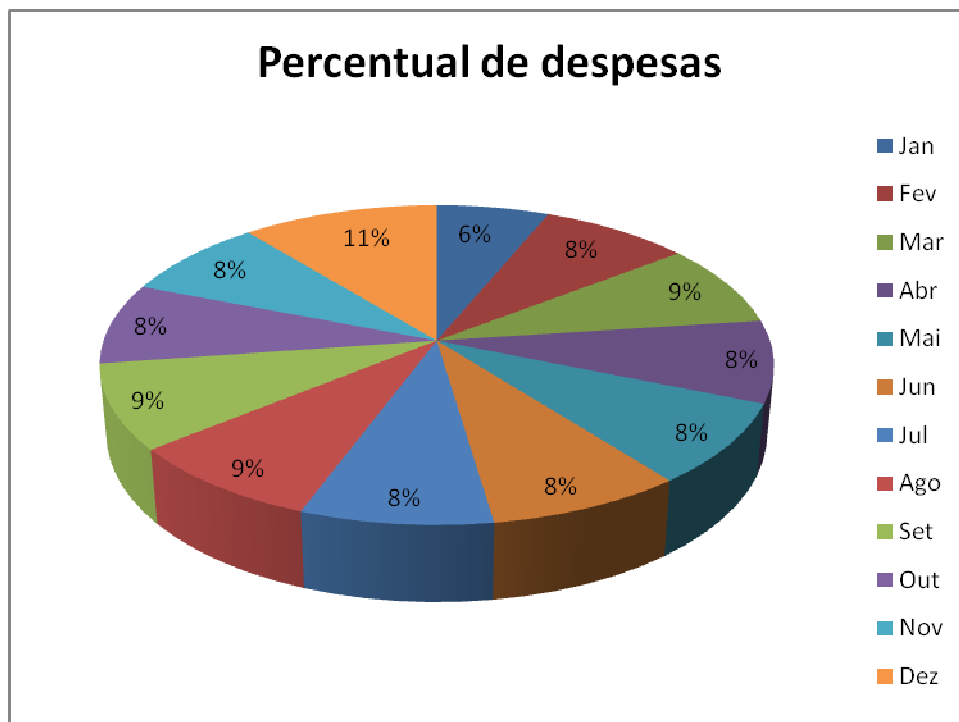


Figura 5 – Visualização das despesas através de um gráfico de pizza

Aprofundando um pouco mais a análise das despesas, pode-se visualizar os mesmos dados através de um gráfico de linhas, como ilustrado na Figura 6. Agora as respostas para as

duas últimas perguntas ficam mais claras: o gasto com celular foi à despesa que mais variou durante o período e o gasto com passagens foi o que apresentou uma tendência de redução.

O mais interessante nessa análise é que, olhando a Figura 6, a pessoa percebeu que houve um momento em que a despesa com as passagens se aproximou de zero. Vendo os meses dessa ocorrência, foi possível recordar que isso se deu as férias no trabalho, quando o mesmo estava em casa sem gastar dinheiro em passagens para ir trabalhar.

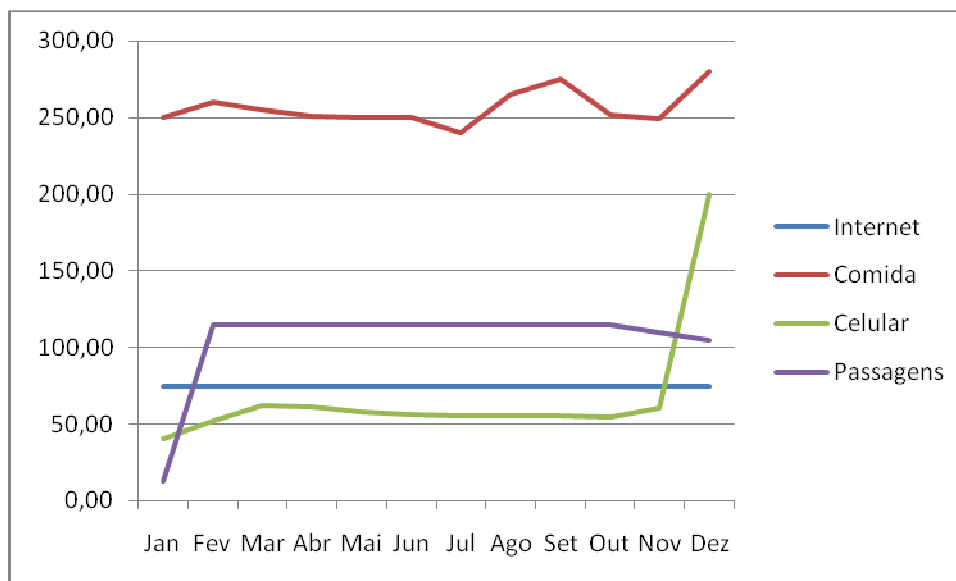


Figura 6 – Visualização das despesas através de um gráfico de linhas

Ao se constatar padrões presentes nas visualizações essas contribuem significativamente para o processo de construção de conhecimento, coisa que não seria tão trivial a observação dos dados em sua forma bruta. Para isso é necessário organizar os dados seguindo algum critério e gerando sua representação visual, como é possível perceber nas figuras anteriores. Geralmente as visualizações acabam recuperando informações relevantes e tornando possível a construção de novos conhecimentos (FRY, 2008).

Ao gerar determinados gráficos pode haver uma grande compreensão dos dados, ocasionando perda de informação, ou também podem ser gerados somente com uma parte dos dados, que foi o caso anterior. Onde a tabela de despesas nos mostra diversas informações: os meses, as despesas, o valor respectivo entre o mês e a despesa e o valor total de cada mês somando todas as despesas. Com essas informações conseguimos responder as quatro perguntas propostas, isso depois de determinado tempo analisando os dados. Ao contrário da tabela de despesas é possível responder as duas primeiras perguntas rapidamente,

simplesmente visualizando o gráfico de pizza, pois nesse gráfico possui como informação somente o percentual de cada despesa. Da mesma forma o gráfico de linhas foi gerado somente com informações de valor versus mês e as linhas realizando a intersecção entre as duas informações, que possibilitou responder o restante das perguntas rapidamente.

Mesmo com uma tabela com poucos dados foi possível demonstrar a dificuldade de representar graficamente os dados, de modo que todas as informações estejam presentes. A sobrecarga dos dados é umas das principais preocupações na área de Visualização de Dados.

Estabelecida a importância de um sistema possibilitar gerar diversos tipos de gráficos, de modo que as representações visuais geradas explorem a capacidade de percepção do homem e este, a partir das relações espaciais exibidas, interprete e compreenda as informações apresentadas gerando conhecimento.

2.5 Formas de Visualização de Dados

Um dos principais objetivos da Visualização de Dados é desenvolver o raciocínio e ajudar na compreensão das informações. Sempre procurando desenvolver métodos para melhor representar os dados e as informações e assim, alcançar o objetivo da Visualização de Dados. Para dados estatísticos, que é formato de dados a ser trabalhado neste trabalho, as formas de visualização mais utilizadas são gráficos de: colunas, barras, pizzas e linhas.

Existem diversas formas de Visualização de Dados. As mais antigas formas de visualização encontram-se na pré-história, quando os homens da época expressavam visualmente o seu viver através de desenhos e pinturas nas cavernas ou rochas. Geralmente essas expressões artísticas registraram atividades e hábitos humanos, sendo hoje um importante subsídio para o entendimento de nossa evolução cultural. Caminhando um pouco mais na linha do tempo, tabelas de posicionamento de estrelas e corpos celestiais, diagramas geométricos para construção de mapas de navegação e exploração. Com o passar do tempo, a geometria analítica, a matemática, a estatística demográfica e a necessidade de demonstrar de forma mais clara as características dos números empíricos, como tendências e padrões, alavancaram o estudo da área de Visualização de Dados (FRIENDLY, 2009).

A Figura 7 exhibe graficamente o mapa da cidade de Konya na Turquia, um dos mais antigos da humanidade, aproximadamente 6200 anos antes de Cristo.



Figura 7 – Mapa da cidade de Konya na Turquia (FRIENDLY, 2004)

Um marco importante na Visualização de Dados foi o mapa de metrô de Londres. O inglês Harry Beck, em 1933, percebeu que os passageiros de metrô não precisavam saber exatamente a posição geográfica por onde as linhas dos trens passavam para poder usufruir eficientemente esse transporte. Verificou-se que, o importante no subsolo, era ter um conhecimento das estações e de quais linhas do metrô se conectavam. Assim, um passageiro poderia planejar a sua viagem considerando apenas as estações de partida e de destino e uma rota viável entre elas. Beck inspirado pela simplicidade dos diagramas elétricos então elaborou um mapa de metrô para Londres que desconsiderava a posição geográfica precisa das linhas ferroviárias, a favor de linhas de conexão mais retas e de espaçamento uniforme entre as estações. Algumas propriedades geográficas foram mantidas no mapa tais como: a direção e a orientação geral das linhas (Norte-Sul e Leste-Oeste) e a distância relativa entre estações adjacentes. A abordagem de visualização desenvolvida por Harry Beck foi publicada em 1933 e é utilizada hoje nos principais metrôs do mundo por ser simples e bastante intuitiva (FRIENDLY, 2009). A Figura 8 ilustra o mapa do metro de Londres desenvolvido por Beck.



Figura 8 – Mapa do metrô de Londres de Beck (FRIENDLY, 2009)

Com o passar do tempo, as técnicas de Visualização de Dados foram evoluindo, motivadas pelo desenvolvimento da computação e pela necessidade crescente de organizar e apresentar cada vez mais, grandes volumes de dados. Contudo, com aumento do poder de processamento dos computadores e pelo acesso a diversas redes, nos geram grandes quantidades de dados e a informação sofre cada vez mais transformações. Por isso se faz necessário, encontrarmos uma forma simples de representá-la, ou seja, desenvolver formas criativas de organizar e visualizar grandes volumes de informações (SANFELICE, 2008).

Quando encontramos a forma de representar e apresentar a informação, seja ela qual for, temos também a responsabilidade de transmitir os resultados de forma clara, eficiente e objetiva. Começam já a surgir diversas formas de representação, sendo algumas delas incrivelmente sofisticadas.

A forma de visualização de árvore permite a representação de informações hierárquicas, ou seja, cada informação pode ter diversos subelementos esses chamados de nodo. A visualização de um grafo é uma forma eficiente para representar as conexões (arestas) entre elementos (nodos). Porém, grafos não são representações ideais para grandes

quantidades de dados, com algumas dezenas de nodos, o grafo se torna ilegível (Figura 9) (FRY, 2008).

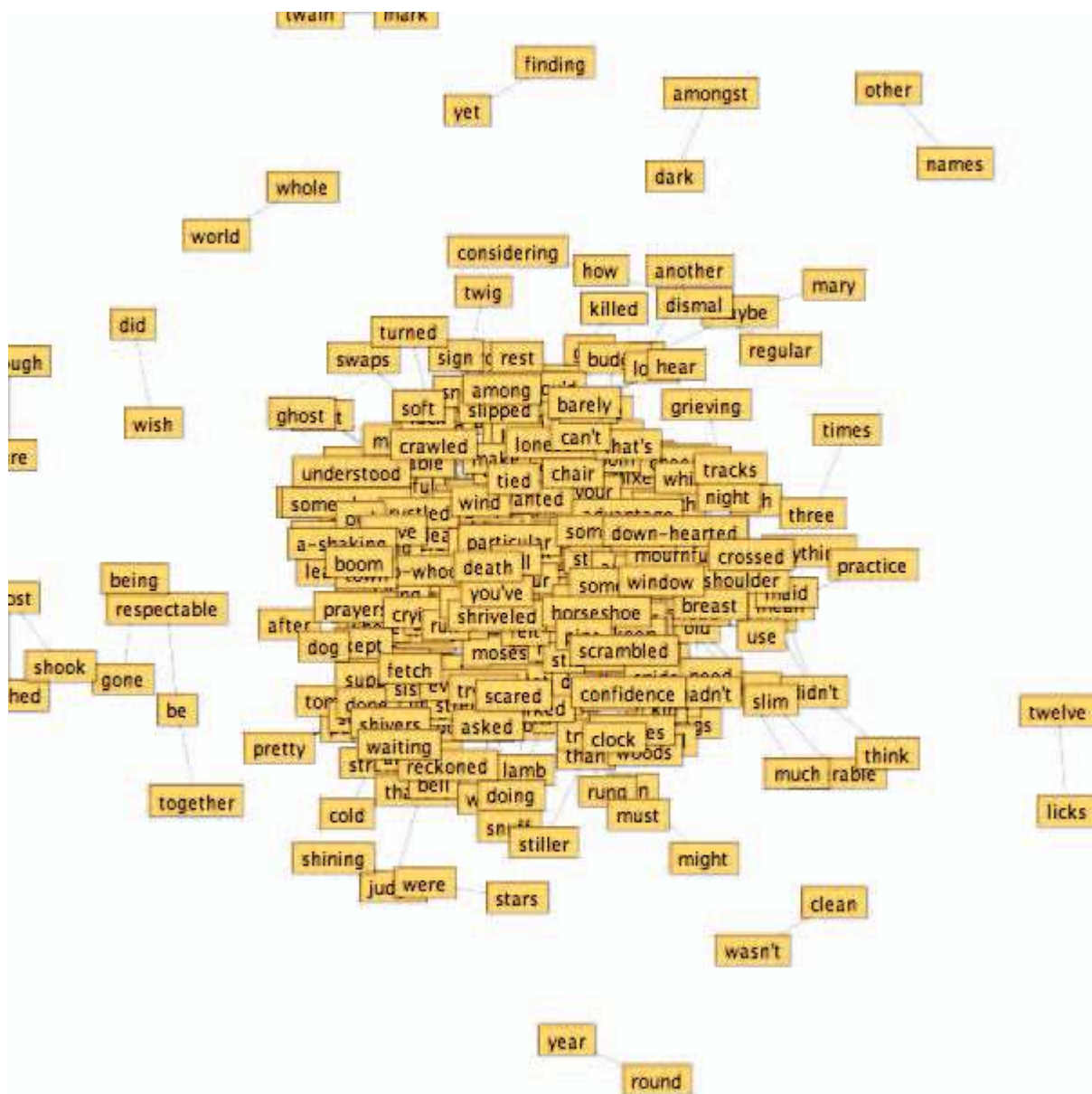


Figura 9 – Exemplo de um grafo com muitos nodos (FRY, 2008)

Para melhorar a representação visual de grafo no que diz respeito à grande quantidade de dados surgem os *Treemaps*: representação de uma árvore com um conjunto de retângulos aninhados. *Treemaps* são mapas bidimensionais que facilitam a visualização de estruturas hierárquicas para propiciar uma rápida tomada de decisão. Tais mapas exploram conceitos básicos de ergonomia que estabelecem que o ser humano foque inicialmente seu olhar em figuras grandes para só depois atentar para figuras menores. As dimensões (área da figura) e cor são utilizadas para codificar os atributos (importância econômica, desempenho, entre

outros) dos nodos folhas. Com essas informações, essa ferramenta gráfica permite uma rápida análise de nodos e subárvores facilitando comparações e permitindo a detecção de padrões e exceções (HOLDEN, 2007). Uso de *treemaps* para visualização das ações no mercado financeiro mundial, representado pela Figura 10.



Figura 10 - Treemaps (HOLDEN, 2007)

O NewsMap (Figura 11) é um serviço que muda constantemente em função da agregação de notícias do *Google News* organizando os títulos em blocos temáticos (mundo, saúde, tecnologia e outros). O tamanho dos blocos de informação reflete a popularidade de um item num determinado momento. Muito útil para quem deseja saber sobre as principais notícias ou assuntos que o mundo está falando.



Figura 11 - Popularidade das notícias (NEWSMAP, 2009)

Baseado no conceito utilizado no histograma e afastando um pouco da tradicional representação de gráficos ou tabelas, a Time Magazine disponibiliza a representação da densidade populacional dos Estados Unidos da América (EUA) sob a forma de picos (Figura 12).

