

**UNIVERSIDADE PAULISTA – UNIP
CIÊNCIA DA COMPUTAÇÃO**

GERADOR DE SISTEMAS WEB AUTOMATICO

NICHOLAS KEVIN NUNES DE OLIVEIRA

**Limeira
2016**

UNIVERSIDADE PAULISTA – UNIP

NICHOLAS KEVIN NUNES DE OLIVEIRA

GERADOR DE SISTEMAS WEB AUTOMATICO

Trabalho de conclusão de curso para obtenção do título de Bacharel em ciência da computação pela Universidade Paulista.
Professores Orientadores: M. Marcos Vinicius Gialdi e M. Antônio Mateus Locci

Limeira

2016

NICHOLAS KEVIN NUNES DE OLIVEIRA

GERADOR DE SISTEMAS WEB AUTOMATICO

Trabalho de conclusão de curso para obtenção do título de Bacharel em ciência da computação pela Universidade Paulista. Professores Orientadores: M. Marcos Vinicius Gialdi e M. Antônio Mateus Locci

Aprovada em ____ de _____ de 201__.

BANCA EXAMINADORA

Prof. Dr. Nome Completo

Prof. Dr. Nome Completo

Prof. Dr. Nome Completo

DEDICATÓRIA

Dedico esse trabalho aos meus pais que me ajudaram e me apoiaram em todo o momento do curso, aos Mestres Professores pela paciência e apoio e aos amigos que estiveram todo o tempo ao meu lado.

“O homem erudito é um descobridor de fatos que já existem - mas o homem sábio é um criador de valores que não existem e que ele faz existir”

Albert Einstein

RESUMO

Com o intuito de facilitar e agilizar o desenvolvimento de softwares, esse projeto tem como objetivo criar um sistema no qual gere automaticamente a estrutura base, telas básicas e relatório de um sistema, sem a necessidade de um programador. Quando um sistema vai ser desenvolvido do zero algumas preocupações devem ser levadas em conta, qual linguagem de programação utilizar, criar a estrutura do sistema (motor), o que demanda um grande tempo. Esse projeto faz com que um simples usuário consiga gerar essa estrutura e criar telas básicas junto a relatórios sem a necessidade de saber nada de programação de sistema.

Palavras-Chave: PHP, Java Script, HTML5, CSS3, Gerador, Sistema, WEB, MVC, Orientação a Objetos.

ABSTRACT

In order to facilitate and speed up the software development, this project aims to create a system which it automatically manages the basic structure, basic screens and reporting of a system, without need a programmer. When a system is on development from zero some concerns should be taken, which programming language use, create the system structure (engine), which demands too much time. This project allows a simple user to generate his structure and create basic screens and reports lists without need to know anything about system development.

Key-Words: PHP, Java Script, HTML5, CSS3, Generator, System, WEB, MVC, Object-oriented.

LISTA DE FIGURAS

Figura 1 Funcionamento Gerador de sistemas.....	17
Figura 2 Gerador de sistema em Desktop	21
Figura 3 Gerador de sistema no celular	21
Figura 4 Exemplo de classe.....	22
Figura 5 Local de instalação do código fonte.....	23
Figura 6 Tela de autenticação	24
Figura 7 Código fonte.....	25
Figura 8 Código fonte backend.....	27

LISTA DE QUADROS

1. TECNOLOGIAS UTILIZADAS	14
---------------------------------	----

LISTA DE ABREVIATURAS

PHP	Personal Home Page (Página Pessoal)
CSS	Cascading Style Sheet, (Folha de estilo em cascata)
	HTML HyperText Markup Language (Linguagem de Marcação de Hipertexto)
MVC	Model, View, Control (Modelo, Visualização, Controle)
SQL	Structured Query Language (Linguagem de Consulta Estruturada)

