

**UNIVERSIDADE FEDERAL DE SANTA MARIA  
CENTRO DE TECNOLOGIA  
CURSO DE CIÊNCIA DA COMPUTAÇÃO**

**GERENCIAMENTO DE CONTEÚDO NA  
WEB USANDO PLONE: APLICAÇÃO  
AO PORTAL DA INFORMÁTICA DA  
UFSM**

**TRABALHO DE GRADUAÇÃO**

**Diego Ribas Adiers**

**Santa Maria, RS, Brasil**

**2007**

**GERENCIAMENTO DE CONTEÚDO NA WEB  
USANDO PLONE: APLICAÇÃO AO PORTAL DA  
INFORMÁTICA DA UFSM**

**por**

**Diego Ribas Adiers**

Trabalho de Graduação apresentado ao Curso de Ciência da Computação  
da Universidade Federal de Santa Maria (UFSM, RS), como requisito  
parcial para a obtenção do grau de  
**Bacharel em Ciência da Computação**

**Orientador: Prof<sup>a</sup> Andrea Schwertner Charão**

**Trabalho de Graduação N. 218**

**Santa Maria, RS, Brasil**

**2007**

**Universidade Federal de Santa Maria  
Centro de Tecnologia  
Curso de Ciência da Computação**

A Comissão Examinadora, abaixo assinada,  
aprova o Trabalho de Graduação

**GERENCIAMENTO DE CONTEÚDO NA WEB USANDO PLONE:  
APLICAÇÃO AO PORTAL DA INFORMÁTICA DA UFSM**

elaborado por  
**Diego Ribas Adiers**

como requisito parcial para obtenção do grau de  
**Bacharel em Ciência da Computação**

**COMISSÃO EXAMINADORA:**

**Prof<sup>a</sup> Andrea Schwertner Charão**  
(Presidente/Orientador)

**Prof<sup>a</sup> Roseclea Duarte Medina (UFSM)**

**Prof. Cláudio Rocha Lobato (UFSM)**

Santa Maria, 28 de Fevereiro de 2007.

*“A felicidade está nas coisas simples da vida.”*

— AUTOR DESCONHECIDO

## **AGRADECIMENTOS**

Agradeço a minha família e a todos que me ajudaram pelo apoio e compreensão.

# RESUMO

Trabalho de Graduação  
Curso de Ciência da Computação  
Universidade Federal de Santa Maria

## **GERENCIAMENTO DE CONTEÚDO NA WEB USANDO PLONE: APLICAÇÃO AO PORTAL DA INFORMÁTICA DA UFSM**

Autor: Diego Ribas Adiers

Orientador: Prof<sup>a</sup> Andrea Schwertner Charão

Local e data da defesa: Santa Maria, 28 de Fevereiro de 2007.

Uma organização sem uma referência na Internet é difícil de se imaginar nos dias de hoje. A maioria das organizações usam seu *site* Web para comunicação com seus colaboradores, publicidade, publicação de notícias, eventos, etc. Nestas situações, há um problema em comum que é o gerenciamento de todo o conteúdo publicado. Esse problema pode custar muito tempo e esforço das pessoas envolvidas nesse gerenciamento. Para suprir as necessidades de fácil e rápido gerenciamento desses portais, surgiram ferramentas conhecidas como Sistemas de Gerenciamento de Conteúdo (*Content Management System* – CMS). Um CMS é basicamente uma aplicação Web designada a auxiliar os usuários não-técnicos a adicionar, editar e gerenciar colaborativamente o conteúdo em um *site* Web. O Plone é um exemplo de CMS de código aberto, que tem como objetivo ser flexível, de fácil uso e instalação. Neste trabalho, apresenta-se um estudo deste CMS com vistas à sua futura aplicação no portal da Informática da UFSM. Discute-se também o desenvolvimento de aplicações Web integradas ao Plone, apresentando-se exemplos inseridos na realidade da instituição.

**Palavras-chave:** Gerenciamento de conteúdo web; Plone; Zope; portal; CMS; Informática UFSM.

# **ABSTRACT**

Trabalho de Graduação  
Curso de Ciência da Computação  
Universidade Federal de Santa Maria

## **WEB CONTENT MANAGEMENT USING PLONE: APPLICATION TO THE INFORMÁTICA UFSM PORTAL**

Author: Diego Ribas Adiers  
Advisor: Prof<sup>a</sup> Andrea Schwertner Charão

An organization without a reference on the Internet is difficult to imagine nowadays. Most organizations use their Web site for communication with collaborators, advertising, news publication, events, etc. In all these portals, there is a recurrent problem concerning the efficient management of all Web content. Indeed, this problem can take much time and effort from people involved in such process. To provide an easy and fast management of Web portals, a number of Content Management Systems (CMS) has been developed in the last few years. A CMS is basically a Web application which allows non-technical users to add, to edit and to manage content in a Web site. Plone is an example of open-source CMS designed to be flexible and easy to use and install. This work analyzes Plone as a future replacement for the current Web site of Informática UFSM. It also presents the process of developing applications for Plone, using examples based on real situations in the institution.

**Keywords:** web content management; Plone; Zope; portal; CMS; Informática UFSM.

## LISTA DE FIGURAS

Figura 2.1 – Arquitetura do servidor de aplicações Zope .....	18
Figura 2.2 – Exemplo de <i>site</i> Plone .....	20
Figura 2.3 – Exemplo de <i>site</i> Plone personalizado .....	21
Figura 2.4 – Tipos de conteúdo na instalação padrão do Plone .....	23
Figura 2.5 – Interface de administração/configuração de um <i>site</i> Plone .....	23
Figura 3.1 – Atual portal da Informática UFSM .....	26
Figura 3.2 – Cadastro/edição de uma notícia no Plone .....	28
Figura 3.3 – Visualização de uma notícia publicada no Plone .....	29
Figura 3.4 – Interface de gerência de permissões do Plone .....	30
Figura 3.5 – Resultado de uma busca no Plone .....	31
Figura 3.6 – Interface de busca avançada no Plone .....	31
Figura 3.7 – Processo de desenvolvimento de aplicações para o Plone .....	33
Figura 3.8 – Modelo UML da aplicação de gerência de informações sobre computadores da rede .....	34
Figura 3.9 – Interface de cadastro de computadores da aplicação exemplo .....	35
Figura 3.10 – Interface de cadastro de locais/salas da aplicação exemplo .....	36
Figura 3.11 – Visualização da tabela Aluno criada no MySQL dentro da ZMI .....	37
Figura 3.12 – Criação de um Z SQL Method dentro da ZMI .....	38
Figura 3.13 – Código SQL que consulta alunos .....	38
Figura 3.14 – Código ZPT que gera o formulário de pesquisa .....	39
Figura 3.15 – Interface de pesquisa de alunos no MySQL .....	39
Figura 3.16 – Código ZPT que mostra o resultado da pesquisa .....	41
Figura 3.17 – Resultado de uma pesquisa por alunos no MySQL .....	42
Figura 3.18 – Código ZPT que gera o formulário de cadastro de aluno .....	43
Figura 3.19 – Interface de cadastro de alunos no MySQL .....	44
Figura 3.20 – <i>Script</i> Python que processa o formulário de cadastro de aluno .....	44
Figura 3.21 – Código ZPT que mostra a resposta da inserção .....	45
Figura 3.22 – Visualização dos arquivos dentro da ZMI .....	45

