



Fundação Educacional do Município de Assis
Instituto Municipal de Ensino Superior de Assis - IMESA

LUCIO CARLOS TOLEDO BALDO

**PESQUISA SOBRE O CMS JOOMLA!, SUA ARQUITETURA, E
COMPONENTES E CRIAÇÃO DE UM SITE MURAL UTILIZANDO O
*FRAMEWORK BOOTSTRAP***

ASSIS

2013

LUCIO CARLOS TOLEDO BALDO

Pesquisa sobre o CMS JOOMLA!, sua arquitetura, e componentes e criação de um site mural utilizando *Framework Bootstrap*.

Trabalho de Conclusão de Curso apresentado ao Instituto Educacional de Ensino Superior de Assis, como requisito de Curso de Graduação.

Orientador: Me. Douglas Sanches da Cunha

Área de Concentração: Informática.

Assis

2013

FICHA CATALOGRÁFICA

BALDO, Lucio Carlos Toledo

Pesquisa sobre o CMS JOOMLA!, sua arquitetura, e componentes e criação de um site mural utilizando *Framework Bootstrap*/ Lucio Carlos Toledo Baldo. Fundação Educacional do Município de Assis – FEMA – Assis, 2013.

Número de páginas:_____

Orientador: Me. Douglas Sanches da Cunha.

Trabalho de Conclusão de Curso – Instituto Municipal de Ensino Superior de Assis – IMESA.

1. CMS. 2. Joomla!. 3. *Bootstrap*. 4. Ferramentas. 5. Mapeando *Template*

CDD:001.61

Biblioteca da FEMA

**PESQUISA SOBRE O CMS JOOMLA!, SUA ARQUITETURA, E
COMPONENTES E CRIAÇÃO DE UM SITE MURAL UTILIZANDO O
*FRAMEWORK BOOTSTRAP***

LUCIO CARLOS TOLEDO BALDO

Trabalho de Conclusão de Curso apresentado ao Instituto Municipal de Ensino Superior de Assis, com requisito do Curso de Bacharelado em Ciência da Computação, analisado pela seguinte comissão examinadora.

Orientador: Me Douglas Sanches da Cunha

Analisador: Célio Desiro

Assis

2013

DEDICATÓRIA

Dedico esse trabalho a minha mãe Vera Maria Toledo, minha Tia Ana Maria Toledo, aos meus irmãos Marco Aurélio e Ana Elisa minha esposa Elisângela Casado Baldo e filhos Rafael e Gabriel.

RESUMO

A proposta deste trabalho foi a de pesquisar sobre o CMS Joomla!2.5 seus componentes e arquitetura, o *Framework Bootstrap* suas classes e características, para geração de um site mural com página responsiva utilizando software livre. Analisando as contribuições teóricas existentes que englobam a modelagem, criação de *templates* e geração de conteúdo de sistemas de informação, com o intuito de levantar as informações e estabelecer os passos necessários para a implementação ou adaptação de um módulo e *template* que atenda as necessidades deste trabalho. Levando em consideração que os *templates* do Joomla!2.5 não são responsivos, será integrada a ele as classes do *Framework Bootstrap* para que aquele recurso seja incorporado ao Joomla. A utilização de ferramentas como o editor Sublime Text2 e a interface web *Layoutit* serão utilizadas para o desenvolvimento deste trabalho.

Palavra-chave: Joomla!, CMS, Mural, *Bootstrap*

ABSTRACT

The purpose of this study was to research the CMS Joomla 2.5 components and architecture, the Framework Bootstrap classes and their characteristics, to generate a collection site with responsive page using free software. Analyzing existing theoretical contributions that encompass modeling, creation of templates and content generation of information systems, in order to collect information and establish the necessary steps for the implementation or adaptation of a module template that meets the needs of this work . Considering that the templates in Joomla! 2.5 are not responsive, it will be integrated into the classes of the Framework Bootstrap to that the feature is built into Joomla. The use of tools like Sublime editor Tex2 Layoutit and web interface will be used for the development of this work.

Keyword: Joomla, CMS, Wall, Bootstrap

LISTA DE ILUSTRAÇÕES

Figura 1 - Mapa Mental	14
Figura 2 - Tela de Administrador	18
Figura 3 – Tela de login do Drupal	20
Figura 4 – CMS no Google Trends Mundial - 14/06/2013	23
Figura 5 – Seleção de Idiomas	24
Figura 6 – Checagem do Sistema	25
Figura 7 – Licença	25
Figura 8 – Instalação do Banco de Dados MySQL	26
Figura 9 – Instalação e configuração do FTP	27
Figura 10 – Instalação e Configurações Finais	27
Figura 11 – Conclusão da Instalação	28
Figura 12 – Tela do Super Usuário	29
Figura 13 – Tela de Gerenciamento de Usuário	30
Figura 14 – Barra de Ferramentas	32
Figura 15 – Template Joomla!	35
Figura 16 – Website sem Padrão W3C	37
Figura 17 – Website com padrão W3C	37
Figura 18 – Doctype HTML	38
Figura 19 – Utilização de Offset	39
Figura 20 – Sobreposição de Colunas	39
Figura 21 – Meta Tag	40
Figura 22 – Mensagens de Alertas	42
Figura 23 – Site LayoutIt!	45
Figura 24 – Tela do LayoutIt! Inicial	46
Figura 25 – Tela de Download do Template	47
Figura 26 – Classe templateDetails.xml	49
Figura 27 – Estrutura do CMS Joomla!	50
Figura 28 – Parte Inicial do Arquivo index.php	51
Figura 29 – Classe Menu	53
Figura 30 – Carregando Arquivo javascript	53
Figura 31 – Função de Módulos HTML Sobrescritos	54
Figura 32 – Disposição dos HTML com 3 Colunas	55
Figura 33 – Arquivo template.css	55
Figura 34 – Arquivo de Tradução	56

LISTA DE ABREVIATURAS E SIGLAS

CMS – *Content Management System.*

SGC – Sistema Gerenciador de Conteúdo.

CSS – *Cascade Style Sheet.*

W3C – *World Wide Web Consortium.*

SEO – *Search Engine Optimization.*

ACL – *Access Control Level.*

XHTML – *Extensible Hypertext Markup Language*

HTML – *Hypertext Markup Language*

XML – *Extensible Markup Language*

RSS – *Really Simple Syndication*

API – *Application Programming Interface*

WYSIWYG – *What You See Is What You Get*

SUMÁRIO

1. INTRODUÇÃO.....	11
1.1 OBJETIVOS	11
1.2 OBJETIVOS ESPECÍFICOS	12
1.3 METODOLOGIA.....	12
1.4 JUSTIFICATIVAS	13
1.5 MOTIVAÇÃO	13
1.6 ESTRUTURA DO TRABALHO.....	14
2. CMS.....	15
2.1 PÁGINAS ESTÁTICAS.....	16
2.2 PÁGINAS WEB COM FOLHAS DE ESTILO EM CASCATA	16
2.3 PÁGINAS DINAMICAS.....	17
2.4 CMS MAMBO	18
2.5 CMS DRUPAL	19
2.6 CMS JOOMLA!.....	20
3. CMS JOOMLA!	21
3.1 INTRODUÇÃO	21
3.2 HISTÓRICO DAS VERSÕES DO JOOMLA!	22
3.3 CARACTERÍSTICA DO JOOMLA!.....	23
3.3.1 Instalando o Joomla!.....	24
3.4 ADMINISTRAÇÃO BÁSICA DO JOOMLA!	28
3.4.1 Gerenciamento de Usuário e Controle de Acesso.....	30
3.4.2 Barra de Menu.....	31
3.4.3 Barra de Ferramentas.....	31
3.5 GERENCIAMENTO DE CONTEÚDO	32
3.5.1 Artigos não Categorizados	33
3.5.2 Artigos Categorizados	33
3.5.3 Módulos.....	34
3.6 TEMPLATE.....	34
3.6.1 Processo de Design do Localhost	36

3.6.2	Padrões W3C.....	36
3.6.3	Desvantagem	37
4.	FRAMEWORK BOOTSTRAP.....	38
4.1	CONFIGURAÇÕES GLOBAIS	38
4.2	SISTEMA DE GRID PADRÃO	38
4.2.1	Layouts	39
4.2.2	Design Responsivo	40
	Tabela 1 – Tabela resolução de dispositivos	40
4.3	CSS BÁSICO.....	41
4.4	COMPONENTES E JAVASCRIPT.....	41
5.	FERRAMENTAS DE EDIÇÃO E CRIAÇÃO	43
5.1	INTERFACE LAYOUTIT!.....	43
5.2	EDITOR SUBLIME TEX2	44
5.3	DESENVOLVENDO UM <i>TEMPLATE</i>	44
5.3.1	Interface LayoutIt!.....	45
6.	MAPEANDO O <i>TEMPLATE</i> PARA O CMS JOOMLA!	48
6.1	ARQUIVO <i>TEMPLATEDETAILS.XML</i>	48
6.2	ESTRUTURA DO PROJETO	50
6.3	ARQUIVO INDEX.PHP.....	51
6.4	MAPEANDO MÓDULOS PARA O CMS JOOMLA!	52
6.4.1	Módulo Menu.....	52
6.4.2	Módulo Personalizar HTML.....	53
6.5	PASTAS DO PROJETO	55
	CONCLUSÃO	57
	REFERÊNCIAS BIBLIOGRÁFICAS	58

1. INTRODUÇÃO

Nos últimos anos tem se visto um grande aumento de vendas de *smartphones* e *tablets* (pequenos computadores pessoais) no Brasil. Só no ultimo trimestre de 2013 o aumento de vendas de *smartphone* chegou a 110% com relação ao mesmo período do ano anterior, o que pressiona o aumento do uso da Internet por usuários comuns buscando informações sobre produtos de uso pessoal ou coletivo como: centro de eventos, *shows* entre outros (olhardigital.uol.com.br).

Porém nem todas as páginas *web* possuem a adaptação para estes dispositivos, que é a característica de ser responsiva.

O Joomla! é um sistema de gerenciamento de conteúdo premiado (CMS), que permite construção de sites e tem duas áreas principais: o *frontend* e *backend*, que facilitam o gerenciamento e controle de conteúdo do *web site*. O Joomla! 2.5 que será abordado neste trabalho, não apresenta a característica responsiva de páginas *web*, a qual permite uma melhor visualização da página em dispositivos de tela pequena, sendo necessária a utilização das classes do *Framework Bootstrap*.

1.1 OBJETIVOS

O objetivo geral deste trabalho é pesquisar e compreender as características do CMS Joomla! 2.5, as classes do *Framework Bootstrap* e as ferramentas para desenvolvimento e edição de *templates* na linguagem PHP.

1.2 OBJETIVOS ESPECÍFICOS

Possibilitar a incorporação das classes do *Framework Bootstrap* na estrutura do Joomla! 2.5 sem prejudicar sua funcionalidade, com a finalidade de desenvolver um *template* que atenda as necessidades de um site mural, utilizando o CMS Joomla!2.5.

Definir os passos necessários para o desenvolvimento do site mural que atenda as necessidades de dispositivos móveis, que possuem telas pequenas e requerem que páginas *web* sejam responsivas para uma melhor visualização das mesmas, ao final desta pesquisa.

Auxiliar um usuário a encontrar a melhor opção que deseja para aquele dia, sem precisar passar em vários eventos, até encontrar o que procura e, também, disponibilizar as pessoas jurídicas um novo meio de propaganda para seu negócio.

1.3 METODOLOGIA

Para atingir os objetivos deste trabalho a execução deste foi dividida em duas fases:

1ª fase – pesquisar e analisar o CMS Joomla! 2.5 e o *Framework Bootstrap*: as características da estrutura do Joomla!, suas classes, componentes e *templates*, as classes do *Framework Bootstrap*, quais seriam necessárias e como utiliza-las.

2ª fase – desenvolver um *template* que atendesse a necessidade de um site mural, utilizando-se de ferramentas de criação e edição, como a interface Layoutit! e o Sublime Tex2.

1.4 JUSTIFICATIVAS

Devido a um constante crescimento, principalmente no setor mobile e de tablets que vem ganhando adeptos por todo o país, as propagandas digitais de eventos, show's para estes dispositivos podem ter grande aceitação tanto para usuários destes como para empreendedores. A importância do desenvolvimento desse trabalho é a busca de um novo recurso de propaganda, cuja principal característica é uma visualização e direcionamento rápido de um determinado produto para um site específico, podendo contribuir com a diminuição de propagandas poluentes, como panfletos, feito de papel.