SUMÁRIO

1. INTRODUÇÃO.....	12
2. TECNOLOGIAS E METODOS UTILIZADOS.....	14
2.1 TECNOLOGIAS.....	14
2.2 MÉTODOS.....	15
2.2.1 MVC.....	15
2.2.2 ORIENTAÇÃO A OBEJETOS.....	15
2.2.3 SISTEMA REPONSIVO E MULTIPLATAFORMA.....	15
4. GERADOR DE SISTEMAS WEB.....	20
4.1 CLASSES DO CODIGO FONTE.....	22
4.2 INSTALANDO O GERADOR.....	22
4.1.1 LOCAL DE INSTALAÇÃO DO CÓDIGO FONTE.....	23
4.1.2 SCRIPT SQL.....	23
4.3 AUTENTICAÇÃO DO SISTEMA.....	24
4.4 TELAS DO SISTEMA.....	25
4.4.1 REGISTRO DE ESTRUTURAS.....	25
4.4.2 REGISTRO DAS TELAS DE CADASTRO.....	26
4.4.3 RELATÓRIOS.....	27
5. CONCLUSÃO.....	28
6. REFERENCIAS BIBLIOGRAFICAS.....	29

1. INTRODUÇÃO

Tendo em vista o lado do usuário de algum sistema no qual trabalha em uma empresa de pequeno ou médio porte e esse mesmo sistema é contratado por uma empresa no qual desenvolve para razões sociais desse nível quando se trata da questão de gestão desses sistemas muitas vezes gera certos constrangimentos no momento em que o mesmo necessita de alterações ou novos desenvolvimentos no sistema, pois ele dependera da demanda do setor de programação.

Na área de desenvolvimento de sistemas para atender os seus clientes, existe uma fila enorme de alterações e correções. Muitas das vezes essas solicitações dos usuários são urgentes ou imediatas, o que impossibilita que o usuário possa esperar, porem como essa espera é inevitável, acaba resultando em descontentamento do usuário com o sistema e a empresa fornecedora do mesmo e conflitos entre eles.

Outro problema relacionado a sistemas da atualizada é a questão da mobilidade, na maioria das vezes o usuário necessita de um sistema para gestão de sua empresa onde não seja necessário estar com um computador, mas sim um sistema onde ele possa acessar de qualquer dispositivo não importando o tamanho da tela ou o sistema operacional do mesmo.

Segundo Gustavo Brigatto da revista Valor (2015, pag. 1), sobre o aumento da mobilidade nos tempos atuais:

“O número de smartphones em uso no Brasil passou o de computadores e tablets. Ao todo, são 154 milhões, contra 152 milhões, segundo pesquisa da FGV-SP. Com esse resultado, o país chegou a 306 milhões de dispositivos conectáveis a internet, o equivalente a três aparelhos para cada dois habitantes. Há cinco anos, a proporção era de um aparelho para cada dois habitantes.

[...]

Outra questão foi a desaceleração das vendas de PCs no ano passado, a primeira em 30 anos. De acordo com Meirelles, foram 20,4 milhões de unidades vendidas, uma queda de 10%".

Com tudo isso em mente foi pensado no desenvolvimento de um gerador sistemas na plataforma web, onde o próprio usuário gera seu próprio sistema sem a necessidade de colocar as mãos em códigos fontes, o mesmo gera as telas dos sistemas e os relatórios para gestão de seus dados e informações. A plataforma web contribui para que os sistemas gerados e também o próprio gerador sejam executados a partir de navegadores, com isso os mesmos já se garantem ser multiplataforma, e utilizando bibliotecas Java Script e CSS para montar o visual dos sistemas, consegue-se com que os mesmos sejam totalmente responsivos.

Outra característica para o gerador de sistemas web é o mesmo ser totalmente de código aberto, ou seja, como essa comunidade vem crescendo ao longo do tempo, qualquer programador pode adquirir esse algoritmo, entende-lo e o otimizando cada vez mais, contribuindo para que o gerador seja mais preciso para demandas de diferentes usuários.

2. TECNOLOGIAS E METODOS UTILIZADOS

A seguir serão listadas todas as tecnologias e métodos utilizados para desenvolver o gerador de sistemas.

2.1 TECNOLOGIAS

Quadro 1 Tecnologias utilizadas

Nome	Versão	Descrição
PHP	7	É uma linguagem em formato de script, desenvolvida totalmente para sistemas web totalmente compatível com as linguagens de programação HTML e Java Script. O PHP soluciona problemas web de forma simples e fica em execução no lado do servidor.
Java Script	-	É uma linguagem de programação em scripts e é a linguagem mais utilizada em <i>client-side</i> nos navegadores web da atualidade.
CSS	3	O CSS (<i>Cascading Style Sheet</i> , em português Folha de estilo em cascata) é uma linguagem integrada ao HTML para definir os estilos dos formulários, telas e campos do sistema.
HTML	5	Linguagem de programação utilizada para construção da parte de <i>frontend</i> de páginas web.
Apache	2	É um servidor web multiplataforma para PHP, também é o mais utilizado do mundo.
MySQL	5.5	MySQL é um banco de dados que utiliza SQL como interface para seu desenvolvimento. É um dos sistemas de banco de dados mais utilizado do mundo pois além de robusto é gratuito.

