

ISSN 1677-9274

## Comparativo do Plone com Gerenciadores de Conteúdo em PHP, Java e Python



## **República Federativa do Brasil**

*Luiz Inácio Lula da Silva*  
Presidente

## **Ministério da Agricultura, Pecuária e Abastecimento**

*Roberto Rodrigues*  
Ministro

## **Empresa Brasileira de Pesquisa Agropecuária - Embrapa**

### **Conselho de Administração**

*Luis Carlos Guedes Pinto*  
Presidente

*Silvio Crestana*  
Vice-Presidente

*Alexandre Kalil Pires*  
*Hélio Tollini*  
*Ernesto Paterniani*  
*Marcelo Barbosa Saintive*  
Membros

### **Diretoria-Executiva da Embrapa**

*Silvio Crestana*  
Diretor-Presidente

*José Geraldo Eugênio de França*  
*Kepler Euclides Filho*  
*Tatiana Deane de Abreu Sá*  
Diretores-Executivos

### **Embrapa Informática Agropecuária**

*Eduardo Delgado Assad*  
Chefe-Geral

*José Ruy Porto de Carvalho*  
Chefe-Adjunto de Administração

*Kleber Xavier Sampaio de Souza*  
Chefe-Adjunto de Pesquisa e Desenvolvimento

*Deise Rocha Martins dos Santos Oliveira*  
Supervisora da Área de Comunicação e Negócios

# ***Documentos 55***

## **Comparativo do Plone com Gerenciadores de Conteúdo em PHP, Java e Python**

Luciana Alvim Santos Romani  
Luciana Aparecida de Campos

Campinas, SP  
2005

**Embrapa Informática Agropecuária**  
**Área de Comunicação e Negócios (ACN)**  
Av. André Tosello, 209  
Cidade Universitária "Zeferino Vaz" – Barão Geraldo  
Caixa Postal 6041  
13083-970 – Campinas, SP  
Telefone (19) 3789-5743 – Fax (19) 3289-9594  
URL: <http://www.cnptia.embrapa.br>  
e-mail: sac@cnptia.embrapa.br

### **Comitê de Publicações**

*Adriana Farah Gonzalez (secretária)*  
*Ivanilde Dispato*  
*Kleber Xavier Sampaio de Souza (presidente)*  
*Luciana Alvim Santos Romani*  
*Marcia Izabel Fugisawa Souza*  
*Renato Fileto*  
*Stanley Robson de Medeiros Oliveira*

### **Suplentes**

*José Iguelmar Miranda*  
*Laurimar Gonçalves Vendrusculo*  
*Maria Goretti Gurgel Praxedis*  
*Silvio Roberto Medeiros Evangelista*

Supervisor editorial: *Ivanilde Dispato*  
Normalização bibliográfica: *Marcia Izabel Fugisawa Souza*  
Editoração eletrônica: *Área de Comunicação e Negócios (ACN)*

### **1<sup>a</sup>. edição on-line - 2005**

**Todos os direitos reservados.**

---

Romani, Luciana Alvim Santos.

Comparativo de Plone com gerenciadores de conteúdo em PHP, Java e Python / Luciana Alvim Santos Romani, Luciana Aparecida de Campos. — Campinas : Embrapa Informática Agropecuária, 2005.

26 p. : il. — (Documentos / Embrapa Informática Agropecuária ; 55).

ISSN 1677-9274

1. Gerenciador de conteúdos. 2. Plone. 3. PHP. 4. Python. I. Campos, Luciana Aparecida de. II. Título. III. Série.

CDD 004.678 (21<sup>st</sup>. ed.)

---

© Embrapa 2005

## **Autoras**

### **Luciana Alvim Santos Romani**

M.Sc. em Ciéncia da Computaçao, Pesquisadora da  
Embrapa Informática Agropecuária, Caixa Postal 6041,  
Barão Geraldo  
13083-970 - Campinas, SP  
Telefone (19) 3789-5792  
e-mail: luciana@cnptia.embrapa.br

### **Luciana Aparecida de Campos**

Estagiária da Embrapa Informática Agropecuária, Caixa  
Postal 6041, Barão Geraldo  
13083-970 - Campinas, SP



# Apresentação

Um gerenciador de conteúdo é uma ferramenta que permite automatizar e integrar todos os procedimentos relacionados desde a criação, catalogação, indexação, personalização, controle de acesso até a publicação de conteúdos em portais *web*. O gerenciador de conteúdo permite organizar informações que têm valor para a empresa, com facilidade e eficiência, tornando-se um componente-chave para manter a vantagem competitiva.

Essa busca contínua por rapidez na publicação de conteúdos na *web* quase que em tempo real acaba por atingir as organizações ainda sem preparo. Em meio à desorganização de recursos de preparação de pessoal e do próprio desconhecimento das ferramentas disponíveis no mercado, e com a maioria dos processos de publicação ainda manuais, aliados a fluxos de trabalho pouco definidos, surge à ineficiência. Os custos e o tempo para a publicação das informações de todos os departamentos da organização disponibilizadas sobem expressivamente e os usuários do *website* ficam insatisfeitos.

Neste sentido, uma série de gerenciadores de conteúdo tem sido desenvolvida com o objetivo de facilitar a criação e gerenciamento de *sites* tanto pessoais como institucionais. Eles podem ser escritos nas mais diversas linguagens de programação que tenham algum tipo de interface com a *web*. Alguns exemplos de linguagens mais utilizadas são: PHP, Java, Phyton e Perl entre outros. Existem inúmeras alternativas para o gerenciamento de conteúdo, é função do gerente a definição da melhor alternativa que seu sistema necessita.

Neste trabalho, é feita uma descrição das principais soluções em termos de gerenciadores de conteúdo disponíveis no mercado com uma comparação destas ferramentas. O trabalho destaca a ferramenta Plone, que é um gerenciador de conteúdos de código aberto que vem sendo amplamente utilizado devido ao seu grande conjunto de recursos e pela sua poderosa máquina de *workflow* que gerencia e organiza a publicação de documentos no *website*.

*Eduardo Delgado Assad*  
Chefe-Geral



# Sumário

<b>Introdução.....</b>	<b>9</b>
<b>Definição e Estrutura de um CMS.....</b>	<b>10</b>
<b>Plataformas de Desenvolvimento.....</b>	<b>11</b>
<b>PHP.....</b>	<b>11</b>
<b>Java.....</b>	<b>12</b>
<b>Python.....</b>	<b>13</b>
<b>Gerenciador de Conteúdo – Plone.....</b>	<b>15</b>
<b>Comparativo de Plone com outros Gerenciadores.....</b>	<b>18</b>
<b>Conclusões.....</b>	<b>24</b>
<b>Referências Bibliográficas.....</b>	<b>25</b>



# Comparativo do Plone com Gerenciadores de Conteúdo em PHP, Java e Python

---

*Luciana Alvim Santos Romani*

*Luciana Aparecida de Campos*

## Introdução

Atualmente, a maioria dos *websites* não é mais construída a partir de ferramentas manuais para geração de páginas estáticas, como o *FrontPage* ou o *DreamWeaver*. Para auxiliar na criação de *sites* foram desenvolvidos os chamados gerenciadores de conteúdo, também conhecidos como CMS - do inglês, Content Management System. Essas ferramentas suportam a criação, gerenciamento, distribuição e publicação de conteúdo na internet (Pokorny, 2001; Robertson, 2003).