Para uma interatividade, e apresentação adequada das informações nos diversos dispositivos e modelos de tamanhos de telas, tem-se a necessidade de uma página responsiva, que se adapte a eles, porém esta não é a característica do Joomla! 2.5. A utilização das classes do *Framework Bootstrap* incorporada ao CMS Joomla! 2.5 proporciona essa característica. Entretanto, para a utilização dos recursos de *backend* do Joomla!, será necessário o mapeamento total da página, seus componentes e módulos.

1.5 MOTIVAÇÃO

Em plena era digital é de fundamental importância um novo tipo de propaganda, tendo em vista também a necessidade de se contribuir para a diminuição de propagandas que utilizam papéis, ou outros tipos de materiais poluentes.

1.6 ESTRUTURA DO TRABALHO

No primeiro capítulo [1], foi abordado o objetivo, metodologia, justificativa e motivação a desenvolver este trabalho. No segundo capítulo [2] será abordado o conceito de CMS, características de websites dinâmicos (CMS), websites estáticos (XHTML), e os que utilizam folhas em estilo em cascata (CSS), ainda neste capítulo, farei um breve resumo, dentre os diversos tipos de CMS existentes no mercado apresentando as características do Mambo, o Drupal e o Joomla!. No terceiro capítulo [3], especificamente o CMS Joomla! 2.5 e sua arquitetura. No capítulo quarto [4], o *Framework Bootstrap*, suas configurações globais, sistemas de grid, CSS básico, componentes e Javascript. No quinto capítulo [5] serão apresentadas as ferramentas de edição e criação; e no sexto capítulo [6] o mapeamento do *template* e componentes para o Joomla!2.5.

O mapa mental representado na figura 1, a seguir, demonstra a estrutura deste trabalho.

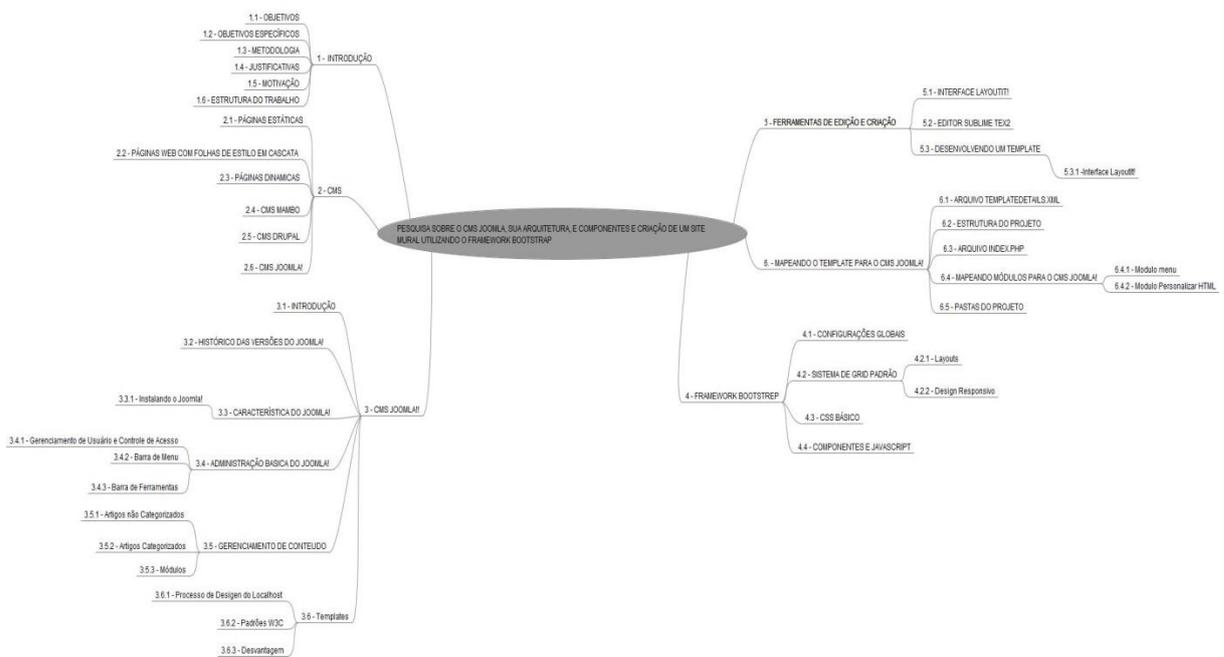


Figura 1 - Mapa Mental

2. CMS

Neste capítulo será abordado o surgimento do CMS, suas características, utilização, diferenças entre website estático (XHTML), que utilizam Folhas de Estilo em Cascata (CSS) e website dinâmico (CMS), e diferentes tipos de CMS.

O CMS (*Content Management System*), ou Sistema de Gerenciamento de Conteúdo (SGC) tem como objetivo o gerenciamento de conteúdo de páginas Web, que permite a criação, administração e armazenamento de forma dinâmica através da internet (NIEDSON, Robson, 2007). Surgiu no final da década de 90, para melhorar a gestão de conteúdo de websites, uma vez que a quantidade de conteúdo digital estava em amplo crescimento (CHAGAS, CARVALHO, SILVA, 2008). Uma característica importante dos CMS é que não precisam de mão de obra especializada para a sua manutenção, ficando a cargo de cada membro da equipe de uma organização a responsabilidade por seu próprio conteúdo. Desde então muitos CMS surgiram, muitos deles com o código aberto, dentre estes o Joomla que será estudado nesse trabalho (PEREIRA, BAX, 2002).

Para Pereira e Bax (2002), os CMS em geral possibilitam a criação, a edição, o gerenciamento, a recuperação de informações, isto é, possibilitam o gerenciamento de todas as etapas, desde a sua criação até a sua publicação. O CMS a ser utilizado deve ser definido sob o ponto de vista das atividades das pessoas e seus objetivos, já que estes possuem particularidades específicas.

A usabilidade do CMS está dividida em duas áreas: a *backend* que é a área administrativa de desenvolvimento e gerenciamento do CMS, para acessá-la é necessário estar autenticado e a *frontend* que é a parte de navegabilidade do website (parte “visível”) destinada principalmente ao usuário comum, este pode estar logado ou não depende de seu privilégio (CHAGAS, CARVALHO, SILVA, 2008).

A segurança reflete a capacidade de um sistema de impedir que dados de usuários e partes do próprio sistema sejam expostos e modificados. Nos CMS's estudados

foram verificados alguns critérios, como validação de e-mail, formas de autenticação (Kerberos, LDAP e NIS) e notificação de problemas.

De uma forma geral não existem CMS que atendam todas as necessidades de uma organização. Sendo assim, fica a critério da organização escolher dentre os CMS disponíveis (Gratuitos ou Pagos), o que melhor atende a organização (BROOKS, BULLET, 1987).

2.1 PÁGINAS ESTÁTICAS

Antes do surgimento dos CMS as páginas eram desenvolvidas de forma estáticas com um conjunto de instrução em XHTML, que apresentavam duas desvantagens principais (BARRIE M. NORTH, 2008):

- Tanto o conteúdo (texto que é apresentado na web) da página como sua apresentação (formato, tamanho e estilo de letra) estavam amarrados, se fosse necessário alterar a cor de um texto o desenvolvedor teria que percorrer todas as páginas no seu site.
- As páginas acabavam tendo tamanhos de arquivos grandes, o que as tornavam pesadas.

2.2 PÁGINAS WEB COM FOLHAS DE ESTILO EM CASCATA

Outra forma de construção de páginas web encontrada adota um conjunto de regras para exibição de conteúdo na tela. O de Folhas de Estilo em Cascata (CSS) – *Cascade Style Sheet*, é um deles, nesta forma de construção de páginas web, todos os arquivos de apresentação como estilo de letra, cores, tamanhos, etc., estão em arquivos diferentes das de conteúdo. Numa definição a grosso modo, é criado um documento CSS com as especificações de estilo do conteúdo. Suas vantagens são:

- Facilidade na edição do estilo da página – basta apenas alterar uma linha do arquivo de edição para que, por exemplo, todos os títulos mudem de cor.
- O conteúdo estar em arquivo separado do de estilo, portanto, ambos se tornam menores, facilitando o carregamento da página.
- O arquivo CSS por sua vez pode ficar armazenado em cache, não sendo necessário o download toda vez que a página é carregada. (BARRIE M. NORTH, 2008)

2.3 PÁGINAS DINAMICAS

Páginas Web Dinâmicas são desenvolvidas utilizando um CMS. Neste caso, o conteúdo é separado da página, ficando aquele armazenado em um banco de dados. O local onde estaria o conteúdo é chamado de “*placeholders*” (espaço reservado para adicionar conteúdo futuro). O CMS também utiliza a técnica de CSS, logo se percebe que todo o website está separado por “categorias”, facilitando assim a sua manutenção, onde cada setor de uma organização ficaria responsável por um “pedaço” do website (BARRIE M. NORTH, 2008).

“Um CMS separa totalmente o conteúdo das páginas do seu formato gráfico. Isto torna mais fácil manter o sistema amplamente coerente no design e fácil de alterar. Também torna a adição de conteúdo fácil para pessoas não técnicas.” (BARRIE M. NORTH, 2008,p5).

2.4 CMS MAMBO

Segundo o site MAMBO, o CMS Mambo surgiu em 2000 e atualmente está na versão 4.6.5, que traz uma série de correções de bugs, melhorias de segurança e de desempenho, esta atualização deve ser executada a-partir da versão 4.6.2. A última versão esta disponível para cópia sob a licença GNU/GPL, e é escrita na linguagem PHP e compatível com o banco de dados MYSQL.

É um CMS premiado de fácil operação, rico em recursos, que é tanto utilizado em websites simples quanto em aplicações corporativas.

As funcionalidades oferecidas pelo Mambo em sua instalação padrão são:

- *Banners*: possibilita gerenciar a exposição de *banners* (imagem publicitária) no *website*.
- *News Feeds*: possibilita a publicação de conteúdos no formato RSS.
- *Polls*: possibilita a criação de pesquisas de opinião

Além das funcionalidades padrão, o Mambo conta com mais de 170 extensões publicadas, é de instalação fácil, porem não é garantida a compatibilidade e funcionamento das extensões.

A seguir será apresentada a tela de administrador do CMS Mambo figura 2.

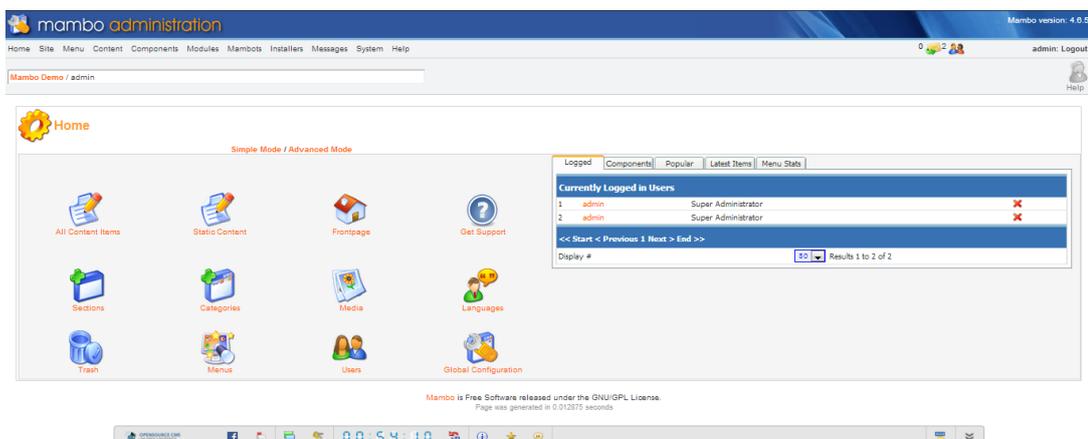


Figura 2 - Tela de Administrador

2.5 CMS DRUPAL

Segundo o site DRUPAL o CMS Drupal teve origem em 2000, quando Dries Buytaert estudante da Universidade de Antuérpia decidiu criar um site de noticia para que seus colegas postassem mensagens. O site inicialmente foi publicado na web com o nome de drop.org, o que fez com que ganhasse novos adeptos principalmente quando os membros começaram a falar sobre novas tecnologias web. Isso fez com que Dries em 2001 lançasse o software por trás do drop.org, o “Drupal”.

Hoje o Drupal é mantido por mais de 630.000 desenvolvedores e usuários, distribuído sob a licença GNU/GPL, escrito em linguagem PHP e compatível com banco de dados MYSQL, *PostgreSQL* e *SQLite*. Sua ultima versão estável é a 7 (DRUPAL).