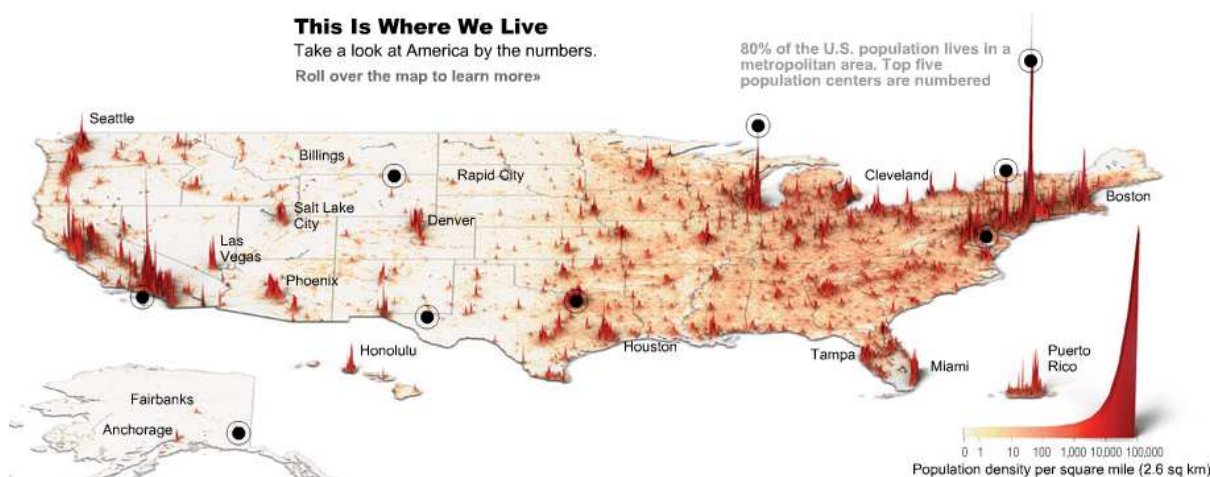


Figura 12 – Densidade populacional dos EUA (TIME MAGAZINE, 2009)

CrazyEgg (2009) dispõe um serviço extremamente útil para simular os cliques dos usuários. Com esse serviço é possível perceber qual o comportamento dos visitantes através da representação de um “mapa de calor”. As seções mais populares (onde os utilizadores mais

clicam) são mostradas com cores mais quentes, próximas do vermelho. Esse serviço tem o intuito de testar diferentes versões de uma página para verificar qual funciona melhor no sentido de qual é o design que melhor encoraja os visitantes a clicar em links que vão para outras páginas do site. Em testes realizados pela própria empresa, durante semanas, um dos resultados que se destaca bastante são os elementos gráficos, particularmente aqueles cuja cor salta à vista, recebem maior atenção. Contudo ficando evidente a grande eficácia da percepção do comportamento dos leitores. A Figura 13 ilustra um exemplo da utilização do mapa de calor num determinado site.



Figura 13 - Mapa de calor utilizado em um blog (CRAZYEGG, 2009)

2.6 Softwares para Visualização

O Visual Thesaurus é um dicionário inglês interativo que visualiza as conexões entre palavras semanticamente relacionadas. Utilizando o conceito de árvores, a representação explícita das relações semânticas por meio de nós interligados e diferentes tipos e cores de traços. Na Figura 14 foi procurado pela palavra “*information*”, essa assumindo a posição central. Ao redor cada bolinha representa um significado diferente. As linhas cheias levam a sinônimos, e as linhas pontilhadas levam a sugestões ou palavras relacionadas. Esse tipo de visualização neural e dinâmica não seria possível em ambientes tradicionalmente propícios para as palavras, como o papel e a fala, tornando a ferramenta bastante lúdica na

aprendizagem das palavras com uma interface inovadora que incentiva a exploração e aprendizagem (JUDELMAN, 2004).

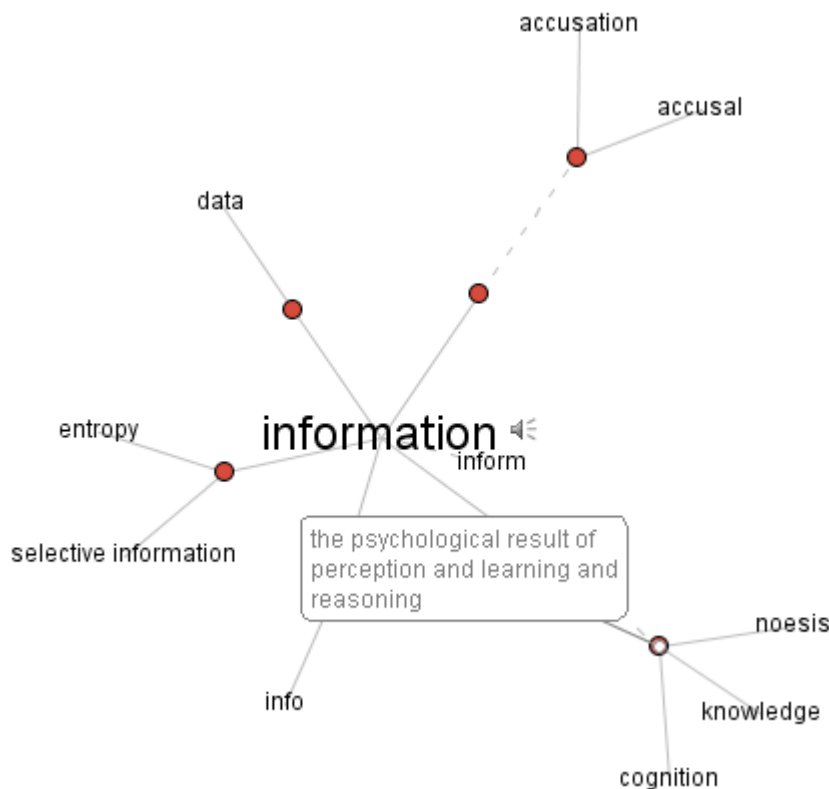


Figura 14 – Relação entre as palavras (VISUAL THESAURUS, 2009)

Os *softwares* de Visualização de Dados estão cada vez mais complexos, por gerar representações visuais organizando e representando grandes quantidades de dados de forma simples, criativa e de fácil compreensão. Para trabalhos mais comuns, existem diversos sistemas disponíveis no mercado tanto livres como pagos. Entretanto, ferramentas que trabalham com grandes volumes de dados ou ferramentas personalizadas requerem tratamento único. Serão apresentados de forma sucinta alguns *softwares* especializados em Visualização de Dados complexos. Com exceção do ReportLab Graphics (REPORTLAB, 2009), esses sistemas não serão empregadas na implementação deste trabalho, uma vez que não atendem aos requisitos necessários da ferramenta.

ReportLab Graphics é uma biblioteca de classes que permite gerar dinamicamente gráficos em diversos formatos através da linguagem Python. Atualmente o ReportLab Graphics é um conjunto de pacotes do ReportLab, sistema que permite a criação de documentos no formato PDF (*Portable Document Format*) também utilizando a linguagem de

programação Python (REPORTLAB, 2009). A Figura 15 exemplifica um gráfico gerado pelo ReportLab Graphics.

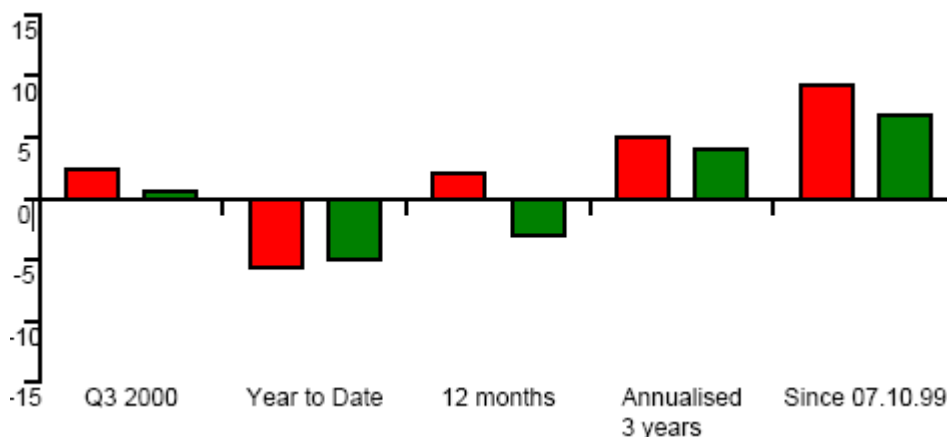


Figura 15 – Gráfico de coluna gerado pelo ReportLab (REPORTLAB, 2009)

Ggobi (2009) é uma ferramenta estatística de código aberto utilizada para visualizar graficamente diferentes tipos de dados. Permite uma análise exploratória dos dados multidimensional ao gerar gráficos dinâmicos no qual pode-se navegar. Seu propósito é que olhando para gráficos diferentes pode revelar mais informações, do que olhando apenas para números ou um resumo de dados. Usando *clusters*, distribuições não-lineares e *outliers* com o objetivo de extrair mais informações dos dados que os métodos tradicionais de visualização. Ele disponibiliza gráficos primeira dimensão (histogramas e gráficos de barras), gráficos de segunda dimensão (gráficos de dispersão), séries temporais, gráficos de terceira dimensão (matriz de dispersão e coordenadas paralelas – ilustradas na Figura 16) e outros (GGOBI, 2009).

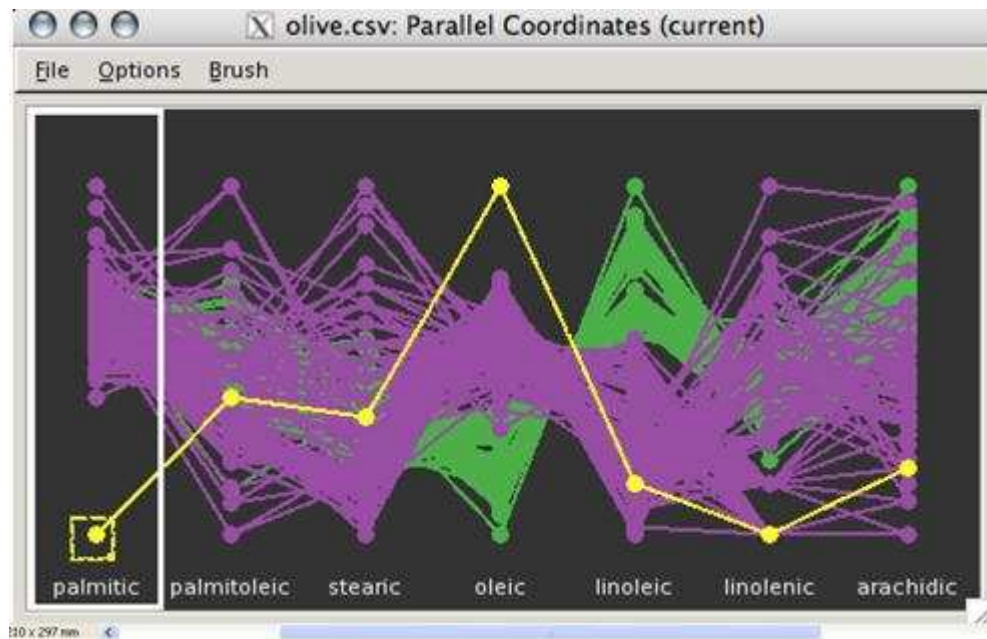


Figura 16 – Gráfico de coordenadas paralelas (GGOBI, 2009)

O Kartoo é um sistema que se baseia em metabuscas, ou seja, são serviços de busca que utilizam diversos mecanismos simultaneamente de pesquisa de informação na *Word Wide Web* (WWW). O resultado de uma pesquisa é reunido e mostrado numa apresentação única em forma de gráfico. A empresa francesa Kartoo especializada em interface visual, recuperação de informação e gerência do conhecimento propôs a utilização de princípios cartográficos para visualizar os resultados. Os sites retornados a partir da busca são montados em ícones menores ou maiores dependendo da sua pertinência. A seguir Figura 17 mostrando uma busca utilizando o Kartoo (KARTOO, 2009).

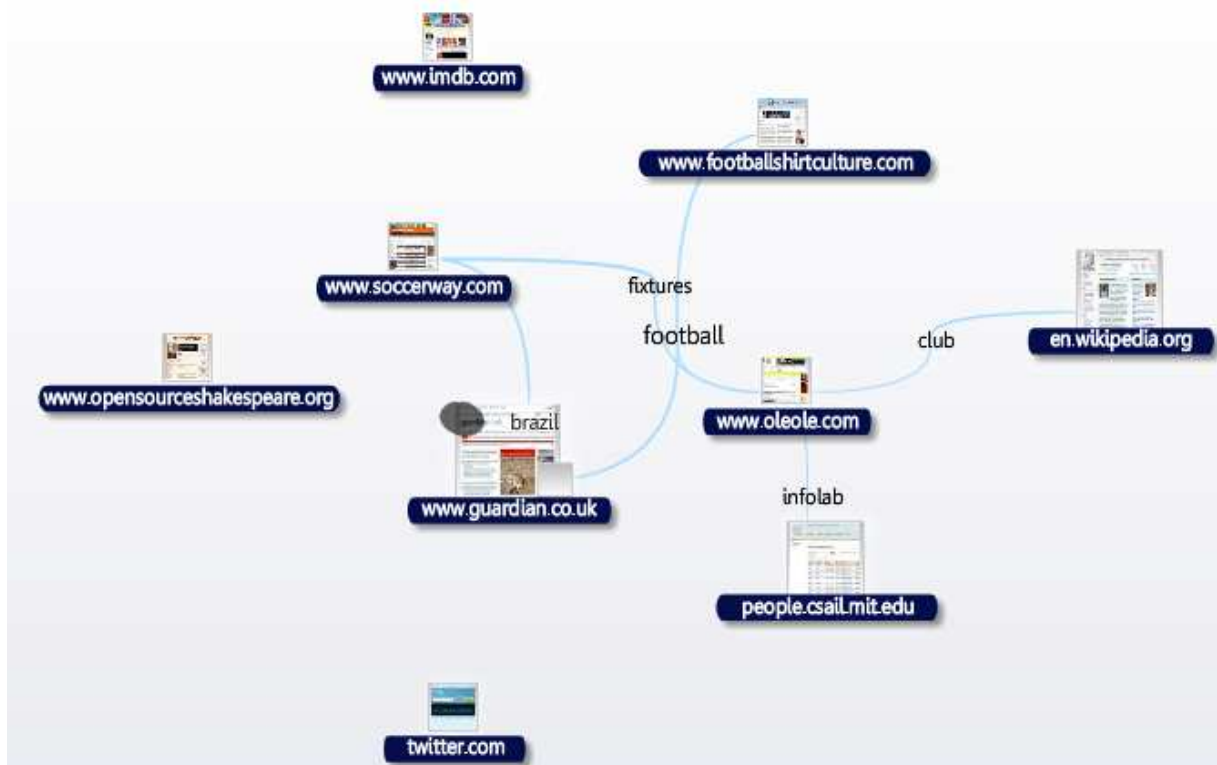


Figura 17 - Visualização da metabusca realizada pelo KARTOO (KARTOO, 2009)

OpenDX é um *software* desenvolvido pela IBM para a visualização de alta qualidade de dados bidimensionais e tridimensionais. Consiste num poderoso sistema por possibilitar gerar visualizações de forma interativa abrangendo dados de diversas áreas de aplicações: antropologia, astronomia, biologia, química, medicina, petróleo, gás, oceanografia, física, matemática, geográfica e outras mais. A Figura 18 apresenta a visualização de alguns vasos sanguíneos de um crânio humano utilizando o OpenDX. É composto por módulos, permitindo tratar os dados, a visualização e os algoritmos separadamente (OPENDX, 2009 apud SALFELICE, 2008).

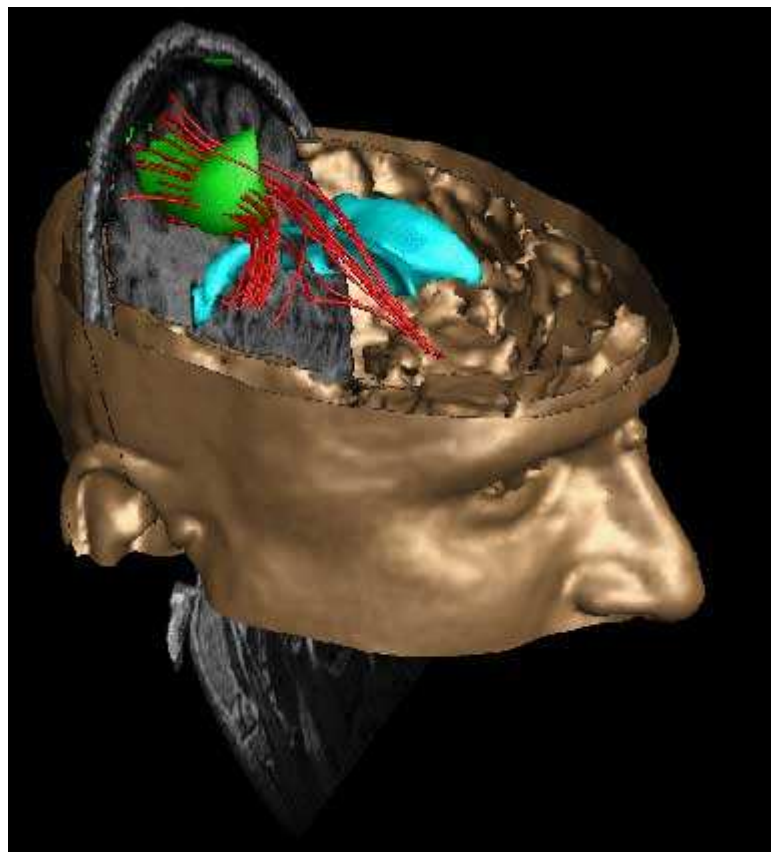


Figura 18 – Vasos sanguíneos (OPENDX, 2009)

2.7 Visualização de Dados na Área da Saúde

Com a ajuda dos atuais recursos de processamento eletrônico, a alimentação dos sistemas de informações em saúde, através da coleta de dados que está sendo realizada, está gerando um volume cada vez maior de dados (MINISTÉRIO DA SAÚDE, 2005).