Autor: Autoria própria

2.2 MÉTODOS

Os métodos utilizados no projeto são as formas como ele foi programado, a forma de estruturação de arquivos de código fonte e para que tipo de dispositivos foram desenvolvidos.

2.2.1 MVC

O MVC é uma metodologia de estruturação dos arquivos de código fonte, que os organiza de forma prática e fácil para manutenção.

Essa metodologia funciona da seguinte maneira: Separar os arquivos de cada tela em arquivos de modelo, controle e visualização. Os arquivos de modelo irão ficar com a parte relacional de comunicação com o banco de dados. Os arquivos de controle armazenam as funções e classes da tela e por fim os arquivos de visualização serão os totais responsáveis pelo *frontend*.

2.2.2 ORIENTAÇÃO A OBJETOS

Orientação a objetos é um método de programação no qual as funções são desenvolvidas em classes, e quando a necessidade de chamar uma dessas funções, basta apenas instanciar essa classe. Tem como propósito reaproveitamento de código fonte.

2.2.3 SISTEMA RESPONSIVO E MULTIPLATAFORMA

Um sistema responsivo é aquele no qual se adequa para qualquer tamanho de tela, possibilitando que o usuário o possa usar em qualquer dispositivo. Quando a parte de multiplataforma é fazer com que o sistema não apenas execute em apenas um sistema operacional, mas sim em todos eles sem a necessidade de haver um código fonte para cada sistema operacional.

3. GERAÇÃO DE SISTEMAS

Um gerador de sistemas é um algoritmo que tem como finalidade gerar códigos fontes, estrutura de arquivos, e até mesmo gerar o banco de dados do sistema tudo de forma automática sem a necessidade de um programador para desenvolver esse mesmo código.

A ideia de geradores de sistemas surgiu em 2003 com o desenvolvimento do GAS (Gerador automático de sistemas) no qual gerava sistemas em visual basic. Logo outros geradores começaram a surgir porém, todos corporativos, e normalmente sempre para o sistema operacional Windows, em alguns deles ainda a necessidade de se realizar uma programação interna, ou seja, do mesmo jeito ainda necessitava de um programador.

Sobre como construir um gerador de sistemas J. Herrington (2012) afirma que se deve:

“Construir o código de saída: deve-se construir o código de saída manualmente, para determinar o que e como deve ser a saída do gerador. Esta etapa facilita a visualização do que deve ser extraído dos arquivos de entrada;

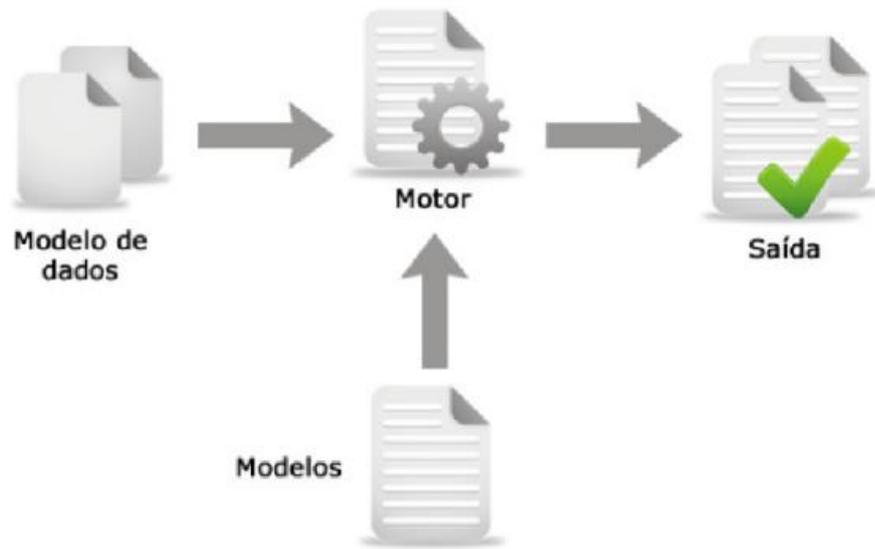
Projetar o gerador: deve-se determinar qual será o formato de entrada, como a entrada será analisada e como o código de saída será gerado, usando templates ou simplesmente escrevendo o código;

Analisar a entrada: deve-se implementar o código para ler o arquivo de entrada e extrair as informações necessárias para construir a saída;

Especificar templates: uma vez que a análise da entrada está definida, o próximo passo é criar os templates que serão utilizados para gerar a saída;

Gerar a saída: este é o último passo, onde deve ser implementada a geração de código utilizando os templates especificados na etapa anterior”.