Os módulos fornecidos pelo Drupal em sua instalação padrão que merecem destaque são:

- *Block*: gerencia o conteúdo dos menus do website.
- *Blog*: possibilita aos usuários registrados manterem um blog (website pessoal).
- *Book*: possibilita a publicação de documentos estruturados.
- Fórum: possibilita criar grupos de discussão.
- *Locale*: possibilita apresentar o website em diferentes idiomas.
- *Poll*: possibilita a criação de enquetes simples.
- *Search*: possibilita efetuar buscas por conteúdos específicos no website.

O *Drupal* conta com mais de 21.850 módulos (6509 para a versão 7), 1.725 Temas, 631 Distribuições e 25.832 Desenvolvedores.

A figura 3 a seguir mostra a tela de *login* do *Drupal*.

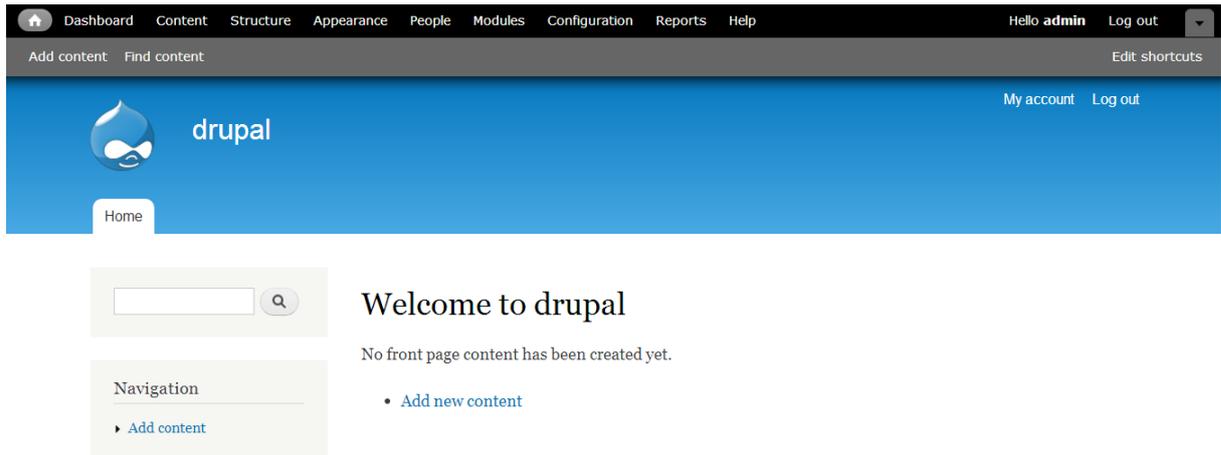


Figura 3 – Tela de login do Drupal

2.6 CMS JOOMLA!

Segundo Joomla.org/about, este é um CMS que permite a construção de sites e poderosas aplicações online de uma maneira muito fácil. Isto fez que o CMS ganhasse muitos adeptos, também, devido a sua grande extensibilidade.

O Joomla surgiu com a separação das equipes de desenvolvimento do projeto Mambo e a Miro - empresa detentora dos direitos sobre o Mambo. A parcela dos desenvolvedores descontentes com a transferência do projeto Mambo para a fundação *Mambo Foundation* criou em 2005 o Joomla 1.0, a partir do código fonte do Mambo 4.5.2.(Wikipedia.org).

O Joomla é um CMS distribuído sob a licença GNU/GPL, escrito na linguagem PHP , compatível com banco de dados MYSQL (MSSQL e PostgreSQL, versão 3.x), podendo ser executado em servidores Apache, Nginx e Microsoft IIS (Joomla.org).

Será explanado mais sobre o Joomla no próximo capítulo.

3. CMS JOOMLA!

Neste capítulo será apresentada a ferramenta CMS Joomla! 2.5.X., Introdução ao Joomla!, Histórico das versões do Joomla!, Características e instalação do Joomla!, Administração básica do Joomla!, Gerenciamento de Conteúdo, *Templates* e Módulos.

3.1 INTRODUÇÃO

O Joomla! possui uma comunidade ativa muito grande, inclusive de terceiros como o joomlaclub.com.br, que possui mais 85.900 membros, atuando no desenvolvimento de Extensões e *Templates*. Isto é, um dos fatores importantes que faz com que o Joomla! seja um projeto de sucesso (Joomla CLUBE).

O Joomla! é um sistema gratuito de código aberto para criação de web sites e está em constante movimento. Segundo Hangel Graf “é imprevisível, por vezes indescritível, parcialmente controverso, muitas vezes altamente sexy e as vezes, um pouco devagar e provinciano. No entanto, ou talvez por causa disso, tem sido extremamente bem sucedido e popular nos últimos seis anos, com milhões de usuário em todo o mundo.”

O projeto Joomla! é uma evolução do Mambo e surgiu da separação da fundação Mambo criada em 2005 e sua equipe de desenvolvimento na época. O Joomla! é um sistema de sucesso e é utilizado para a criação desde homepages simples até sites corporativos complexos. (Graf Hangel, 2012).

3.2 HISTÓRICO DAS VERSÕES DO JOOMLA!

Muitos dos usuários do Mambo migraram seus sites para o Joomla! e permanecem fiéis. A seguir a história das versões do Joomla!, ao longo do tempo (Graf Hangel, 2012).

- Joomla!1.0 – lançada em 2005 em paralelo com o Mambo 4.5.23, esteve ativa até junho de 2009
- Joomla!1.5 – é a evolução do Joomla! 1.0 e foi publicada oficialmente em 2008 com o intuito de resolver os problemas de segurança e bugs. O suporte para esta versão esteve ativo até 2012.
- Joomla!1.6 – é a evolução do Joomla!1.5 que foi lançada em janeiro de 2011 com suporte até agosto de 2011. Melhorias no upload de imagem, nesta versão o próprio Joomla! procurava por atualizações.
- Joomla!1.7 – lançado em julho de 2011 e com suporte curto de 6 meses, entre outras características, foi melhorado o suporte a site multilíngüe. Seu ciclo encerrou-se em fevereiro de 2012.
- Joomla!2.5 – lançado em janeiro de 2012 e o substituto oficial do Joomla!1.5, tem suporte previsto até a primavera de 2014.
- Joomla!3.1 – a mais nova versão do Joomla!, recomendada para a maioria das novas instalações, ele inclui os recursos mais recentes como a incorporação do Twitter *Bootstrap*.

O principal concorrente do Joomla! hoje, é o *Wordpress* que também é gratuito e em 2010 ultrapassou o Joomla! como mostrado na figura 4.



Figura 4 – CMS no Google Trends Mundial - 14/06/2013

3.3 CARACTERÍSTICA DO JOOMLA!

O arquivo do Joomla! é de aproximadamente de 7,6MB que precisa ser instalado em um servidor web, algumas das características presentes na instalação são: (BARRIE M. NORTH, 2008, p8).

- Simples revisão / criação do conteúdo usando um editor de texto do website ou de um site oculto de administração.
- Registro de usuário e habilidade para restringir a visitação de páginas baseado no nível de usuário.
- Controle de edição e publicação de conteúdo baseado em níveis de administrador
- Pesquisa.
- Formulário de contato simples.
- Site de estatística público.
- Detalhes privados de estatísticas de visitação do site.
- Padrões RSS
- Sistema simples de qualificação de conteúdo.
- Apresentação de notícias (*newsfeeds*) de outros sites.

Além destas características, o Joomla! possui uma grande quantidade de desenvolvedores espalhados pelo mundo, que contribuem de diversas formas dentre elas, as extensões, sendo a maioria gratuitas.

Algumas delas são:

- Fóruns
- Carrinhos de Compras
- Calendários
- Gerenciador de documentos
- Formulários
- Diretórios

3.3.1 Instalando o Joomla!

Após baixar o arquivo do Joomla em joomla.org, deve-se descompactá-lo na pasta root do servidor- no caso do *Xampp* (*c:\xampp\htdocs\joomla*), feito isto, ative seu servidor e digite no navegador *http://localhost/joomla*(Graf Hangel, 2012).

Aparecerá o instalador do Joomla! com uma seleção de idiomas, conforme figura 5.

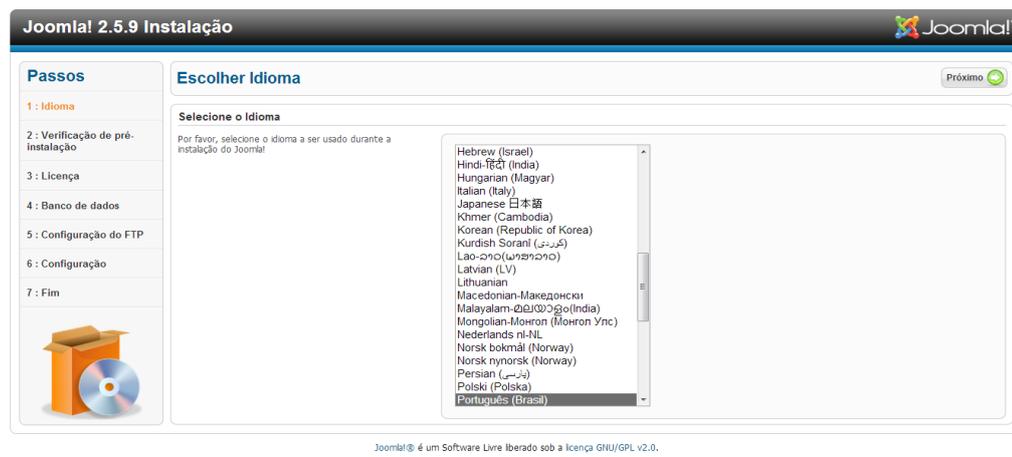


Figura 5 – Seleção de Idiomas

A próxima tela é a de checagem do sistema, nela é verificado se o servidor escolhido é compatível, figura 6.

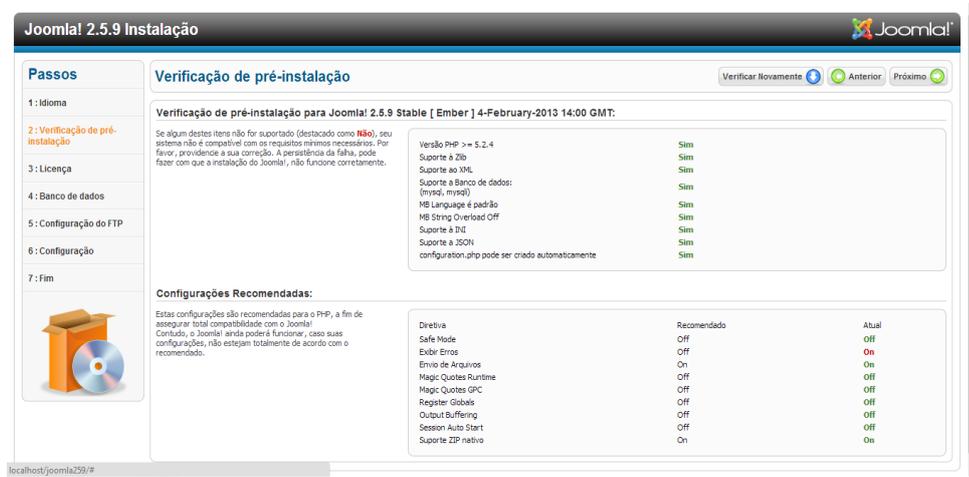


Figura 6 – Checagem do Sistema

Esta tela pode variar um pouco, dependendo do servidor que esta sendo utilizado. Neste caso, foi utilizando do servidor Apache (*Xampp*) instalado na máquina local. As informações apresentadas, na figura 6, representam um bom sinal.

A próxima tela, figura 7, é a da licença. O Joomla! é licenciado de acordo com a *GNU General Public Licence*.

