UMA FERRAMENTA WEB PARA AUXILIAR O DESENVOLVIMENTO DE RELATÓRIOS EM PDF PARA SISTEMAS EM PHP

João Lucas P. Soares[1], Marcelo V. C. Rosa[1], Francisco A. Silva[1]

[1]Faculdade de Informática (FIPP) – Universidade do Oeste Paulista (UNOESTE) – Presidente Prudente-SP. E-mail: chico@unoeste.br

RESUMO

Este artigo apresenta uma ferramenta para auxiliar na customização e geração de relatórios em PDF destinados a sistemas de internet baseados na tecnologia Open source PHP. Foi desenvolvido um editor visual para confecção interativa dos relatórios de acordo com as necessidades do desenvolvedor. São gerados arquivos no formato da linguagem PHP para serem adicionados aos sistemas, minimizando o trabalho do desenvolvedor. O editor visual, usado em navegadores de internet, foi desenvolvido utilizando a junção de diversas tecnologias Open source.

Palavras-chave: Relatórios em PDF - PHP -Editor visual

A WEB TOOL TO AID THE DEVELOPMENT OF PDF REPORTS FOR SYSTEMS IN PHP

ABSTRACT

This article presents a tool to help in customization and generation of reports in PDF to internet systems that are based in Open-Source technologies PHP. It was developed a visual editor for interactive confection of reports, according to the needs of the developer. Files are generated in the PHP file to be added to systems, minimizing the work of the developer. The visual editor, used in browsers, was developed using several Open source technologies.

Keywords: PDF records - PHP - Visual editor

INTRODUÇÃO

Nos últimos anos o desenvolvimento de sistemas para internet tem crescido muito, isso devido ao barateamento dos custos para acesso à internet banda larga, a facilidade de integração destes sistemas com aparelhos móveis (celulares e *smartphones*). De acordo com a pesquisa realizada em 2009 pela agencia NIC.br (Núcleo de Informação e Coordenação do Ponto BR) e pela Cetic.br (Centro de Estudos as Tecnologias da Informação e da Comunicação) entre os anos de 2008 e 2009 houve uma taxa de crescimento de 35% no acesso à internet entre os usuários do Brasil [Cetic 2009].

Assim como soluções desenvolvidas para ambientes *desktops*, soluções para a internet necessitam de diversos relatórios, que vão desde simples listagens de clientes até os mais complexos como financeiros, contábeis, gestão integrada, dentre outros.

Desde o surgimento da internet em meados de 1990, evoluiu-se muito em relação às linguagens de programação para esse ambiente, melhorando principalmente em aspectos como a segurança e a GUI (*Graphics User Interface*).

A partir de 1994 o desenvolvedor canadense-dinamarquês Rasmus Lerdorf, começou o desenvolvimento da linguagem de programação PHP (*Hypertext Preprocessor*), para auxiliar no desenvolvimento de sua página pessoal. A liberdade de código fonte aberto e *software* gratuito do PHP garantida por um grupo de esquemas de licença, denominadas GPL (*General Public License*), colocou o PHP como uma das linguagens mais usadas para desenvolvimento web, tendo grande reconhecimento mundial [Converse and Park 2003].

Anualmente a empresa *Tiobe Software* apresenta uma lista com as 20 linguagens mais usadas para desenvolvimento de soluções de *software*. Conforme apresentado na tabela 1, a linguagem PHP ocupa a 3ª posição, está à frente de linguagens com fortes investimentos de grandes companhias de *softwares* como C# (*C Sharp*) que é desenvolvida pela Microsoft e Delphi (*Object Pascal*) que é desenvolvida pela *Embarcadero* entre outras [Tiobe 2010].

Atualmente, o PHP dispõe de IDE´s (*Integrated Development Environment*) próprias, como por exemplo, o phpDesign da MpSoftware, o Zend Studio da Zend, e também suporte em IDE´s como Netbeans desenvolvido pela Oracle, dentre outras. Portanto, nenhuma dessas IDE´s possui um editor gráfico ou um *Framework* de *software*, que auxilie o desenvolvedor na criação e/ou geração de relatórios para o PHP.