## **LISTA DE TABELAS**

Tabela 2.1 – Comparativo entre alguns CMS populares .....	17
---	----

## **LISTA DE ABREVIATURAS E SIGLAS**

CMF	Content Management Framework
CMS	Content Management System
GPL	General Public License
IIS	Microsoft Internet Information Services
LDAP	Lightweight Directory Access Protocol
SGC	Sistema de Gestão de Conteúdo
SQL	Structured Query Language
UML	Unified Modeling Language
WCMS	Web Content Management System
XMI	XML Metadata Interchange
XML	Extensible Markup Language
ZMI	Zope Management Interface
Zope	Z Object Publishing Environment
ZPL	Zope Public License
ZPT	Zope Page Template

# SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b> .....	12
<b>2</b>	<b>GESTÃO DE CONTEÚDO</b> .....	14
<b>2.1</b>	<b>Sistemas de gestão de conteúdo</b> .....	15
2.1.1	Características e funcionalidades .....	15
2.1.2	Ferramentas .....	16
<b>2.2</b>	<b>Zope</b> .....	18
2.2.1	Arquitetura .....	18
2.2.2	Recursos .....	19
<b>2.3</b>	<b>Plone</b> .....	19
2.3.1	Interface .....	20
2.3.2	Recursos e características .....	21
<b>3</b>	<b>APLICAÇÃO AO PORTAL DA INFORMÁTICA UFSM</b> .....	25
<b>3.1</b>	<b>O <i>site</i> atual da Informática UFSM</b> .....	25
<b>3.2</b>	<b>O portal da Informática com o uso do Plone</b> .....	27
3.2.1	Publicação de conteúdo .....	27
3.2.2	Indexação e pesquisa de conteúdo .....	30
<b>3.3</b>	<b>Desenvolvimento de aplicações para o Plone</b> .....	32
3.3.1	Processo de desenvolvimento .....	32
3.3.2	Exemplo: gerenciador de informações sobre computadores da rede .....	33
3.3.3	Exemplo: gerência de alunos usando MySQL .....	36
<b>4</b>	<b>CONCLUSÃO</b> .....	46
	<b>REFERÊNCIAS</b> .....	48
	<b>APÊNDICE A PERGUNTAS FREQUENTES SOBRE PLONE</b> .....	51

# 1 INTRODUÇÃO

O volume e a importância das informações digitais têm aumentado muito nos últimos anos. Esse fato é provocado pela popularização da Internet e principalmente da World Wide Web, além das facilidades do consumo e produção de informações neste ambiente. Como exemplo dessa popularização tem-se os grandes portais de conteúdo que possuem milhares de assinantes, onde as notícias do mundo inteiro são publicadas de minutos em minutos. Esses portais servem como repositório de informações de quase todo tipo, como textos, vídeos, fotos, etc. Tais portais necessitam de atualização de conteúdo constante.

A maioria dos portais segue o modelo de gestão centrado em um *webmaster*, que é responsável em gerenciar todo conteúdo do *site*. Quando o volume das informações a serem gerenciadas cresce, como é o caso nesses grandes portais, o *webmaster* sofre uma sobrecarga a ponto de comprometer a gestão eficiente do conteúdo do *site*. Para tentar contornar esse problema surgiram os sistemas de gestão de conteúdo (CMS) (BAX; PEREIRA, 2002).

Um CMS (*Content Management System*) é basicamente uma aplicação Web projetada para auxiliar os usuários não-técnicos a adicionar, editar e gerenciar conteúdo em um Web *site* de forma rápida e simples. Dessa forma, os responsáveis pela produção do conteúdo do portal podem facilmente gerenciar o conteúdo de forma colaborativa e distribuída. Dentre os principais objetivos de um CMS tem-se a separação entre conteúdo e apresentação, a aplicação de regras que definem quem pode publicar o quê e quando, e a pesquisa e indexação de informações de forma inteligente. Existem diversas ferramentas para gestão conteúdo atualmente, sendo que o CMS Plone (PLONE FOUNDATION, 2007) é uma das mais populares alternativas de Software Livre.

O *site* da Informática da UFSM<sup>1</sup> reúne muitas informações e segue hoje um mod-

---

<sup>1</sup>Informática UFSM: <http://www.inf.ufsm.br>

elo centralizado de publicação de conteúdo. Uma pessoa que deseje incluir conteúdo no *site* deve repassar as informações ao *webmaster*, sugerindo onde elas devem ser incluídas. Em seguida, o *webmaster* avalia e inclui o novo conteúdo no *site*. A remoção do conteúdo também é feita manualmente, sendo que é comum a permanência de notícias desatualizadas no *site*.

Para substituir esse modelo centrado no *webmaster* e facilitar o gerenciamento do conteúdo, vem-se cogitando a utilização de um CMS para transformar o *site* atual em um portal Web que integre diferentes aplicações e que possa ter suas informações atualizadas de forma colaborativa. Um dos objetivos desse trabalho é, portanto, mostrar as vantagens da utilização do CMS Plone em relação ao modelo de gestão centralizado atualmente utilizado no *site* da Informática, fornecendo subsídios para uma futura implantação do portal da Informática UFSM. O outro objetivo do trabalho é demonstrar e discutir o desenvolvimento de aplicações para o Plone, que possam ser úteis aos alunos e professores.

Esse trabalho é dividido basicamente em 2 partes. A primeira parte é uma introdução à gestão de conteúdo e às ferramentas usadas neste trabalho. A segunda parte descreve como o *site* da Informática é hoje e as vantagens que poderiam ser alcançadas com a implantação do CMS Plone. Ainda nesta parte, demonstra-se o processo de desenvolvimento de aplicações para o Plone, descrevendo-se a implementação de aplicações que possam ser úteis à instituição.

## 2 GESTÃO DE CONTEÚDO

A informação hoje em dia é produzida, consumida e disseminada num ritmo veloz. Essa afirmação é comprovada pela existência bem-sucedida de grandes portais tais como Globo.com<sup>1</sup>, Terra<sup>2</sup>, Uol<sup>3</sup>, entre outros. Para que esses portais satisfaçam seus usuários, é imprescindível o emprego de uma solução automatizada de gestão que permita aos diversos jornalistas publicar informação de forma fácil e rápida, independentemente de onde estejam.

No contexto acima delineado, a gestão de conteúdo torna-se de grande importância não só para portais de notícias, mas também para instituições de ensino, grandes e médias empresas (HAZRA, 2002; WEITZMAN et al., 2002). De fato, *"com os inúmeros avanços tecnológicos, muitas instituições sofrem com o excesso de informações, sendo imprescindível a aplicação da gestão da informação para administrar esse caos informacional do mundo digital."* (DIAS, 2001). Assim, a gestão de conteúdo é aplicável a qualquer organização que possua muita informação em seus *sites* e precise mantê-los sempre atualizados, permitindo buscas avançadas e incluindo aplicações que facilitem as tarefas do dia-a-dia dos seus colaboradores e usuários. Essa é a principal razão do surgimento de sistemas para gestão de conteúdo.