Este novo paradigma de desenvolvimento permite que pessoas leigas em tecnologia da informação e computação possam atualizar e publicar conteúdo de forma rápida, segura e com qualidade. Esses gerenciadores possibilitam que várias pessoas incluam ou alterem conteúdo no *site* mantendo a integridade e o *design* das páginas. Para isso, basta o usuário acessar o CMS por meio de um navegador. O sistema gerenciador instalado no servidor fará acessos ao banco de dados acoplado e à estrutura predefinida de *layout* das páginas para manter o *site* atualizado e padronizado.

Nos últimos anos, foram desenvolvidos inúmeros CMSs em diversas linguagens. A grande maioria deles baseia-se em PHP e MySql. No entanto, existem CMSs em Perl, Java e Python. Os CMSs desenvolvidos em PHP são bastante utilizados gerando uma numerosa comunidade de usuários que contribuem para sua melhoria e manutenção. No entanto, o gerenciador Plone desenvolvido na linguagem Python tem alcançado muitos adeptos nos últimos anos. Plone é um gerenciador de conteúdos bastante completo com uma interface amigável, uma máquina de *workflow* que permite organizar o conjunto de usuários e uma preocupação com a acessibilidade dos *sites* criados.

O objetivo deste artigo é apresentar um comparativo de Plone com outros gerenciadores de conteúdo desenvolvidos também em Python e em outras linguagens como PHP e Java. Com isso, pretende-se dar subsídios para a escolha de um CMS que mais se adeque às necessidades do leitor. No próximo tópico é apresentada de forma mais detalhada a estrutura de um gerenciador de conteúdos.

A seguir, são descritas as plataformas de desenvolvimento de CMSs; descrito também com mais detalhes o gerenciador Plone; é feita uma comparação de Plone com outros CMSs e apresentadas as conclusões do trabalho.

## Definição e Estrutura de um CMS

Um CMS permite que um *site* seja criado de forma facilitada por meio de ferramentas para edição de conteúdo, alteração de *layout* da página, controle de acesso de usuários, revisão de conteúdo e publicação na *internet*. O gerenciamento da estrutura e a navegação também são controlados pelo gerenciador sem ônus para o editor e revisor de conteúdo.

Segundo Robertson (2003), um CMS possui as seguintes funcionalidades: criação e gerenciamento do conteúdo, publicação e apresentação. Um ambiente de autoria permite que diversos tipos de conteúdos sejam criados ou importados para a estrutura do CMS. Desta forma, tanto a criação de novas páginas quanto a atualização de páginas já existentes é feita de forma simples sem necessidade de conhecimento da sintaxe da linguagem na qual o CMS foi desenvolvido. Em geral, esses ambientes de autoria são acessados remotamente por meio de um navegador (Mozilla, FireFox, Internet Explorer, etc.). Com isso, é possível que o *site* seja atualizado por diferentes usuários simultaneamente, sem o risco de conflito ou de geração de páginas despadronizadas.

O conteúdo a ser apresentado nas páginas do *site* é armazenado em um repositório central. O uso de uma base de dados permite:

- manter um histórico das versões das páginas, informações sobre quem e quando essas páginas foram alteradas;
- controlar o acesso às páginas do *site* apenas pelos usuários que tenham permissão para inclusão, alteração e revisão daquela parte do *site*; e
- integrar o *site* com outras fontes de informação e sistemas.

Todo conteúdo disponível na base de dados do CMS está apto para publicação. No entanto, em alguns CMSs, a publicação só é efetuada após autorização do revisor. Em outros casos, o próprio editor pode autorizar a publicação do conteúdo.

A aparência do *website* é definida pelo administrador ou editor responsável e é carregada automaticamente pelo CMS no momento em que o *site* é aberto por um navegador. Todo CMS utiliza um mecanismo para aplicação de aparência e *layout* das páginas. Este mecanismo é uma programação dinâmica interpretada ou executada no momento em que o usuário acessa o *site* (Pokorny, 2001), ou seja, o CMS busca no banco de dados o conteúdo a ser apresentado e aplica a ele a estrutura e aparência do *site* no momento do acesso. Esta aparência pode ser diferente para cada *site* criado com o mesmo CMS, pois é possível personalizá-la

utilizando opções de configuração do gerenciador, arquivos CSS - Cascade Style Sheets ou alterações utilizando comandos em HTML. Apesar de ser possível, não é necessária a modificação do código fonte do CMS para fazer alterações na aparência final do *website*. A navegação também é criada pelo CMS a partir das definições da estrutura contidas no repositório de conteúdos.

Uma característica importante é a possibilidade e facilidade de incorporação de novos módulos ou produtos. Isto ocorre pelo fato da maioria dos CMSs ser construída utilizando o princípio dos "blocos encaixe", ou seja, possuir uma estrutura modular. Esses novos recursos podem ser gerados pela própria equipe responsável pelo desenvolvimento do CMS ou por qualquer outro colaborador de forma independente. Uma vez desenvolvido o módulo ou produto, ele pode ser anexado à estrutura do gerenciador de forma fácil quando necessário. Essa estrutura permite que os CMSs possam ser adaptados para melhor atender aos objetivos de cada *síté* sem necessidade de alterações no núcleo do gerenciador, o que poderia causar problemas ou ruptura de serviços fornecidos pelo *síté*.

## Plataformas de Desenvolvimento

Diferentes linguagens de programação que possuam interface para a *web* podem ser usadas para desenvolvimento de gerenciadores de conteúdo. A grande maioria deles é escrita em PHP. No entanto, existem CMSs em Perl, Java e Python. A seguir, é apresentado um descriptivo de cada linguagem com exemplos dos CMSs mais utilizados.

### PHP

É uma linguagem de *script*, de código aberto, utilizada para o desenvolvimento de aplicações *web* que pode ser embutida dentro de arquivos HTML. O nome PHP é um acrônimo recursivo para "PHP: Hypertext Preprocessor". A linguagem foi criada a partir de um produto mais antigo denominado PHP/FI proposto por Rasmus Lerdorf em 1995 como um conjunto de *scripts* em Perl para gerar estatísticas de acesso para seu currículo *on-line*. Ele denominou esta série de *scripts* como Personal Home Page Tools e para atender a mais funcionalidades, escreveu uma nova versão em C com acesso a base de dados e possibilidade de criação de pequenos aplicativos dinâmicos para *web*. Após disponibilizar o código fonte do PHP/FI, o mesmo foi sendo melhorado até culminar em uma versão lançada em 1997 (Lerdorf, 2005). Posteriormente, foi substituído pelo PHP 3.0 que foi evoluído até a versão 5.0 disponível para uso atualmente.