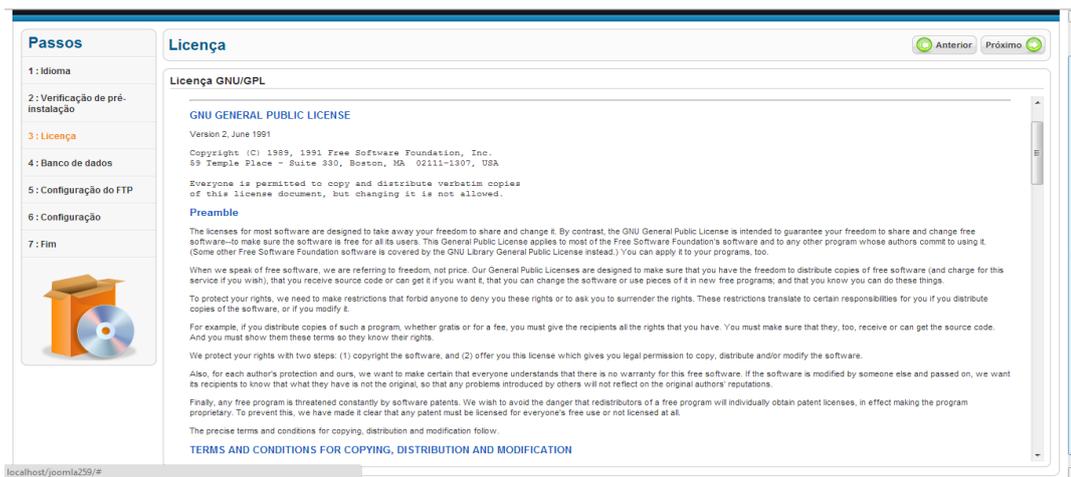


Figura 7 – Licença

O próximo passo é a instalação e configuração do banco de dados MySQL, figura 8.

Joomla! 2.5.9 Instalação

Passos

- 1: Idioma
- 2: Verificação de pré-instalação
- 3: Licença
- 4: Banco de dados
- 5: Configuração do FTP
- 6: Configuração
- 7: Fim

Configuração do Banco de Dados

Configurações da Conexão

Um site Joomla! armazena todos seus dados em um banco de dados. Esta tela fornece ao programa de instalação as informações necessárias para a criação deste banco de dados.

Se estiver instalando o Joomla! em um servidor remoto, você deve obter estas informações de sua hospedagem.

Alguns servidores precisam que você crie o banco de dados antes de instalar o Joomla!. Se você receber a mensagem "Não foi possível conectar ao banco de dados", verifique se seu nome de usuário e senha estão corretos. Se ainda receber esta mensagem, verifique com seu servidor se você precisa criar o banco de dados antes da instalação do Joomla!.

Configurações Básicas

Tipo de Banco de dados *
 Normalmente é "mysql"

Nome do servidor *
 Normalmente é localhost ou o nome de servidor fornecido por sua hospedagem

Nome de Usuário *
 Pode ser algo como "root", ou um nome de usuário fornecido por sua hospedagem

Senha
 Para segurança do site, é obrigatório utilizar uma senha para a conta MySQL. Normalmente, é a mesma senha usada para acessar seu banco de dados.

Nome do banco de dados *
 Alguns servidores permitem somente um BD por site. Se esse for o seu caso, use o prefixo de tabela para diferenciar cada site Joomla! instalado no mesmo banco de dados.

Prefixo das tabelas *
 Selecione um prefixo para as tabelas do banco de dados ou use o gerado aleatoriamente. Preferencialmente, não ou quatro caracteres, contendo apenas caracteres alfanuméricos, e DEVEM terminar com um underscore (sublinhado). Tenha certeza que o prefixo escolhido, não é usado por nenhuma outra tabela.

Tabelas Antigas *
 Backup Remover Qualquer cópia de segurança das tabelas de uma instalação anterior do Joomla! serão perdidas

Joomla! © é um Software Livre liberado sob a licença GNU/GPL v2.0.

Figura 8 – Instalação do Banco de Dados MySQL

Para uma instalação local devem ser inseridos os seguintes dados:

- Nome do computador: *localhost*
- Usuário: *root*
- Senha: [seu critério]
- Data
- *Database Name*: Nome do Banco
- *Table Prefix*: o sistema gera um prefixo automático, para o nome das tabelas, podendo ser alterado pelo usuário.

O fato do sistema inserir este prefixo, é evitar que um mesmo banco de dados tenha o mesmo nome de tabela para diferentes sites.

O próximo passo é a configuração do FTP do PHP, figura 9, para manipulação de arquivos. Esta configuração não é necessária em uma instalação local.

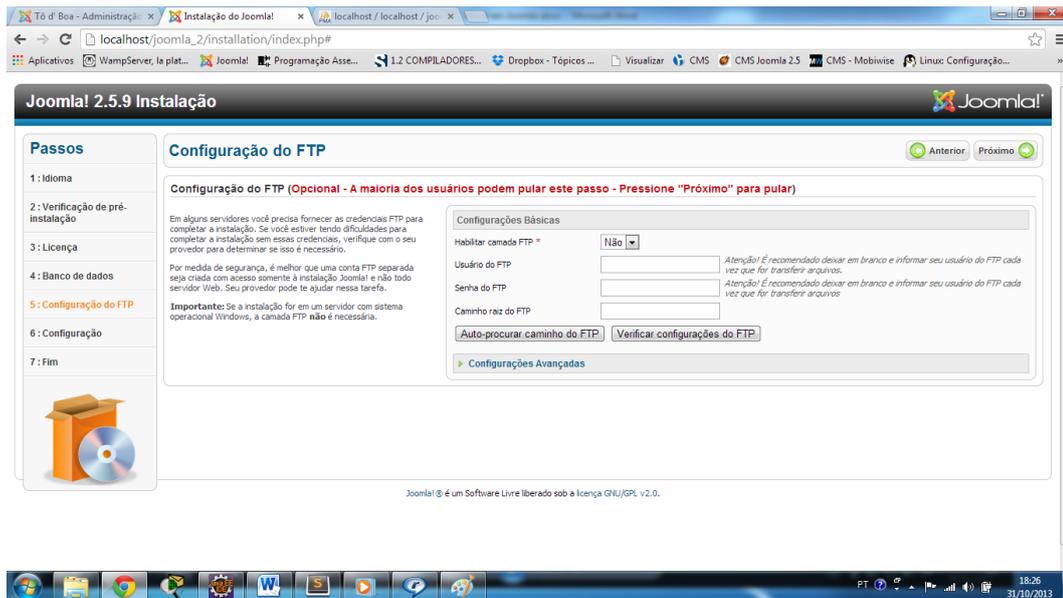


Figura 9 – Instalação e configuração do FTP

O último passo de instalação, mostrado na figura 10, é a de configuração de senha de *Super Users* e nome do site, aquele que aparecerá no topo da página e de configurações de e-mail, para usuários cadastrados. Neste ponto o usuário poderá optar pela instalação de conteúdo de exemplo ou não.

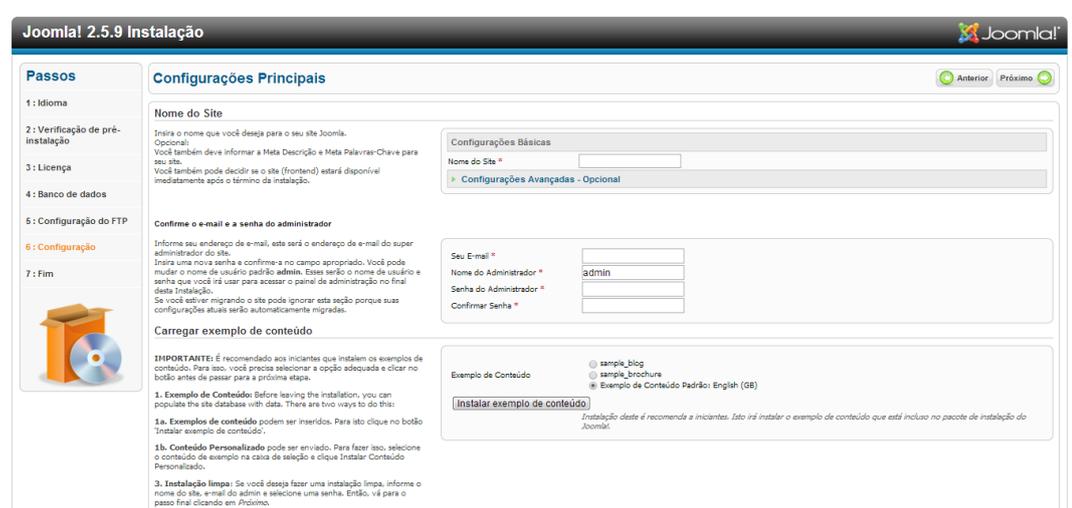


Figura 10 – Instalação e Configurações Finais

A última tela mostrada, na figura 11, é a de conclusão da instalação. Nela aparecerá uma mensagem em vermelho parabenizando-o pela instalação.

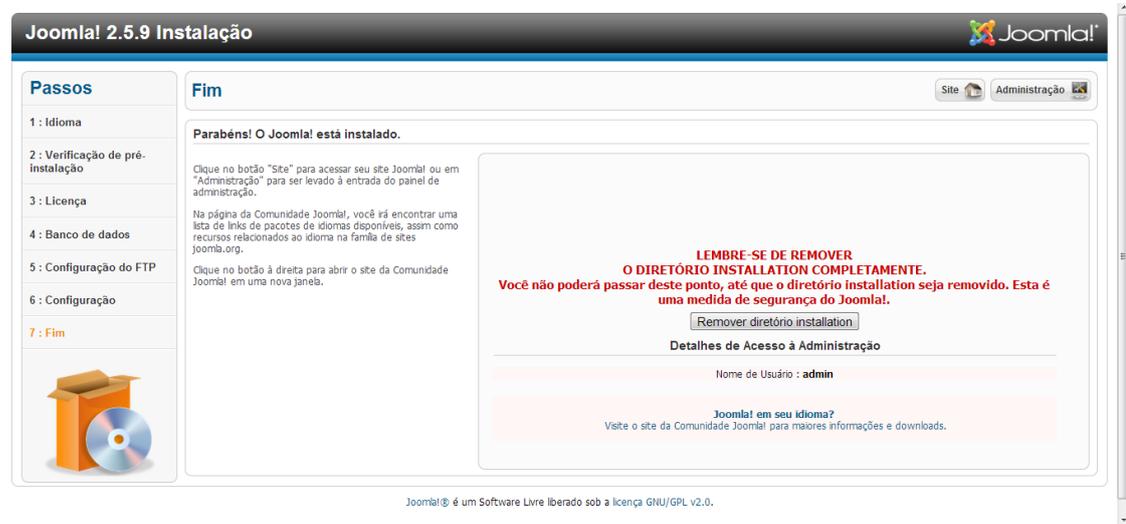


Figura 11 – Conclusão da Instalação

3.4 ADMINISTRAÇÃO BÁSICA DO JOOMLA!

Neste tópico, será abordada a administração básica do CMS Joomla!. Tais como: as diferenças entre *frontend* e *backend*, principais funções do menu administrador e quais os tipos de usuários que existem. A documentação completa pode ser encontrada no site www.joomla.org.

O Joomla!, possui duas áreas de trabalho: o *frontend* (público) e o *backend* (privado). O *frontend* é destinado a usuários comuns. É a vitrine de uma loja, por exemplo, onde o usuário pode observar os objetos, mas não trocá-los de lugar nem excluí-los. Dependendo do nível de privilégio de um usuário, este pode até administrar um site via *frontend*, o que não é muito aconselhável, sendo mais eficiente via *backend*. (BARRIE M. NORTH, 2008, p29).

“Quando falamos do frontend, queremos dizer as áreas do website como visitantes ou usuários registrados vêem. Um usuário registrado normalmente funciona apenas no frontend. É como em uma loja, onde as mercadorias são exibidos em vitrines e nas prateleiras. Aqui o usuário pode dar uma olhada.” (Rafael Santana em 3. Março 2012).

Para acesso ao *backend*, é necessário que o usuário tenha sido cadastrado previamente por um super administrador. Após entrar com *login* e senha, o usuário será redirecionado a tela de administração, conforme figura 12 abaixo, dependendo do nível de privilegio, haverá pequenas variações.