Assim, temos este grande volume de dados distribuído entre diversos sistemas e a necessidade crescente por informações estruturadas, de fácil visualização e em tempo real, fundamentais para a gestão dos serviços. Estes fatores apontam, então, para a necessidade de sistemas que ajudam na Visualização dos Dados, geralmente implementados em sistemas de apoio a decisão.

A seguir SBIS (2009), complementa citando sobre a área da saúde necessitar de informações para tomada de decisão.

“A Saúde é uma das áreas onde há maior necessidade de informação para a tomada de decisões. A Informática Médica é o campo científico que lida com recursos, dispositivos e métodos para otimizar o armazenamento, recuperação e gerenciamento de informações biomédicas. O crescimento da Informática Médica como uma disciplina deve-se, em grande parte: aos avanços nas tecnologias de computação e comunicação, à crescente convicção de que o conhecimento médico e as informações sobre os pacientes são ingerenciáveis por métodos tradicionais baseados em papel, e devido à certeza de que os processos de acesso ao conhecimento e tomada de decisão desempenham papel central na Medicina moderna.”

Nos Estados Unidos segundo o HCFA (2009), no ano 2006 foi gasto um trilhão de dólares em saúde, um ano mais tarde esse valor chegou aproximadamente dois trilhões. Estimam ainda que vinte e cinco por cento desse valor é gasto devido a ineficiências, redundâncias e realização de procedimentos médicos desnecessários. A área da saúde convive com um mercado altamente distribuído, que necessita do compartilhamento de informações. Isso na maioria das vezes é feito de forma manual ou de forma ineficiente. Além disso, as informações estão dispersas em diversos sistemas diferentes e em diversos tipos de bancos de dados, que armazenam um grande volume de dados, sem estabelecimento de um padrão para representação das informações e há uma demanda crescente por informações estruturadas, de fácil visualização e em tempo real (DIAS, 2006; SANFELICE, 2008).

Os crescentes avanços tecnológicos no desempenho, capacidade de armazenamento dos computadores têm gerado grandes volumes de dados na área saúde. Torna-se um desafio a apresentação dessa informação de uma maneira intuitiva para o usuário. Se os programas não contemplarem a apresentação efetiva dos dados e interação com o usuário, possivelmente as informações contidas não poderão ser utilizadas por falta de compreensão ou por se esperar um diagnóstico mais rápido e completo. Espera-se uma mudança na forma pela qual a informação é visualizada. Neste cenário, este trabalho tem como objetivo desenvolver formas de visualização que auxiliem a construção do conhecimento por parte do usuário. Melo (2007), cita que a informática em saúde busca mudanças na forma como a informação é visualizada fazendo uso de novas tecnologias.

“A informática em Saúde vem fazendo uso de novas ferramentas e tecnologias da computação, como é o caso da Inteligência Artificial (IA), Realidade Virtual (RV), Multimídia, Hipermídia e Internet. Ainda não sendo suficiente o uso de tais ferramentas e tecnologias, atualmente, busca-se a associação das mesmas, com o objetivo de obter maior precisão das informações médicas manipuladas.”

Para aliar o uso dessas novas tecnologias e facilitar a comunicação entre sistemas e equipamentos na área da saúde estão surgindo vários padrões. Um deles referente a imagens

médicas é *Digital Imaging and Communications in Medicine* (DICOM). É um conjunto de normas especificadas pela *International Standard Organization* (ISO) para tratamento, armazenamento e comunicação entre sistemas que utilizam imagens médicas. Foi criado, com a finalidade de padronizar a formatação das imagens em sistemas que trabalham com informação biomédica, diagnóstica e terapêutica. Também com a finalidade melhorar a eficiência do fluxo de trabalho entre hospitais, computadores, equipamentos de diagnóstico e sistemas de informações na área da saúde (DICOM, 2009).

Um sistema interessante de ser mencionado no que diz respeito à sistemas que fazem o acompanhamento médico de pacientes, seus medicamentos e internações é o Lifelines. É um sistema de visualização de informações que registra os atendimentos e tratamentos médicos realizados pelos pacientes. O sistema fornece ambientes onde podem ser visualizadas diversas informações sobre paciente (Figura 19), tais como: registro de atendimento, problemas de saúde, hospitalizações e medicamentos, consultas, exames entre outras informações.

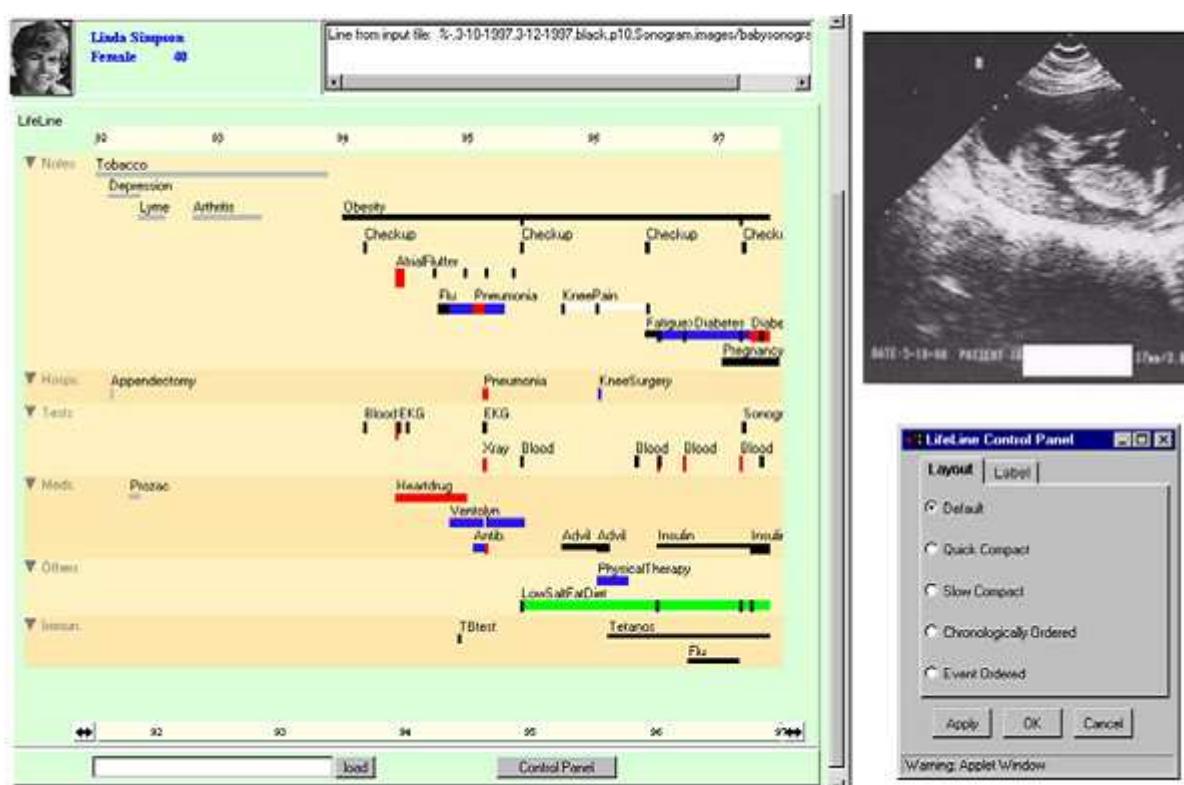


Figura 19 - Informações do paciente geradas pelo Lifelines (IEPSEN, 2008)

Funções específicas Lifelines de visualização permitem ao usuário focar em partes da imagem para descobrir mais informações. Além de uma série de outras interações que

auxiliam na redução de chances de esquecimento e erros de análise de alguma informação sobre a saúde do paciente. Com isso, facilitando a percepção de anomalias e tendências de doenças do paciente.

Massad et al (2003) deixa claro na citação seguinte a falta de padronização nas diferentes organizações de saúde no Brasil.

“As diferentes organizações de saúde e suas diversas unidades ou departamentos provavelmente nunca utilizarão o mesmo sistema, variando a sua estrutura de dados, interface e aplicações conforme a sua cultura, especificidade, conhecimento, parque tecnológico, etc. Porém, conforme já citado anteriormente, é fundamental que o conjunto essencial de dados, que também é utilizado para o gerenciamento da unidade de saúde e para a gestão do sistema, siga padrões previamente estabelecidos a fim de possibilitar a troca de mensagens entre diferentes sistemas e instituições.”

No Brasil esta padronização tem sido solicitada pelo Ministério da Saúde, através de importantes projetos como o Comitê Temático Interdisciplinar de Padronização de Registros Clínicos (CTI/PRC). A entidade responsável pelos sistemas de informações em saúde é o Departamento de Informática (DATASUS) do Sistema Único de Saúde (SUS), órgão da Secretaria Executiva do Ministério da Saúde, com a responsabilidade de coletar, processar e disseminar informações sobre saúde.

O Brasil se encontra, hoje, num momento especial para a área de informática em saúde. Finalmente se tem perspectiva de se conectar toda a rede assistencial e compartilhar a informação de saúde em prol do paciente. Os sistemas de informação do SUS, voltados para a atenção básica, média e alta complexidade, gerenciamento hospitalar e ambulatorial, cujo desenvolvimento fica a cargo do DATASUS, encontram-se, em fase de re-desenho, utilizando tecnologias baseadas em arquitetura de *software* multicamada e cem por cento voltadas à *web* (DATASUS, 2009).

2.8 Considerações Finais do Capítulo

Este capítulo teve por finalidade apresentar os principais conceitos sobre Visualização de Dados, uma breve explicação de cognição e porque ela é tão importante nessa área, quais são as vantagens de se utilizar sistemas de representação visual dos dados, formas e ferramentas de visualização e por último a importância da visualização na área da saúde e na área médica.

Através deste estudo foi possível compreender o processo de visualização, suas vantagens se comparado com formas visualização com dados brutos e a necessidade maçante dessas ferramentas serem utilizadas na saúde.

O próximo capítulo apresenta o projeto OTICSSS, sua ferramenta de consulta a base de dados da área da saúde e a ferramenta de Visualização de Dados que Sanfelice (2008) desenvolveu. Também apresenta as tecnologias utilizadas no desenvolvimento das ferramentas. Visto que o objetivo geral deste trabalho é desenvolver uma ferramenta de Visualização de Dados que possibilite ao usuário a construir conhecimento.

3 FERRAMENTA DE CONSULTA

Este capítulo apresenta o projeto OTICSSS, destacando em especial o sistema de consulta a base de dados da área da saúde. Apresenta-se também um sistema de Visualização de Dados que adicionou uma importante ferramenta de construção de conhecimento ao projeto OTICSSS, bem como sua estrutura e tecnologias utilizadas. Finaliza-se o capítulo focando nos objetivos e em ideias para construção e desenvolvimento do sistema de Visualização de Dados a ser desenvolvido.

3.1 Projeto OTICSSS

Os bancos de dados e os sistemas de informação em saúde constituem uma ferramenta importante para avaliação das políticas de saúde no Brasil. Os sistemas nacionais de informação disponíveis hoje são distribuídos e heterogêneos. No entanto falhas estão acontecendo, gerando indicadores que nem sempre reproduzem o quadro real das situações de saúde da população. Obtendo informações não concretas, com isso possibilitando existir uma grande chance dos gestores tomarem decisões errôneas. Além do mais, os dados são apresentados na maioria das vezes, em formato de tabela, não gerando dados estatísticos e informações que possam produzir informações úteis para a tomada de decisão. Estas são premissas que dão sentido e significado ao projeto OTICSSS (Observatório de Tecnologias de Informação e Comunicação em Sistemas e Serviços de Saúde).

O projeto OTICSSS está focado em analisar e sistematizar dados da área da saúde visando desenvolver uma plataforma que integre sistemas de informação com aplicações para o ensino e gestão em saúde. Ele tem como objetivos: desenvolver e implementar um Observatório de Tecnologias de Informação e Comunicação em Sistemas e Serviços de Saúde, composto por um conjunto de soluções tecnológicas e operacionais, visando qualificar o monitoramento e avaliação de indicadores de saúde para quatro regiões (sul, sudeste, nordeste e norte); criar meios que facilitem o acesso e a troca de informações entre os atores envolvidos com informação em saúde, estabelecendo a base para um processo permanente e contínuo de gestão no sistema de saúde. Este projeto está sendo desenvolvido pela

Universidade de Caxias do Sul em parceria com outras instituições de pesquisa do Brasil (Relatório projeto OTICSSS, apud SANFELICE, 2008).

Para interagir e divulgar o projeto OTICSSS, um portal na *web* permite que gestores, pesquisadores e estudantes tenham acesso às ferramentas já desenvolvidas, permitindo realizar consultas mais abrangentes e completas aos indicadores de mortalidade (categoria C) determinados pela RIPSA (Rede Interagencial de Informações para a Saúde).

Apresentando o sistema de consulta do projeto OTICSSS a Figura 20 mostra o resultado de uma consulta (indicadores de mortalidade) realizada pelo sistema de consulta as bases de dados, desenvolvido no projeto. É possível refinar a consulta utilizando diversos filtros: por regiões, estados, local de nascimento, sexo, ano e outros filtros.

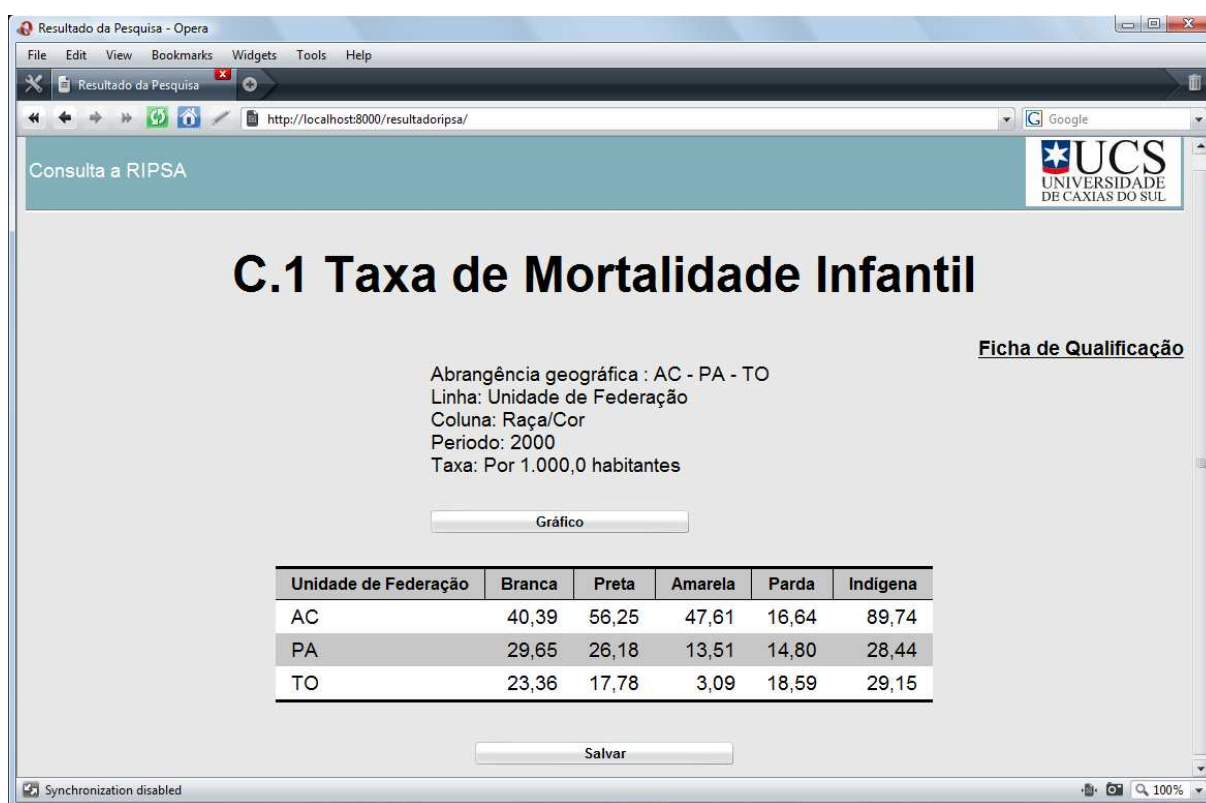


Figura 20 - Resultado da consulta (SANFELICE, 2008)

O sistema de consulta possibilita realizar consultas à base de dados da área da saúde gerando aos usuários visualizações em formato de tabelas. Apesar de ser um dos formatos mais utilizados em sistemas de informação, o formato de tabela falha ao representar um grande volume de dados, pois muitos dados são apresentados simultaneamente ao usuário. Ao

agregar novos formatos de visualização, o sistema de consulta será mais intuitivo, provendo mais subsídios para os processos de gestão dos serviços de saúde.