O PHP é uma linguagem interpretada, ou seja, não há necessidade de compilar (transformar em executável) as aplicações. O interpretador lê cada linha de código do arquivo e as executa. Seu código é delimitado por *tags* iniciais (*<?php*) e finais

(?>) que indicam se ele está ou não no "modo PHP". O que distingui o PHP de programas em Javascript, por exemplo, no lado do cliente é que o código será executado no servidor. Com PHP é possível obter as mesmas funcionalidades geradas com programas CGI como: coletar dados de formulários, gerar páginas com conteúdo dinâmico ou enviar e receber *cookies* (PHP Group, 2005).

Uma das vantagens de uso desta linguagem é pelo fato de ser muito simples para um iniciante, e também por oferecer diversos recursos para o programador profissional. Uma outra vantagem é que os programas podem ser rodados em qualquer sistema operacional, sem a necessidade de comandos adicionais ou diferentes. A grande desvantagem é a velocidade de execução, pois os programas interpretados são mais lentos em comparação aos compilados. Para desenvolver programas em PHP é necessária a instalação de um servidor *web* como o Apache, por exemplo e o próprio *software* PHP. Além disso, deve-se instalar um banco de dados como o MySQL ou PostgreSQL.

Devido a suas características e facilidades de implementação, a linguagem PHP é a mais utilizada para desenvolvimento de gerenciadores de conteúdo. O mais conhecido deles é o PHP-Nuke que foi desenvolvido a partir de um sistema de publicação - Nuke - desenvolvido em Perl. O PHP-Nuke é um sistema de portal bastante completo com uma estrutura modular, que facilita a integração rápida de novas ferramentas (PHP-Nuke, 2005). Outros gerenciadores de destaque também desenvolvidos em PHP são: PostNuke (PostNuke.Com, 2005), Xoops (2005), Drupal (2005), eZ Publish (eZ Systems AS, 2005), Guppy (2005), Joomla! (Joomla! Accessibility Statement, 2005) e Mambo (Miro International Pty Ltd., 2005).

## Java

É uma linguagem de programação de alto nível que tem como principais características ser: simples, orientada a objetos, distribuída, interpretada, robusta, segura, de arquitetura neutra, portável, de alta performance, *multithreaded* e dinâmica. Diferentemente das demais linguagens de programação, Java é compilada e depois interpretada. O compilador traduz o código fonte em uma linguagem intermediária chamada "Java bytecodes", um código independente de plataforma. O interpretador traduz e executa cada instrução em Java bytecode no computador. A compilação ocorre uma única vez e a interpretação é realizada toda vez que o programa é executado.

O Java bytecode pode ser considerado como as instruções de código de máquina para a Máquina Virtual Java (Java Virtual Machine - Java VM). Todo programa compilado para bytecodes em qualquer plataforma que tenha um compilador Java pode ser executado em diversos sistemas operacionais como Windows, Solaris, Linux, Macintosh, etc.

A plataforma Java é composta por dois componentes: a Java VM e Java API - Java Application Programming Interface. A Java API é uma extensa coleção de componentes de *software* que provê funcionalidades muito úteis, como por exemplo interface gráfica. Ela é agrupada dentro de bibliotecas de classes e interfaces que são chamadas de pacotes. Para o desenvolvimento de aplicações *web* com a linguagem Java são utilizados as Servlets ou JSP - Java Server Pages.

Servlets são programas Java que são carregados dinamicamente para atender solicitações de um servidor *web*, gerando conteúdo dinâmico e interagindo com os clientes, utilizando o modelo *request/response* (Zeiger, 2005). Em qualquer Servlet deve-se implementar direta ou indiretamente a interface javax.servlet.Servlet. O ciclo de vida de um servlet é determinado por três de seus métodos: *init*, *service* e *destroy*. Os servidores carregam e executam o servlet, que por sua vez aceita nenhuma ou mais requisições de clientes e retorna dados para o mesmo. Os servidores podem também remover os servlets.

JSP é uma tecnologia para desenvolvimento de aplicações *web* que permite ao desenvolvedor de *sites* produzir aplicações com o acesso a banco de dados, a arquivos-texto, a captação de informações a partir de formulários, sobre o visitante e sobre o servidor, o uso de variáveis e *loops* dentre outros. A codificação em JSP é simples, o que facilita a elaboração e manutenção de uma aplicação. Além disso, essa tecnologia permite separar a programação lógica (parte dinâmica) da programação visual (parte estática), facilitando o desenvolvimento de aplicações mais robustas, onde programador e *designer* podem trabalhar no mesmo projeto, mas de forma independente. Outra característica do JSP é produzir conteúdos dinâmicos que possam ser reutilizados (J2EEBrasil, 2005).

Apesar da grande maioria dos CMSs serem escritos em PHP, existem gerenciadores desenvolvidos também em plataforma Java, utilizando XML, XSLT, Javascript e JSP. O mais conhecido é o OpenCms que permite a criação e gerenciamento de *websites* complexos e possui um editor com uma interface gráfica. Este tipo de interface WYSIWYG - what you see is what you get - auxilia os usuários na criação do conteúdo (Alkacon Software Gmbh, 2005). Outros CMSs escritos em Java também utilizados são o Apache Lenya (Apache Software Foundation, 2005) e o Enonic Vertical Site (Enonic, 2005).

## Python

É uma linguagem de programação orientada a objetos, interativa e interpretada. Ela é uma linguagem poderosa, mas de fácil aprendizado. O início do seu desenvolvimento foi em 1990 por Guido Van Rossum, no Centrum voor Wiskunde en Informática - CWI na Holanda e continua sendo desenvolvido pela Python Software Foundation - PSF. Possui estruturas de dados de alto nível eficientes e adota uma abordagem efetiva para programação orientada a objetos. Python é ideal para desenvolvimento rápido de aplicações em diversas áreas devido a sua sintaxe elegante, tipagem dinâmica e pelo fato de ser interpretada (Rossum & Drake, 2004).

A incorporação de novos módulos escritos em C ou C++ é feita de forma fácil. Ela também é usada como uma linguagem de extensão para aplicações que precisam de uma interface programável. A implementação Python é portável pois pode ser executada em diferentes sistemas operacionais como Windows, Linux, Unix, OS/2, Mac, Amiga, etc.

Python serviu de base para o desenvolvimento do Zope, um *software* livre de código aberto, criado pela Digital Creations - atual Zope Corporation. Ele permite gerenciar e desenvolver *sites* e aplicações *web* com conteúdo dinâmico, de forma relativamente simples se comparado aos *sites* estáticos (Zope Corporation, 2005). Da mesma forma que Python, Zope é orientado a objetos. Ele permite a criação de *scripts* diretamente em Python para resolver problemas mais complexos de uma aplicação. No entanto, a maioria dos problemas é resolvida com DTML - Document Template Markup Language. O Zope possui também a ZPT - Zope Page Template que é uma ferramenta para geração de páginas *web*. Elas ajudam os programadores e *designers* a colaborar na produção de páginas *web* dinâmicas para aplicações *web* em Zope. Os *designers* podem usá-las para manter as páginas sem ter que abandonar suas ferramentas de edição. O objetivo da ZPT é permitir que programadores e *designers* trabalhem juntos de forma mais facilitada. O *designer* pode usar um editor visual de HTML para criar um *template* e o programador pode editá-lo e torná-lo parte de uma aplicação.