Tabela 1. Lista de algumas das linguagens mais usadas em Março de 2010 [Tiobe 2010]

Linguagem de programação	Posição Março 2010	Posição Março 2009	Mercado (%) março 2010	Delta em relação a 2009
Java	1	1	17.50%	-2.29%
C	2	2	17.27%	+1.42%
PHP	3	3	9.90%	+0.42%
C++	4	4	9.51%	-0.75%
(Visual) Basic	5	5	6.57%	-1.71%
C#	6	7	4.26%	-0.06%
Python	7	6	4.23%	-0.95%

Assim, a motivação de pesquisa desse projeto concentra-se no desenvolvimento de uma ferramenta para auxiliar a geração de relatórios dinâmicos para sistemas desenvolvidos em PHP internet, utilizando-se de uma interface gráfica de fácil uso com a integração de diversos sistemas de gerenciamento de banco de dados. Esse editor gráfico possibilita o desenvolvedor escolher o cabeçalho e o rodapé dos relatórios assim como ter toda a relação de campos de dados dos bancos. Nele, o desenvolvedor também tem a possibilidade de estruturar o SQL (*Structured*

Query Language) para retornar os dados formatados conforme necessário.

Objetiva-se com o uso dessa ferramenta aproximar a facilidade no desenvolvimento de relatórios em PDF (*Portable Document Format*) com o PHP de outras linguagens de programação que já possuem em muitos casos um ambiente com recursos baseados em interfaces para a criação de relatórios em PDF (*JReport* para linguagem *Java*, *Crystal Reports* para C# no *Visual Studio*, etc). Entretanto, na linguagem de programação PHP uma interface gráfica na geração de relatórios PDF seria muito desejável e de grande valia.

Os relatórios são baseados na utilização da biblioteca FPDF (*Free Portable Document Format*). Uma classe gratuita que não requer extensões e funciona tanto com as versões 4 e 5 do PHP, que permite gerar arquivos em PDF sem a necessidade da utilização da biblioteca PDFlib que para uso comercial requer uma licença [Oliver 2006], [Paraíso 2010].

O processo convencional para confeccionar relatórios em PDF é de forma complexa devido à biblioteca FPDF não disponibilizar recursos visuais para elaboração de arquivos PDF, onde o desenvolvedor deve estruturar todos os elementos do relatório que deseja que apareça linha a linha. A atividade de construir relatórios de forma manual em PHP (processo convencional) onera em muito o tempo de desenvolvimento dos sistemas.

Além do fator despendido para confecção dos relatórios, um dos principais problemas é de que os navegadores utilizam de dois principais posicionamentos de objetos na tela, a posição relativa ou absoluta. Para impressão utiliza-se da posição absoluta, mas esta faz referência conforme a resolução do monitor onde está sendo desenvolvido.

Dessa forma, foi prevista na ferramenta a construção de uma interface gráfica de auxílio no desenvolvimento de relatórios em PDF, que não muda de acordo com a resolução do monitor e utiliza recursos de "arrastar e soltar" de componentes existentes, como cabeçalho, rodapé, tabela entre outras. A interface da ferramenta visa minimizar o esforço do desenvolvedor, pois disponibiliza o suporte necessário aos dados diretamente dos bancos de dados [Jaques 2010].

Na seção seguinte serão tratadas e demonstradas as tecnologias empregadas para desenvolver a interface da ferramenta, assim como, o uso das tecnologias responsáveis pelo tratamento da conversão e geração dos relatórios em PHP.

TECNOLOGIAS

A importância de extrair informações por meio de relatórios em sistemas vem como umas das principais necessidades para o usuário final. Mesmo no mais básico dos sistemas, os relatórios sempre serão necessários, pois o usuário final necessita de algo que traga informações que são previamente cadastradas. Essas informações muitas vezes precisam ser organizadas, agrupadas para produzir os resultados desejados.

O resultado final, após o processo de *design* do relatório, é apresentado ao usuário por meio da geração do relatório utilizando a o padrão PDF, que fora desenvolvido em 1993, pela *Adobe Systems*. O PDF foi largamente padronizado como formato para relatórios, devido a sua portabilidade entre sistemas operacionais, hardware, resoluções diversas de monitores, até em dispositivos móveis, mantendo seu formato original gerado tanto na visualização gráfica em monitor como também na versão impressa, esse foi um dos motivos da escolha do padrão PDF para utilização na ferramenta desse trabalho.