O termo "gestão de conteúdo" relaciona-se a um amplo conjunto de áreas e tecnologias, incluindo gestão do conhecimento, gestão do arquivamento e ciclo de vida de documentos, gestão de conteúdo Web, gestão de mensagens e correio eletrônico, gestão de fluxos de trabalho e processos de negócio, portais, pesquisa, equipes e colaboração (BAX; PEREIRA, 2002; LERNER, 2003; BALASUBRAMANIAN et al., 1998).

Para cada uma dessas áreas existem diferentes tipos de ferramentas de gestão. Dada

---

<sup>1</sup>Portal Globo.com: <http://globo.com>

<sup>2</sup>Portal Terra: <http://www.terra.com.br>

<sup>3</sup>Universo Online: <http://www.uol.com.br>

a amplitude do tema, este trabalho se limita aos sistemas de gestão de conteúdo Web ou WCMS (*Web Content Management System*)<sup>4</sup>, com ênfase na ferramenta Plone (PLONE FOUNDATION, 2007; ENFOLDSYSTEMS, 2005).

O processo de gestão de conteúdo pode ser representado por três etapas básicas: criação, gestão e publicação (BAX; PEREIRA, 2002). A etapa de gestão abrange as etapas de indexação e controle de qualidade, revisão, arquivamento e eliminação de conteúdo. Um CMS tem como função ajudar os usuários em cada uma dessas etapas, através de uma interface simples e de fácil uso.

Na próxima seção apresenta-se algumas características e exemplos de ferramentas de gestão de conteúdo. Em seguida, apresenta-se a plataforma Zope (ZOPE CORPORATION, 2007; PELLETIER; LATTEIER, 2005) e a ferramenta Plone para gestão de conteúdo Web. É importante ressaltar que Plone depende de Zope, o que justifica a descrição desta plataforma neste trabalho.

## **2.1 Sistemas de gestão de conteúdo**

Um CMS (*Content Management System*) ou SGC (Sistema de Gestão de Conteúdo) é uma ferramenta que permite que várias pessoas criem, editem, gerenciem e finalmente publiquem uma variedade de conteúdos na Web, como texto, gráficos, vídeo, documentos, etc. Isso ocorre seguindo uma lista de regras, processos e fluxos de trabalho centralizados que garantem coerência e validade do conteúdo eletrônico (BAX; PEREIRA, 2002).

### **2.1.1 Características e funcionalidades**

A lista abaixo enumera as principais características e funcionalidades de um CMS:

- prepara o conteúdo de uma página da sua apresentação;
- permite certos usuários a adicionar e editar conteúdo;
- aplica regras sobre quem pode o quê e quando;
- permite criação descentralizada de conteúdo;
- possibilita aplicação de regras de negócio ao conteúdo;
- possibilita pesquisa e indexação de informação inteligentemente.

---

<sup>4</sup>Ao longo deste texto, utiliza-se as siglas WCMS e CMS indistintamente para designar sistemas de gestão de conteúdo Web.

Essas são algumas características desejáveis em um CMS, mas nem todos os sistemas as possuem. A escolha da ferramenta de gestão certa depende da realidade onde ela será aplicada. A seção seguinte apresenta brevemente algumas ferramentas de gestão de conteúdo.

### 2.1.2 Ferramentas

Existe atualmente no mercado um grande número de ferramentas de gestão de conteúdo. Neste meio, algumas soluções de código aberto vêm se tornando populares, tais como Mambo (MAMBO FOUNDATION, 2007), XOOPS (XOOPS PROJECT, 2007; ATWAL, 2006), PHP-Nuke (PHPNUKE.ORG, 2007), Joomla (Joomla TEAM, 2007; GRAF, 2007) e Plone. Para esse trabalho foi escolhida a ferramenta Plone devido a vários fatores, incluindo maturidade, flexibilidade, documentação, experiência de uso e licença. Além disso, sabe-se que esta ferramenta vem sendo empregada em diversos casos de sucesso no Brasil (COELHO, 2004), como por exemplo o portal Interlegis<sup>5</sup> e o portal Serpro<sup>6</sup>.

Um recurso para comparação de ferramentas de gestão de conteúdo é o portal CMS Matrix<sup>7</sup>, que mantém um banco de dados com características detalhadas sobre diversas ferramentas. Dentre as características que podem ser comparadas tem-se segurança, suporte, requisitos de sistema, facilidade de uso, desempenho, interoperabilidade, flexibilidade, entre outros.

A Tabela 2.1, gerada a partir do portal CMS Matrix e adaptada para mostrar as características mais relevantes, compara algumas características de alguns CMS populares. Nesta tabela, observa-se que o Plone é uma ferramenta que possui a maioria das características desejadas em um CMS. Outra ferramenta que se destaca na comparação é o Joomla. A ferramenta que menos se destaca nessa comparação é o PHP Nuke, que não apresenta alguns recursos que as outras ferramentas possuem.

É importante ressaltar que cada ferramenta possui suas vantagens e desvantagens, logo, a escolha da ferramenta certa depende especificamente para que finalidade ela será usada. Dessa forma, a Tabela 2.1 apenas ilustra características de alguns gerenciadores de conteúdo, e deve ser interpretada especificamente para o projeto de gestão de conteúdo

---

<sup>5</sup>Portal dos poderes legislativos federal, estaduais e municipais: <http://www.interlegis.gov.br>

<sup>6</sup>Portal do Serviço Federal de Processamento de Dados: <http://www.serpro.gov.br>