3.2 Ferramenta de Consulta

O sistema de consulta desenvolvido no projeto OTICSSS trabalha com dois grandes bancos de dados SIM (Sistema de Informações sobre Mortalidade) e SINASC (Sistema de Informações sobre Nascidos Vivos). O SIM com objetivo de obter dados sobre mortalidade das pessoas é utilizado por gestores de saúde, pesquisadores e entidades que utilizam as informações oferecidas pelo sistema para realização de programas de prevenção e controle de doenças. Os dados coletados pelo sistema são fornecidos através das informações contidas nas declarações de óbitos coletadas pelas Secretarias Estaduais de Saúde. O SINASC apresenta dados sobre os nascidos vivos e características consideradas como mais importantes, como sexo, local onde ocorreu o nascimento, tipo de parto, peso ao nascer entre outras. Os dados coletados pelo SINASC são fornecidos através das informações contidas nas declarações de nascimento (DATASUS, 2009).

Sanfelice (2008), em seu trabalho de conclusão de curso, desenvolveu uma ferramenta de Visualização de Dados da área da saúde que foi integrada a ferramenta de consulta, propiciando uma alternativa à visualização em formato de tabelas com intuito de auxiliar no processo de tomada de decisão. A Figura 21 mostra telas de filtros utilizados para consultar o indicador *Taxa de Mortalidade Infantil por Raça/Cor* para os estados do Acre, Pará e Tocantins no ano de 2000. Outra tela de filtros é solicitada, as informações selecionadas nessa tela serão visualizadas como linha ou colunas: *Unidade da Federação* para o campo Linha, *Raça/Cor* para o campo Coluna, *Taxa Mortalidade Infantil* para o campo Conteúdo e *2000* para o campo Ano.

Informe a Abrangência da Consulta

Unidades da Federação: BA, RJ, RS, SE, TO

Regiões: Desabilitado, Todas as Regiões, Região Norte, Região Nordeste, Região Sudeste

Base Locorregional: Todas as categorias, Amazônia Legal, Área de Fronteira

Selecione Dados para Consulta

Linha: Mês de Nascimento, Sexo, Raça/Cor, Peso, Unidade de Federação

Coluna: Tipo de Gravidez, Tipo de Parto, Mês de Nascimento, Sexo, Raça/Cor

Subcoluna: Nenhuma, Idade da Mãe, Tipo de Gravidez, Tipo de Parto, Mês de Nascimento

Conteúdo: Taxa Mortalidade Infantil, Taxa Mortalidade Neonatal Precoce, Taxa Mortalidade Neonatal Tardia, Taxa Mortalidade Pós-Neonatal, Taxa Mortalidade Perinatal

Ano: 1996, 1997, 1998, 1999, 2000

Figura 21 - Seleção dos filtros da consulta (SANFELICE, 2008)

Ao informar todos os filtros desejados e clicar em *Pesquisar* é possível visualizar os dados em formato de tabela, conforme demonstra a Figura 20. Esses dados também podem ser visualizados em formato de gráfico de barras (Figura 22).

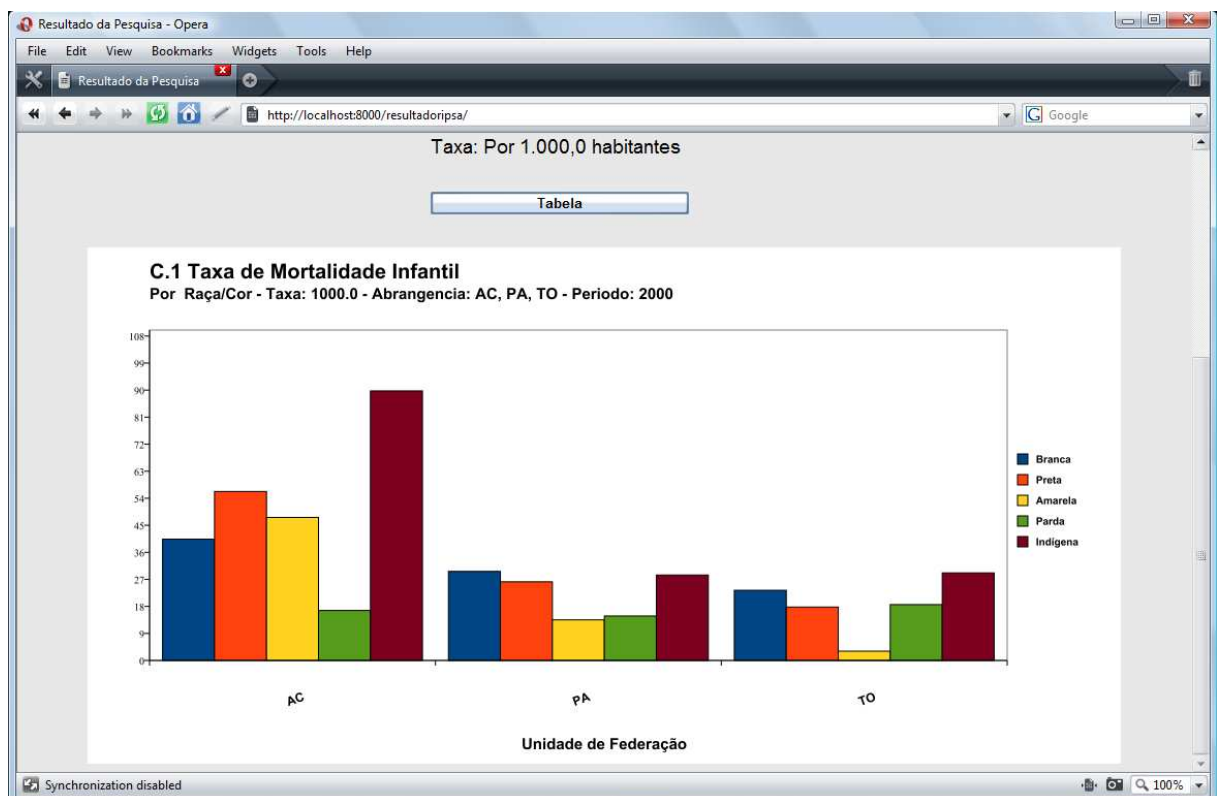


Figura 22 - Resultado da consulta em formato de gráfico (SANFELICE, 2008)

É possível constatar que através do formato gráfico Sanfelice (2008) consegue auxiliar o usuário a compreender as informações contidas na visualização gerando novos

conhecimentos. O autor deixa claro na citação seguinte a importância da representação visual gráfica.

"Através da visualização dos dados no formato de gráfico notamos rapidamente que o índice de mortalidade infantil da raça/cor Indígena no estado do Acre é o maior entre os das unidades da federação e das raças pesquisadas. Podemos concluir também que o Acre apresenta os maiores índices de mortalidade infantil das raças pesquisadas, com exceção do índice de mortalidade da raça parda que se mantém em todos estados pesquisados.

O desenvolvimento da ferramenta de visualização de dados ocorreu do ponto de vista de um sistema de apoio à decisão, ou seja, com o intuito de auxiliar o gestor na interpretação das informações disponibilizadas, permitindo extrair novos conhecimentos."

3.3 Arquitetura de *Software*

A arquitetura de *software* utilizada na ferramenta de consulta de dados do projeto OTICSSS é composta pela linguagem de programação Python, *framework* Django e o banco de dados PostgreSQL. Sanfelice (2008) deixa claro na citação seguinte o motivo da utilização dessa arquitetura.

"A escolha de tais ferramentas deu-se em função das características dos softwares, as quais atendem os requisitos necessários para a implementação da ferramenta de consulta à bases de dados da área da saúde, e em função de suas licenças de uso, que são softwares livres."

A Figura 23 ilustra a representação em camadas da arquitetura utilizada no sistema de consulta de dados. As seguintes camadas fazem parte dessa arquitetura:

- Armazenamento: o banco de dados PostgreSQL é responsável pelo armazenamento das bases de dados (SIM, SINASC e outras);
- Negócio: é responsável pelas regras do sistema. Regras essas que são implementadas utilizando a linguagem de programação Python e *framework* Django;
- Apresentação: responsável pela interface com usuário, ou seja, como os dados são apresentados para os usuários. Nesta camada é utilizada a linguagem de programação HTML (*HyperText Markup Language*), a linguagem de estilos CSS (*Cascading Style Sheets*) e algumas *templates* predefinidas.

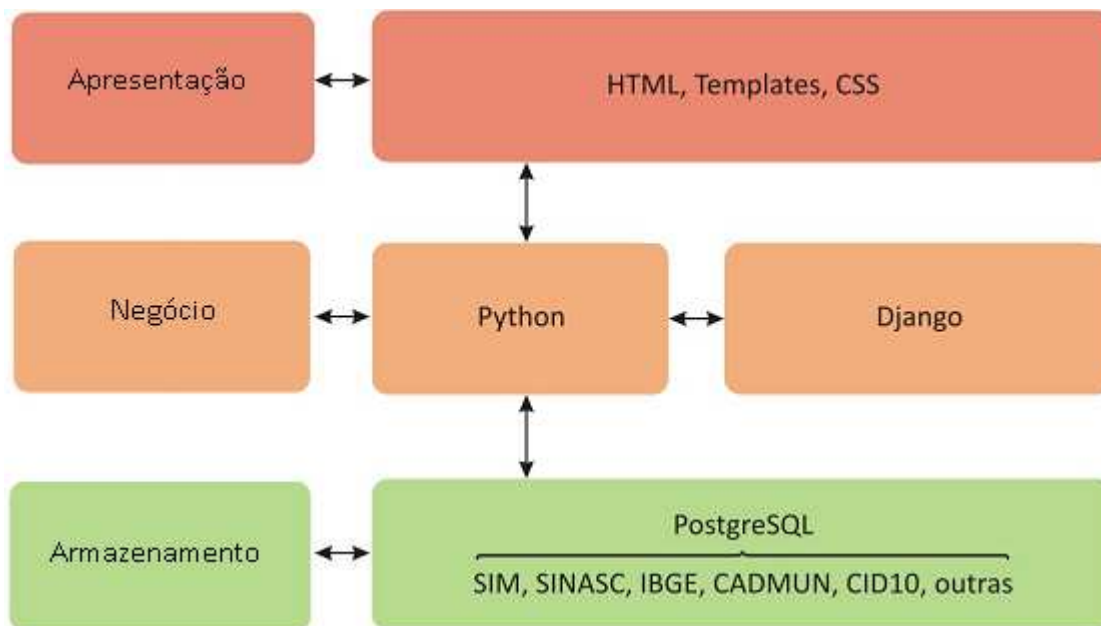


Figura 23 - Arquitetura em camadas da ferramenta de consulta (SANFELICE, 2008)

Com o intuito de oferecer uma alternativa na representação visual dos dados em formato de tabelas, que o projeto OTICSSS oferecia, até então era o único formato disponível de visualização, Sanfelice (2008) decidiu gerar gráficos como alternativa de visualização. Para tal foi escolhido *ReportLab Graphics*, um pacote gráfico que funciona integrado ao *framework* Django. Para visualização gráfica foi preciso alterar somente a camada de negócio. A Figura 24 apresenta a modificação na arquitetura das camadas da ferramenta de consulta.

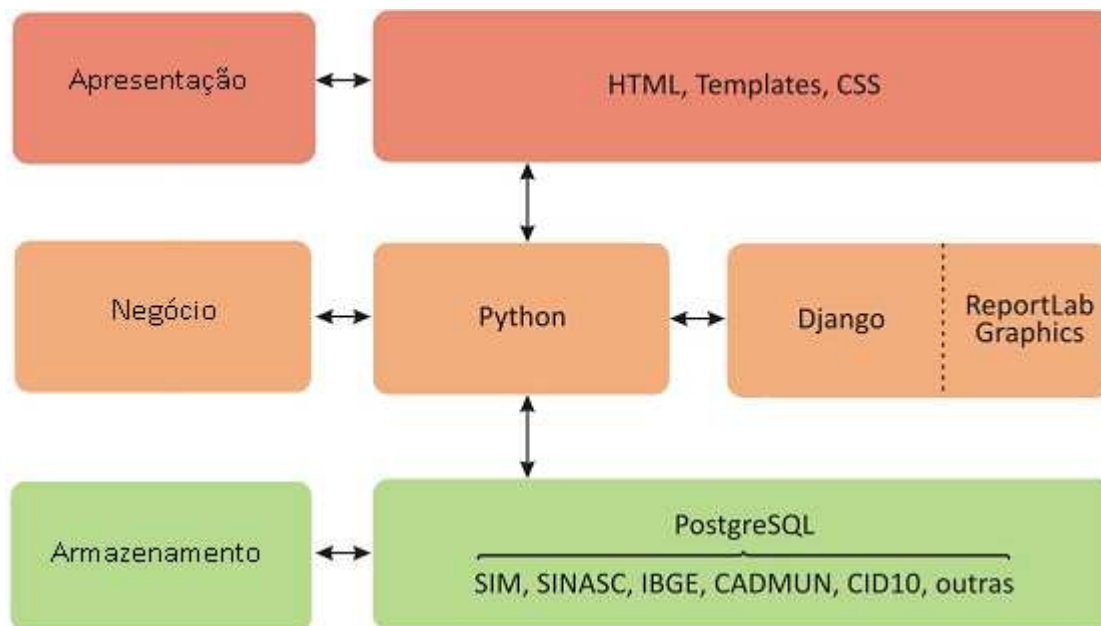


Figura 24 - Arquitetura em camadas da ferramenta de consulta com o ReportLab Graphics (SALFELICE, 2008)

A seguir serão apresentadas as tecnologias utilizadas para implementação da ferramenta de consulta como também a implementação do recurso gráfico.

3.3.1 Linguagem de Programação Python

A linguagem Python foi criada em 1991, por Guido van Rossum, com a filosofia de enfatizar a importância do esforço do programador sobre o esforço computacional, priorizando a legibilidade do código com uma sintaxe elegante, concisa e clara. É uma linguagem interpretada de alto nível, orientada a objetos, porém procedural e funcional, de tipagem dinâmica, tipagem forte e multiplataforma. Uma característica interessante que a linguagem Python possui referente à sintaxe léxica que a maioria das linguagens de programação não tem são os blocos de códigos delimitados por indentação, ou seja, inexistência de delimitadores do tipo BEGIN, END ou { e } (PYTHON, 2009).

Entre suas características, destacam-se também que os tipos de dados de alto nível permitem expressar operações complexas em um único comando e não há necessidade de declaração de variáveis ou parâmetros formais.

O Python possui uma licença livre compatível com a GPL (*General Public License*), não impondo nenhum tipo de restrição quanto à utilização e venda. Entre as diversas

características interessantes citadas anteriormente, esta agrega muita força fazendo com que as pessoas da área a estudem e a utilizem como ferramenta profissional de trabalho. (PYTHON, 2009).

3.3.2 Framework Django

O *framework* Django foi criado por Adrian Holovaty e Simon Willison em 2003, com intuito de gerenciar um site jornalístico. Tornou-se uma ferramenta de alto nível utilizada para desenvolvimento rápido e limpo de aplicativos *web*, escritos em Python. Foi projetado para lidar com dois desafios: os prazos apertados de uma redação e os requisitos rígidos dos experientes desenvolvedores. Hoje Django opera sobre licença BSD (*Berkeley Software Distribution*), ou seja, é considerado como domínio público podendo ser alterado a qualquer momento até para fins comerciais (DJANGO, 2009).

A seguir as principais características do *framework* Django (DJANGO, 2009):

- Não se repita: facilita o desenvolvimento de aplicações web com mais qualidade e menos código aderindo ao princípio de DRY (*Don't Repeat Yourself*);
- Mapeamento objeto-relacional: é possível definir os modelos de dados inteiramente em Python e uma API rica e dinâmica facilita a manipulação com o banco de dados;
- URLs elegantes: projeto *url's* (*Uniform Resource Locator*) livres de sujeira, baseadas em expressões regulares, facilitando o acesso às aplicações;
- Sistema de *templates*: permite usar uma linguagem para *templates* separando a camada de negócio da camada de apresentação;
- Sistema de *cache*: sistema distribuído de alto desempenho para o cacheamento de objetos na memória e diminuir a carga no banco de dados;
- Internacionalização: suporte para aplicações multi-idiomas.