CMF - Content Management Framework é um recurso disponível na plataforma Zope desenvolvido para resolver questões relativas à gerência de conteúdo. Este *framework* inclui serviços para catalogação, *workflow*, integração, teste, criação e controle de usuários, dentre outros. Zope é multiplataforma e possui as mesmas funcionalidades que linguagens como PHP, ASP ou ColdFusion. Tem como principal vantagem a facilidade de atualização do conteúdo por parte dos responsáveis pelo *site*, pois um *site* em Zope é integralmente mantido e administrado pela própria *web*, por meio de uma interface similar a um gerenciador de arquivos, usando *frames*, *logins* e senhas.

Apesar de possuir seu próprio banco de dados orientado a objetos (ZODB), é possível utilizar o Zope com outros bancos como Oracle, MySQL, SQL Server, por exemplo. Todo o conteúdo é armazenado no banco de dados, mesmo páginas simples, contendo apenas HTML. Estas páginas e todos os documentos armazenados são fornecidos dinamicamente ao navegador, deixando de lado o conceito de páginas estáticas. O Zope também funciona como servidor *web*, denominado ZServer, porém é possível utilizar outro servidor para hospedar uma aplicação ou um *site* desenvolvido em Zope.

O gerenciador de conteúdos Plone foi desenvolvido em Python/Zope e é um CMSs bastante utilizado atualmente. Uma descrição mais completa é apresentada no tópico seguinte deste documento. O GeCon é um outro exemplo de gerenciador de conteúdo desenvolvido em Zope, com o objetivo de permitir que pessoas com conhecimento técnico relativamente restrito, tenham a possibilidade de

administrar seus *websites* sem a necessidade de intervenções diretas de desenvolvedores ou técnicos (Romani, 2004). O PyLucid é um outro gerenciador de conteúdos desenvolvido em Python puro que utiliza MySql como banco de dados para armazenar o conteúdo do *site* gerenciado (PyLucid CMS, 2005).

## Gerenciador de Conteúdo – Plone

É um gerenciador de conteúdos livre e de código aberto, desenvolvido sobre o servidor de aplicações Zope. Da mesma forma que no Zope, todo conteúdo criado no Plone é tratado como um objeto. Ele destaca-se como o CMS desenvolvido em Zope mais utilizado atualmente. Segundo McKay (2005), Plone tem por objetivo agregar valor a cada nível da organização, por meio de um *workflow* para publicação de páginas e documentos. Esse *workflow* é baseado em permissões para os diferentes tipos de usuário que podem ser cadastrados e no estado dos objetos. Existem quatro estados possíveis para os objetos: visível, pendente, publicado e privado. Os usuários podem ser cadastrados como anônimos, membros, revisores, proprietários e gerentes.

Todo conteúdo criado ou inserido no *site* como por exemplo textos, imagens, arquivos, etc. são considerados objetos e assumem inicialmente o estado "visível". Neste momento, apenas o proprietário e o gerente do *site* podem editá-lo. Nesse estado, qualquer usuário pode visualizar o objeto por meio da função de busca ou digitando a URL do objeto diretamente no navegador. Entretanto, ele não aparece como um *link* na árvore de navegação do *site*. Os objetos "visíveis" podem assumir o estado "pendente" quando são submetidos para publicação e aguardam uma aprovação para serem publicados. Neste caso, eles podem ser editados pelos revisores que poderão aprová-lo ou rejeitá-lo. Uma vez no estado "publicado", os objetos passam a ser visíveis na árvore de navegação do *site* podendo ser editados apenas pelo gerente. O proprietário do documento pode apenas reverter seu estado para "visível". No estado "privado", o objeto passa a ser visível e editável apenas pelo seu proprietário e pelo gerente.

O perfil do usuário define o que ele pode ver e fazer dentro do *site*. Anônimos são todos os usuários que acessam o *site* sem se identificar. Eles podem visualizar objetos nos estados publicado, pendente e visível por meio de busca ou digitando a URL do objeto no navegador. Usuários cadastrados no *site* têm alguns privilégios. Eles são proprietários de uma pasta na qual podem criar conteúdo no estado visível e submetê-los para publicação. O revisor pode editar um conteúdo pendente, aprová-lo ou rejeitá-lo. As regras para o proprietário são as mesmas que um membro possui em relação a um objeto criado por ele. Os gerentes possuem permissão máxima de acesso ao conteúdo do *site*, ou seja, podem ver todos os conteúdos em todos os estados. Eles podem adicionar, editar, apagar e mover qualquer conteúdo ou usuário.

Para criar conteúdos no *site* gerenciado, o usuário cadastrado precisa se identificar, informando o *login* e senha. Para adicionar o conteúdo, o usuário ativa o modo de "visão de conteúdo" (Fig. 1). Na lista que lhe será exibida, ele escolhe o tipo de conteúdo a ser incluído e seleciona a opção "selecionar novo item".

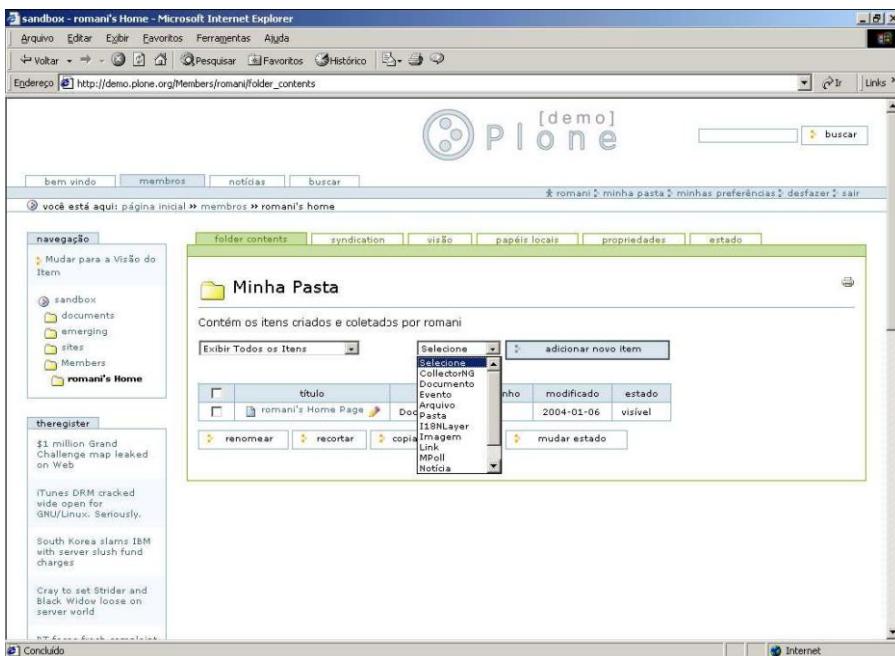


Fig. 1. Tela com a pasta de um usuário membro do Plone (Plone Foundation, 2005).