A portabilidade do PDF e flexibilidade para implementações futuras também foram

consideradas na fundamentação teórica deste trabalho. O conjunto de bibliotecas disponibilizado pela tecnologia FPDF permitem o mapeamento completo e absoluto dos objetos de interface do lado cliente (*client-side*) para o conversor em PDF do lado servidor (*server-side*) em PHP.

O *MySQL* e o *PostgreSQL*, principalmente, foram empregados no projeto com banco de dados padrão, devido a ampla utilização com a linguagem PHP, além da proposta inicial do projeto de utilizar apenas tecnologias de software livre, utilizou-se da extensão PDO (PHP *Data Object*), com isso possibilitou um nível alto de abstração no acesso aos dados, permitindo que outros banco de dados, integrem a ferramenta [Jaques 2010].

O arquivo gerado pela ferramenta tem como padrão a linguagem de programação PHP, que atualmente se encontra na versão 5.3.2.

Entre as razões para a escolha da linguagem PHP para o desenvolvimento deste trabalho, podem ser destacadas as seguintes: linguagem *open-source;* portabilidade (pode ser acessada por vários sistemas operacionais) e escalabilidade (é uma das linguagens de programação na internet, com mais de 40% das aplicações desenvolvidas). A linguagem PHP está instalada em mais de 22 milhões de servidores e possui mais de 2,5 milhões de desenvolvedores [Lerdorf; Tatroe; Macintyre; 2006].

A ferramenta também conta com implementações da biblioteca *jQuery*, desenvolvida por John Resig, que possibilita a geração do conteúdo dinâmico da ferramenta e também todas as funções de interatividade entre o desenvolvedor e a interface como interação no uso de formulários, parâmetros de entrada, dentre outros [Silva 2008].

Para o controle de todos os objetos, foi escolhido o padrão para troca de dados computacionais *JSON* (*Java Script Object Notation*), que faz o controle para serializar todos os objetos do editor e fazer a troca de dados dos objetos da tela no formato de linguagem de programação *Java Script* para o PHP. Foi definida uma estrutura básica em JSON para todos os objetos criados, onde são guardadas tanto informações básicas de estilos do mesmo como também informações de texto do objeto. Sem o JSON, a possibilidade de serialização dos objetos do Java Script para PHP se tornaria muito custoso para a ferramenta [Crockford 2006].

Esta estrutura foi utilizada como base para que o desenvolvedor conseguisse gravar e abrir projetos na ferramenta. Adotou-se a linguagem de marcação XML (*Extensible Markup Language*), padrão recomendado pela W3C para gerar uma infraestrutura de marcação de dados. Foi escolhido este padrão de marcação de dados XML devido a sua flexibilidade. Este arquivo XML contém todas as informações dos objetos e também das configurações que o desenvolvedor previamente configurou.

Problema de Posicionamento Absoluto

Um dos maiores problemas em termos da geração de relatórios em sistemas para internet e mesmo no desenvolvimento de *layout* de *websites* concentra-se nas diferentes configurações de monitor, resoluções e navegadores diversos, sendo comum o mesmo código-fonte em um local aparecer diferentemente em outro dependendo das configurações.

Os dois principais tipos de posições mais usadas neste projeto são a posição absoluta e relativa.

Posicionamento absoluto {*position: absolute*} é determinado em relação ao canto superior esquerdo da janela do navegador. As coordenadas de um ponto exprimem-se então de cima para baixo (*top-down*) e da esquerda para a direita (*left-right*).

A posição relativa é determinada em

relação aos outros elementos, como uma imagem, o que quer dizer que os elementos contidos nas *tags* DIV ou SPAN serão posicionados na sequência dos elementos HTML depois dos quais se encontram.

O problema está em que objetos com posições absolutas podem parecer corretos em uma determinada resolução, mas totalmente distorcida em outro.