3.3.3 Banco de Dados PostgreSQL

O PostgreSQL é um dos resultados de uma ampla evolução que se iniciou com o projeto Ingres, desenvolvido na Universidade de Berkeley, Califórnia, tendo como líder do

projeto Michael Stonebraker. Hoje o PostgreSQL é um sistema gerenciador de banco de dados (SGBD) objeto relacional de código aberto mais avançados. Apresenta todas as características dos sistemas de bancos de dados comerciais tradicionais. O PostgreSQL também é de código aberto licenciado sob os termos da licença BSD. Além das funcionalidades de um SGBD relacional, o PostgreSQL também apresenta características como: gatilhos, visões, linguagem procedural, orientação a objetos, herança e tipos personalizados entre outras (POSTGRESQL, 2009).

O projeto é coordenado pelo PostgreSQL Global Development Group, tendo em sua maioria desenvolvedores voluntários de toda parte do mundo. A parte financeira é o principal motivo do projeto ter bastante voluntários. O projeto PostgreSQL se sustenta de doações e de patrocínio. O SGBD vem ganhando cada vez mais espaço no mercado e é utilizado tanto em organizações públicas quanto organizações privadas. Um dos fatores que gerou um aumento significativo de utilização foi a sua inclusão na maioria das distribuições do Linux (POSTGRESQL, 2009).

3.3.4 Pacote Gráfico ReportLab Graphics

O ReportLab é um *software* que permite criar documentos no formato PDF, utilizando a linguagem de programação Python. Já o ReportLab Graphics é um subpacote do ReportLab que fornece recursos gráficos que também geram em formato PDF ou em formato de imagem. O ReportLab e suas bibliotecas são *softwares* livres, disponíveis sob licença BSD. O ReportLab Graphics permite a criação de gráficos estatísticos, como os de barras, de colunas, de linha e de pizza. Por utilizar a linguagem Python, os gráficos podem ser exibidos diretamente nos navegadores *web* ou serem salvos em alguns formatos de imagens ou em PDF.

A seguir as principais características do ReportLab Graphics (REPORTLAB, 2009):

- *Design* simples;
- Fácil de utilizar;
- Código Python puro;
- Geração de grandes volumes de relatórios;
- Mecanismo de impressão embutido em outros *softwares*;
- Necessidade de conversão do formato XML para PDF.

3.4 Considerações Finais do Capítulo

A saúde e a medicina moderna são áreas onde apresentam grandes necessidades de informação. E essas informações podem auxiliar gestores e outros profissionais da saúde como: na resolução de problemas, nas informações dos pacientes, acesso ao conhecimento, em pesquisa, na tomada de decisão entre outras. Em parte é onde as bases de dados do DATASUS e as ferramentas do projeto OTICSSS entram para suprir algumas dessas necessidades, possibilitando realizar consultas visualizadas em formato de tabelas. Um dos objetivos desse projeto é adicionar novas funcionalidades para ferramenta de consulta do projeto OTICSSS. Funcionalidades essas que auxiliam o usuário a analisar, interpretar e transformar conjuntos de dados em informações úteis de forma sistemática, gerando gráficos.

Sanfelice (2008) desenvolveu uma ferramenta que gera somente dois tipos de gráficos (barras e temporais), dependendo do conjunto de dados a ferramenta possibilita visualizar um único tipo de gráfico. Como visto no capítulo 2 sobre as vantagens da Visualização dos Dados é extremamente importante disponibilizar vários tipos de gráficos, pois cada um possui uma característica e informações diferentes possibilitando construir novos conhecimentos. Com esse intuito o próximo capítulo é apresentado o desenvolvimento de uma ferramenta de Visualização de Dados, que forneça para o usuário a opção de escolher qual tipo de gráfico gostaria de visualizar determinado conjunto de dados gerados pela ferramenta de consulta do projeto OTICSSS.

4 DESENVOLVIMENTO

Este capítulo apresenta o desenvolvimento de um sistema que possibilita aos usuários escolher qual tipo de gráfico deseja visualizar as informações que são retornadas pela ferramenta de consulta do projeto OTICSSS. Apresenta-se também a implementação do sistema de visualização utilizando a linguagem Python com auxílio da biblioteca gráfica ReportLab Graphics. Finaliza-se o capítulo visualizando um cenário de uso para ilustrar as funcionalidades do sistema focando nos resultados obtidos com o sistema.

4.1 Tecnologias Utilizadas

As tecnologias utilizadas no desenvolvimento desse sistema são as mesmas utilizadas no desenvolvimento do sistema de consultas a base de dados da saúde pelo projeto OTICSSS e as mesmas utilizadas no sistema de visualização desenvolvido por Sanfelice (2008). A linguagem de programação Python e *framework* de desenvolvimento em Python (Django) permaneceram na implementação desse trabalho por se tratar de uma linguagem muito boa para o desenvolvimento *web*, ainda mais, com a agilidade que o *framework* Django lhe proporciona. Até porque todo o desenvolvimento das consultas de dados da saúde estão escritos em Python.

A arquitetura de camadas utilizada também permaneceu inalterada uma vez que foi possível suprir todas as necessidades do sistema desenvolvido. As camadas são muito importantes, pois, ao visualizar o código é possível rapidamente identificar se a classe trata de uma interação com o usuário, de regras de negócio ou banco de dados. Até regras de visualização de dados na forma gráfica estão explicitamente separadas. Com isso o código fonte acaba ficando bem legível de fácil interpretação e manutenção.

Para a geração de gráfico optou-se por dar continuidade com o pacote gráfico ReportLab Graphics que auxiliou na construção de duas novas formas de visualização: gráfico de barra e gráfico de pizza. Pesquisou-se outros pacotes gráficos e levantou-se a hipótese de adicionar mais uma forma gráfica utilizando o pacote “Matplotlib”, mas como a dificuldade de se implementar gráfico em Python esbarrou diretamente com o tempo, inviabilizando a implementação de novos gráficos.

Na próxima seção é apresentado um exemplo da utilização do sistema.

4.2 Caso de Uso

Na seção 2.4, que fala sobre as vantagens da visualização de dados é demonstrada as vantagens de se representar graficamente os dados. Na demonstração são abordados três formas de visualização, são elas: formato de tabela, gráfico de pizza e gráfico de linha. A demonstração das vantagens é realizada de forma comparativa entre os gráficos. O objetivo é provar que a forma de representação gráfica torna a aprendizagem mais intuitiva com isso facilitando a descoberta de novos conhecimentos. A seguir é realizada uma simulação do sistema implementado baseando-se no seção 2.4.

Nesta simulação utilizou-se uma consulta ao indicador C.1 Taxa de Mortalidade Infantil. Os seguintes estados (Acre, Pará, Tocantins, Sergipe, Minas Gerais, Rio de Janeiro, Rio Grande do Sul) foram selecionados como área da abrangência da consulta (Figura 25).

Informe a Abrangência da Consulta

Unidades da Federação

- Desabilitado
- Todas as Unidades de Federação
- AC
- MG
- PA

Regiões

- Desabilitado
- Todas as Regiões
- Região Norte
- Região Nordeste
- Região Sudeste

Base Locorregional

- Todas as categorias
- Amazônia Legal
- Área de Fronteira

Continuar >>>

Figura 25 - Área de abrangência da consulta

Ao clicar no botão *Continuar* é chamada outra tela onde o usuário pode selecionar algumas informações como: linha, coluna, sobcoluna, conteúdo, ano e apresentar população por UF/Município. Os seguintes filtros foram selecionados nessa tela e podem ser visualizados na Figura 26:

- Linha: Unidade de Federação;
- Coluna: Mês de Nascimento;
- SubColuna: Nenhuma;
- Conteúdo: Taxa de Mortalidade Infantil;
- Ano: 2000;
- Apresentar População por UF/Município: Não foi selecionado.

Selecione Dados para Consulta

Linha

- Mês de Nascimento
- Sexo
- Raça/Cor
- Peso
- Unidade de Federação**

Coluna

- Idade da Mãe
- Tipo de Gravidez
- Tipo de Parto
- Mês de Nascimento**
- Sexo

Subcoluna

- Nenhuma**
- Idade da Mãe
- Tipo de Gravidez
- Tipo de Parto
- Mês de Nascimento

Conteúdo

- Taxa Mortalidade Infantil**
- Taxa Mortalidade Neonatal Precoce
- Taxa Mortalidade Neonatal Tardia
- Taxa Mortalidade Pós-Neonatal
- Taxa Mortalidade Perinatal

Ano

- 1996
- 1997
- 1998
- 1999
- 2000**

Apresentar População da UF/Município

Pesquisar

Figura 26 - Filtros e campos de seleção para realizar a consulta

Ao clicar no botão *Pesquisar* é chamado métodos da ferramenta de consulta que retornaram os dados em formato de tabela. A Figura 27 apresenta as informações de retorno da pesquisa.

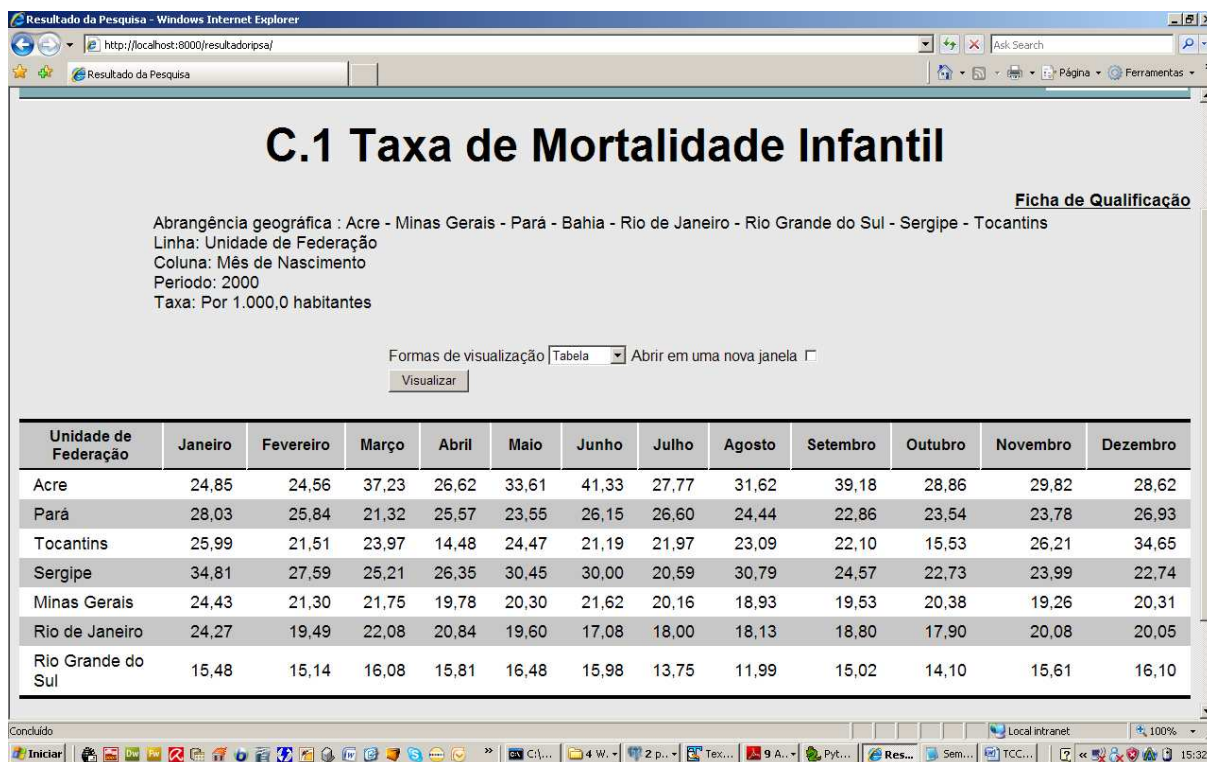


Figura 27 - Retorno da pesquisa em formato de tabela

Neste exemplo é possível visualizar as informações da tabela em uma única tela. Entretanto, em determinadas consultas, quando a tabela gerada for muito grande, o usuário poderá encontrar dificuldades para ler e assimilar tantas informações. Neste caso, ele pode contar com a opção de visualização gráfica. Para ilustrar as vantagens da representação gráfica, considere duas perguntas: a) qual é o mês que apresenta a maior taxa de mortalidade infantil somando-se todos os Estados listados? ; b) entre os Estados listados, qual foi o Estado que apresentou a maior tendência de redução da taxa de mortalidade infantil durante o ano de 2000? Note que, para responder a essas perguntas através da tabela, são necessários alguns minutos disponíveis e a capacidade de realizar vários cálculos.

Através de recursos gráficos, vamos realizar o mesmo teste. Observe na Figura 27 o campo *Formas de visualização*. Neste exemplo vamos escolher a opção gráfico de pizza, selecionando-se a seguir a opção *abrir a visualização em uma nova janela*. Ao clicar no botão *Visualizar* uma nova janela é aberta com os dados no formato de pizza (Figura 28).

C.1 Taxa de Mortalidade Infantil

Por Mês de Nascimento segundo Unidade de Federação

Acre, Minas Gerais, Pará, Bahia, Rio de Janeiro, Rio Grande do Sul, Sergipe, Tocantins - 2000

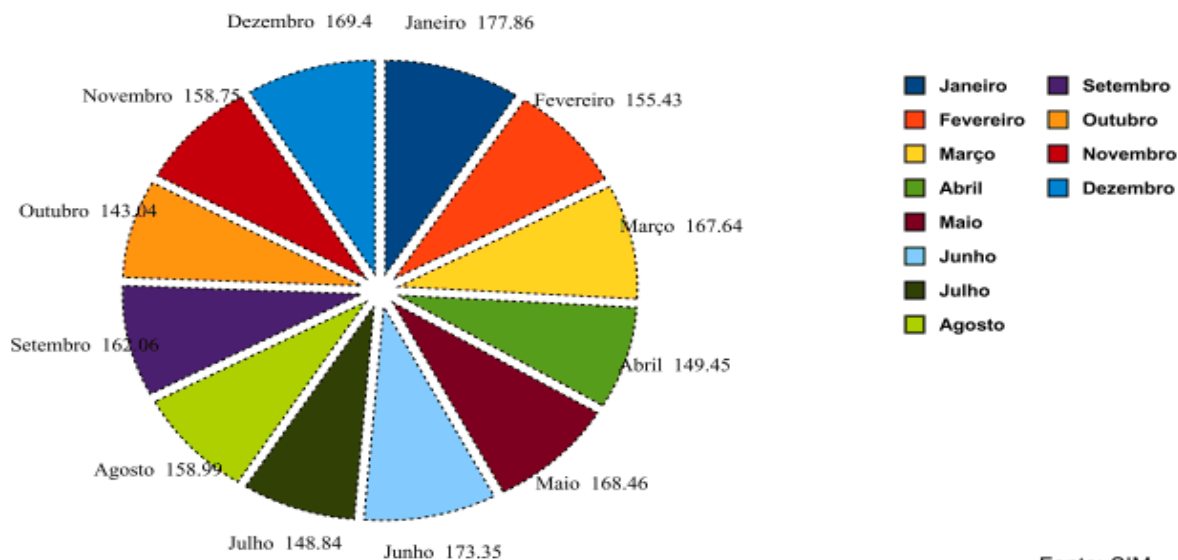


Figura 28 - Taxa de mortalidade infantil em gráfico de pizza

Com base no gráfico de Pizza pode-se responder sem dificuldade a primeira pergunta. Janeiro foi o mês que teve maior taxa de mortalidade infantil entre os Estados listados na tabela, com 177,86 mortes. Nota-se que para responder a segunda pergunta visualizando o gráfico de pizza fica inviável. Para responder essa pergunta é necessário ter informações que permite realizar uma comparação entre os meses, com isso, pode-se determinar se o Estado está aumentando ou diminuindo a taxa de mortalidade infantil. Para isso será gerado um gráfico no formato de linha.

No campo *Formas de visualização* é escolhido o tipo de gráfico linha e somente clicado no botão *Visualizar*, uma nova janela é aberta com os dados num gráfico de linha (Figura 29).