Figura 1 Funcionamento Gerador de sistemas



Autor: David (<http://www.informatiqueiro.com.br/geracao-automatica-de-codigo/>)

A necessidade de geradores automático de sistemas surge quando toda vez que seja necessário o desenvolvimento de um novo sistema ou a manutenção do mesmo, precisa-se que essa solicitação de desenvolvimento entre em uma fila de projetos para depois passar por reuniões para repassar a ideia do solicitante para os responsáveis pela criação do sistema, analistas de sistemas, desenvolvedores e setores de testes, porém esse processo todo é muito demorado e o usuário no qual necessita do sistema não tem disponibilidade de espera gerando um constrangimento para o mesmo. Algumas vezes as solicitações de alterações no sistema realizadas pelo usuário não é algo muito complexo nem menos possui uma demanda de tempo dos desenvolvedores e analistas muito alta, mas como essa solicitação era entrar na fila de desenvolvimento a espera do solicitante pode-se estender para meses, e caso seja um desenvolvimento de urgência, como um relatório para entregar para a gerencia de uma empresa por exemplo, o solicitante terá um descontentamento com o sistema e gerar certos tipos de conflitos.

Tendo tudo isso em mente, nasce a ideia de se criar gerador de sistemas automático, no qual o usuário gera sistemas de forma rápida e prática conforme sua demanda sem a necessidade da intervenção de programadores ou analistas de sistemas. Um gerador de sistemas pode criar estruturas e sistemas automaticamente, telas, relatórios entre outros

componentes para o sistema sempre quando surgisse a sua necessidade. Outra vantagem do gerador de sistemas é sua estrutura, todos os sistemas e seus componentes serão gerados na mesma estrutura, que é a qual o gerador é programado para fazer, gerando assim um padrão entre os sistemas.

J. Herrington (2012) sobre as vantagens de se utilizar geradores de código para o desenvolvimento de software:

“Sobre qualidade, código escrito manualmente tende a ter um nível de qualidade muito irregular visto que, durante o desenvolvimento da aplicação, podem ser propostas melhores abordagens para solucionar os problemas;

Sobre produtividade: quando são usados geradores de código, o volume de código digitado manualmente é bem menor se comparado ao volume de código escrito para aplicações desenvolvidas sem o uso dessas ferramentas. Sendo assim, tem-se mais tempo para outras etapas do projeto, como, por exemplo, a fase de testes;

Sobre abstração: a definição de templates é bem mais simplificada que o código alvo. Com o uso de templates, pode-se corrigir erros do projeto ou incluir novas funcionalidades apenas reescrevendo os templates. Além disso, o gerador pode ser facilmente reprojetoado para outras linguagens ou tecnologias”.

3.1 ESTRUTURAS DE SISTEMAS

Estrutura de um sistema de forma técnica significa como todo o código fonte do sistema é organizado, normalmente quando surge um projeto para o desenvolvimento de um sistema, os analistas junto aos programadores ficam horas, ou até mesmo dias decidindo em qual linguagem de programação o mesmo será feito e qual estrutura de organização será utilizada, essa decisão é importante pois será toda a base do sistema. Quando a empresa possui muitos sistemas é difícil manter o padrão de utilizar em todos os sistemas a mesma estrutura.

A vantagem de um gerador de sistemas retira todo esse problema já que ele irá sempre manter a mesma estrutura de arquivos padronizada em sua programação para tudo que ele gerar.