Partindo do principio que essas propriedades, referentes ao posicionamento, são determinadas por meio do CSS (*Cascading Style Sheets*) usado para formatar os conteúdos estruturados de *layouts* de documentos HTML. Portanto, como a ferramenta necessita das posições de cada objeto na tela para gerar o relatório, apenas as propriedades CSS dos objetos não conseguiriam tratar todas as particularidades necessárias para a geração do PDF.

Padronização do arquivo PDF com a interface da ferramenta

Outro problema que surgiu durante o desenvolvimento da ferramenta foi as diferentes unidades de medida utilizadas, pois na interface gráfica do navegador todas as informações são tratadas e obtidas em *pixels*, já nos relatórios as informações tem que ser posicionadas utilizando-se do plano cartesiano, e medido em milímetros.

Não existe uma fórmula que converta *pixels* em milímetros exatamente, devido ao fato do *pixel* ser definido dependendo da resolução e tamanho do monitor. De acordo com Samy (2004) existe uma média do tamanho de pixel em milímetros, mas como a ferramenta necessita das coordenadas exatas conforme o desenvolvedor determinou não seria possível a utilização dessa média.

Foi realizado o tratamento de conversões de *pixels* para milímetros, mantendo-se a proporção do tamanho original do papel A4 (210x297mm) onde o comprimento tem 70,7% do tamanho da sua altura. A proporção de *pixel* x milímetros ficou com a largura e altura na tela (566x800 *pixels*) onde cada pixel equivale a 0,37mm.

EDITOR VISUAL UTILIZANDO TECNOLOGIAS ASSÍNCRONAS PARA INTERNET

Esta seção apresenta e descreve o editor visual que faz parte da ferramenta desenvolvida nesse trabalho. A ferramenta foi construída de forma a permitir ao desenvolvedor de sistemas para internet em PHP, crie seus relatórios necessários para seu sistema sem a necessidade de que seja feita a programação linha a linha, como acontece no processo convencional.

A ferramenta visa diminuir o trabalho do desenvolvedor assim como também o tempo gasto pelo mesmo, para gerar relatórios, devido ao grande número de informações que são necessárias em relatórios. Para se construir uma ferramenta eficaz e que realmente seja eficiente foram utilizadas diversas tecnologias na interface como, jQuery, JSON, XML e CSS.

Apesar de o desenvolvedor obter o arquivo em PHP apenas quando aciona a opção “Gerar relatório”, grande parte do processamento da ferramenta está junto à interface gráfica, no momento em que o desenvolvedor está elaborando o relatório.

Para equilibrar a usabilidade e o desempenho da ferramenta, diversas tarefas são executadas de forma assíncrona, ou seja, transparente ao desenvolvedor, possibilitando assim que os processos mais custosos sejam tratados à parte. Assim a ferramenta funciona normalmente enquanto o processo é executado em paralelo e, quando o mesmo finalizar, recupera-se o resultado que é exibido ao desenvolvedor [Aece 2008].

Para realizar as chamadas assíncronas utilizou-se da tecnologia AJAX (*Asynchronous*

Javascript And XML) utilizando um objeto nativo dos navegadores, XMLHttpRequest, que permite essa comunicação assíncrona com o servidor. No projeto, toda a comunicação e verificação são realizadas por funções do *framework jQuery*, necessitando-se apenas informar os dados [Chaffer J and Swedberg 2009].

Módulo Visual

No editor visual, o JSON é utilizado para encapsular informações como tipo de objeto, estilos definidos pelos desenvolvedores e informações do banco de dados escolhido.

Para aproximar-se de um padrão semelhante dos editores visuais já existentes no mercado de outras linguagens, mantendo para o desenvolvedor uma interface familiar sem dificultar sua utilização, foi mantido o padrão de "Arrastar-Soltar" (*drag-drop*) de objetos por meio do complemento jQuery UI (*User Interface*) [Ferreira 2010].

Com o uso da função de arrastar os componentes, é feita toda a manipulação de suas informações de posicionamento que são atualizadas apenas quando o objeto é solto pelo desenvolvedor. Isso é feito de forma assíncrona, assim o desenvolvedor não percebe que todo o processamento da ferramenta é feito ao mesmo tempo em que ele desenvolve um relatório em sua interface.