C.1 Taxa de Mortalidade Infantil
Por Mês de Nascimento segundo Unidade de Federação
Acre, Minas Gerais, Pará, Bahia, Rio de Janeiro, Rio Grande do Sul, Sergipe, Tocantins - 2000

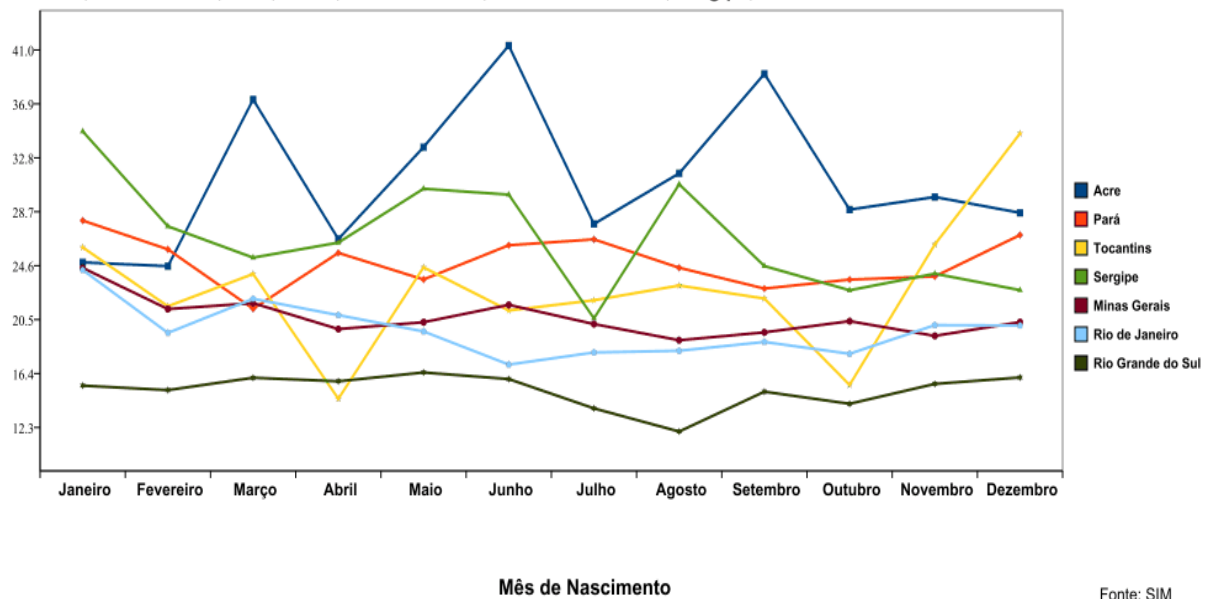
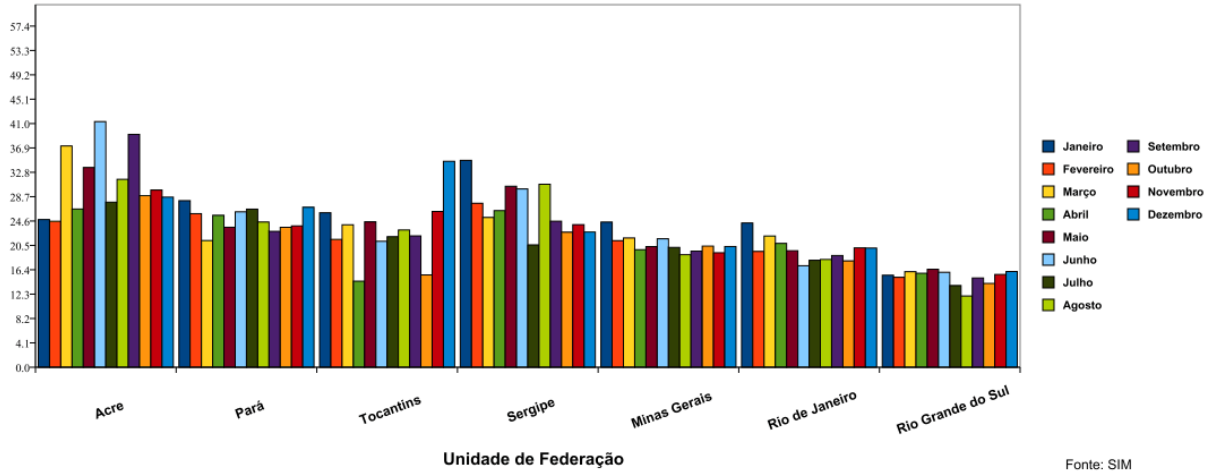


Figura 29 - Taxa de mortalidade infantil em gráfico de linha

Com o gráfico de linha é possível responder a segunda pergunta, pois, temos informações necessárias para realizar uma comparação da taxa de mortalidade infantil entre os meses do ano por Estado. Sergipe foi o Estado em que apresentou a maior redução na taxa de mortalidade infantil durante o ano, iniciou em janeiro com aproximadamente com 35 mortes e finalizou em dezembro com aproximadamente 22 mortes.

É possível selecionar outras duas formas de visualização em formato de gráfico: gráfico de coluna e gráfico de barra. As Figuras 30 e 31 apresentam esses dois gráficos gerados com os mesmos filtros utilizados para gerar os gráficos e pizza e o gráfico de linhas.

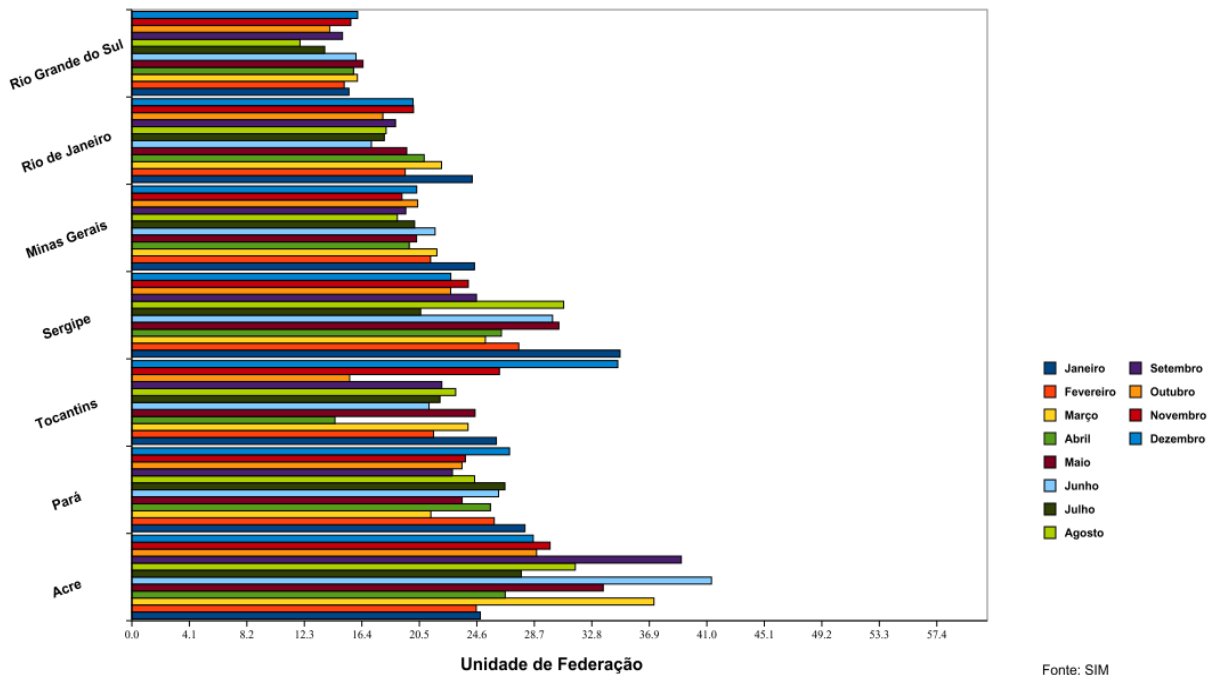
C.1 Taxa de Mortalidade Infantil
Por Mês de Nascimento segundo Unidade de Federação
Acre, Minas Gerais, Pará, Bahia, Rio de Janeiro, Rio Grande do Sul, Sergipe, Tocantins - 2000



Fonte: SIM

Figura 30 - Taxa de mortalidade infantil em gráfico de coluna

C.1 Taxa de Mortalidade Infantil
Por Mês de Nascimento segundo Unidade de Federação
Acre, Minas Gerais, Pará, Bahia, Rio de Janeiro, Rio Grande do Sul, Sergipe, Tocantins - 2000



Fonte: SIM

Figura 31 - Taxa de mortalidade infantil em gráfico de barra

Ambos os gráficos de coluna e barra estão mostrando as mesmas informações, porém o gráfico de barra está em sentido horizontal e o gráfico de coluna está no sentido vertical.

Com esses gráficos é possível visualizar intuitivamente a taxa de mortalidade infantil de todos os meses de cada Estado.

Apesar da representação visual em forma de gráficos não conseguir comportar grandes quantidades de dados, eles se destacam por serem intuitivos fazendo com que a cognição do ser humano seja trabalhada, ajudando a construir conhecimento de forma que sem elas exigiria mais tempo. Este é um exemplo de caso que contribui para reforçar a ideia de que a Visualização de Dados pode ser mais eficaz do que a representação em tabelas.

4.3 Descrição da Implementação

Nesta seção é explicado a implementação realizada neste trabalho de conclusão. É abordado desde a implementação referente à interface com usuário até a parte de comunicação com o banco de dados.

A estrutura geral das pastas onde contém os arquivos de códigos é bem simplificada, cada pasta contém fontes de sua respectiva camada (Figura 32). Utilizando a linguagem Python juntamente com o *framework* Django e adotando o modelo de camadas proposto é possível dar grande legibilidade ao código, permitindo agilizar o entendimento do código.

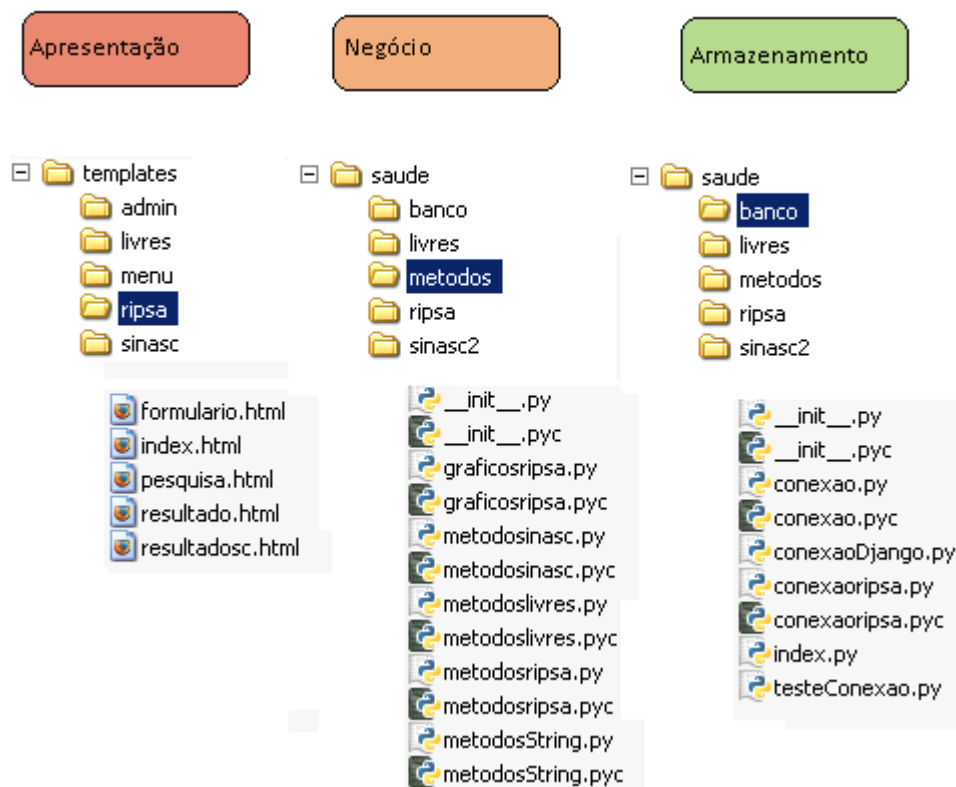


Figura 32 - Estrutura de pastas que contém os códigos fontes

A seguir é explicado as pastas principais de cada camada:

- Camada de apresentação: possui uma pasta chamada *templates*, que contém outras subpastas de cada segmento (*menu*, *ripsa* entre outras) que basicamente possuem arquivos HTML. Esses arquivos são responsáveis pela interface com o usuário (interação com o usuário, visualização dos dados, mensagens);
- Camada de Negócio: possui uma pasta chamada *saude*, que contém outras subpastas de cada segmento (*metodos*, *ripsa* entre outras) que basicamente contém arquivos Python. Esses arquivos com extensão *.py* são responsáveis pelo processamento das interações do usuário e retornarem alguma resposta. Exemplo : processar uma determinada consulta e retornando os dados em formato de gráfico;
- Camada de Armazenamento: possui uma pasta chamada *banco* que também está localizada dentro da pasta *saude*. Os arquivos Python localizados dentro dessa pasta tem a finalidade de realizar toda a comunicação com o banco de dados PostgreSQL.

O sistema desenvolvido complementou o sistema desenvolvido por Sanfelice (2008), porém gerando mais formas de visualização e não somente gráficos de barras e histogramas. O sistema possibilita ao usuário ou gestor da área da saúde visualizar informações diferentes, com isso, construindo novos conhecimentos. No sistema desenvolvido é possível gerar as consultas para os seguintes tipos de gráficos: barra, coluna, linha e pizza. Alguns resultados podem ser melhor visualizados através de certos tipos de gráficos. Isso porque o tipo de gráfico selecionado pode apresentar as informações de maneira diferente do esperado.

A Figura 33 apresenta a tela onde são visualizados os resultados da consulta. Nessa figura é possível visualizar os componentes desenvolvidos que estão destacados em vermelho. A tela é semelhante a sua versão anterior, porém ela permite que o usuário selecione a forma com que deseja visualizar através de um componente do tipo *combobox*. Também permite que seja mostrada a forma de visualização em outra janela através de um componente *checkbox*. Caso o *checkbox* estiver marcado, então abrirá uma nova janela contendo a visualização selecionada, caso contrário a visualização será mostrada na mesma tela. E o último componente é o botão *Visualizar* que tem por finalidade executar as opções selecionadas pelo usuário.

Abrangência geográfica : Acre - Pará - Rio de Janeiro - Rio Grande do Sul - Tocantins
 Linha: Unidade de Federação
 Coluna: Sexo
 Período: 2000
 Taxa: Por 1.000,0 habitantes

Formas de visualização: Tabela Abrir em uma nova janela

Visualizar

- Gráficos
- Barra
- Coluna
- Linha
- Pizza
- Tabela

Unidade de Federação	Ignorado	formado	Masculino	Feminino
Acre		125,0	16,87	11,65
Pará		73,17	14,64	11,00
Tocantins		333,33	13,28	9,00
Rio de Janeiro		17,75	11,39	8,95
Rio Grande do Sul		846,15	7,63	6,09

Figura 33 - Opção para o usuário selecionar a forma de visualização

O Código 1 mostra o código fonte escrito em HTML do campo *Formas de visualização* com suas opções de escolha. O código completo do arquivo *resultado.html* está disponível no anexo I.

Código 1 – Campo formas de visualização em código HTML

```
<tr>
  <td align="right">
    Formas de visualização
  </td>
  <td>
    <select size=1 id="formasVisualizacao" name="formasVisualizacao"
class="Attribute" >
      <optgroup label="Gráficos">
        <option value="Barra">Barra
        <option value="Coluna">Coluna
        <option value="Linha">Linha
        <option value="Pizza">Pizza
      </optgroup>
        <option value="Tabela">Tabela
    </select>
  </td>
</tr>
```

Ao clicar no botão *Visualizar* que está visível na Figura 33 é passado uma *URL* elegante */resultadoGrafico/* para a classe *url.py*, essa classe interpreta a *url* e direciona para um determinado método. Esse conceito é um dos recursos disponibilizados pelo *framework* Django para que o código fonte fique limpo. Seu objetivo é simples, realizar chamadas de métodos de forma simples utilizando expressões regulares. O Código 2 apresenta trecho da implementação da classe *url.py*. Exemplo de funcionalidade dessa classe: toda a vez que for chamada a *url indexripsa*, será desviado para método *index* da classe *views*. O código fonte completo da classe *url.py* está disponível no anexo II.

Código 2 – Trecho de código referente à URL elegante

```
# -*- coding: UTF-8 -*-
from django.conf.urls.defaults import *

urlpatterns = patterns('',
    # Conteúdo do RÍPSA
    (r'^indexripsa/$', 'saude.ripsa.views.index'),
    (r'^pesquisarripsa/$', 'saude.ripsa.views.pesquisa'),
    (r'^resultadorripsa/$', 'saude.ripsa.views.resultado'),
    (r'^csv/$', 'saude.ripsa.views.csv'),
    # Formulário
    (r'^indexformulario/$', 'saude.ripsa.views.indexformulario'),
    (r'^formulario/$', 'saude.ripsa.views.formulario'),
    (r'^resultadoGrafico/(?P<formasVisualizacao>\w+)/$',
'saude.ripsa.views.resultadoGrafico'),
    # Uncomment this for admin:
```



```
(r'^admin/', include('django.contrib.admin.urls')),  
)
```

Ao realizar a chamada da *url /resultadoGrafico/*, a classe *url.py* vai interpretar e desviar para o método *resultadoGrafico* dentro da classe *views.py*. Esse método vai processar a solicitação do usuário e conforme o tipo de gráfico solicitado irá chamar determinado método da classe *graficosripsa.py*. Exemplificando, se o usuário escolheu o formato de pizza, então é chamado o método *graficoPizza*. Pode-se visualizar um trecho do método que gera o gráfico de pizza (Código 3). O método *graficoPizza* completo é apresentado no anexo III.