Com isso, uma página de edição é aberta e o usuário pode finalmente incluir o conteúdo (Fig. 2). A página de edição contém quatro abas. A primeira apresenta uma pré-visualização do conteúdo que está sendo editado, a segunda exibe o formulário de edição de conteúdo. A terceira aba mostra as propriedades associadas àquele conteúdo, como idioma, palavras chaves, formato, *copyright* e outras dependendo do tipo de conteúdo. A quarta permite mudar o estado do conteúdo dando início ao processo de publicação onde ele será revisado por um editor/revisor e finalmente publicado (Fig. 3).

O Plone permite que o gerente altere a aparência do *site*, inclua novos produtos, deixando o *site* personalizado. A questão de acessibilidade é considerada no Plone que segue padrões internacionais. Todo *site* desenvolvido nesse CMS já traz, *a priori*, as recomendações definidas pelo W3C Web Accessibility, como por exemplo possibilidade de alteração de tamanho do fonte, *hint* explicativo para imagens, etc. Também é possível criar uma cópia em disco de toda estrutura, aparência e conteúdo do *site* para que possa ser armazenada como *back-up* ou para ser importada em outro servidor Plone.

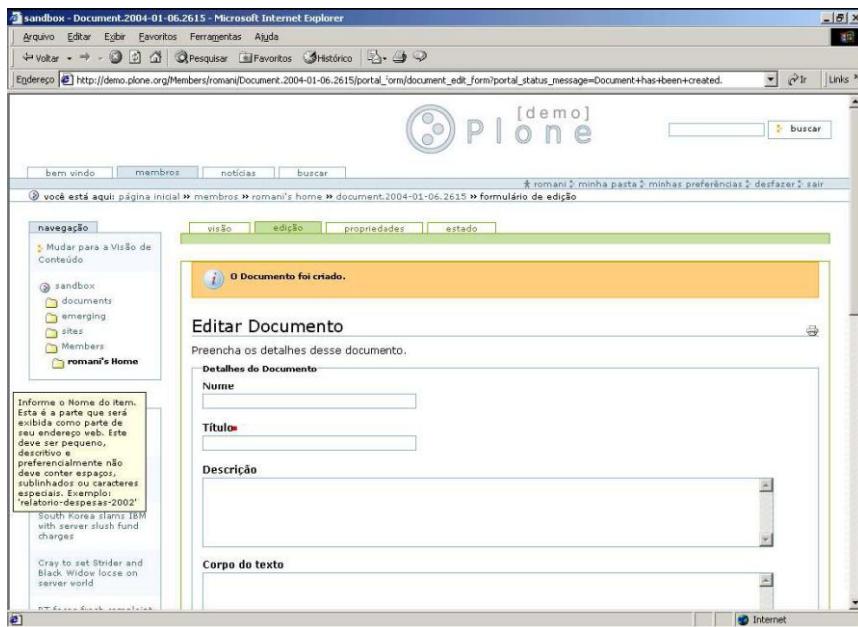


Fig. 2. Tela de edição de um conteúdo recém-criado no Plone (Plone Foundation, 2005).

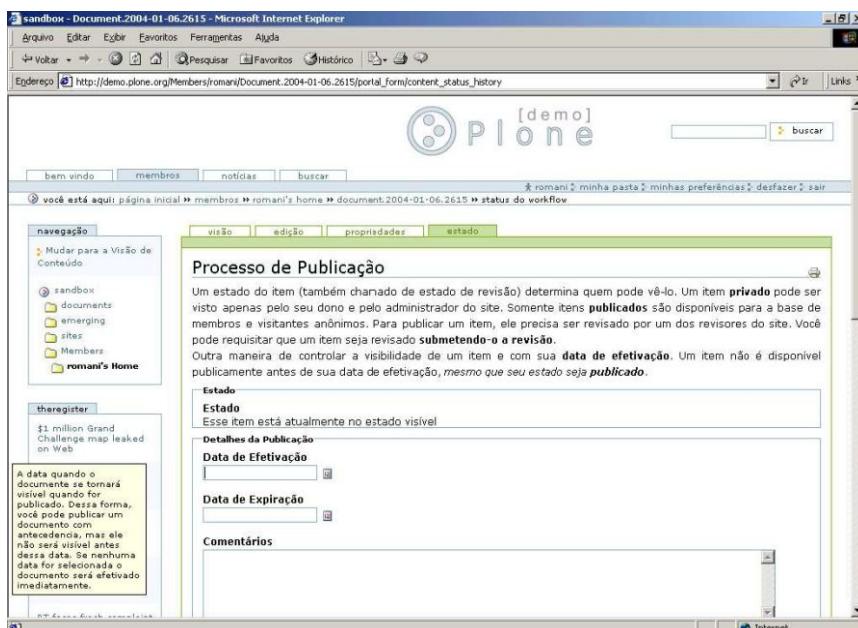


Fig. 3. Tela que permite alterar o estado do conteúdo no Plone (Plone Foundation, 2005).

## Comparativo de Plone com outros Gerenciadores

Como já mencionado anteriormente, o Plone é uma poderosa ferramenta para auxiliar na criação, gerenciamento e publicação de conteúdo na *web*. Possui uma série de funcionalidades e pode gerenciar vários *sites* simultaneamente. No entanto, existem outros CMSs disponíveis para uso que foram desenvolvidos em diferentes plataformas. A seguir é apresentado um quadro comparativo de Plone com gerenciadores desenvolvidos em PHP, Java e Python.

Foram selecionados os CMSs mais utilizados, já que a lista de CMSs é bastante extensa. Os tópicos sobre requisitos do sistema, segurança, suporte, facilidade de uso, performance, gerenciamento, interoperabilidade, flexibilidade e aplicações embutidas foram considerados nas comparações. Foi utilizada como referência parte da lista de quesitos do *site* CMSMatrix (CMSMatrix, 2005), no qual pode-se comparar até 10 CMSs diferentes simultaneamente. No entanto, foram feitas atualizações tanto nos quesitos quanto nas informações sobre os gerenciadores apresentados neste documento. Também foi incorporado o *software* GeCon que não consta da lista disponível no *site* do CMSMatrix (2005).

A Tabela 1 apresenta a comparação do CMS Plone com os CMSs PHP-Nuke, PostNuke e Xoops desenvolvidos em PHP.