A edição de um objeto, assim como, sua exclusão podem ser realizadas com o clique no objeto que trás uma caixa de edição com as opções para o desenvolvedor, conforme a Figura 1.



Figura 1. Caixa de edição e exclusão do objeto

Todos os objetos podem ser redimensionados sem ter a necessidade de editar o objeto, apenas arrastando sua base inferior direita, aumentando ou diminuindo seu tamanho.

Ao redimensionar um objeto todas suas propriedades de tamanho, altura e largura, assim como suas posições são atualizadas.

A exclusão do objeto foi tratada de forma lógica, ou seja, mudando seu *status* de 1 (*um*) para ativo e 0 (*zero*) para inativo, isto foi realizado dessa forma pensando-se em atender uma futura implementação da função de voltar ou desfazer a ação do usuário (*Ctrl-Z*).

A Figura 2 mostra a interface gráfica da ferramenta desenvolvida neste trabalho.

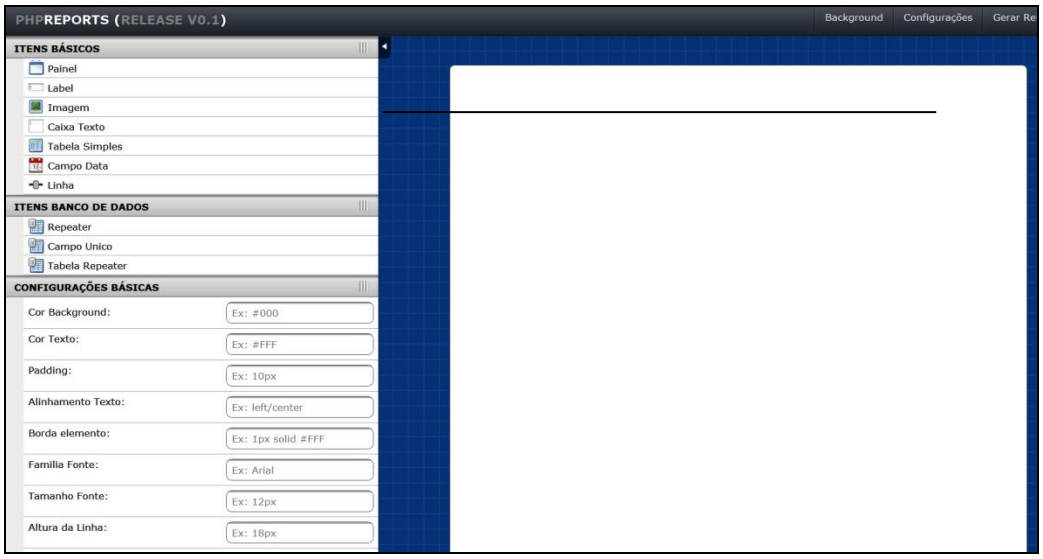

Figura 2. Interface gráfica da ferramenta

Mapeamento para PHP e a geração dos relatórios

Todo o processamento (*back-end*) é feito tanto na interface enquanto o desenvolvedor realiza a confecção do relatório como também na conversão que é realizada após a serialização dos objetos e esses convertidos do JSON para um *Array* serializado, cujos dados serão tratados pelo PHP.

De início, foi criada uma estrutura que fizesse com que os objetos criados pelo usuário fossem serializados, possibilitando assim a passagem de toda a estrutura da interface gráfica para a próxima tela onde é gerado o arquivo PHP.

A estrutura foi feita para que fosse compatível com todas as informações dos diversos objetos. A estrutura em JSON foi dividida em duas partes principais, a parte de estilo, onde são guardadas todas as características do objeto, como cores, tamanho, fontes, e também a parte de informações, onde ficam as propriedades do objeto especificando sua utilização, como mostra a Tabela 2.