4. GERADOR DE SISTEMAS WEB

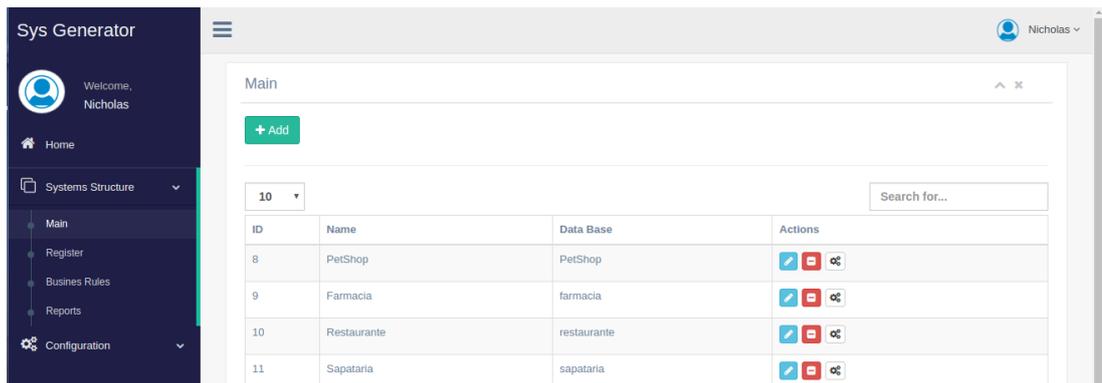
Esse projeto foi baseado com uma estrutura para gerar sistemas WEB com o *backend* desenvolvido totalmente em PHP7, que tem a finalidade de se comunicar o banco de dados e devolver os dados para o *frontend* no qual as telas são montadas via Java Script, HTML e CSS, por um código estruturador em PHP orientado a objetos. O *middleware* para que o *frontend* se comunique com o *backend* foi desenvolvido em uma tecnologia de Java Script chamada comunicações AJAX, no qual deixa o sistema mais rápido, flexível e seguro, os sistemas gerados a partir do gerador terão o mesmo layout e estrutura que o próprio gerador.

O gerador de sistemas WEB é totalmente de código aberto permitindo para que o usuário que o utilize possa fazer alterações em seu algoritmo para deixa-lo cada vez melhor, mais preciso e seguro ou até mesmo para atender necessidades específicas para ele. Também permite o usuário compartilhar suas alterações nas comunidades de programação de código aberto, para que outros usuários o ajude e torne o sistema cada vez melhor.

Outro atributo que gerador de sistema WEB e os sistemas gerados a partir do mesmo possuem é serem totalmente responsivos e multiplataforma, ou seja, ele se adequa a qualquer tipo e tamanho de tela de qualquer dispositivo, algo que é muito importante com a demanda de hoje em dia, com celulares, *tablets*, computadores, notebooks e cada vez mais dispositivos vem surgindo sempre com diferentes sistemas operacionais. A tecnologia que torna isso real, é a biblioteca CSS e Java Script de código aberto chamada *BootStrap* junto com a lógica de programação utilizada no sistema.

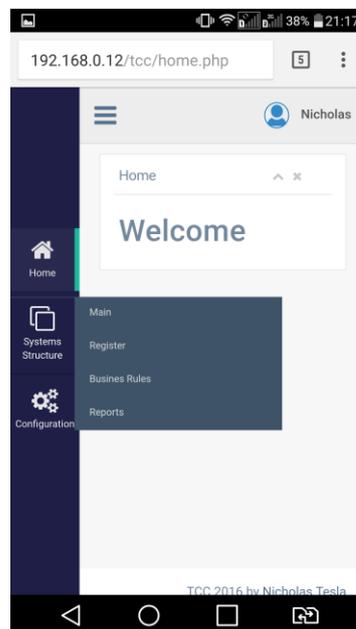
O banco de dados no qual o gerador utiliza para armazenar seus dados, e onde cria os esquemas onde os sistemas gerados também irão utilizar é o MySQL, pois é um banco de dados robusto, simples para utilizar a manusear dados e também não tem nenhum custo.