**Tabela 1.** Comparação do Plone com PHP-Nuke, PostNuke e Xoops.

Plone versus Gerenciadores em PHP				
Site	Plone	PHP-Nuke	PostNuke	Xoops
Oficial Brasileiro	http://plone.org/ http://br.plone.org	http://phpnuke.org/ http://phpnuke.org.br	http://postnuke.com http://postnuke.pro.br	http://www.xoops.org http://xoopsbr.org/
Requisitos do sistema	Plone	PHP-Nuke	PostNuke	Xoops
Servidor de aplicação	Zope	mod_php	PHP 4.3.0 +	Nenhum
Custo Banco de Dados	Free ZODB, MySQL, Oracle, outros	Free MySQL, Postgres, mSQL, Interbase, Sybase	Free MySQL	Free MySQL
Licença Sistema Operacional Linguagem de programação	GNU GPL Qualquer um Python	GNU GPL Qualquer um PHP	GNU GPL Qualquer um PHP	GNU GPL Qualquer um PHP
Servidor Web	Apache, IIS, ZServer	Apache, IIS	Apache, IIS	Apache, IIS
Segurança	Plone	PHP-Nuke	PostNuke	Xoops
Mantém histórico das ações	Sim	Não	Pode ser adicionado	Não
Revisão de conteúdo	Sim	Não	Sim	Sim
Granularidade de privilégios	Sim	Limitado	Sim	Sim
Histórico de login	Pode ser adicionado	Não	Não	Pode ser adicionado
Notificação de problemas	Não	Não	Sim	Limitado

## Continuação da Tabela 1.

Inclusão de conteúdo no novo em modo privado	Sim	Não	Sim	Limitado
Gerenciamento de sessão	Pode ser adicionado	Não	Não	Limitado
Versioning	Sim	Sim	Sim	Não
<b>Supporte</b>				
Manuais comerciais	Sim	Não	Sim	Sim
Suporte comercial	Sim	Não	Sim	Sim
Treinamento comercial	Sim	Não	Sim	Não
Comunidade de Desenvolvedores	Sim	Sim	Sim	Sim
Ajuda on-line	Sim	Sim	Sim	Sim
Fóruns de discussão	Sim	Sim	Sim	Sim
Conferência usuário	Sim	Sim	Não	Não
<b>Facilidades de uso</b>				
e-mail para discussão	Pode ser adicionado	Não	Pode ser adicionado	Não
URLs amigáveis	Sim	Não	Sim	Pode ser adicionado
Linguagem Macro	Sim	Não	Não	Limitado
Suporte a idiomas	Sim	Sim	Sim	Sim
Corretor ortográfico	Pode ser adicionado	Não	Não	Não
Função de Undo	Sim	Não	Não	Não
Editor WYSIWYG	Sim	Não	Sim	Pode ser adicionado
<b>Gerenciamento</b>				
Clipboard	Sim	Não	Não	Sim
Programação de conteúdo	Sim	Não	Sim	Sim
Administração on-line	Sim	Sim	Sim	Sim
Subsites	Sim	Não	Sim	Pode ser adicionado
Temas e alteração de aparência	Sim	Sim	Sim	Sim
Lixeira	Pode ser adicionado	Não	Não	Não
Estatísticas Web	Pode ser adicionado	Sim	Sim	Pode ser adicionado
Gerenciamento de templates baseado na Web	Sim	Limitado	Sim	Sim
Workflow	Sim	Não	Pode ser adicionado	Não
<b>Interoperabilidade e Flexibilidade</b>				
Suporte a FTP	Sim	Não	Pode ser adicionado	Sim
Suporte a UTF-8	Sim	Não	Sim	Não
XHTML	Sim	Não	Sim	Não
Reuso de conteúdo	Sim	Não	Não	Pode ser adicionado
Extensão de Perfis de usuário	Sim	Não	Sim	Não
<b>Pacotes</b>				
Blog	Sim	Sim	Sim	Pode ser adicionado
Chat	Pode ser adicionado	Não	Pode ser adicionado	Pode ser adicionado
Formulário de contatos	Pode ser adicionado	Não	Sim	Limitado
Fórum	Sim	Sim	Pode ser adicionado	Sim

## Continuação da Tabela 1.

Calendário	Sim	Não	Pode ser adicionado	Pode ser adicionado
FAQ	Pode ser adicionado	Sim	Sim	Sim
Groupware	Pode ser adicionado	Não	Pode ser adicionado	Não
Help desk	Pode ser adicionado	Não	Pode ser adicionado	Não
Formulário de e-mail	Pode ser adicionado	Não	Sim	Sim
Newsletter	Pode ser adicionado	Não	Pode ser adicionado	Não
Galeria de fotos	Pode ser adicionado	Não	Pode ser adicionado	Pode ser adicionado
Mecanismo de busca	Sim	Não	Sim	Não
Mapa do site	Pode ser adicionado	Não	Pode ser adicionado	Não
Testes	Pode ser adicionado	Não	Pode ser adicionado	Não
Contribuições de usuários	Sim	Sim	Sim	Sim

A Tabela 2 mostra a comparação do CMS Plone com os CMSs Apache Lenya e OpenCMS desenvolvidos em Java/XML.

**Tabela 2.** Comparação do Plone com Apache Lenya e OpenCMS.

Plone versus Gerenciadores em Python/Zope			
Site	Plone	Apache Lenya	OpenCMS
Oficial	<a href="http://plone.org/">http://plone.org/</a>	<a href="http://lenya.apache.org/index.html">http://lenya.apache.org/index.html</a>	<a href="http://www.opencms.org">http://www.opencms.org</a>
Brasileiro	<a href="http://br.plone.org">http://br.plone.org</a>	---	----
Requisitos do sistema			
Servidor de aplicação	Zope	Vários tipos	Tomcat, Jboss, Bea Weblogic
Custo	Free	Free	Free
Banco de Dados	ZODB, MySQL, Oracle, outros	Qualquer um	MySQL, PostGreSQL, Oracle, MSSQL
Licença	GNU GPL	Apache-style	GNU LGPL
Sistema Operacional	Qualquer um	Qualquer um	Qualquer um
Linguagem de programação	Python	Java/XML/XSLT/ Javascript, JSP	Java 1.3+
Servidor Web	Apache, IIS, ZServer	Qualquer um	Apache, IIS
Segurança			
Mantém histórico das ações	Sim	Sim	Sim
Revisão de conteúdo	Sim	Sim	Sim
Granularidade de privilégios	Sim	Sim	Sim
Histórico de login	Pode ser adicionado	Sim	Sim
Notificação de problemas	Não	Sim	Sim
Inclusão de conteúdo no novo em modo privado	Sim	Sim	Sim
Gerenciamento de sessão	Pode ser adicionado	Sim	Não
Versioning	Sim	Sim	Sim

Continuação da Tabela 2.