Tabela 2. Estrutura de dados JSON dos objetos

```
var data = {
        "style": [
          {
"background" : "" ,
            "color" : "",
            "border"    :
"",
            "fontfamily"
: "",
            "padding"  :
"",
            "textalign"  :
"",
            "fontsize"  :
"",
            "width" : "",
            "height" : "",
            "zindex"    :
"",
            "top":"",
            "left":""
          }
        ],
        "info":[
          {
            "id" : "" ,
            "opts"   :
"" ,
            "status"
: "",
            "texto"  :
"" ,
            "local"  :
"" ,
            "coluna"
: "" ,
            "SQL"   :
"" ,
            "linha"  :
""
          }
        ]
      };
```

Nessas estruturas estão todas as configurações do documento que serão a base do relatório, como nome do autor, tipo do papel (horizontal ou vertical), nome do documento e conexões com o banco de dados. São realizados os tratamentos do cabeçalho e o rodapé para que sejam padronizados em todas as páginas do relatório.

Objetos que necessitam de conexão com o banco de dados usam o tratamento do próprio PDO para exceções e erros de conexão.

CONCLUSÕES

A inexistência de ferramentas destinadas à confecção de relatórios com o intuito de facilitar e minimizar o tempo e o trabalho do desenvolvedor de aplicações web em PHP deu origem a este trabalho.

Os experimentos foram baseados em testes funcionais realizados utilizando o gerador de relatórios comparativamente ao desenvolvimento de relatórios no modo tradicional. O gerador apresentou resultados satisfatórios no que diz respeito a facilidade de uso, se comparado ao tradicional trabalho do programador diretamente em código PHP. O principal ganho observado foi: no modo tradicional enquanto o programador testava suas coordenadas (X e Y), para melhor distribuir os objetos em tela; no editor além de não haver mais a necessidade de posicionamentos de objetos em tela baseando-se em coordenadas supostas, a área de trabalho do editor representa os resultados em uma página muito próxima ao resultado final em PDF.

A facilidade promovida pela interface gráfica do projeto faz com que o usuário através da possibilidade de arrastar e soltar (*drag-and-drop*) os objetos na posição desejada e também de poder redimensionar como deseja esses objetos se revelou de grande importância, pois evita que o usuário tenha que ir programando e testando várias vezes ate que se tenha o resultado desejado para realizar esta tarefa.

Com a ferramenta deste trabalho percebeu-se, também, uma melhora significativa no desenvolvimento dos relatórios. Isso devido à interface gráfica que auxilia o desenvolvedor, não necessitando que seja feito da forma convencional, posicionando e estruturando cada objeto manualmente.

Assim, percebe-se que o objetivo da pesquisa foi atingido satisfatoriamente com a geração de arquivos com extensão PHP que produzem como resultado final relatórios no formato PDF, e que esses possam ser incluídos em sistemas desenvolvidos em PHP.

Na ferramenta também são disponibilizados itens para a criação de relatórios, básicos como *label*, imagem, tabelas simples, linhas, painel e também itens com conexões ao banco de dados como campo único SQL e tabela *repeater*. Conforme são apresentados na Figura 3.