Figura 2 Gerador de sistema em Desktop



Autor: Autoria própria

Figura 3 Gerador de sistema no celular

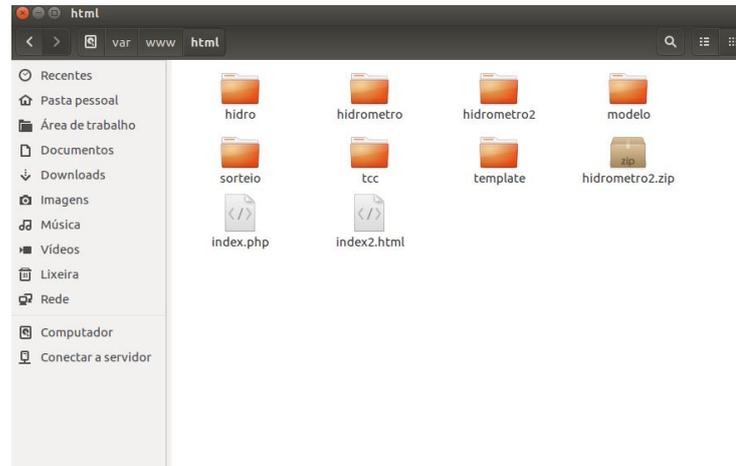


Autor: Autoria própria

Um exemplo prático para a necessidade do uso desse gerador de sistemas seria a gestão de informações de uma escola, onde todas as telas não possuem nenhuma complexidade. O próprio diretor ou coordenador da escola poderia gerar o sistema para atender as suas necessidades, como cadastro de alunos, cursos, matérias, turmas, etc. Supondo que depois surja a necessidade da criação de telas e relatórios para controle de armário de alunos, ao invés de ter todo o trabalho de abrir solicitação para uma empresa responsável por um sistema, esperar as análises e desenvolvimento e muitas das vezes algum valor adicional, o próprio responsável pelo controle dos

4.1.1 LOCAL DE INSTALAÇÃO DO CÓDIGO FONTE

Figura 5 Local de instalação do código fonte



Autor: Autoria própria

Nesse caso foi utilizado como servidor o sistema operacional Linux *Ubuntu* 16.04 LTS.

4.1.2 SCRIPT SQL

```
use tcc;
```

```
create table users (  
id int(10) primary key auto_increment,  
login varchar(20),  
password varchar(100),  
name varchar(100)  
);
```

```
create table menu (  
id int(10) primary key auto_increment,  
name varchar(100),  
id_pai int(10)  
);
```

```
create table register (  
id int(10) primary key auto_increment,  
name varchar(100)  
);
```

```
create table fields (  

```

```
id int(10) primary key auto_increment,  
name varchar(100),  
id_register int (10)  
);
```

```
create table reports (  
id int(10) primary key auto_increment,  
name varchar(100)  
);
```

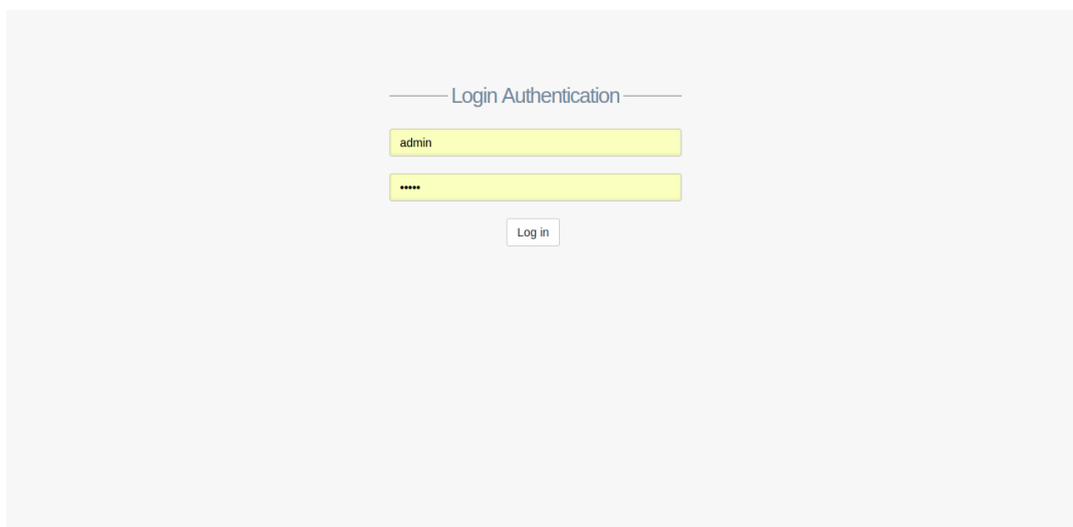
```
create table reports_field (  
id int(10) primary key auto_increment,  
id_report int (10),  
data varchar(100),  
field varchar(100)  
);
```

```
alter table menu add column icon varchar(100);
```

4.3 AUTENTICAÇÃO DO SISTEMA

Como descrito no capítulo 4, todos os sistemas gerados terão o mesmo **layout** que o gerador, então a tela de autenticação será igual para todos os sistemas, mas lembrando que o código está aberto para alterações caso o usuário necessite.