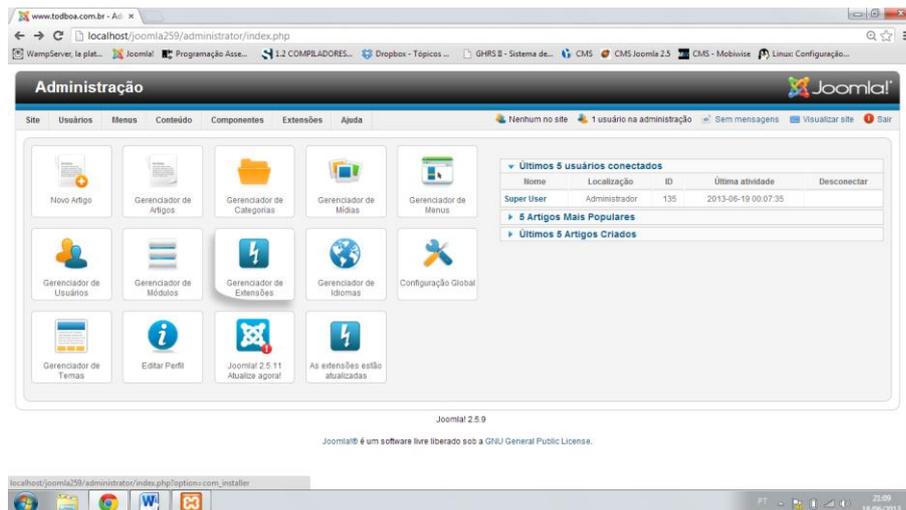


Figura 12 – Tela do Super Usuário

Na figura 12, encontra-se a *Barra de Menu* no topo da página e logo abaixo a *Barra de Ferramentas* e abaixo dela a área de trabalho (*workspace*).

3.4.1 Gerenciamento de Usuário e Controle de Acesso

O Joomla! possui nove níveis de usuário. Esses níveis controlam o privilégio do usuário acessar determinadas áreas do site, ACL(*Access Control Level*), ou em português, Controle de Acesso de Usuário, nome dado para designar tais funções.

Para gerenciar o controle de usuário, deve-se acessar no menu a opção – Usuário - onde existe uma lista de funções, dentre elas, a de gerenciador de usuário, ao acessa-la aparecerá uma tela, conforme a figura 13 abaixo.

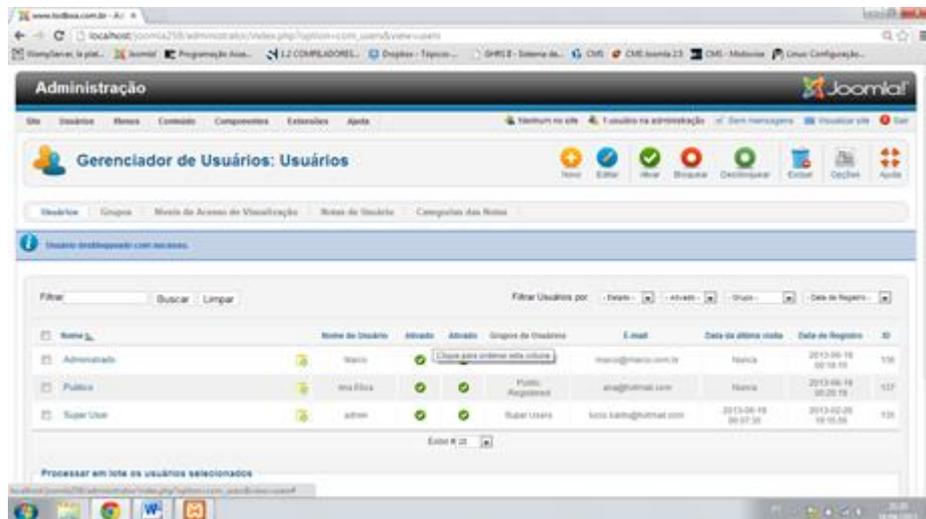


Figura 13 – Tela de Gerenciamento de Usuário

Usuários com permissões de *Public*, *Registered*, *Autor*, *Editor*, *Publisher*, possuem acesso apenas ao *frontend* do site. O *Public* possui privilégio apenas de visualização, o *Registered* pode ver conteúdos restritos e públicos, o *Autor* pode criar conteúdo e todos os privilégios do *Registered*, assim por diante. Os usuários com permissões de *Maneger*, *Administrator*, e *Super User*, possuem acesso tanto ao *frontend*, como ao *backend*. Seguindo a mesma linha de raciocínio, o *Super User* possui privilégio máximo, isto é, o que tem controle total (BARRIE M. NORTH, 2008, p35).

3.4.2 Barra de Menu

A Barra de Menu é a principal ferramenta de gerenciamento do CMS Joomla!. Nela o usuário encontra todas as funcionalidades e recursos disponíveis, específica para qual nível de privilégio acessado. Os menus disponíveis são: (BARRIE M. NORTH, 2008, p31).

- Site
- Usuário
- Menus
- Conteúdo
- Extensões
- Ajuda

No lado direito deste, estão algumas informações adicionais, como:

- Link de visitação no site
- Link de preview do site
- Link de quantidade de usuários de administração logados
- Link de mensagens
- Sair

3.4.3 Barra de Ferramentas

Esta ferramenta surge assim que o administrador acessa alguns dos links que aparecem no *Workspace*. Para o link de Gerenciamento de Categorias, aparecem as opções mostradas na figura. 14

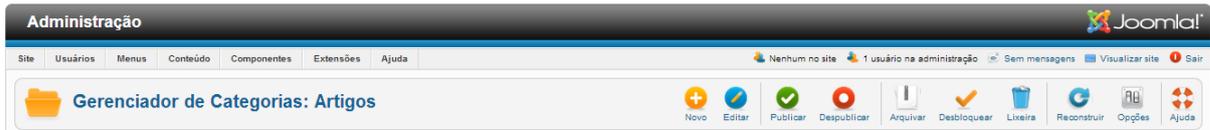


Figura 14 – Barra de Ferramentas

Na tela de Gerenciamento de Categorias observam-se as seguintes funcionalidades para um super administrador: (BARRIE M. NORTH, 2008, p32).

- Novo – abre uma tela para a publicação de uma nova categoria ou sub-categoria.
- Editar – deve-se selecionar uma das categorias listadas abaixo PA edição da mesma.
- Publicar – publica uma categoria selecionada.
- Despublicar – operação inversa a de publicar.
- Arquivar – arquiva categoria selecionada.
- Desbloquear – desbloqueia categoria bloqueada
- Lixeira – apaga categoria selecionada.
- Reconstruir – recupera dados corrompidos.
- Opções – abre uma tela de configurações para gerenciamento e configurações de categorias.
- Ajuda

3.5 GERENCIAMENTO DE CONTEÚDO

O Joomla! cria uma página, apenas quando se clica em um link, desta forma fica fácil trocar o conteúdo de uma determinada página, alterando apenas o link desta. (BARRIE M. NORTH, 2008, p51).

O Joomla! possui duas maneiras básicas de criação de conteúdo:

- Componentes
 - Artigos(categorizados ou não).
 - Outros componentes como links web
- Módulos

Os Componentes geralmente são encontrados no centro do corpo principal de uma página web, e os Módulos em sua lateral.

3.5.1 Artigos não Categorizados

É a maneira mais fácil e simples de organizar um site em Joomla!. É aconselhável apenas para sites pequenos, pois não apresentam nenhuma estrutura hierárquica.

Analogamente, pode-se compará-la com uma gaveta de um armário onde se guardam arquivos. Caso não haja uma quantidade muito grande de arquivos, esta é uma maneira bem prática de guardá-los. Se o volume começa a aumentar e a variedade de tipos de artigos, também, fica complicado o gerenciamento destes. (BARRIE M. NORTH, 2008, p53).

3.5.2 Artigos Categorizados

Neste tipo de organização os artigos são separados por tipos e subtipos, É mais indicado para sites de tamanho médio e grandes. O Joomla! oferece uma estrutura hierárquica que facilita a organização dos artigos. (BARRIE M. NORTH, 2008, p54).

Neste modo os artigos são separados em Categorias e Sub Categorias, por exemplo:

- Esporte (categoria)
 - Basquete (sub Categoria)

- Vídeos
- Entrevistas
- Vôlei
 - Vídeos
 - Entrevistas
- Futebol
 - Vídeos
 - Entrevistas

Percebe-se que fica mais fácil o gerenciamento dos arquivos e administração do site desta forma, quando se trata de um site de tamanho considerável.

3.5.3 Módulos

Módulos são elementos de conteúdo, que podem ser posicionados ao lado de artigos. Os módulos são elementos de design, e estão presente na maior parte dos *templates*. Por exemplo: os módulos de pesquisa e menu têm posições pré-definidas num *template*.

Isso não significa que não se pode inserir outros tipos de módulos em um *template*. Justamente o contrario, pode-se criar módulos com funções inteligentes e inserí-los em locais pré-determinados nos *templates*.(Graf Hangel,2012)

3.6 TEMPLATE

O *Template* é a cara do site, é ele quem vai receber as informações em forma de artigo, módulos inteligentes, menus. A maioria é construída em HTML e CSS. O Joomla! oferece vários temas de *templates*, à escolha. Os *templates* permitem configurações como: mudar logotipo, alterar cor de fundo, etc. Cada *template* dispõe

de posições (áreas) diferentes onde os módulos podem ser posicionados (Graf Hangel,2012). A figura 15 nos mostra um *template* com informações em formato de artigo.

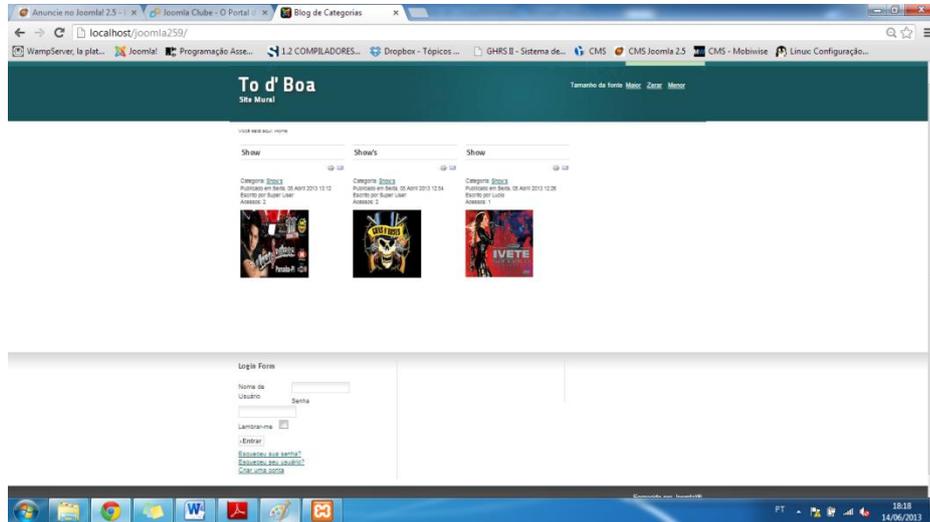


Figura 15 – Template Joomla!

Observa-se na figura 15, acima o logotipo; mais abaixo os artigos e no final da página, um módulo de *login*.

Em outras palavras, o *template* ou modelo é o alicerce básico para visualização de seu site. Para que se possa ter uma visualização de um website completo, tem-se que inserir a este modelo, o conteúdo que fica alojado no banco de dados do Joomla! (BARRIE M. NORTH, 2008, p189).

“O template é uma das mais importantes peças do site. Este é que providencia a aparência, o design. Este motiva novos visitantes a ficar no seu site e explorá-lo. Visitantes regulares e usuários apreciam voltar em um site com um bonito e útil design. Pense em outros produtos, por exemplo. Um carro precisa de um bom motor e pneus, mas a mais importante razão para comprar um é normalmente o design. Ainda que o design não seja a principal razão, essa é normalmente a razão para se possuir algo e pode fazer com que um comprador tenha mais razões palpáveis para isso. Se o design é bem feito, as pessoas esperam que o resto também seja” (Graf Hangel,2012).

3.6.1 Processo de Design do Localhost

A página web em um website Joomla! é gerada dinamicamente pelo conteúdo que está alocado em seu banco de dados. Esta é criada através de vários comandos PHP, o que representa alguma dificuldade no processo inicial de desenvolvimento.

Atualmente, é comum a utilização de um editor HTML WYSIWYG, como o Dreamweaver, o que possibilita ao Designer não modificar o HTML. Porém, pelo fato da página de um CMS Joomla! ser gerada dinamicamente, isso não é possível, tendo o Designer que codificar “manualmente” e visualizar o output, através do PHP em uma página (BARRIE M. NORTH, 2008, p191).

Para facilitar esse processo de visualização a maioria dos *designers* utiliza um *localhost*.

Por causa dessa limitação, o Designer deve seguir alguns passos;

1. Fazer a edição com um editor HTML, salvar as modificações.
2. Ter um *localhost* operando em background.
3. Visualizar as edições em um navegador web.
4. Retornar ao primeiro passo.

3.6.2 Padrões W3C

Este padrão especifica os mecanismos e regras necessários para que a criação de uma *webpage* que atenda as necessidades atuais de usabilidade, acessibilidade, e mecanismo de otimização de busca (SEO).