<sup>7</sup>CMS Matrix: <http://www.cmsmatrix.org>

Tabela 2.1: Comparativo entre alguns CMS populares

<b>Requisitos</b>	<b>Joomla</b>	<b>Mambo</b>	<b>PHP Nuke</b>	<b>Plone</b>	<b>Xoops</b>
servidor Web	Apache	Apache	Apache	Zope	Apache
banco dados	MySQL	MySQL	MySQL	ZODB	MySQL
linguagem	PHP	PHP	PHP	Python	PHP
sistema operacional	qualquer	qualquer	qualquer	qualquer	qualquer
<b>Segurança</b>	<b>Joomla</b>	<b>Mambo</b>	<b>PHP Nuke</b>	<b>Plone</b>	<b>Xoops</b>
auditoria	não	não	não	sim	limitada
<i>sandBox</i>	não	não	não	sim	sim
versionamento	sim	limitado	não	sim	sim
autenticação LDAP	<i>plug-in</i>	sim	não	<i>plug-in</i>	sim
<b>Facilidade de Uso</b>	<b>Joomla</b>	<b>Mambo</b>	<b>PHP Nuke</b>	<b>Plone</b>	<b>Xoops</b>
URLs amigáveis	sim	sim	não	sim	<i>plug-in</i>
<i>upload</i> massivo	não	não	não	sim	<i>plug-in</i>
<i>wizard</i>	não	não	não	não	não
níveis interface	não	sim	não	sim	sim
linguagem de <i>template</i>	sim	sim	não	sim	sim
desfazer	não	sim	não	sim	limitado
<b>Desempenho</b>	<b>Joomla</b>	<b>Mambo</b>	<b>PHP Nuke</b>	<b>Plone</b>	<b>Xoops</b>
<i>caching</i> avançado	sim	sim	não	sim	não
balanceamento carga	não	não	não	sim	sim
<b>Gerenciamento</b>	<b>Joomla</b>	<b>Mambo</b>	<b>PHP Nuke</b>	<b>Plone</b>	<b>Xoops</b>
administração <i>online</i>	sim	sim	sim	sim	sim
ferramenta <i>workflow</i>	não	não	não	sim	não
estatísticas Web	sim	sim	sim	<i>plug-in</i>	<i>plug-in</i>
<b>Interoperabilidade</b>	<b>Joomla</b>	<b>Mambo</b>	<b>PHP Nuke</b>	<b>Plone</b>	<b>Xoops</b>
RSS	sim	sim	não	sim	sim
suporte FTP	<i>plug-in</i>	não	não	sim	sim
XHTML	não	sim	não	sim	não
<b>Flexibilidade</b>	<b>Joomla</b>	<b>Mambo</b>	<b>PHP Nuke</b>	<b>Plone</b>	<b>Xoops</b>
suporte CGI	não	não	sim	<i>plug-in</i>	não
vários <i>sites</i>	<i>plug-in</i>	<i>plug-in</i>	não	sim	não
metadados	sim	sim	não	sim	<i>plug-in</i>
<b>Aplicações</b>	<b>Joomla</b>	<b>Mambo</b>	<b>PHP Nuke</b>	<b>Plone</b>	<b>Xoops</b>
fórum	sim	sim	sim	sim	sim
gerência documentos	<i>plug-in</i>	<i>plug-in</i>	não	sim	<i>plug-in</i>
calendário eventos	<i>plug-in</i>	<i>plug-in</i>	não	sim	<i>plug-in</i>
galeria fotos	<i>plug-in</i>	<i>plug-in</i>	não	<i>plug-in</i>	<i>plug-in</i>
motor busca	sim	sim	não	sim	sim

Fonte: CMS Matrix, disponível em <http://www.cmsmatrix.org> (adaptação).

desejado. Também é importante ressaltar que as características nessa tabela podem mudar conforme a evolução no desenvolvimento desses sistemas de gestão de conteúdo.

Como o objetivo desse trabalho é analisar as possibilidades de aplicação do Plone em

relação ao sistema atual usado pela Informática UFSM, as seções seguintes discutem mais detalhadamente o servidor de aplicações Zope e o WCMS Plone.

## 2.2 Zope

Zope (*Z Object Publishing Environment*) (ZOPE CORPORATION, 2007; PELLETIER; LATTEIER, 2005) é um servidor de aplicações que provê um ambiente para publicação de objetos na Web. Trata-se de uma ferramenta de código aberto, escrita na linguagem Python. Zope permite realizar tarefas de administração do servidor através de uma interface Web denominada *Zope Management Interface* (ZMI). Os objetos que o Zope publica na Web são escritos em linguagem Python, e tipicamente armazenados num banco de dados orientado a objetos, o *Zope Object Database* (ZODB), que é integrado ao sistema. Objetos básicos, tais como documentos, imagens e *templates* (modelos de páginas) podem ser criados ou modificados via Web. Além disso, o Zope é um produto multiplataforma, ou seja, funciona em Unix, Linux, Mac OS e Windows. Atualmente está sob a licença ZPL (*Zope Public License*), uma licença compatível com a GPL (*General Public License*).

As seções seguintes apresentam a arquitetura do Zope e detalham alguns dos seus recursos mais relevantes para este trabalho.

### 2.2.1 Arquitetura

O Zope possui uma arquitetura flexível constituída basicamente de um servidor Web, um servidor de banco de dados, Zclasses e produtos pré-construídos. A Figura 2.1 ilustra esta arquitetura que em seguida é discutida componente a componente.

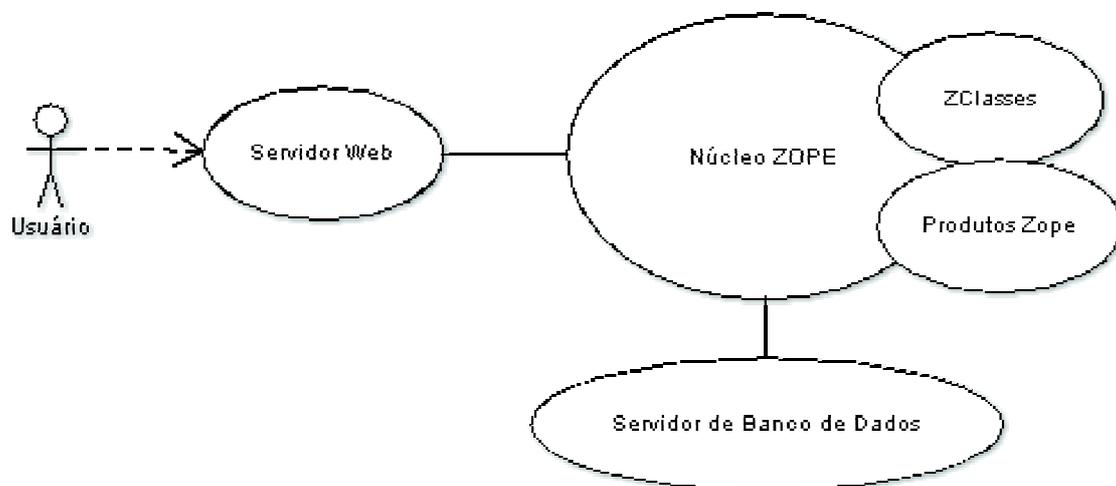


Figura 2.1: Arquitetura do servidor de aplicações Zope

- **Servidor Web:** Zope possui seu próprio servidor Web chamado ZServer, mas permite também o uso de outro servidor tal como Apache ou Microsoft IIS (*Internet Information Server*).
- **Servidor de banco de dados:** Zope possui seu próprio servidor de banco de dados orientado a objetos (ZODB). Também é possível integrá-lo a outros banco de dados relacionais como Oracle, PostgreSQL, Sybase, MySQL, entre outros.
- **ZClasses:** consistem em modelos para novos objetos no Zope. Um objeto Zope pode ser por exemplo uma pasta, um documento, um método, etc. As ZClasses podem ser criadas e ampliadas via interface Web (ZMI).
- **Produtos Zope:** são pacotes de aplicativos e/ou novos tipos de objetos que podem ser adicionados ao Zope. Esses produtos são escritos em linguagem Python e são facilmente instalados pela interface Web de administração do Zope. O Plone é um dos mais importantes produtos Zope.

### 2.2.2 Recursos

Um importante recurso do Zope é a sua interface Web administrativa (ZMI). Nela é possível instalar novos produtos Zope (como por exemplo um novo *site* Plone), reinitializar o servidor, adicionar *scripts*, gerenciar os usuários, permissões, etc. Através da ZMI pode-se estender o Zope instalando-se novos produtos, incluindo componentes que permitem que Zope se comunique com outros bancos de dados.

Outro recurso relevante é a ferramenta de geração de páginas Web, através da linguagem de *script* ZPT (*Zope Page Templates*). As ZPTs são processadas ao lado do servidor e geram páginas HTML dinâmicas. Com elas, é possível desenvolver aplicações que fazem acessos a um banco de dados, por exemplo, conforme será discutido no capítulo 3 desse trabalho.