Código 3 – Trecho do método que gera o gráfico em formato de pizza

```
def graficoPizza(self, conteudo, linhas, colunas, taxa, uf, anos, lresultado, cresultado, tresultado):  
  
    # caso o cresultado nao conter nada, retorna uma figura padrao  
    ## alguma outra solucao melhor?  
    if ((len(cresultado)-1)==0):  
        return 'noresult.gif'  
  
    # inicializa o grafico  
    largura = 650  
    altura = 450  
  
    self.add(Pie(), name='grafico')  
  
    # cores  
    cores = []  
    cores.append(colors.HexColor(0x004586))  
    cores.append(colors.HexColor(0xff420e))  
    cores.append(colors.HexColor(0xffd320))  
    cores.append(colors.HexColor(0x579d1c))  
    cores.append(colors.HexColor(0x7e0021))  
  
    # inicia classe com metodos auxiliares  
    metodos = metodosripsa()  
  
    # dimensoes do grafico  
    self.width = largura;  
    self.height = altura;  
  
    # atribuindo largura e altura do grafico  
    self.grafico.x = 100  
    self.grafico.y = 50  
    self.grafico.width = largura-400  
    self.grafico.height = altura-200  
    self.grafico.slices.strokeWidth = 1  
  
    # formatacao do titulo  
    titulo = metodos.ConvTabelaCampo(conteudo)  
    self.add(String(20,420,titulo), name = 'titulo')  
    self.titulo.fontName = 'Helvetica-Bold'  
    self.titulo.fontSize = 14  
    (...)
```

Cada tipo de gráfico tem atributos diferentes disponíveis no momento da implementação, mas que não serão detalhados aqui. Outras características importantes, tais como a composição dos dados para cada tipo de gráfico são apresentados a seguir:

- Gráfico de Pizza: cada pedaço da pizza é composto pelo somatório das colunas da tabela. A legenda desse tipo de gráfico é formado pelos cabeçalhos de cada coluna.
- Gráfico de Coluna: no eixo x (horizontal) são adicionados os valores da taxa de mortalidade infantil. No eixo y (vertical) são adicionadas as linhas e como um subconjunto dessas linhas são adicionadas as colunas da tabela. A legenda é composta pelo subconjunto (colunas da tabela).
- Gráfico de Barra: no eixo x são adicionadas as linhas e como um subconjunto dessas linhas são adicionadas as colunas da tabela. No eixo y são colocados valores da taxa de mortalidade infantil. A legenda é composta pelo subconjunto (colunas da tabela).
- Gráfico de Linha: no eixo x são adicionadas as colunas da tabela. No eixo y são adicionados os valores da taxa de mortalidade infantil. As linhas do gráfico são formadas pelas linhas da tabela. Nesse tipo de gráfico a legenda é composta pelas linhas da tabela.

O pacote gráfico ReportLab Graphics dá suporte para gerar gráficos de duas e três dimensões, mas a implementação dos gráficos não é trivial, até para gerar gráficos em duas dimensões a tarefa se torna trabalhosa. É necessário setar diversos parâmetros de configurações e disponibilizar várias informações para se conseguir gerar um gráfico simples. Foi comentando cada parte do código para melhorar o seu entendimento.

Na Figura 33 existe o campo *Abrir em uma nova janela*, que se estiver marcado permite visualizar o gráfico em outra janela. Foi implementada essa funcionalidade com o intuito de o usuário visualizar diversas formas de gráficos ao mesmo tempo, possibilitando cruzar as informações contidas nos gráficos, com isso construindo novos conhecimentos antes não possíveis.

Uma forma de coleta de informações dos usuários foi disponibilizar um questionário abaixo da visualização gráfica gerada. Esse questionário está disponível somente quando for marcado a opção para visualizar o gráfico em uma nova janela. Para isso foi implementado

uma tela nova *formulário.html*. Toda a vez que for gerada uma forma de visualização está sendo solicitado algumas perguntas ao usuário (Figura 34), tais como: “Essa forma de visualização está adequada para estes dados?”, “Essa visualização lhe auxiliou?” e “Espaço para observações e comentários:”.

Tipo de Gráfico: Pizza
Filtros: Abrangência geográfica : Rio Grande do Sul - Tocantins
Linha: Unidade de Federação
Coluna: Sexo
Período: 2000
Taxa: Por 1.000,0 habitantes

Essa forma de visualização está adequada para estes dados? Moderadamente

Essa visualização lhe auxiliou? Sim

Espaço para observações e comentários:

Enviar

Figura 34 - Formulário para o usuário preencher

As respostas dessas perguntas serão extremamente úteis para um trabalho futuro, aonde as respostas servirão como base para o desenvolvimento de um sistema que irá sugerir ao usuário as formas ideais de visualização para determinado conjunto de dados. Ou seja, as respostas dos especialistas serão analisadas baseando-se nos filtros escolhidos, no tipo de gráfico e nas respostas referente ao *feedback* da visualização gerada. É uma forma de os usuários ou especialistas na área da saúde contribuírem de forma construtiva, para melhorar o sistema desenvolvido. Várias informações são gravadas no banco de dados na tabela *tperguntas*, tais como: título do gráfico, tipo de gráfico, filtros e as perguntas. Com o objetivo que alguns usuários da saúde (especialistas) testarem a aplicação e responderem as perguntas. O Código 4 apresenta a estrutura da tabela *tperguntas*.

Código 4 – Estrutura da tabela *tperguntas*

```
CREATE TABLE tperguntas
(
  per_id int8 NOT NULL DEFAULT nextval('per_id'::regclass),
  per_pergunta1 varchar(15), -- Essa forma de visualização está adequada para estes dados?
  per_pergunta2 varchar(3), -- Essa visualização lhe auxiliou?
```

```
per_pergunta3 varchar(1000), -- Espaço para observações e comentários:
per_dados varchar(1000), -- Dados
per_filtros varchar(5000), -- Filtros
per_tipografico varchar(15),
per_titulo varchar(100), -- Titulo da consulta
CONSTRAINT tperguntas_pkey PRIMARY KEY (per_id)
)
WITHOUT OIDS;
ALTER TABLE tperguntas OWNER TO postgres;
COMMENT ON COLUMN tperguntas.per_pergunta1 IS 'Essa forma de visualização está adequada para estes dados?';
COMMENT ON COLUMN tperguntas.per_pergunta2 IS 'Essa visualização lhe auxiliou?';
COMMENT ON COLUMN tperguntas.per_pergunta3 IS 'Espaço para observações e comentários: ';
COMMENT ON COLUMN tperguntas.per_dados IS 'Dados';
COMMENT ON COLUMN tperguntas.per_filtros IS 'Filtros';
COMMENT ON COLUMN tperguntas.per_titulo IS 'Titulo da consulta
exemplo: C.1 Taxa de Mortalidade Infantil';
```

Ao clicar no botão *enviar* é chamando um método da classe *views.py*, capturando as informações que o usuário digitou. Após é chamando o método *gravaPerguntas* da classe da camada de armazenamento *conexaoRipsa.py*, que irá gravar as informações na tabela *tperguntas*. O Código 5 apresenta o método que grava as perguntas na tabela.

Código 5 - Método que grava as informações do questionário

```
def gravaPerguntas(self, titulo, pergunta1, pergunta2, pergunta3, tipoGrafico,
filtros):
    cur = self.getCursor()
    comando = "insert into tperguntas (per_titulo, per_pergunta1, per_pergunta2,
per_pergunta3, per_tipografico, per_filtros) values ('" + titulo + "','" +
pergunta1 + "', '" + pergunta2 + "', '" + pergunta3 + "', '" + tipoGrafico + "', '"
+ filtros + "')"
    cur.execute(comando)
    cur.execute("commit")
```

4.4 Considerações Finais do Capítulo

Esse capítulo abordou o desenvolvimento do sistema de visualização de dados, bem como os passos de utilização do mesmo e os resultados obtidos. Python é uma linguagem bastante simples e objetiva, que possibilitou alcançar os objetivos, implementando o sistema de visualização sem contratempos, apesar de não conhecer a linguagem antes desse trabalho.

Na seção cenário, mostrou-se uma simulação da ferramenta de consulta do projeto OTICSSS juntamente com as formas de visualização implementadas. Ao mesmo tempo foi realizado um comparativo entre a forma de visualização tradicional (tabela) com as formas de visualizações gráficas disponíveis no sistema. Claramente é mostrado como as representações gráficas tornam as informações intuitivas partindo de uma série de dados, possibilitando uma melhor compreensão das informações disponibilizadas e conseqüentemente auxiliando o usuário a construir conhecimento.

A intenção também era analisar as respostas que os especialistas da área da saúde responderiam, para saber se os resultados do sistema de visualização também seriam positivos para eles. Por motivos de sobrecarga de atividades do pessoal responsável pela manutenção do projeto OTICSSS o sistema de visualização não pode ser atualizado no servidor da UCS, com isso, inviabilizando a utilização do sistema pelos especialistas e ficando essa tarefa para um trabalho futuro.

O projeto OTICSSS e o sistema desenvolvido foram bem estruturados e projetados para serem extensíveis, ou seja, de fácil integração, permitindo a inclusão de outros formatos de Visualização de Dados e a inclusão de novas ferramentas em que o usuário possa interagir com a visualização. Essas integrações podem contribuir com a evolução do sistema, tornando-a cada vez mais de uso profissional, quem sabe uma necessidade para os usuários do projeto OTICSSS e uma referência para outros pesquisadores e estudantes de diferentes áreas de visualização.

5 CONCLUSÃO

5.1 Síntese

Este trabalho apresentou um estudo sobre a Visualização de Dados da área da saúde, destacando a sua importância no processo de construção do conhecimento e tomada de decisão. Destacou-se a importância da Visualização de Dados nos sistemas que tendem ajudar o usuário a compreender as informações apresentadas. A cognição como um importante passo para o reconhecimento de padrões, auxiliando o usuário compreender as informações contidas na representação visual.

Neste trabalho desenvolveu-se uma ferramenta de *software* que permite aos usuários especialistas da área da saúde visualizar as informações em vários formatos de gráficos. Antes desse trabalho era possível visualizar somente em um único formato as informações retornadas pela ferramenta de consulta do projeto OTICSSS. Essas consultas podem retornar grandes quantidades de dados e somente um único formato de gráfico não consegue absorver todas as informações. Com a possibilidade de gerar os dados em vários gráficos a chance de conseguir visualizar todas as informações contidas no retorno da consulta aumenta. Além disso, ao permitir-se várias formas de gráficos, o usuário pode explorar diferentes formas de visualização dos dados, comparar resultados, analisar pontos de vista, cruzar informações e, assim, tirar conclusões que irão apoiar processos de tomada de decisão.

Com o estudo de caso foi possível mostrar os benefícios da utilização da Visualização dos Dados. O uso das visualizações possibilitou aos avaliadores a realização de análises comparativas, intuitivas e simultâneas dos gráficos. Neste contexto, os novos conhecimentos gerados são resultado do uso da ferramenta, uma vez que sem ela dificilmente seriam construídos.

Este trabalho constituiu um desafio por integrar duas áreas de conhecimento (computação e saúde), proporcionando a aprendizagem de novas tecnologias (linguagem Python, *framework* Django, entre outras). Essa integração permitiu o desenvolvimento do sistema proposto, permitindo que os objetivos propostos fossem atingidos, construindo um sistema que permite ao profissional da saúde, interpretar os dados de forma intuitiva,

melhorando o seu entendimento, tornando-se uma ferramenta alternativa para a construção do conhecimento no projeto OTICSSS.

5.2 Contribuições do Trabalho

Uma das contribuições da realização desse trabalho de conclusão é poder disponibilizar um sistema que seja útil, que fortaleça o projeto OTICSSS, que ofereça aos usuários do projeto, alternativas diferentes de Visualização de Dados, permitindo contribuir significativamente para o aprimoramento e construção do conhecimento.

Entretanto, em um contexto mais abrangente, o projeto OTICSSS juntamente com o sistema desenvolvido pode ser um ótimo meio dos gestores da área da saúde utilizarem como base para tomada de decisões. Uma vez que a gestão das políticas públicas de saúde no Brasil dispõe de dezenas de sistemas de informações, coletando dados de forma heterogênea e sem estabelecimento de um padrão para representação das informações, com isso, dificultando a geração de indicadores reais.

Espera-se que este trabalho sirva de motivação e como uma nova fonte de pesquisa para trabalhos futuros que envolvam o conceito de Visualização de Dados.

5.3 Perspectivas de Trabalhos Futuros

Este trabalho proporcionou a implementação de um sistema que permite o usuário selecionar qual forma de Visualização de Dados deseja, mas há ainda muitos aspectos importantes que podem ser analisados gerando uma série de trabalhos futuros, apresentados a seguir:

- Pesquisar e estudar outras bibliotecas em Python que auxiliem na geração de gráficos, com o intuito de gerar outros formatos, para aperfeiçoar o sistema desenvolvido. Quanto mais formas de visualização o sistema dispôr para o usuário, maiores são as chances de informações novas serem descobertas.
- Analisar as respostas do questionário que os usuários ou especialistas da área da saúde responderam, com objetivo de desenvolver um sistema que permita

recomendar ao usuário qual forma de visualização é mais adequada para determinados dados.

- Desenvolver um sistema que permita o usuário escolher quais informações da tabela devem ser visualizados no gráfico. Acredito que essa parte é interessante para os usuários que já sabem quais informações estão procurando, com isso torna a visualização mais simplificada e objetiva.

REFERÊNCIAS

ALEXANDRE, D. S. **Percepção Humana na Visualização de Grandes Volumes de Dados: Estudo, Aplicação e Avaliação**. Dissertação de Mestrado (Mestrado em Tecnologia Multimédia). Universidade do Porto, 2006.

AURÉLIO. **Dicionário da Língua Portuguesa**. Disponível em: www.dicionariodoaurelio.com. Acesso em: 6 set. 2009.

BARBOSA, G. R. **Sistemas de Apoio a Decisão sob o Enfoque de Profissionais de Tecnologia da Informação e Decisores**. Dissertação de Mestrado (Engenharia de Produção). Universidade Federal de Pernambuco, Recife, Pernambuco, 2003.

CAMPO, M.; OROSCO, R.; TEYSEYRE, A. **Automatic Abstraction Management in Information Visualization Systems**. Proceedings IEEE Conference on Information Visualization. 1997.

CRAZYEGG. **Graziegg**. Disponível em <http://crazyegg.com>. Acesso em 18 set. 2009.

DATASUS. **Departamento de Informática do SUS**. Disponível em: <http://www.datasus.gov.br>. Acesso em 3 jun. 2009.

DIAS, R. D. M. **O Governo e a Padronização de Informação na Saúde Suplementar: uma Análise Comparativa**. Rio de Janeiro, 2006.

DICOM. **Digital Imaging and Communications in Medicine**. Disponível em <http://medical.nema.org/>. Acesso em 20 de set. 2009.

DJANGO. **Django Brasil**. Disponível em <http://www.djangobrasil.org>. Acesso em 07 set. 2009.

FAYYAD, U. M.; GRINSTEIN, G. G.; WIERSE, A. **Information Visualization in Data Mining and Knowledge Discovery**. San Francisco: Morgan Kaufmann, 2002.

FRIENDLY, M. **Milestones in the history of data visualization: A case study in statistical historiography**. In W. Gaul and C. Weihs, eds., *Studies in Classification, Data Analysis, and Knowledge Organization*. New York: Springer, 2004.

FRIENDLY, M. **Milestones in the history of thematic cartography, statistical graphics and data visualization**. Disponível em: <http://www.math.yorku.ca/SCS/Gallery/milestone>. Acesso em 7 set. 2009.

FRY, B. **Visualizing Data**. 2. ed. Sebastopol: O'Reilly, 2008.

GGOBI. **Ggobi Data Visualization System**. Disponível em: <http://www.ggobi.org>. Acesso em 7 set. 2009.

HABER, R. B.; MCNABB, D. A. **Visualization Idioms: A conceptual model for scientific visualization systems**. *Visualization in Scientific Computing*, 1990.

HCFA. **Health Care Financing Administration**. Disponível em <http://www.hcfa.org>. Acesso em 19 de set. 2009.

HOLDEN, G. **Implementing Decision-support Portals based on Data Visualization Best Practices**. [S.l.], Jun. 2007.

IDB. **Indicadores e Dados Básicos**. Disponível em <http://www.datasus.gov.br/idb>. Acesso em 3 jun. 2009.