Padrões com HTML, XHTML, e o CSS são muito comuns, porém, em muitos casos são usados de forma errada, devido ao não conhecimento do padrão W3C. Caso, esse padrão não seja respeitado, poderá ocorrer barreiras tecnológicas para

diversas pessoas, desestimulando o acesso a sua página. Uma página web sem o padrão é mostrada na figura 16 e uma utilizando o padrão na figura 17.

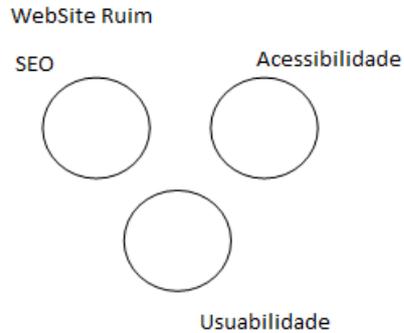


Figura 16 – Website sem Padrão W3C

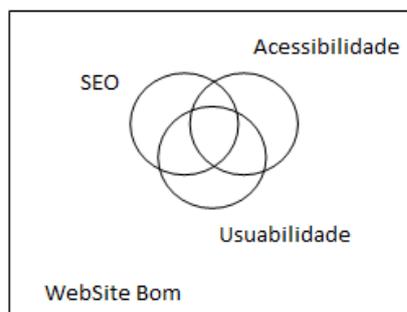


Figura 17 – Website com padrão W3C

3.6.3 Desvantagem

A principal desvantagem do Joomla! é a falta de flexibilidade, com relação a disposição de elementos visuais, pois o usuário fica preso a uma estrutura pré-definida pelo *template* escolhido.

4. FRAMEWORK BOOTSTRAP

Este capítulo tem como objetivo apresentar o *Framework Bootstrap*, configurações globais, sistema de *grid*, *layout* e *design* responsivo. Vale ressaltar que toda documentação e especificações estão disponíveis no site oficial <http://getBootstrap.com/> ou em <http://globocom.github.io/Bootstrap/scaffolding.html>, que possui todo o conteúdo traduzido.

4.1 CONFIGURAÇÕES GLOBAIS

Segundo o site oficial do *Bootstrap*, este possui estilo básico de tela, tipografia e estilo de link, faz uso de CSS que utiliza *doctype* de HTML5. Isto requer que se inicie o projeto como mostrado a seguir na figura 17:

```
<!DOCTYPE html>
<html lang="en">
...
</html>
```

Figura 18 – Doctype HTML

4.2 SISTEMA DE GRID PADRÃO

O sistema padrão do *Bootstrap*, segundo <http://getBootstrap.com/>, utiliza uma *grid* de 12 colunas com um *container* de 940 *pixel* (px) para funcionalidade de responsivo desabilitada. 724px e 1170px para funcionalidade habilitada e 767px as colunas ficam estagnadas verticalmente. As colunas podem ser deslocadas horizontalmente

para direita utilizando a função de *offset*. Por exemplo, para deslocar um *span* em 4 colunas faz-se conforme a figura 18:

```
<div class="row">
<div class="span4">...</div>
<div class="span4 offset4">...</div>
</div>
```

Figura 19 – Utilização de Offset

Observe que no exemplo acima, tem-se um *layout* simples de duas colunas. O número de *span*, mais o deslocamento de *offset* devem somar 12 que é o número de colunas da grid.

Outro recurso é o de sobreposição de colunas, isto é sobrepor uma coluna sobre a outra, para isto deve-se adicionar um novo *row* ao existente e escolher quantas colunas *span* desejar desde que não ultrapasse o número da coluna existente. Esses novos *span* devem conter o mesmo número de colunas que a coluna pai, como exemplificado no código abaixo na figura 19.

```
<div class="row-fluid">
  <div class="span12">
    Fluid 12
    <div class="row-fluid">
      <div class="span6">
        Fluid 6
        <div class="row-fluid">
          <div class="span6">Fluid 6</div>
          <div class="span6">Fluid 6</div>
        </div>
      </div>
      <div class="span6">Fluid 6</div>
    </div>
  </div>
</div>
```

Figura 20 – Sobreposição de Colunas

4.2.1 Layouts

Segundo o site oficial, *Bootstrap* possui dois tipos de layout que são os fixos, que possuem uma largura fixa (opcionalmente responsiva) e as fluidas bastando apenas indicá-las desta forma `<div class="container-fluid"><div class="container">`.

4.2.2 Design Responsivo

Como já mencionado, o *Bootstrap* possui a funcionalidade responsiva, porém esta funcionalidade não é ativa por padrão, pois nem tudo precisa ser responsivo.

Para habilitar esta funcionalidade deve-se incluir em nosso projeto CSS a meta *tag* e folhas de estilos adicionais dentro do `<head>` do documento. Caso esteja com o *Bootstrap* customizado deve-se incluir esta meta *tag*.

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<link href="assets/css/Bootstrap-responsive.css" rel="stylesheet">
```

Figura 21 – Meta Tag

De acordo com <http://getBootstrap.com/>, o *Bootstrap* suporta uma utilidade de mídia query em um único arquivo, o que facilita o desenvolvimento de projetos para diferentes dispositivos, a tabela 1 abaixo, mostra o que está incluso.

A mídia query contém uma ou mais expressões envolvendo características de uma mídia com definições para diversos dispositivos.

Nome	Largura do layout	largura da coluna	Largura do espaçamento
Resolução de tela grande	1200px e superior	70px	30px
Padrão	980px e superior	60px	20px
Tablets no formato na orientação horizontal	768px e acima	42px	20px
Fones até tablets	767px e abaixo	Colunas fluidas, larguras não fixas	
Fones	480px e abaixo	Colunas fluidas, larguras não fixas	

Tabela 1 – Tabela resolução de dispositivos

Utilitários responsivos não devem ser usados com tabelas, isso não é suportado.

4.3 CSS BÁSICO

Encontram-se aqui elementos HTML fundamentais, estilizados e aprimorados com classes extensivas.

Na parte tipográfica encontra-se toda a documentação com imagens ilustrativas e códigos prontos de exemplos usuais para sua aplicação como: tamanho de cabeçalho, formatação de parágrafo, estilo de fonte, como criar listas não ordenadas, ordenadas, não estilizadas, *inline*, listas com descrição, construção de tabelas, formulários, diversos tipos de botões (formato, tamanho e cor), imagens e ícones *glyphs* disponíveis em um padrão cinza escuro e branco, segundo o site oficial.

4.4 COMPONENTES E JAVASCRIPT

Em componentes há dezenas de componentes reusáveis, segundo o site *Bootstrap* construídos para prover navegação, alertas e outros mais. Dentre estes:

- Menus *dropdown* – Menus contextuais e alternáveis para exibir uma lista de links, opções de alinhamento e *sub menus* em um *dropdown*.
- Grupo de botões – únicos, múltiplos, verticais e também *checkbox* e rádios
- Botões de menu *dropdown* – diversos tipos, tamanhos e orientação, para alguns casos é necessário a utilização do *Bootstrap dropdown plugin (JavaScript)* para funcionar.

- Navegação – Todos os componentes de navegação, abas, pílulas e listas compartilham a mesma marcação base e estilo através da classe `.nav`. Exemplos: tabelas (`.nav-tabs`), pílulas (`.nav-pills`).
- Caso deseje desativar algum item de navegação basta adicionar ao código `.disabled`.
- Alertas – Inserir mensagens de alerta, como: mensagens de erro, de advertência e de sucesso. Para uma mensagem de alerta básico, basta inserir o código mostrado na figura 21:

```
<div class="alert">  
  <button type="button" class="close" data-dismiss="alert">x</button>  
  <strong>Advertência!</strong> Melhor checar campos novamente.  
</div>
```

Figura 22 – Mensagens de Alertas

O *Bootstrap* possui 13 *plug-ins jQuery* personalizados, estes podem ser incluídos individualmente ou todos de uma vez. Ambos, *Bootstrap.js* e *Bootstrap.min.js*, contêm todos os *plug-ins* em um único arquivo.

A utilização dos *plug-ins* do *Bootstrap* pode ser feita através da API sem a necessidade de escrever uma única linha de *JavaScript*. Da mesma forma pode-se desabilitar itens de navegação, desligando em algumas situações esta funcionalidade. O *JavaScript* é responsável por trazer “vida” aos componentes do *Bootstrap*.

5. FERRAMENTAS DE EDIÇÃO E CRIAÇÃO

Durante o processo de pesquisa, foram encontrados diferentes tipos de ferramentas para construção e edição de páginas *frontend/backend*, como por exemplo: o *Pingendo* que é uma aplicação visual *desktop* para criação de protótipos de página web, com base no *Bootstrap*, não abordado neste trabalho; a Interface *LayoutIt!* uma ferramenta *web*, de desenvolvimento de protótipos para páginas web e o editor de código Sublime Tex2, de edição de códigos *html, php*, entre outros.

Ainda neste capítulo será descrito o desenvolvimento de um *template* utilizando o *LayoutIt!* e o editor Sublime Tex2.

5.1 INTERFACE LAYOUTIT!

Interface para construção de código *frontend* utilizando o *Bootstrap*.

Esta interface, segundo site oficial *LAYOUTIT!*, não é um construtor de site, mas é um ponta pé inicial, para construção de código *frontend* sem precisar escrever uma única linha de código.

Todos seus projetos podem ser CSS *Responsive* e *Fluid*, nesta ferramenta que contém todos os componentes e elementos do *Bootstrap*.

Sua principal característica é a de arrastar e soltar os componentes do *Bootstrap*, para o seu próprio código no local que desejar, respeitando o padrão do *Bootstrap*.

5.2 EDITOR SUBLIME TEX2

É um editor de texto proprietário, porém, com prazo indeterminado para registro. Sofisticado e rápido com tela limpa e diversos recursos como atalhos: *plugin* e suporte para expressões regulares. Toda a documentação e arquivo para download pode ser encontrada no site oficial <http://www.sublimetext.com/2>. O usuário pode fazer configurações personalizadas e guardá-las. O Sublime guarda essas configurações em um arquivo separado, sendo possível colocá-lo no *Dropbox* e fazer um link simbólico para o Sublime, ficando este com as configurações iguais, em qualquer lugar que necessite.

O *snippets* é um recurso do Sublime, onde o usuário começa a digitar uma palavra e o editor lança o restante do código para que o usuário possa terminá-lo. Esses *snippets* podem ser apenas um *head*, ou uma configuração *xml* completa. O mais interessante é que o usuário pode criar seu próprio *snippet*. No menu *tools*, *new snippets*, o editor irá preparar a tela para que o usuário comece a escrever o código. Depois de pronto o usuário pode usá-lo a qualquer momento.

Por ser um editor, o Sublime só apresenta o auto complete em palavras e classe presentes, apenas na página programada, o que pode desanimar programadores que utilizam o eclipse, mas compensa com outros recursos.

5.3 DESENVOLVENDO UM *TEMPLATE*

Será descrito aqui os passos para o desenvolvimento de um *template responsiv e fluid*, utilizando a Interface *LayoutIt! Bootstrap* e o editor Sublime Tex2 para fazer o mapeamento para o CMS Joomla! 2.5.

5.3.1 Interface LayoutIt!

A interface *LayoutIt!* é gratuita e online, não sendo necessário fazer qualquer instalação, basta apenas um navegador web e conexão de internet.

No browser *digitar* <http://www.layoutit.com/>, e inicializar o *LayoutIt!* como mostrado na figura 22 a seguir.



Figura 23 – Site LayoutIt!

Nesta tela são apresentados alguns recursos e um vídeo demo na opção *Watch the live demo*.

Em *Start Now* será iniciado o aplicativo e a tela da interface será apresentada, já com uma configuração inicial de layout, como mostrado na figura 23. Para limpar basta clicar em *remove* ou em *clear*.

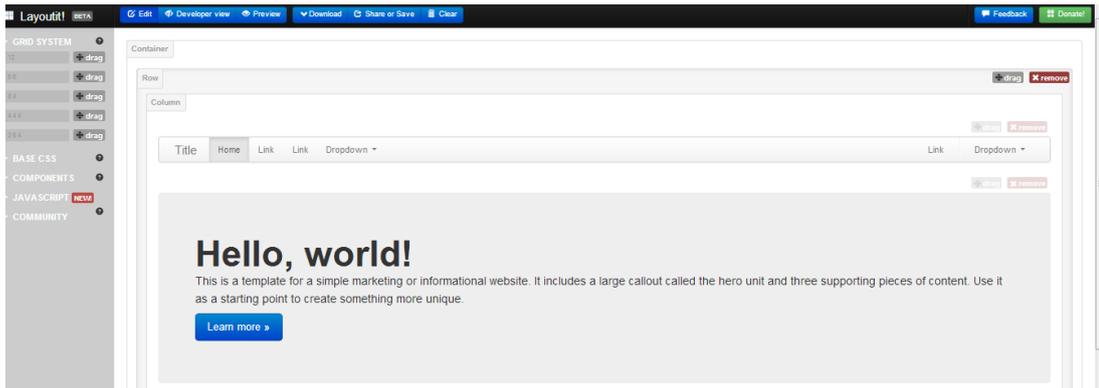


Figura 24 – Tela do LayoutIt! Inicial

Do lado esquerdo é apresentada uma barra de componentes contendo os seguintes itens do *Bootstrap*.

Grid system – contém a formatação do *Row* e *column* do *template*, lembrando que o *Bootstrap* possui uma grid de 12 colunas.

Nesta opção o usuário pode alterar o número de colunas mostrada como exemplo, desde que a soma seja 12.

Base CSS – contém diversas bases de dados do CSS como título da página, textos, imagens entre outros.

Components - contém os componentes do *Bootstrap* como grade de botões, thumbnails (versões reduzidas de imagem) e outros.

JavaScript – conteúdo como *carousel*, entre outros.

Community - em desenvolvimento até a data de entrega deste material.

Para utilizar estes componentes basta escolher um e arrastar para o container do *LayoutIt!*.

Após ter finalizado, ou a qualquer momento, o usuário pode através da opção de visualização da interface, visualizar o *frontend*. Para retornar ao modo de edição basta selecionar a opção *edit*.

Para baixar o *template frontend* basta clicar em download, uma nova janela aparecerá solicitando um cadastro. Após este cadastro uma nova janela será apresentada mostrando todo o código presente do *template*. Nesta hora o usuário poderá optar por um *template fluid* ou não. O usuário poderá escolher se prefere baixar o arquivo compactado ou em formato *html*, como mostrado na figura 24.

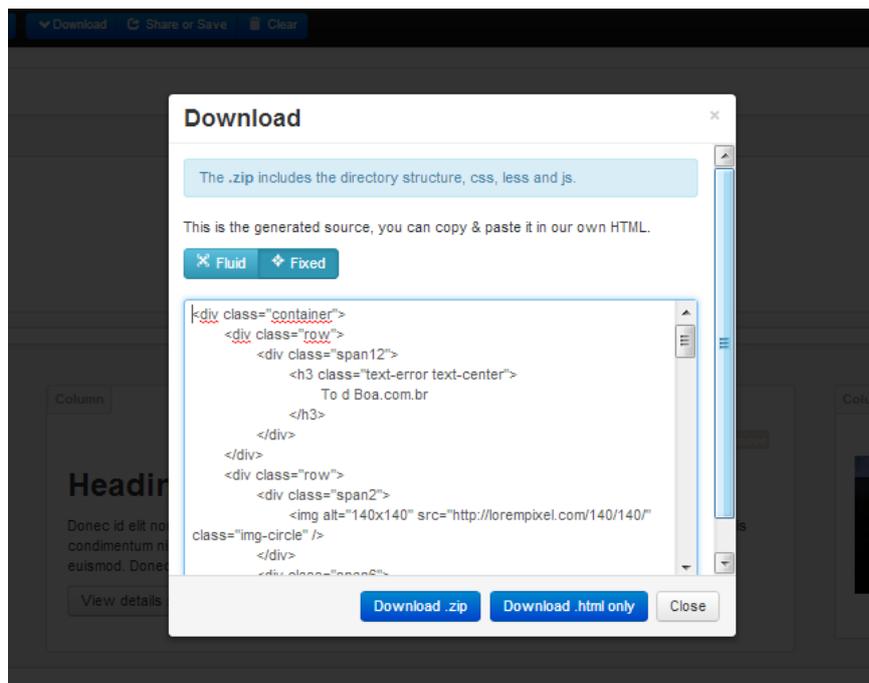


Figura 25 – Tela de Download do Template.

Vale lembrar que o formato Zip está no formato CSS e não em CMS. Para este trabalho foi utilizado o arquivo *html*. Este arquivo deverá ser renomeado para *index.html*.

Percebe-se que a construção do *template* é bem simples, não possui nada de extraordinário. O próximo passo é o mapeamento deste *template* para o CMS Joomla! 2.5.

6. MAPEANDO O *TEMPLATE* PARA O CMS JOOMLA!

Utilizando o Sublime Text2 mapeia-se o *template* e seus componentes para o CMS Joomla. Para isto será necessário a utilização de classes como *JFactory*, *JRequest* e *jConfig*, entre outras, além de métodos como *getApplication()*, *GetVar()*, *getMessageQueue()*, entre outros.

A classe *JFactory::getApplication()* é responsável por mapear o núcleo do *Joomla*, é através dela, que se recuperam informações da classe *jConfig*, onde, por exemplo, encontra-se o nome do site. Classe *JRequest* é usada para solicitar dados de filtragem. A classe *JRequest*, com o método *GetVar()*, é utilizada para recuperar dados de uma variável.

6.1 ARQUIVO *TEMPLATEDETAILS.XML*

Arquivo responsável pelo mapeamento de instalação / alteração / desinstalação do *template* no CMS Joomla. Nele estão presentes quais arquivos e pastas que serão criados durante a instalação, como mostra a figura 25.

Este arquivo também é responsável por carregar arquivos de tradução do sistema, como o *pt-BR/pt-BR.tpl_Bootstrap.sys.ini*, que executado pelo sistema e não pelo carregamento do componente.

Todos os arquivos e pastas, presente neste arquivo deve necessariamente estar no arquivo de *template* criado, caso contrario o este não será instalado, o contrario também é válido.

```

    <?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE install PUBLIC "-//Joomla! 1.6//DTD template 1.0//EN"
"http://www.joomla.org/xml/dtd/1.6/template-install.dtd">
<extension version="2.5" type="template" client="site">
  <name>Bootstrap</name>
  <creationDate>31/07/2013</creationDate>
  <author>Lucio Carlos</author>
  <authorEmail>lucio.baldo@hotmail.com</authorEmail>
  <authorUrl</authorUrl>
  <copyright>Copyright (C) . All rights reserved.</copyright>
  <license>GNU General Public License version 2 or later; see LICENSE.txt</license>
  <version>2.5.0</version>
  <description>TPL_TODBOA_XML_DESCRIPTION</description>
  <files>
<folder>css</folder>
  <folder>html</folder>
  <folder>images</folder>
  <folder>img</folder>
  <folder>js</folder>
  <folder>language</folder>s
  <filename>component.php</filename>
  <filename>error.php</filename>
  <filename>favicon.ico</filename>
  <filename>index.html</filename>
  <filename>index.php</filename>
  <filename>offline.php</filename>
  <filename>template_preview.png</filename>
  <filename>template_thumbnail.png</filename>
  <filename>templateDetails.xml</filename>
</files>
  <positions>
    <position>Bootstrap-mainmenu</position>
    <position>Bootstrap-search</position>
    <position>Bootstrap-sidebar</position>
    <position>Bootstrap-footer</position>
    <position>Bootstrap-footermenu</position>
    <position>Bootstrap-blocks</position>
    <position>Bootstrap-welcome</position>
  </positions>
  <languages folder="Language">
    <language tag="pt-BR">pt-BR/pt-BR.tpl_Bootstrap.ini</language>
    <language tag="pt-BR">pt-BR/pt-BR.tpl_Bootstrap.sys.ini</language>
  </languages></extension>

```

Figura 26 – Classe templateDetails.xml

Observa-se neste código a presença das pastas, dos arquivos e posições de módulos no *template* (*position*). Estas posições podem ser alteradas pelo desenvolvedor a qualquer momento.

A pasta e arquivos de *Language* são responsáveis pela tradução dos componentes e do *template*. Os arquivos *pt-BR.tpl_Bootstrap.ini* e *pt-BR.tpl_Bootstrap.sys.ini* devem estar presentes no arquivo de *manifeste* (*templateDetails.xml*) de instalação, caso contrário, estes não serão instalados. O arquivo *pt-BR.tpl_Bootstrap.sys.ini* é o arquivo do sistema, é compacto e é carregado automaticamente pelo sistema para

tradução de menus, por exemplo. Já o *pt-BR.tpl_Bootstrap.ini* é o arquivo completo, que é carregado com a execução do componente.

6.2 ESTRUTURA DO PROJETO

A estrutura de um *template* em CMS Joomla deve conter as pastas e arquivos como mostrado na figura 26.

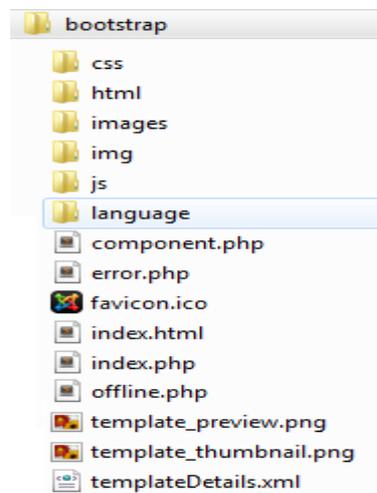


Figura 27 – Estrutura do CMS Joomla!

Como dito anteriormente o arquivo *templateDetails.xml* é responsável pela instalação do *template* no Joomla. O arquivo *index.html* é um arquivo praticamente em branco, possui apenas a declaração `<!DOCTYPE html><title></title>`. Este arquivo é responsável por bloquear acessos diretos não autorizados e deve estar em todas as pastas do projeto.

6.3 ARQUIVO INDEX.PHP

Este arquivo é o *template* propriamente dito, deve possuir a declaração `<!DOCTYPE htm>` logo no início. A declaração não é uma tag HTML, e sim uma instrução para o navegador sobre qual versão HTML está escrito o código fonte da página. No cabeçalho do script dentro da tag `<?php>` deve ser inserida a declaração `defined('_JEXEC') or die`, que é responsável por verificar se o arquivo está sendo chamado de dentro de uma sessão do Joomla!, protegendo o site de ataques externos. A figura 27 mostra parte do arquivo *index.php*.

```
<?php
defined('_JEXEC') or die;
$path = $this->baseurl . '/templates/' . $this->template;
$app = JFactory::getApplication();
?>
<!DOCTYPE html>
<html lang="<?php echo $this->language ?>" dir="<?php echo $this->direction ?>">
<head>
    <jdoc:include type="head" />
    <link href="<?php echo $path ?>/css/Bootstrap.min.css" rel="stylesheet">
    <link href="<?php echo $path ?>/css/Bootstrap-responsive.min.css" rel="stylesheet">
    <link href="<?php echo $path ?>/css/style.css" rel="stylesheet">

    <!-- HTML5 shim, for IE6-8 support of HTML5 elements -->
    <!--[if lt IE 9]>
        <script src="js/html5shiv.js"></script>
    <![endif]-->

    <!-- Fav and touch icons -->
    <link rel="apple-touch-icon-precomposed" sizes="144x144" href="<?php echo $path
?>image/apple-touch-icon-144-precomposed.png">
    <link rel="apple-touch-icon-precomposed" sizes="114x114" href="<?php echo $path
?>image/apple-touch-icon-114-precomposed.png">
    <link rel="apple-touch-icon-precomposed" sizes="72x72" href="<?php echo $path
?>image/apple-touch-icon-72-precomposed.png">
    <link rel="apple-touch-icon-precomposed" href="<?php echo $path ?>image/apple-touch-
icon-57-precomposed.png">
    <link rel="shortcut icon" href="<?php echo $path ?>image/favicon.png">

    <script
src="https://ajax.googleapis.com/ajax/libs/jquery/1.10.2/jquery.min.js"></script>
    <script src="<?php echo $path ?>/js/Bootstrap.min.js"></script>
</head>
```

Figura 28 – Parte Inicial do Arquivo index.php

A definição da variável `$path = $this->baseurl . '/templates/' . $this->template;` presente dentro da tag `<?php>` é a responsável pelo direcionamento da página pela

arquitetura do Joomla! e localização dos arquivos necessários para o carregamento correto da página, através da *url*.

6.4 MAPEANDO MÓDULOS PARA O CMS JOOMLA!

Para se mapear módulos é necessário a declaração *jdoc:include* – que é um modelo de exibição de conteúdo específico para a página que está sendo visualizada. Ele carrega informações de módulos, cabeçalhos e *javascript* do Joomla! necessários, para um correto funcionamento da página. Para visualizar essas informações incluídas pela declaração *jdoc:include*, tecla-se *ctrl+u* na página do site.

6.4.1 Módulo Menu

Como dito anteriormente, é necessária a utilização de declarações do tipo *jdoc:include* para o mapeamento correto da página para o Joomla!. A instrução `<jdoc:include type="modules" name="Bootstrap-mainmenu" style="mainmenu" />`, é responsável para carregar e referenciar os menus que forem criados no *backend* do Joomla. Porém não é tão simples assim, pois a classe do Joomla (*class menu*) é diferente da do *Bootstrap* (*class nav*), o que torna necessário reescrever o módulo. Para isso é necessário copiar os arquivos da pasta *mod_menu*, menos a pasta *tpl* da aplicação, para dentro da pasta *html* do *template*. É importante ressaltar que, ao reescrever o módulo, isto vai valer para todos os menus do site. Deve-se alterar o nome da classe *menu* para *nav* do arquivo *default.php* que foi reescrito. Como mostrado na figura 28.