A seção seguinte apresenta o Plone, um CMS que executa sobre o servidor de aplicações Zope.

## 2.3 Plone

Plone (PLONE FOUNDATION, 2007; ENFOLDSYSTEMS, 2005) é um CMS baseado em um *framework* para gestão de conteúdo (*Content Management Framework* – CMF) desenvolvido sobre o servidor de aplicações Zope. Assim como Zope, Plone é um produto

de código aberto que beneficia-se de uma ampla comunidade de usuários e desenvolvedores. Plone possui uma interface amigável e personalizável, podendo ser utilizado para publicação de conteúdo na Internet ou Intranet, servindo também como ferramenta para trabalho colaborativo via Web.

### 2.3.1 Interface

Toda a administração de um *site* Plone pode ser feita via Web. Um exemplo da estrutura típica de um tal *site* pode ser observada na Figura 2.2. Como se pode observar nesta figura, é possível adicionar novos conteúdos, editar conteúdos existentes, pesquisar, alterar estado de certas publicações feitas, etc. A figura ainda ilustra algumas das características e recursos do Plone que serão discutidos mais a fundo na seção seguinte.

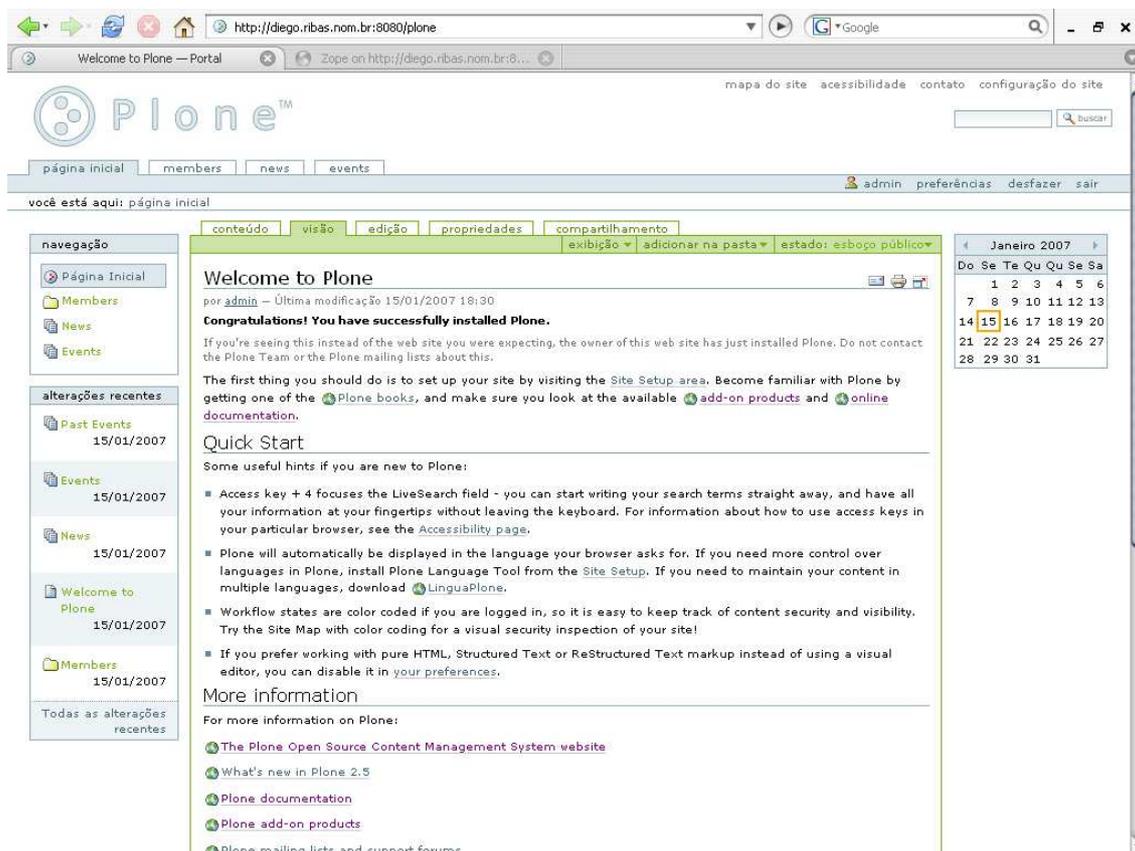


Figura 2.2: Exemplo de *site* Plone

É importante ressaltar que a estrutura e apresentação do *site* podem ser personalizadas a ponto de ficar difícil de identificar que o mesmo foi desenvolvido usando Plone. A Figura 2.3 ilustra essa afirmação, apresentando um exemplo de *site* Plone personalizado.



Fonte: Network of Oregon Watershed Councils (<http://www.oregonwatersheds.org/>).

Figura 2.3: Exemplo de *site* Plone personalizado

### 2.3.2 Recursos e características

O Plone possui diversos recursos e características que foram decisivos na escolha deste CMS para este trabalho. A seguir são descritos os principais recursos e uma breve descrição de cada um deles, conforme apresentados no guia de referência do Plone (EN-FOLDSYSTEMS, 2005):

- **Instalação:** para facilitar e gerenciar a instalação, Plone mantém ferramentas de instalação para Windows, Linux e Mac.
- **Internacionalização:** a interface com usuário foi traduzida para mais de 20 idiomas incluindo português.
- **Usabilidade:** Plone oferece uma interface com alto nível de acessibilidade e usabilidade, além de ser compatível com padrões de indústria e governo, tais como WAI-AAA (W3C, 1999) e U.S. Section 508 (US GOVERNMENT, 2007).
- **Personalização:** Plone permite a personalização completa do *site* através de temas

que separam o conteúdo da apresentação.

- **Autenticação:** Plone possui um sistema de autenticação e cadastro que pode ser personalizado. Além disso permite importação de usuários de bases de dados como LDAP (*Lightweight Directory Access Protocol*).
- **Fluxo de trabalho e segurança:** o fluxo de trabalho pode ser configurado usando ferramentas gráficas. Para segurança, pode-se criar listas de controle para decidir o que um usuário tem permissão de fazer com um item do *site*, como editar, ver, comentar, interagir, acessar.
- **Extensibilidade:** como o Plone é uma ferramenta de código aberto, ele pode ser facilmente modificado. Existem ainda diversos produtos independentes que podem ser adicionados ao Plone aumentando suas funcionalidades. Com ferramentas de desenvolvimento como *Archetypes* pode-se gerar ou alterar código Plone usando ferramentas de modelagem UML (*Unified Modeling Language*). Ainda existe a possibilidade de integração do Plone com soluções empresariais como LDAP, Apache, Microsoft IIS (*Internet Information Server*), Macromedia Dreamweaver, etc.
- **Personalização de conteúdo:** desenvolvedores Plone podem criar seus próprios tipos de conteúdo e gerenciá-los.
- **Documentação:** o projeto Plone mantém uma rica documentação, incluindo um livro sobre o assunto. Também existe uma iniciativa de tradução para o português da documentação no portal da comunidade Zope e Plone brasileira (TCHEZOPE, 2007), onde já se encontram alguns documentos traduzidos.
- **Comunidade:** Plone conta com uma extensa comunidade de desenvolvedores e empresas que oferecem suporte e desenvolvem sobre a plataforma.