Figura 6 Tela de autenticação



Autor: autoria própria

4.4 TELAS DO SISTEMA

O sistema é formado pelas telas de registro de estruturas, registro das telas de cadastro comum, registro de relatórios e cadastro de usuários do sistema.

4.4.1 REGISTRO DE ESTRUTURAS

Nessa tela será onde o usuário irá começar a criar seus sistemas, onde para criar uma estrutura nova precisa-se apenas do nome do sistema e do nome do banco de dados para o mesmo.

Figura 7 Código fonte

```
//inicia pagina
$pagina = new Pagina();
$pagina->iniciaPagina();
$pagina->iniciaConteudo();
//painel grid
$pagina->iniciaPainel(12, 'Main', '', 'menu'); //(tamanho, titulo)
//cria o form (modal)
$form = new Form();
$form->campoTexto('Name', 'name', 'name', true, '');
$form->campoTexto('Data Base', 'data-base', 'data-base', true, '');
$form->startForm('form', 'Add Main');
$form->formDeletar();
//outros componentes da pagina
echo '<div id="alerta"></div>';
echo '<button type="button" id="btnAdd" class="btn btn-success"><span class="fa fa-plus" style="color:white;" aria-hidden="true"></span></button>';
echo '<hr>';
//grid
$grid = new Grid();
$grid->setGrid('grid_main', 'ID', 'Name', 'Data Base', 'Actions');
//finaliza os conteudos da pagina
$pagina->finalizaConteudos(3);
$pagina->finalizaConteudos(2);
//chama as bibliotecas
$pagina->finalizaPagina();
$grid->callGrid('ajax.Main.php', 'type':'select', 'POST');
$form->callForm();
//unset as variaveis
unset($pagina);
unset($grid);
unset($form);
//chama o js da pagina
echo '<script src="js/telas/main.js"></script>';
```

Autor: Autoria própria

Após registrar a estrutura, o registro novo aparece na grade, para que a geração do sistema seja efetuada será necessário clicar na engrenagem na grade. Após isso, o gerador executara um algoritmo que copiara uma estrutura padrão dos arquivos do código fonte para o diretório do apache, mudara o nome do projeto para o nome que o usuário colocou para o sistema, dentro dos arquivos do código fonte terão variáveis que serão alteradas com as

informações do sistema novo, como o título do sistema e o banco de dados que o sistema conectara. Para finalizar a geração o algoritmo executa um comando SQL no banco de dados para criar o esquema do sistema, criar a tabela de usuários e já definir um usuário administrador padrão.

4.4.2 REGISTRO DAS TELAS DE CADASTRO

Para adicionar telas novas ao sistema, basta apenas seguir para área de registro de telas de cadastro do gerador de sistemas. Nessa tela será necessário apenas informar o nome da tela, nome da tabela referente no banco de dados, cada campo da tela, as formatações dos campos e informar se esse campo é vinculado à alguma outra tela, para que então esse campo seja exibido ao usuário como uma caixa de seleção.

Para criar a tela o gerador de sistemas assume os campos informados para a tela e cria um objeto relacional da tela com o a tabela do banco dos de dados, nos quais os atributos serão os campos informados que terão os mesmos nomes na base de dados. Nesse objeto também já cotem os comandos *SQL*'s que serão utilizados durante os processos da tela que seriam: Selecionar os dados das telas, inserir um novo registro, editar um registro e delete um registro. Além do objeto relacional é criado o arquivo *JavaScript* da tela no qual é responsável por fazer as transições das e as chamadas para *backend* (objeto relaciona e banco de dados).