```

//classe Joomla! original
<ul class="nav<?php echo $class_sfx;?>">?php
    $tag = '';
    if ($params->get('tag_id')!=NULL) {
        $tag = $params->get('tag_id').'';
        echo ' id="'. $tag. '";'
    }
//classe Joomla! Reescrita.
<ul class="nav<?php echo $class_sfx;?>">?php
    $tag = '';
    if ($params->get('tag_id')!=NULL) {
        $tag = $params->get('tag_id').'';
        echo ' id="'. $tag. '";'
    }
}

```

Figura 29 – Classe Menu

Para que os módulos funcionem corretamente é necessário carregar os arquivos *javascripts* e *jquery* na página, o que é feito através das instruções mostradas na figura 29, presentes no arquivo *index.php*.

```

<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.10.2/jquery.min.js"></script>
<script src="<?php echo $path ?>/js/Bootstrap.min.js"></script>

```

Figura 30 – Carregando Arquivo javascript

O arquivo *jquery.min.js*, foi direcionado para ser utilizado o do google por motivos de cache e de desempenho. Este é responsável por efetuar buscas. Já o *Bootstrap.min.js* é o arquivo que é baixado do site do *Bootstrap*.

6.4.2 Módulo Personalizar HTML

Foi utilizado o padrão HTML para os dois módulos de visualização (título e mural) e sobrescrita à função destes, para que seja possível utilizar o *backend*. Estes módulos receberão o nome de *Bootstrap-welcome* e *Bootstrap-blocks*, tendo suas funcionalidades adaptadas para utilização no *template*. Para a função *welcome* foi alterado a *tag<h1>* e a *class*, e para a função *blocks* foi definido um padrão de três (3) colunas para os módulos, possibilitando desta forma uma melhor organização dos módulos pelo *template*, como mostrado na figura 30. Para isso foi copiado as

function modChrome_xhtml de sua pasta original (*template / system / html / modules.php*) para a pasta *html* do projeto e alterado o nome da função para *welcome* e *blocks*.

```

/* welcome*/
function modChrome_welcome($module, &$params, &$attribs)
{
    if (!empty ($module->content)) : ?>
        <div class="hero-unit"<?php echo htmlspecialchars($params->get('moduleClass_sfx'));
?>">
            <?php if ($module->showtitle != 0) : ?>
                <h1><?php echo $module->title; ?></h1>
            <?php endif; ?>
            <?php echo $module->content; ?>
        </div>
    <?php endif;
}
/* blocks */
function modChrome_blocks($module, &$params, &$attribs)
{
    if (!empty ($module->content)) : ?>
        <div class="span"<?php echo $params->get('moduleClass_sfx') ?
htmlspecialchars($params->get('moduleClass_sfx')) : '3' ;?>">
            <?php if ($module->showtitle != 0) : ?>
                <h2><?php echo $module->title; ?></h2>
            <?php endif; ?>
            <?php echo $module->content; ?>
        </div>
    <?php endif;
}

```

Figura 31 – Função de Módulos HTML Sobrescritos

A função *blocks* na imagem anterior define que caso a instrução *\$params->get('moduleclass_sfx')* não tenha sido definida pelo administrador no *backend*, o *span* (área destinada a imagem da propaganda) terá um padrão de 3 colunas (*span3*) no corpo da página, o que facilita na formação do mural conforme figura 31.

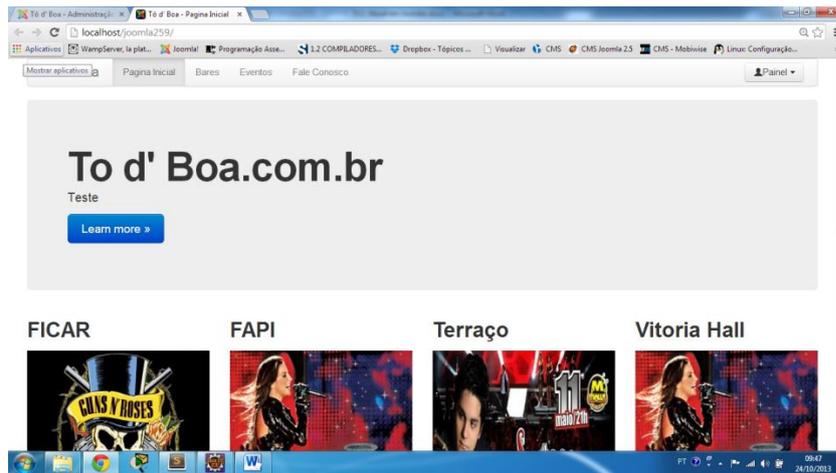


Figura 32 – Disposição dos HTML com 3 Colunas

6.5 PASTAS DO PROJETO

A pasta CSS contém os arquivos *Bootstrap-responsive.min.css*, *Bootstrap.min.css* e *template.css*. Arquivos, estes, responsáveis pela flexibilidade de adaptação da página em dispositivos de telas de tamanhos variáveis, arquivos de plug-ins do *Bootstrap* e de alinhamento dos componentes na página, respectivamente. O arquivo *template.css* é mostrado na figura 32.

```

/* Tooltips */
.tip-wrap{
  z-index: 10000;
}
.tip {
  float: left;
  background: #ffc;
  border: 1px solid #D4D5AA;
  padding: 5px;
  max-width: 200px;
}

```

Figura 33 – Arquivo template.css

Este arquivo possui diversas definições de módulos e componentes espalhados pela página. Através dele pode-se alterar as características de posicionamento de *labels*, *checkbox* e *tooltips*, entre outros. Para isto basta inserir inicialmente o nome da *class* precedido por um ponto (*.inline label*) e as definições dentro de colchetes, caso seja uma *id* esta deve ser precedida por um #.

Na pasta *html* estão presentes os módulos e componentes que foram manipulados/editados como módulos de menu, módulos *welcome* e *blocks* e componentes de contato. Arquivos estes que são responsáveis pela apresentação da página web.

Pasta *images* contém os ícones do *Bootstrap* em um único arquivo. Ícones como de impressora, e-mail e outros, além de arquivos de imagens que podem ser inseridos para melhorar a apresentação, como o arquivo de imagem *selector-arrow.png* que é um *tooltips*.

Pastas *img* e *js* do *Bootstrap*, que contém os arquivos de imagens de ícones com fundo claro e escuro e o arquivo de *Javascript*, respectivamente.

Na pasta de *language* são encontrados os arquivos de tradução do sistema *pt-BR.tpl_Bootstrap.sys.ini* e *pt-BR.tpl_Bootstrap.ini*, já mencionados anteriormente, estes arquivos contém a tradução dos componentes e módulos abordados, como mostra a figura 33.

```
Bootstrap="Bootstrap"

TPL_BOOTSTRAP_PAINEL="Painel"

TPL_BOOTSTRAP_XML_DESCRIPTION="Template para CMS Joomla!"

TPL_BOOTSTRAP_MESSAGE_NOTICE="Aviso! "

TPL_BOOTSTRAP_MESSAGE_ALERT="Alerta"

TPL_BOOTSTRAP_MESSAGE_ERROR="Erro! "}
```

Figura 34 – Arquivo de Tradução

CONCLUSÃO

Com este trabalho conclui-se que apesar de o CMS Joomla!, não ser o mais utilizado hoje em dia, é um excelente CMS. Possui excelente área administrativa, totalmente em *backend*, o que permite que o proprietário do site possa fazer alterações em seu conteúdo sem a necessidade da intervenção de terceiros.

Com uma grande quantidade de módulos e componentes disponíveis, o Joomla permite que o proprietário possa personalizar sua página conforme sua necessidade, desde que atendendo as características do *template*.

Diferentemente do que foi visto nos materiais utilizados para este trabalho, o Joomla não é uma ferramenta em que qualquer usuário vai sair produzindo um site, pois este possui um *backend* com muitas funcionalidades, fazendo com que o usuário inexperiente se confunda com tantos recursos. Para se utilizar o Joomla o usuário deve entender pelo menos um pouco de páginas *web* e ter algum material de apoio atualizado, pois o Joomla altera muito suas funcionalidades e particularidades de uma versão para outra.

No caso do Joomla!2.5 que não é responsivo, a utilização do *Framework Bootstrap* para tornar seus *templates* responsivos parece uma boa saída, mas não é trabalho para qualquer um. Quem quiser se aventurar deve conhecer bem a linguagem PHP, HTML, além de um conhecimento amplo da arquitetura do Joomla!, caso contrario terá problemas no momento de mapear os componentes e módulos do *template* responsivo para o *backend* do Joomla!

Para projetos novos vale mais apenas instalar o Joomla! 3.x, pois este já possui as classes do *Bootstrap*, incorporadas em sua arquitetura.

Como trabalho futuro fica a possibilidade do desenvolvimento de um componente a ser integrado ao *template* desenvolvido como: um chat que interaja ao componente do site, onde usuários podem deixar mensagens a outros usuários, melhorias na parte visual, criação de menus para categorizar o site mural.

REFERÊNCIAS BIBLIOGRÁFICAS

BARRIE, M.Norton – **Guia do Operador**: Construindo um website com Joomla, 1. ed. Tradução de Raquel Marques e Arcanjo Miguel, Rio de Janeiro: Atlas Book, 2008.

BOOTSTRAP, Site Oficial disponível <<http://getBootstrap.com>> acessado em 3/09/2013.

BOOTSTRAP, Site Oficial Traduzido <<http://globocom.github.io/Bootstrap/scaffolding.html>> acessado em 17/09/2013.

BROOKS, F; BULLET, N. Essence and Accidents of Software Engineering. IEEE Computer, 20(4):10–19, 1987.

DRUPAL, Site Oficial disponível <<http://drupal.org/about> > acessado em 17/06/2013.

GERENCIAMENTO DE CONTEUDO <<http://cocoate.com/pt-br/j25pt/gerenciando-conteudo> > acessado em 28/05/2013

GRAF,HAGEN;KRAMER,JEA. Joomla! 2.5 Livro do Iniciante, 20(4):10–19, 1987. <http://www.mobivise.com.br/website/sistemas-de-gerenciamento-de-conteudo/> acessado em 15/05/2013.

JOOMLA!, Site Oficial disponível em <<http://www.joomla.org/about-joomla.html> > acessado em 22/05/2013.

JOOMLA CLUBE <<http://www.joomlaclube.com.br/site/comunidade.html> > acessado em 22/05/2013

LAYOUTIT!, Site Oficial disponível em <<http://www.layoutit.com/> > acessado em 04/09/2013.

MAMBO, Site Oficial disponível <<http://mambo-code.org/gf> > acessada em 17/06/2013.

NIEDSON, Robson, 2007 – **SISTEMA DE GERENCIAMENTO DE CONTEUDO** <<http://www.vivaolinux.com.br/artigo/Sistema-de-Gerenciamento-de-Conteudo/>> acessado em 20/05/2013.

OLHARDIGITAL, Site de notícia < <http://olhardigital.uol.com.br/pro/noticia/37111/37111>> acessada em 27/11/2013.

PEREIRA, J.C.L.; BAX, M. P. Introdução à Gestão de Conteúdos. In: KM BRASIL, 2002,

PINGENDO, Site Oficial disponível em <<http://www.pingendo.com/> > acessado em 01/10/2013.

São Paulo. Anais (CD-ROM)... São Paulo: [s.n.], 2002.

SISTEMA DE GERENCIAMENTO DE CONTEUDO EM SOFTWARE LIVRE
<<http://www.hardware.com.br/dicas/sgc-livre.html>> acessado em 15/05/2013.