O Plone, na instalação padrão, já possui diversos tipos de conteúdo que são úteis para a maioria dos *sites* Web. A Figura 2.4 mostra que é possível adicionar notícias, eventos, páginas e demais conteúdos comuns a maioria dos *sites*. Também é possível, através da instalação de produtos Plone, incorporar novos tipos de conteúdo multimídia, como músicas, vídeos e álbuns de fotografias. Além disso, existe a possibilidade da criação de novos conteúdos adaptados às necessidades de cada *site*.

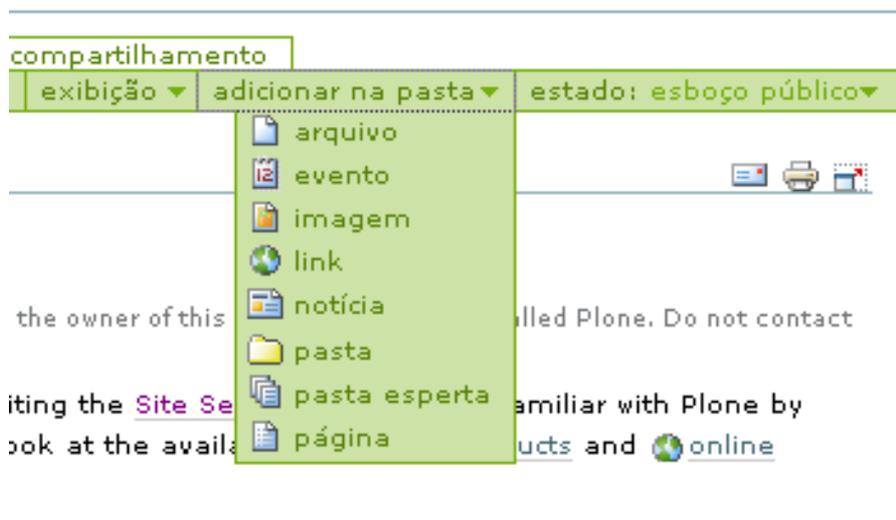


Figura 2.4: Tipos de conteúdo na instalação padrão do Plone

Conforme mencionado anteriormente, o Plone pode ser administrado e configurado via interface Web. A Figura 2.5 apresenta a tela de configuração de um *site* Plone. Com

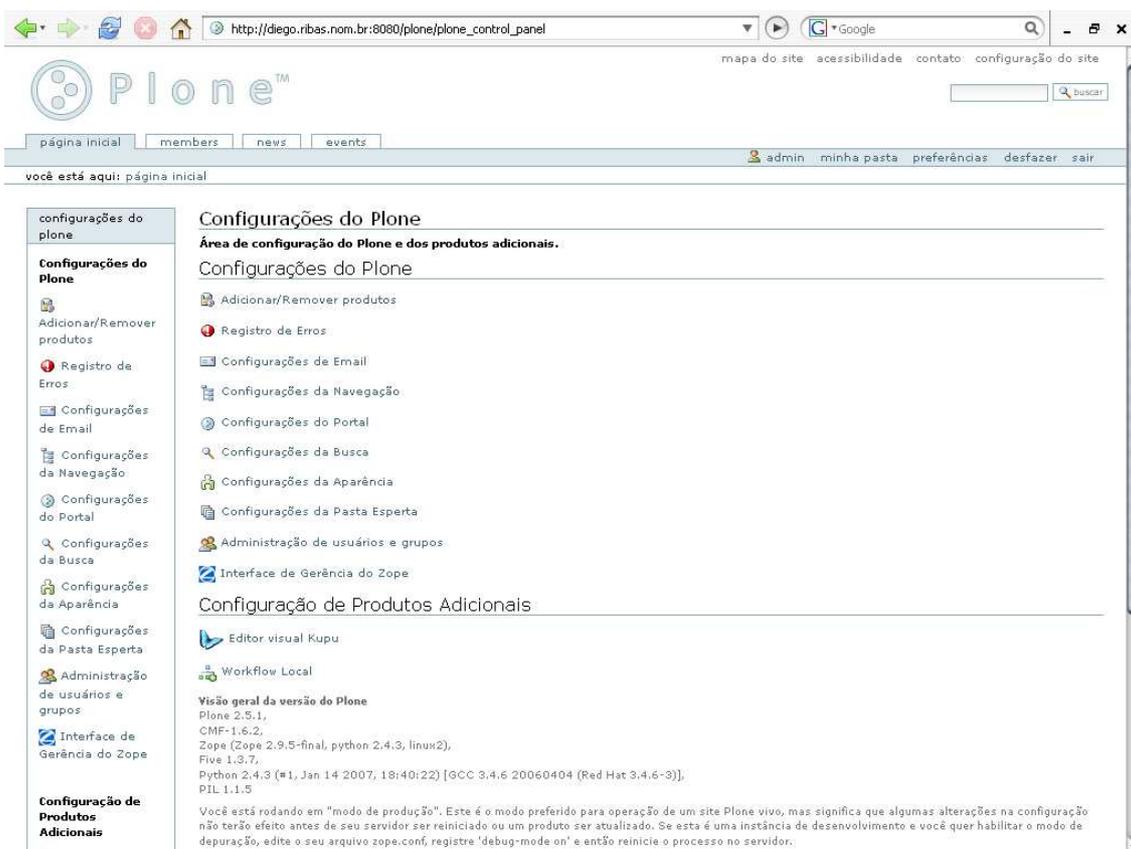


Figura 2.5: Interface de administração/configuração de um *site* Plone

essa interface é possível, por exemplo, adicionar e remover produtos, ver registros de erros, configurar busca, e-mail, navegação, adicionar novos fluxos de trabalho e, ainda,

administrar usuários e grupos.

### 3 APLICAÇÃO AO PORTAL DA INFORMÁTICA UFSM

Este capítulo analisa as vantagens do uso do Plone em relação ao sistema utilizado atualmente pela Informática da UFSM. Inicialmente apresenta-se o ambiente atual em que se encontra o *site* e os processos usados para o seu gerenciamento. Posteriormente discute-se a aplicação de recursos do Plone para gerenciamento deste portal. Ainda neste capítulo, demonstra-se o processo de desenvolvimento de aplicações para o Plone, com exemplos de aplicações que se enquadram na realidade da instituição.

#### 3.1 O *site* atual da Informática UFSM

O *site* Web da Informática da UFSM foi desenvolvido no ano de 2002, pela empresa Prumos Interativa<sup>1</sup>, que na época fazia parte da Incubadora Tecnológica da UFSM. Atualmente, este *site* possui diversas informações organizadas em seções. Algumas seções possuem ainda subseções para melhor organização do seu conteúdo. Essas informações são referentes, por exemplo, aos cursos de graduação e pós-graduação, aos laboratórios, endereços e telefones, publicações, eventos, etc. A Figura 3.1 mostra o *site* atual, permitindo visualizar sua estrutura e interface.