<b>Supor te</b>	<b>Plone</b>	<b>Apache Lenya</b>	<b>OpenCMS</b>
Manuais comerciais	Sim	Não	Sim
Supor te comercial	Sim	Sim	Sim
Treinamento comercial	Sim	Sim	Sim
Comunidade de Desenvolvedores	Sim	Sim	Sim
Ajuda on-line	Não	Sim	Não
Fóruns de discussão	Sim	Sim	Sim
Conferência usuário	Sim	Sim	Não
<b>Facilidades de uso</b>	<b>Plone</b>	<b>Apache Lenya</b>	<b>OpenCMS</b>
e-mail para discussão	Pode ser adicionado	Não	Não
URLs amigáveis	Sim	Sim	Sim
Linguagem Macro	Sim	Não	Não
Supor te a idiomas	Sim	Sim	Sim
Corretor ortográfico	Pode ser adicionado	Não	Não
Função de Undo	Sim	Sim	Sim
Editor WYSIWYG	Sim	Sim	Sim
<b>Gerenciamento</b>	<b>Plone</b>	<b>Apache Lenya</b>	<b>OpenCMS</b>
Clipboard	Sim	Sim	Não
Programação de conteúdo	Sim	Sim	Sim
Administração on-line	Sim	Sim	Não
Subsites	Sim	Sim	Sim
Temas e alteração de aparência	Sim	Não	Não
Lixeira	Pode ser adicionado	Sim	Não
Estatísticas Web	Pode ser adicionado	Sim	Não
Gerenciamento de templates baseado na Web	Sim	Não	Limitado
Workflow	Sim	Sim	Não
<b>Interoperabilidade e Flexibilidade</b>	<b>Plone</b>	<b>Apache Lenya</b>	<b>OpenCMS</b>
Supor te a FTP	Sim	Não	Não
Supor te a UTF-8	Sim	Sim	Sim
XHTML	Sim	Sim	Limitado
Reuso de conteúdo	Sim	Sim	Sim
Extensão de Perfis de usuário	Sim	Não	Não
<b>Pacotes</b>	<b>Plone</b>	<b>Apache Lenya</b>	<b>OpenCMS</b>
Blog	Sim	Sim	Não
Chat	Pode ser adicionado	Não	Não
Formulário de contatos	Pode ser adicionado	Não	Não
Fórum	Sim	Não	Não
Calendário	Sim	Não	Custo extra
FAQ	Pode ser adicionado	Não	Custo extra
Groupware	Pode ser adicionado	Não	Não
Help desk	Pode ser adicionado	Não	Não
Formulário de e-mail	Pode ser adicionado	Não	Sim
Newsletter	Pode ser adicionado	Não	Não
Galeria de fotos	Pode ser adicionado	Não	Não
Mecanismo de busca	Sim	Sim	Sim
Mapa do site	Pode ser adicionado	Não	Sim
Testes	Pode ser adicionado	Não	Não
Contribuições de usuários	Sim	Não	Não

A Tabela 3 apresenta comparação do CMS Plone com os CMSs PyLucid e GeCon desenvolvidos em Python/Zope.

**Tabela 3.** Comparação do Plone com PyLucid e GeCon.

<i>Plone versus Gerenciadores em Python/Zope</i>			
<i>Site</i>	<i>Plone</i>	<i>PyLucid</i>	<i>GeCon</i>
Oficial Brasileiro	<a href="http://plone.org/">http://plone.org/</a> <a href="http://br.plone.org">http://br.plone.org</a>	<a href="http://www.pylucid.org/">http://www.pylucid.org/</a> ---	<a href="http://www.ead.unicamp.br/GECON">http://www.ead.unicamp.br/GECON</a> <a href="http://www.ead.unicamp.br/GECON">http://www.ead.unicamp.br/GECON</a>
<i>Requisitos do sistema</i>	<i>Plone</i>	<i>PyLucid</i>	<i>GeCon</i>
Servidor de aplicação	Zope	Apache e qualquer que suporte Python CGI	Zope
Custo	Free	Free	Free
Banco de Dados	ZODB, MySQL, Oracle, outros	MySQL	ZODB, MySQL, Oracle, outros
Licença	GNU GPL	GNU GPL	GNU GPL
Sistema Operacional	Qualquer um	Qualquer um	Qualquer um
Linguagem de programação	Python	Python	Python
Servidor Web	Apache, IIS, ZServer	Qualquer um	Apache, IIS, ZServer
<i>Segurança</i>	<i>Plone</i>	<i>PyLucid</i>	<i>GeCon</i>
Mantém histórico das ações	Sim	Não	Sim
Revisão de conteúdo	Sim	Não	Não
Granularidade de privilégios	Sim	Não	Sim
Histórico de login	Pode ser adicionado	Sim	Não
Notificação de problemas	Não	Não	Sim
Inclusão de conteúdo novo em modo privado	Sim	Não	Não
Gerenciamento de sessão	Pode ser adicionado	Sim	Não
Versioning	Sim	Não	Sim
<i>Suporte</i>	<i>Plone</i>	<i>PyLucid</i>	<i>GeCon</i>
Manuais comerciais	Sim	Não	Sim
Supporte comercial	Sim	Não	Não
Treinamento comercial	Sim	Não	Sim
Comunidade de Desenvolvedores	Sim	Sim	Não
Ajuda on-line	Não	Sim	Não
Fóruns de discussão	Sim	Sim	Não
Conferência usuário	Sim	Não	Não

Continuação da Tabela 3.

<b>Facilidades de uso</b>	<b>Plone</b>	<b>PyLucid</b>	<b>GeCon</b>
e-mail para discussão	Pode ser adicionado	Não	Não
URLs amigáveis	Sim	Sim	Não
Linguagem Macro	Sim	Sim	Sim
Suporte a idiomas	Sim	Não	Não
Corretor ortográfico	Pode ser adicionado	Não	Não
Função de Undo	Sim	Não	Não
Editor WYSIWYG	Sim	Não	Pode ser adicionado
<b>Gerenciamento</b>	<b>Plone</b>	<b>PyLucid</b>	<b>GeCon</b>
Clipboard	Sim	Não	Não
Programação de conteúdo	Sim	Não	Não
Administração online	Sim	Sim	Sim
Subsites	Sim	Não	Sim
Temas e alteração de aparência	Sim	Não	Sim
Lixeira	Pode ser adicionado	Sim	Não
Estatísticas Web	Pode ser adicionado	Não	Não
Gerenciamento de templates baseado na Web	Sim	Sim	Sim
Workflow	Sim	Não	Não
<b>Interoperabilidade e Flexibilidade</b>	<b>Plone</b>	<b>PyLucid</b>	<b>GeCon</b>
Suporte a FTP	Sim	Não	Sim
Suporte a UTF-8	Sim	Sim	Sim
XHTML	Sim	Sim	Sim
Reuso de conteúdo	Sim	Sim	Sim
Extensão de Perfis de usuário	Sim	Não	Sim
<b>Pacotes</b>	<b>Plone</b>	<b>PyLucid</b>	<b>GeCon</b>
Blog	Sim	Não	Não
Chat	Pode ser adicionado	Não	Pode ser adicionado
Formulário de contatos	Pode ser adicionado	Não	Pode ser adicionado
Fórum	Sim	Não	Pode ser adicionado
Calendário	Sim	Não	Pode ser adicionado
FAQ	Pode ser adicionado	Não	Pode ser adicionado
Groupware	Pode ser adicionado	Não	Pode ser adicionado
Help desk	Pode ser adicionado	Não	Pode ser adicionado
Formulário de e-mail	Pode ser adicionado	Não	Pode ser adicionado
Newsletter	Pode ser adicionado	Não	Pode ser adicionado
Galeria de fotos	Pode ser adicionado	Não	Pode ser adicionado
Mecanismo de busca	Sim	Sim	Pode ser adicionado
Mapa do site	Pode ser adicionado	Sim	Sim
Testes	Pode ser adicionado	Não	Pode ser adicionado
Contribuições de usuários	Sim	Não	Sim

Analizando as Tabelas de comparação dos vários CMSs, aqueles escritos em Python/Zope contemplam a maioria dos quesitos. Além disso, é possível incorporar produtos ou *plugins* para torná-los mais completos. Os gerenciadores desenvolvidos em Java também possuem muitos recursos mas ainda não existem pacotes e/ou produtos que possam ser adicionados. Apesar de muito utilizados, os CMSs em PHP não possuem tantos recursos como os demais, além de problemas com segurança devido a característica de não-verificação de tipos da linguagem PHP.