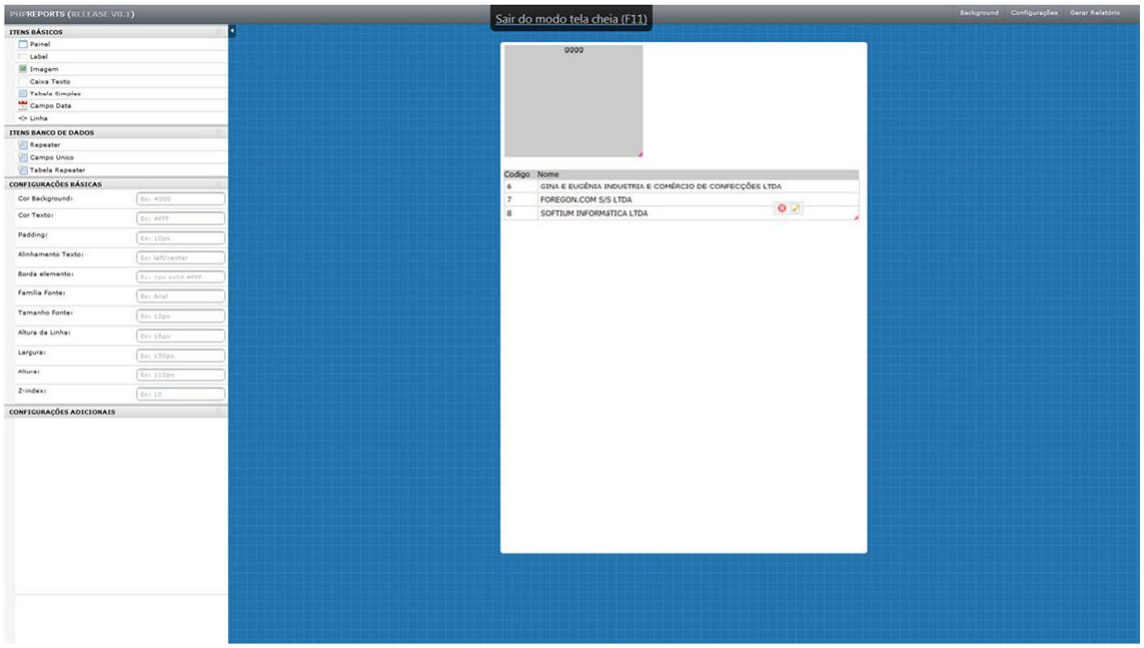

Figura 3. Interface gráfica da ferramenta com imagens, SQL e *repeater*

Para casos de não atendimento de todas as especificidades de um determinado relatório como: gráficos e planilhas, a ferramenta auxilia o desenvolvedor na geração do o arquivo base do relatório em PHP. O arquivo base em PHP é estruturado com todos os mapeamentos necessários elaborados no editor, e pela característica do PHP ser uma linguagem *script*, onde o código é interpretado somente no momento da requisição do arquivo pelo usuário no servidor, tal arquivo permite a personalização por parte do desenvolvedor.

Para implementações futuras, espera-se a agregação de novas funcionalidades no editor, além da possibilidade de restaurar a partir de arquivos base PHP, o relatório para o editor. Esse processo é complexo devido a necessidade de reconstruir os objetos de tela a partir de descrições não formais destes objetos em PHP.

Uma das principais dificuldades encontradas está no reposicionamento de todos os objetos na tela após a modificação da orientação do papel de horizontal para vertical ou vice-versa.

A partir do uso da ferramenta desenvolvida, a atividade de confecção de relatórios torna-se mais interativa e menos onerosa em termos de tempo.

REFERÊNCIAS

AECE, I. (2008). **"**WCF - Chamadas Assíncronas". http://www.israelaece.com/post/WCF-Chamadas-Assincronas.aspx. Setembro.

Cetic.br, and Nic.br (2009) "Pesquisa sobre o uso das tecnologias da informação e da comunicação no Brasil". http://www.cetic.br/tic/2009/index.htm. Março.

CHAFFER J and SWEDBERG, K (2009), "Learning JQuery 1.3". Packt Publishing. 444p.

CONVERSE, T. and PARK, J. (2003) "Php - A Biblia". Editora: Campus, 904p.

CROCKFORD, D. (2006), "Definições de Notação JavaScript Object (JSON)", RFC 4627.

FERREIRA, D. (2010). "Interface Drag and Drop". http://www.daviferreira.com/blog/post/9/interface-drag-and-drop-com-jquery.html, Março.

JAQUES, R. (2010). "PHP e Unicode - O caminho das pedras". http://www.phpit.com.br/artigos/php-e-unicode-o-caminho-das-pedras.phpit. Agosto.

LERDORF, Tatroe and MACINTYRE (2006) "Programming PHP – 2nd Edition". O´Reilly. 500p.

OLIVER (2006), "FPDF, a PHP class which allows to generate PDF without using the PDFlib library". http://www.weberdev.com/ViewArticle/FPDF,--a-PHP-class-which-allows-to-generate-PDF-without-using-the-PDFlib-library, Abril.

PARAISO, D. (2010) "A Brazilian Portuguese version of the 1.6 manual FPDF". http://www.fpdf.org/en/dl.php?id=97. Março.

SAMY, M. (2004) "As medidas CSS de comprimento". http://maujor.com/tutorial/medidascss.php. Abril.

SILVA, M. S. (2008) "JQuery: A biblioteca do desenvolvedor Javascript". Editora Novatec.432p

TIOBE (2010) "TIOBE Programming Community index for March 2010". http://www.tiobe.com/index.php/content/paperinfo/tpci/index.html. Março.