Um ponto positivo do *site* atual é que ele possui uma interface objetiva e limpa, o que torna fácil a navegação para os seus visitantes. Já o processo de publicação de novas informações no *site* é feita através de solicitações ao *webmaster*. Esse processo centralizado é um ponto negativo, pois no portal várias informações são incluídas, atualizadas e alteradas constantemente, levando a uma sobrecarga do administrador do *site* e impaciência dos que aguardam os pedidos solicitados. Assim, informações como as notícias, reunidas na página principal, que são atualizadas constantemente, podem sofrer atrasos de publicação. Além disso, a inclusão de conteúdo é feita diretamente em linguagem HTML e

---

<sup>1</sup>Prumos Interativa: <http://www.prumos.com.br>



Figura 3.1: Atual portal da Informática UFSM

PHP (THE PHP GROUP, 2007), dificultando a manutenção e uniformidade do *layout* do *site*. Essa dificuldade de publicação de informação acaba refletindo em uma impressão externa de um *site* pouco dinâmico.

Outro aspecto importante a se considerar é a pesquisa eficiente no *site*. Este recurso é bastante importante quando a quantidade de informações é grande. Em testes realizados no sistema de busca atual do *site* da Informática, não obteve-se resultados satisfatórios. Em particular, constatou-se que o sistema de busca funciona apenas com determinados navegadores.

Portais Web geralmente agregam aplicações de interesse da organização, como registro de alunos/professores, fóruns, registro de patrimônios, publicação de notas/trabalhos, fluxo de documentos, etc. No atual *site*, não observou-se nenhuma aplicação integrada para uso dos alunos ou professores.

## 3.2 O portal da Informática com o uso do Plone

Conforme já visto, o Plone possui um conjunto de tipos de conteúdos comuns à maioria dos portais. A maioria deles, tais como notícias, eventos, páginas, imagens e arquivos, pode ser aproveitada diretamente para desenvolvimento de um novo portal da Informática. O processo de publicação dessas novas informações é simples e descentralizado com o uso do Plone, pois um dos objetivos de um CMS é permitir que usuários não-técnicos possam realizar tudo ou boa parte do que um *webmaster* geralmente faz. A seguir discute-se em maiores detalhes o processo de publicação e indexação de conteúdo usando o Plone.

### 3.2.1 Publicação de conteúdo

O plone permite o uso de diferentes estados ao conteúdo. Entende-se por conteúdo uma página HTML, um documento, uma figura, etc. Para compreender-se como funciona o processo de inclusão de conteúdo usando o Plone é necessário conhecer-se os estados que um conteúdo pode assumir:

- **Visível:** o conteúdo é criado nesse estado. Ele não é mostrado na árvore de navegação, mas pode ser acessado e editado pelo seus donos (criadores) e pelos gerenciadores do portal.
- **Pendente:** compreendem os ítems que aguardam revisão para poderem ser publicados. Revisores do portal são notificados para publicar, rejeitar ou editar os ítems pendentes.
- **Publicado:** são ítems visíveis em todo o portal. Eles aparecem na árvore de navegação e nas pesquisas. Ítems publicados só podem ser editados pelos gerenciadores do portal, mas seus donos podem editá-los voltando ao estado pendente para revisão.
- **Privado:** ítems nesse estado só são vistos e editados por seus donos e gerenciadores do portal. Eles não aparecem em resultados de buscas e na árvore de navegação.

Quando um conteúdo é criado no Plone, ele assume o estado visível. Assim que o criador solicitar a publicação do conteúdo ele passa para o estado pendente. Os revisores do portal são notificados através de uma lista de revisões, e decidirão se o conteúdo pode ou não ser publicado, podendo inclusive fazer alterações no mesmo.

As figuras a seguir ilustram o cadastro de uma notícia no Plone, simulando situações cotidianas do *site* da Informática. Pode-se observar na Figura 3.2 que não é necessário nenhum conhecimento técnico para adicionar uma notícia no portal. Basta preencher os campos necessários e publicar a notícia. Nota-se que existe ainda um editor disponível, para dar agilidade e evitar erros de programação, além de facilitar a publicação por parte de usuários que desconhecem a linguagem HTML.



Figura 3.2: Cadastro/edição de uma notícia no Plone

A notícia pode ainda ser editada em caso de erros, sendo que esse procedimento é muito semelhante à inclusão da notícia. Após a solicitação de publicação da notícia, ela aguardará o processo de revisão para poder passar ao estado público. Uma vez no estado público, a notícia já aparecerá na árvore de navegação, nos resultados das pesquisas e já estará visível a todos os usuários. A Figura 3.3 mostra a notícia já publicada. Pode-se também ver um histórico das ações e comentários dos revisores da notícia antes de sua publicação.

Os recursos apresentados acima permitem que diferentes usuários, cada qual com sua função, possam ficar responsáveis pela atualização das informações no portal. O papel do

The screenshot shows a web browser displaying a news article on the Plone CMS. The browser's address bar shows the URL: `http://diego.ribas.nom.br:8080/inf/noticias/selecao-de-tutor-para-o-pet-ciencia-da-computacao`. The website header includes the logo for 'informática Universidade Federal de Santa Maria' and navigation links like 'página inicial', 'members', and 'notícias'. The article title is 'Seleção de Tutor para o PET/Ciência da Computação', published by 'admin' on 15/01/2007. A 'Histórico' table shows the publication action. The main text discusses the selection process for the PET program, mentioning a deadline of 18/01/2007. A sidebar on the left shows recent changes, and a calendar on the right shows the date 15/01/2007 highlighted. The footer contains copyright information for Plone CMS.

Figura 3.3: Visualização de uma notícia publicada no Plone

revisor, em particular, garante a qualidade e veracidade das informações que serão publicadas. Como os usuários podem assumir diversos papéis no Plone, existe uma interface de gerência de permissões em que é possível estipular papéis dentro da arquitetura do *site*. A Figura 3.4 apresenta essa interface, que permite especificar as permissões dos usuários por tipo de conteúdo. No caso da seção notícias do *site*, por exemplo, poderiam ser especificados os usuários responsáveis por publicar novas notícias e os revisores que iriam aprovar ou não essas publicações.

Outras operações comuns ao *site* atual da Informática, tais como criação de seções, subseções, páginas novas, cadastro de eventos, etc., podem ser realizadas de forma semelhante à publicação de uma notícia. Dessa maneira, fica fácil distribuir a tarefa de gestão do conteúdo do *site* entre diversos colaboradores, atribuindo a eles papéis e permissões particulares dentro das seções do portal. Com tudo isso, o portal ficaria mais dinâmico, mais simples e fácil de administrar.

The screenshot shows the 'Permissões de compartilhamento atuais de Notícias' interface. At the top, there are tabs for 'conteúdo', 'visão', 'edição', 'propriedades', and 'compartilhamento'. Below the tabs, there are buttons for 'ações', 'exibição', 'adicionar novo item', and 'estado: esboço público'. The main heading is 'Permissões de compartilhamento atuais de Notícias'. Below this, there is a text block stating: 'Você pode compartilhar direitos tanto para pastas (incluindo seu conteúdo) quanto para itens individuais. Estes usuários possuem direitos nessa pasta:'. The main content area is titled 'Papéis atribuídos para Notícias' and contains a table with columns: 'nome', 'tipo', 'papéis herdados', and 'papéis locais'. The table has two rows: one for 'admin' (Usuário) with 'Dono' as inherited and local roles, and one for 'usuario1 (usuario1)' (Usuário) with 'Revisor' as a local role. Below the table, there are radio buttons for 'Administrador', 'Membro', and 'Revisor'. There are two buttons: 'atribui os papéis selecionados para os usuários/grupos selecionados.' and 'exclui papéis selecionados e usuários/grupos'. Below this is the section 'Adicionar permissões de compartilhamento para usuários' with the heading 'Resultados da busca'. It says 'Selecione um ou mais Membros para atribuir um papel.' and shows a table of 'Membros Disponíveis' with columns 'nome completo' and 'nome do usuário (user id)'. The table has four rows: 'Revisor' (revisor), 'Diego Ribas Adiers' (ribas), 'usuario1' (usuario1), and 'usuario2' (usuario2). Below the table, there is a dropdown menu for 'Papel a atribuir' with options: 'Administrador', 'Membro', 'Dono', and 'Revisor'. There is a button 'atribuir papéis locais para os usuários selecionados'.