Antes da adoção de um CMS é importante considerar e analisar as características e recursos disponíveis para cada um (Romani et al., 2003). De posse dos requisitos necessários para criação do *site* e observando as comparações entre os CMSs mais utilizados, é possível identificar aquele que melhor se adeque às necessidades da empresa.

## Conclusões

1. O mercado apresenta uma grande quantidade e variedade de CMSs o que torna difícil a seleção de um gerenciador de conteúdos que seja mais adequado para cada tipo de *site*.
2. Estudos comparativos e análises das características e recursos dos CMSs são ferramentas importantes que auxiliam o gerente na tomada de decisão.
3. Plone é um forte candidato para empresas de médio e grande porte nas quais o trabalho de atualização do site seja realizado por muitas pessoas ou que possuam *sites* mais complexos ou mais dinâmicos.
4. O Plone se destaca por possuir uma máquina de *workflow* bem elaborada que permite gerenciar todo o fluxo de criação, revisão e publicação de documentos.
5. O Plone também possui características e recursos relativos à acessibilidade, fator muito importante para auxiliar na inclusão digital.

## Referências Bibliográficas

ALKACON SOFTWARE GMBH – THE OPENCMS EXPERTS. **OpenCMS. Org:** OpenCms - The open source Content Management System. Disponível em: <<http://www.opencms.org>>. Acesso em: ago. 2005.

APACHE SOFTWARE FOUNDATION. **Apache Lenya - Open Source Content Management (Java/XML).** Disponível em: <<http://lenya.apache.org/index.html>>. Acesso em: set. 2005.

DRUPAL. **Drupal.org:** community plumbing. Disponível em: <<http://drupal.org>>. Acesso em: out. 2005.

ENONIC. **Enonic - the vertical site product range.** Disponível em: <<http://www.enonic.com/products>>. Acesso em: out. 2005.

EZ SYSTEMS AS. **eZ.no/br – content management system eZ publish (Open Source CMS).** Disponível em: <<http://ez.no/br>>. Acesso em: out. 2005.

GUPPY. **GuppY [home page].** Disponível em: <<http://www.freeguppy.org>>. Acesso em: out. 2005.

J2EEBRASIL. **J2EEBrasil - o site da comunidade J2EE no Brasil:** tutoriais - Java server pages e servlets. Disponível em: <<http://www.jspbrasil.com.br/jsp/tutoriais/tutorial.jsp>>. Acesso em: 04 out. 2005.

JOOMLA! ACCESSIBILITY STATEMENT. **Joomla!** Disponível em: <<http://www.joomla.org>>. Acesso em: out. 2005.

LERDORF, R. **Do you PHP?.** Disponível em: <[http://www.oracle.com/technology/pub/articles/php\\_experts/rasmus\\_php.html](http://www.oracle.com/technology/pub/articles/php_experts/rasmus_php.html)>. Acesso: 04 out. 2005.

MIRO INTERNATIONAL PTY LTD. **Mamboserver.com - home.** Disponível em: <<http://www.mamboserver.com>>. Acesso em: out. 2005.

McKAY, A. **The definitive guide to Plone.** Disponível em: <<http://docs.neuroinf.de/PloneBook>>. Acesso em: 30 set. 2005.

PHP GROUP. **PHP: manual do PHP.** Disponível em: <[http://www.php.net/manual/pt\\_BR/index.php](http://www.php.net/manual/pt_BR/index.php)>. Acesso: 03 out. 2005.

**PHP-NUKE. PHP-Nuke [home page].** Disponível em:  
<<http://www.phpnuke.org/>>. Acesso em: 05 out. 2005.

**PLAIN BLACK CORPORATION. CMSMATRIX.** Disponível em:  
<<http://www.cmsmatrix.org>>. Acesso em: set. 2005.

**PLONE FOUNDATION. Plone:** a user-friendly and powerful open source content management system. Disponível em: <<http://plone.org/>> Acesso em: ago. 2005.

**POKORNY, J.** Static pages are dead: how a modular approach is changing interaction design. **Interactions**, New York, v. 8, n. 5, p.19-24, 2001.

**POSTNUKE. COM. Postnuke:** navegation. Disponível em:  
<<http://www.postnuke.com>>. Acesso em: out. 2005.

**PYLUCID CMS. PyLucid – a OpenSource CMS written in pure Python.** Disponível em: <<http://www.pylucid.org/>>. Acesso em: out. 2005.

**ROBERTSON, J.** **So, what is a content management system?** [Sydney]: Step Two Designs, 2003. 5 p. Disponível em:  
<[http://www.steptwo.com.au/papers/kmc\\_what/](http://www.steptwo.com.au/papers/kmc_what/)>. Acesso em: ago. 2003.

**ROMANI, L. A. S.; EVANGELISTA, S. R. M.; KASHIWAGURA, F.** **Diretrizes para escolha de um sistema de gerenciamento de conteúdos.** Campinas: Embrapa Informática Agropecuária, 2003. 5 p. (Embrapa Informática Agropecuária. Comunicado Técnico 53). Disponível em:  
<<http://www.cnptia.embrapa.br/publica/2003/comtec53.pdf>>. Acesso em: nov. 2005.

**ROMANI, R.** **GECON:** um gerenciador de conteúdos na Web para não especialistas em informática. 2004. 155 p. Dissertação (Mestrado em Ciência da Computação) – Universidade Estadual de Campinas, Campinas.

**ROSSUM, G. van; DRAKE JUNIOR, F. L.** **Tutorial de Python:** release 2.1. [S. l.]: Python Software Foundation, 2004. 66 p. Disponível em:  
<<http://www.gpr.com.br/download/python21.pdf>>. Acesso em: 29 set. 2005.

**XOOPS. Official XOOPS website.** Disponível em: <<http://www.xoops.org>>. Acesso em: set. 2005.

**ZEIGER, S.** **Servlets essentials:** version 1.3.5. Disponível em:  
<<http://www.novocode.com/doc/servlet-essentials/>>. Acesso em: 04 out. 2005.

**ZOPE CORPORATION. Zope.org.** Disponível em: <<http://www.zope.org>>. Acesso em: set. 2005.



---

*Informática Agropecuária*

Ministério da Agricultura,  
Pecuária e Abastecimento