IEPSEN, E. F.. **Estudo e desenvolvimento de uma Ferramenta de Visualização de Informações Temporais para Banco de Dados estruturados no formato Mestre/Detalhe**. Dissertação de Mestrado. Universidade Católica de Pelotas, Pelotas, Rio Grande do Sul, 2008.

JUDELMAN, G. B. **Knowledge Visualization Problems and Principles for Mapping the Knowledge Space**. Jun. 2004.

MALHOTRA, Y. **Expert Systems for Knowledge Management Crossing the Chasm between Information Processing and Sense Making**. Expert Systems with Application 20, 2001. Disponível em <http://www.brint.org/expertsystems.pdf>. Acesso em 3 jun. 2009.

MASSAD, E.; MARIN, H. F.; NETO, R. S. A. **O Prontuário Eletrônico do Paciente na Assistência, Informação e Conhecimento Médico**. 2003.

MATTOS, N. P. **Sistema de Apoio à Decisão para Planejamento em Saúde**. Dissertação de Mestrado (Informática Aplicada). Pontifícia Universidade Católica do Paraná, Curitiba, Paraná, 2003.

MELO, Jairo Simão Santana. **Integrando Realidade Virtual em Sistemas Tutores Inteligentes no Domínio da Saúde**. Dissertação de Pós-Graduação, Brasília, 2007.

MICHAELIS. **Moderno Dicionário da Língua Portuguesa**. Disponível em: <http://michaelis.uol.com.br>. Acesso em 17 set. 2009.

MINISTÉRIO DA SAÚDE. Secretaria de Vigilância em Saúde. **Guia de Vigilância Epidemiológica**. 6. ed. Brasília: Ministério da Saúde, 2005. Disponível em http://portal.saude.gov.br/portal/arquivos/pdf/Guia_vig_Epid_novo2.pdf. Acesso em: 7 set. 2009.

NASCIMENTO, H. A. D.; FERREIRA, C. B. R. **Visualização de Informações – Uma Abordagem Prática**. XXV Congresso da Sociedade Brasileira de Computação, XXIV JAI. UNISINOS, São Leopoldo, Rio Grande do Sul, 2005.

NEWSMAP. **Newsmap**. Disponível em <http://newsmap.jp>. Acesso em 19 set. 2009.

OPENDX. **The Open Source Software Project Based on IBM's Visualization Data Explorer**. Disponível em <http://www.opendx.org/>. Acesso em 7 set. 2009.

POSTGRESQL. **PostgreSQL Brasil**. Disponível em: <http://www.postgresql.org.br>. Acesso em 4 out. 2009.

PYTHON. **Python Programming Language**. Disponível em <http://www.python.org/>. Acesso em 4 out. 2009.

REPORTLAB. **ReportLab**. Disponível em: <http://www.reportlab.org>. Acesso em 15 set. 2009.

RIPSA. **Rede Interagencial de Informações para a Saúde**. Disponível em: <http://www.ripsa.org.br>. Acesso em 3 jun. 2009.

SANFELICE, S. **Visualização de Dados da Área da Saúde para Apoio à Decisão**. Caxias do Sul, Brasil. Dezembro 2008.

SBIS. **Sociedade Brasileira de Informática em Saúde**. Disponível em <http://www.sbis.org.br/>. Acesso em 20 set. 2009.

STOREY, Margaret-Anne. **Information Visualization and Knowledge Managment**. Disponível em http://webhome.cs.uvic.ca/~mstorey/teaching/infovis/course_notes/introduction.pdf. Acesso em 5 jun. 2009.

TIME MAGAZINE. **Time: America By The Numbers**. Disponível em: http://www.time.com/time/covers/20061030/where_we_live. Acesso em 17 set. 2009.

TURBAN, E.; ARONSON, J. E. **Decision Support Systems and Intelligent Systems**. New Jersey: Prentice Hall, 2000 (Sixth Edition).

VISUAL THESAURUS. **Visual Thesaurus**. Disponível em <http://www.visualthesaurus.com/>. Acesso em 19 set. 2009.

WARE, C. **Information Visualization: Perception for Design**. San Francisco, CA: Morgan Kaufmann, 2004 (Second Edition).

ANEXOS

Anexo I – Código fonte do arquivo *resultado.html*.

Anexo II – Código fonte da classe *url.py* que implementa o conceito de *URL* elegante.

Anexo III – Código fonte completo do método *graficoPizza*.

Anexo I

Código fonte do arquivo *resultado.html*.

```
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
    <link rel="stylesheet" href="http://127.0.0.1:8000/media/css/saude.css">
    <title> Resultado da Pesquisa </title>
    <script language="javascript">

      function init(formas){
        formas.value = "Tabela"
      }

      function toggle(chk, formas) {

        var dColuna = document.getElementById("divColuna");
        var dBarra = document.getElementById("divBarra");
        var dLinha = document.getElementById("divLinha");
        var dPizza = document.getElementById("divPizza");
        var dTabela = document.getElementById("divTabela");
        var b = document.getElementById("graftab");
        var te = document.getElementById("formasVisualizacao");
        te = formas.value;

        if(chk.checked == 1)
        {

          window.open("/resultadoGrafico/"+te+"/",null,
"height=500,width=700,status=yes,toolbar=no,menubar=no,location=yes, scrollbars=yes, resizable=yes");

        }
        else
        {
          if(formas.value == "Coluna")
          {
            dColuna.style.display="block";
            dBarra.style.display="none";
            dLinha.style.display="none";
            dPizza.style.display="none";
            dTabela.style.display="none";
          }
          else
          {
            if(formas.value == "Barra")
            {
              dColuna.style.display="none";
              dBarra.style.display="block";
              dLinha.style.display="none";
              dPizza.style.display="none";
              dTabela.style.display="none";
            }
            else
            {
              if(formas.value == "Linha")
              {
                dColuna.style.display="none";
                dBarra.style.display="none";
                dLinha.style.display="block";
                dPizza.style.display="none";
                dTabela.style.display="none";
              }
              else
              {
                if(formas.value == "Pizza")
                {
                  dColuna.style.display="none";
                  dBarra.style.display="none";
                  dLinha.style.display="none";
                  dPizza.style.display="block";
                }
              }
            }
          }
        }
      }
    </script>
  </head>
</html>
```



```

</table>
<table align="center">
  <tr>
    <td align="right">
      Formas de visualização
    </td>
    <td>
      <select size=1 id="formasVisualizacao" name="formasVisualizacao"
class="Attribute" >
        <optgroup label="Gráficos">
          <option value="Barra">Barra
          <option value="Coluna">Coluna
          <option value="Linha">Linha
          <option value="Pizza">Pizza
        </optgroup>
        <option value="Tabela">Tabela
      </select>
    </td>
    <td>
      Abrir em uma nova janela
    </td>
    <td>
      <input type="checkbox" id="outraJanela" name="outraJanela" value="outraJanela">
    </td>
  </tr>
  <tr>
    <td colspan="4">
      <input type="button" id="grftab" onclick='toggle(outraJanela,
formasVisualizacao); value="Visualizar" >
    </td>
  </tr>
  <tr>
    <td colspan="4">
      &nbsp;
    </td>
  </tr>
</table>
<div id="divTabela">
  {% if not conteudotabela %}
    <h1> Não existem dados disponíveis para estas restrições </h1>
  {% else %}
    <table class="resultado" align="center" cellpadding="0" cellspacing="0">
      <tr>
        {% for c in cabecalho %}
          {% if forloop.first %}
            <th class="primeiro">{{ c }}</th>
          {% else %}
            <th class="padrao" > {{ c }} </th>
          {% endif %}
        {% endfor %}
      </tr>
      {% for l in conteudotabela %}
        <tr class= "{% cycle linha1,linha2 %}">
          {% for c in l%}
            {% if forloop.first %}
              <td class="conteudoesquerda">
            {% else %}
              <td class="conteudodireita">
            {% endif %}
              {{ c }}
            </td>
          {% endfor %}
        </tr>
      {% endfor %}
    </table>
    <br>
    <br>
    <form name="form1" action="/csv/" method="post" onSubmit="return ibgecheckbox()">
      <input type="submit" name="resultado" value="Salvar" >
    </form>

```



```
        {% endif % }
    </div>
    <div          id="divColuna"
src="http://127.0.0.1:8000/media/img/temp/{{ graficoColuna }}"></div>
    <br>
    <div          id="divBarra"
src="http://127.0.0.1:8000/media/img/temp/{{ graficoBarra }}"></div>
    <br>
    <div          id="divLinha"
src="http://127.0.0.1:8000/media/img/temp/{{ graficoLinha }}"></div>
    <br>
    <div          id="divPizza"
src="http://127.0.0.1:8000/media/img/temp/{{ graficoPizza }}"></div>
    </body>
</html>
```

Anexo II

Código fonte da classe *url.py* que implementa o conceito de *URL* elegante.

```
# -*- coding: UTF-8 -*-

from django.conf.urls.defaults import *

urlpatterns = patterns("",
    # Conteúdo do RIPSa
    (r'^indexripsa/$', 'saude.ripsa.views.index'),
    (r'^pesquisarripsa/$', 'saude.ripsa.views.pesquisa'),
    (r'^resultadorripsa/$', 'saude.ripsa.views.resultado'),
    (r'^csv/$', 'saude.ripsa.views.csv'),
    # Conteúdo do SINASC/SIM
    (r'^sinasc/index/$', 'saude.sinasc2.views.index'),
    (r'^sinasc/pesquisa/$', 'saude.sinasc2.views.pesquisa'),
    (r'^sinasc/resultado/$', 'saude.sinasc2.views.resultado'),
    (r'^sinasc/csv/$', 'saude.sinasc2.views.csv'),
    # Conteúdo da consulta livre 2.0
    (r'^livres/index/$', 'saude.livres.views.index'),
    (r'^livres/tabulacao/$', 'saude.livres.views.tabulacao'),
    (r'^livres/totalizadores/$', 'saude.livres.views.totalizadores'),
    (r'^livres/filtros/$', 'saude.livres.views.filtros'),
    (r'^livres/relacoes/$', 'saude.livres.views.relacoes'),
    (r'^livres/visualizacao/$', 'saude.livres.views.visualizacao'),
    (r'^livres/resultado/$', 'saude.livres.views.resultado'),
    # Formulário
    (r'^indexformulario/$', 'saude.ripsa.views.indexformulario'),
    (r'^formulario/$', 'saude.ripsa.views.formulario'),
    (r'^resultadoGrafico/(?P<formasVisualizacao>\w+)/$', 'saude.ripsa.views.resultadoGrafico'),
    # Uncomment this for admin:
    (r'^admin/', include('django.contrib.admin.urls')),
)
```

Anexo III

Código fonte completo do método *graficoPizza*.

```
def graficoPizza(self, conteudo, linhas, colunas, taxa, uf, anos, lresultado, cresultado, tresultado):

    # caso o cresultado nao conter nada, retorna uma figura padrao
    ## alguma outra solucao melhor?
    if ((len(cresultado)-1)==0):
        return 'noresult.gif'

    # inicializa o grafico
    largura = 650
    altura = 450

    self.add(Pie(), name='grafico')

    # 12 barras formatadas com cores personalizadas
    # porem, nao ha limites para o numero de barras
    cores = []
    cores.append(colors.HexColor(0x004586))
    cores.append(colors.HexColor(0xff420e))
    cores.append(colors.HexColor(0xffd320))
    cores.append(colors.HexColor(0x579d1c))
    cores.append(colors.HexColor(0x7e0021))
    cores.append(colors.HexColor(0x83caff))
    cores.append(colors.HexColor(0x314004))
    cores.append(colors.HexColor(0xaecf00))
    cores.append(colors.HexColor(0x4b1f6f))
    cores.append(colors.HexColor(0xff950e))
    cores.append(colors.HexColor(0xc5000b))
    cores.append(colors.HexColor(0x0084d1))

    # inicia classe com metodos auxiliares
    metodos = metodosripsa()

    # dimensoes do grafico
    self.width = largura;
    self.height = altura;

    # atribuindo largura e altura do grafico
    self.grafico.x = 100
    self.grafico.y = 50
    self.grafico.width = largura-400
    self.grafico.height = altura-200
    self.grafico.slices.strokeWidth = 1

    # formatacao do titulo
    titulo = metodos.ConvTabelaCampo(conteudo)
    self.add(String(20,420,titulo), name = 'titulo')
    self.titulo.fontName = 'Helvetica-Bold'
    self.titulo.fontSize = 14

    # subtitulo1 = Por 'coluna' segundo 'linha'
    subtitulo1='Por '+metodos.ConvTabelaCampo(colunas)+' segundo '+metodos.ConvTabelaCampo(linhas)
    self.add(String(20,400,subtitulo1), name='subtitulo1')
    self.subtitulo1.fontName='Helvetica-Bold'
    self.subtitulo1.fontSize=14

    # subtitulo2: area e anos de abrangencia
    subtitulo2=uf[0]
    ## se houver mais de um estado selecionado
    if(len(uf)>1):
        for estado in uf[1:len(uf)]:
            subtitulo2=subtitulo2+', '+estado

    subtitulo2=subtitulo2+' - '+str(anos[0])
    ## se houver mais de um ano selecionado
    ## TO-DO: formatar os anos da seguinte forma:
    ##     2000-2005: para o periodo de 2000 a 2005
    ##     2000/2005: para os anos de 2000 e 2005
```

```

if (len(anos)>1):
    for ano in anos[1:len(anos)]:
        subtítulo2=subtítulo2+', '+str(ano)
self.add(String(20,380,subtítulo2), name='subtítulo2')
self.subtítulo2.fontName='Helvetica-Bold'
self.subtítulo2.fontSize=14

# fonte dos dados
## e se trocar a fonte? tem que ser dinamica...
fonte='Fonte: SIM'
self.add(String(570,10,fonte), name='fonte')
self.fonte.fontName='Helvetica'
self.fonte.fontSize=12

# preparando dados do grafico...
tot=[]
conteudoGrafico=[]
for i in range(len(cresultado)-1):
    out=[]
    tota = 0
    for linha in tresultado:
        if linha[i+1]=='.':
            out.append(0)
        else:
            # se tiver formato #.###,##, remove 'ponto'
            if ( linha[i+1].find('.') > -1 ):
                linha[i+1]=linha[i+1].replace('.',',')
            # substitui virgula
            n=linha[i+1].replace(',','.',-3)
            out.append(float(n))
            tota += float(n)

    # insere o conteudo 'out' na posicao 'i'
    conteudoGrafico.insert(i,out)
    tot.append(tota)

    #print "total: " + str(tot[cont])
self.grafico.data = tot

print tot
print conteudoGrafico

# setando propriedades para cada parte da pizza
cont = 0
#while cont < len(self.grafico.slices):
while cont < 12:
    self.grafico.slices[cont].popout = 10
    self.grafico.slices[cont].strokeWidth = 1
    self.grafico.slices[cont].strokeDashArray = [2,2]
    #self.grafico.slices[cont].labelRadius = 0.75
    self.grafico.slices[cont].fontSize = 11
    #self.grafico.slices[cont].fontColor = colors.yellow
    self.grafico.slices[cont].fontColor = colors.black
    self.grafico.slices[cont].fillColor = cores[cont]
    cont += 1

# legendas
legenda = Legend()
legenda.alignment = 'right'
legenda.x = 500
legenda.y = 300
legenda.dxTextSpace = 5
legenda.strokeColor = colors.black
legenda.columnMaximum = 8
legenda.fontName = 'Helvetica-Bold'
legenda.fontSize = 10
legenda.colorNamePairs = []
c=0
legendas=cresultado[1:len(cresultado)]

```

```

teste=False
for l in legendas:
    if(len(l)>15):
        teste=True
        legenda.colorNamePairs.append((self.grafico.slices[c].fillColor,l))
        c=c+1

# caso a legenda for grande, aumenta a area do grafico
## TO-DO: encontrar uma forma mais 'decente' de fazer isso
if(teste):
    self.width=1100
    self.add(legenda,'legenda')

# descrições de cada parte da pizza
legendas2 = legendas
cont = 0
for l in legendas2:
    legendas2[cont] = legendas2[cont] + ' ' + str(tot[cont])
    cont += 1
self.grafico.labels = legendas2

# cores
#self.grafico.strokeColor=colors.gray

# salvando grafico
## TO-DO: interessante encontrar uma forma de retornar o grafico (gif/binario)
##     assim, nao haveria a necessidade de armazenar em disco a imagem temporario
##     porem, nao consegui exibir o grafico utilizando o render_to_response (so atraves do HttpResponse)
arquivo=self.save(formats=['gif'],outDir=settings.MEDIA_ROOT+'/img/temp',fnRoot='tmp'+str(random.getrandbits(40)))
# recuperando nome do arquivo temporario
t=arquivo.split('\')
t=t[(len(t)-1)]
return t

```