Figura 8 Código fonte backend

```
function manipulateData(type) {
    var $form = $('#form');

    if (type == 'delete') {
        data = $('#dataBase').val() + $('#data-base').val() + '&name=' + $('#name').val();
        if (type == 'insert') {
            alerta = '<div class="alert alert-success"><button type="button" class="close" data-dismiss="alert" aria-label="Close"><span ari
        )else
        {
            alerta = '<div class="alert alert-success"><button type="button" class="close" data-dismiss="alert" aria-label="Close"><span ari
        }
    }else
    {
        alerta = '<div class="alert alert-danger"><button type="button" class="close" data-dismiss="alert" aria-label="Close"><span ari-hid
        data = '';
    }

    if ($form.parsley().validate() || type == 'delete') {
        $.ajax({
            url: 'components/ajax/ajax.Main.php',
            data: 'type=' + type + '&data=' + data + '&id=' + __id,
            type: 'POST',
            success: function () {
                if (type == 'delete')
                {
                    $('#myDeleteModal').modal('toggle');
                }
                else
                {
                    $('#myModal').modal('toggle');
                }
                $('#alerta').html(alerta);
                $('#grid_main').DataTable().ajax.reload();
            }
        });
    } else
    return false;
}
}
```

Autor: Autoria própria

4.4.3 RELATÓRIOS

Para gerar relatórios, basta ir a área de registro de relatórios do gerador de sistemas, então ao selecionar para criar novo relatório, basta informar de qual tela será o mesmo e dar um nome a ele. Como a tela já está vinculada a um objeto relacional de uma tabela no código fonte do sistema gerado, o próprio gerador ira montar automaticamente o SQL do relatório. Também é possível informar quais são os filtros para gerar esse relatório e dizer se são obrigatórios ou não.

Os relatórios gerados aparecerão no menu lateral com os devidos nomes dados a eles como submenu da opção relatórios

5. CONCLUSÃO

Com o gerador de sistemas web, usuários nos quais possuem empresas de pequeno ou médio porte não precisarão se preocupar em comprar algum sistema para sua gestão, podendo de uma maneira simples gerar seus próprios sistemas da melhor maneira pelo menos e poderão fazer manutenções sempre que necessário. Como os sistemas gerados são totalmente responsivos, os usuários podem acessar seus sistemas de qualquer lugar. E sendo também um sistema de código aberto o gerador de sistemas web poderá ser sempre ser atualizado de acordo com as necessidades de seus usuários caso precisem, e compartilhando suas alterações os sistemas gerados serão cada vez mais precisos.

6. REFERENCIAS BIBLIOGRAFICAS

GERADOR de sistemas. 2012. 1 p. Gerador de sistemas (Ciência da computação) - Universidade Paulista - UNIP, Limeira, 2016. Disponível em: <<http://www.informatiqueiro.com.br/geracao-automatica-de-codigo/>>. Acesso em: 05 set. 2016.

AUMENTO do uso de *smartphones*. 2012. 1 p. Gerador de sistemas (Ciência da computação) - Universidade Paulista - UNIP, Limeira, 2016. Disponível em: <<http://www.valor.com.br/empresas/4010440/numero-de-smartphones-passa-o-de-computadores-e-tablets-no-brasil>>. Acesso em: 10 ago. 2016.

AUMENTO do uso de *smartphones*. 2012. 1 p. Gerador de sistemas (Ciência da computação) - Universidade Paulista - UNIP, Limeira, 2016. Disponível em: <<http://exame.abril.com.br/tecnologia/numero-de-smartphones-supera-o-de-computadores-no-brasil/>>. Acesso em: 15 ago. 2016.

AUMENTO do uso de *smartphones*. 2012. 1 p. Gerador de sistemas (Ciência da computação) - Universidade Paulista - UNIP, Limeira, 2016. Disponível em: <<http://www.webartigos.com/artigos/avanco-tecnologico-dos-celulares/106443/>>. Acesso em: 16 ago. 2016.

CICLO de vida do software. 2012. 1 p. Gerador de sistemas (Ciência da computação) - Universidade Paulista - UNIP, Limeira, 2016. Disponível em: <<http://www.devmedia.com.br/ciclos-de-vida-do-software-artigo-revista-engenharia-de-software-magazine-36/21099>>. Acesso em: 10 set. 2016.

PROGRAMAÇÃO Orientada a Objetos. 2014. p. 20. Gerador de Sistemas (Ciência da Computação) - UNIP - Universidade Paulista, Limeira, 2016.

DESENVOLVIMENTO Profissional Multiplataforma para Smartphone, Iphone, Android, Windows Mobile e Blackberry. 2012. p. 26. Gerador de Sistemas (Ciência da Computação) - UNIP - Universidade Paulista, Limeira, 2016.