Figura 3.4: Interface de gerência de permissões do Plone

### 3.2.2 Indexação e pesquisa de conteúdo

Um ponto forte do Plone é a indexação automática do seu conteúdo. Toda informação no estado “publicado” é automaticamente indexada e já irá aparecer em resultados de pesquisa. Após uma notícia ser incluída, por exemplo, ela já será retornada como resultado de pesquisas relacionadas conforme mostra a Figura 3.5. Além disso, Plone possui uma interface de pesquisa avançada, que pode ser visualizada na Figura 3.6. Através dessa interface é possível especificar na pesquisa o tipo de conteúdo, autor, combinar palavras chaves, estado da revisão, etc. A pesquisa avançada é um recurso útil que não existe no atual *site* da Informática e poderia ser explorado no futuro portal.

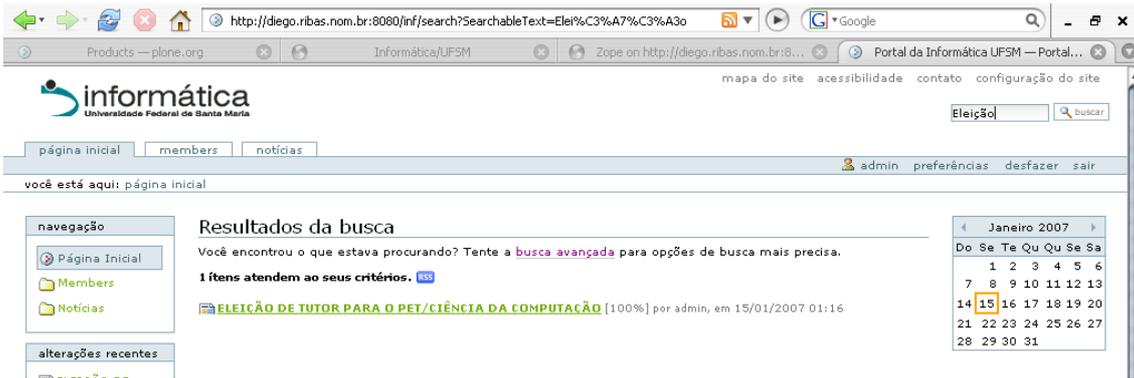


Figura 3.5: Resultado de uma busca no Plone

Buscar Termos

**Texto da Busca**  
Para uma simples busca de texto, informe seu termo de busca aqui. Múltiplas palavras podem ser encontradas combinando-as com **AND** e **OR**. O texto desse campo será comparado com o conteúdo dos itens, título e descrição.

**Título**  
Retorna itens que atendam esse título.

**Palavras-chave**  
Retorna itens que atendam a uma ou mais dessas palavras-chave. Múltiplas palavras podem ser selecionadas pressionando **Ctrl** (ou **Apple** no Mac) enquanto você clica nas palavras-chave.

Combinar algumas  
 Combinar todas

**Descrição**  
Retorna itens que atendam essa descrição. Múltiplas palavras podem ser encontradas combinando-as com **AND** e **OR**.

**Notícias desde**  
Retorna os itens adicionados desde seu último acesso, última semana, etc.

Última semana ▾

**Tipo de Item**  
Retorna itens de um tipo específico.

Inverter

<input checked="" type="checkbox"/> Computador	<input checked="" type="checkbox"/> Imagem
<input checked="" type="checkbox"/> Computadores	<input checked="" type="checkbox"/> Pasta Gigante
<input checked="" type="checkbox"/> Página	<input checked="" type="checkbox"/> Link
<input checked="" type="checkbox"/> Evento	<input checked="" type="checkbox"/> Locais
<input checked="" type="checkbox"/> Favorito	<input checked="" type="checkbox"/> Local
<input checked="" type="checkbox"/> Arquivo	<input checked="" type="checkbox"/> Notícia
<input checked="" type="checkbox"/> Pasta	<input checked="" type="checkbox"/> Pasta Esperta

**Autor**  
Retorna os itens criados por uma pessoa em particular.

Qualquer Autor ▾

**Estado de revisão**

Figura 3.6: Interface de busca avançada no Plone

### 3.3 Desenvolvimento de aplicações para o Plone

Um aspecto importante em um portal é a integração de aplicações. Isso é cada vez mais comum nas organizações, principalmente pelo benefício proporcionado aos seus usuários. Tratando-se de uma organização de ensino, aplicações que gerenciassem as notas e trabalhos dos alunos facilitariam a vida de ambos. O mesmo vale para aplicações mais simples de gerenciamento de patrimônio, cotas de impressão, contato de alunos, entre várias outras aplicações possíveis. Existem diversas ferramentas que facilitam o desenvolvimento dessas aplicações, que podem ser facilmente instaladas em qualquer *site* Plone. As seções seguintes detalham o processo de desenvolvimento de novas aplicações para o Plone.

#### 3.3.1 Processo de desenvolvimento

A Figura 3.7 ilustra o processo de desenvolvimento de uma aplicação para o Plone. Para compreender-se esse processo, é necessário conhecer as ferramentas envolvidas: Archetypes, ArchGenXML e um *software* para modelagem UML.

Archetypes (PLONE.ORG, 2007) é um *framework* que facilita a criação de aplicações para o Plone. Seu objetivo principal é oferecer uma maneira comum de criar conteúdo, baseado em definições de esquemas. Essa ferramenta também é útil para geração de formulários e validação automática de dados.

ArchGenXML é uma ferramenta de linha de comando que gera produtos funcionais para o Plone, baseados no *framework* Archetypes. A entrada para esta ferramenta é um modelo UML armazenado em formato XMI (*XML Metadata Interchange*).

Existem diversos programas para modelagem UML que geram arquivos em formato XMI. Um dos mais citados na literatura é o *software* ArgoUML (TIGRIS.ORG, 2007; RAMIREZ et al., 2007), uma ferramenta de código aberto que executa em todas plataformas que suportem Java. Outro exemplo é o Poseidon (GENTLEWARE AG, 2007) que é um *software* proprietário também desenvolvido em Java. Existe uma edição deste *software* que pode ser usada sem custos (*Community Edition*), mas com algumas limitações nas funcionalidades.

O processo de desenvolvimento de uma aplicação, na prática, transcorre da seguinte maneira: usando um programa de modelagem UML, o desenvolvedor desenha seu modelo UML de aplicação, com suas classes, atributos, métodos e relacionamentos. A seguir,

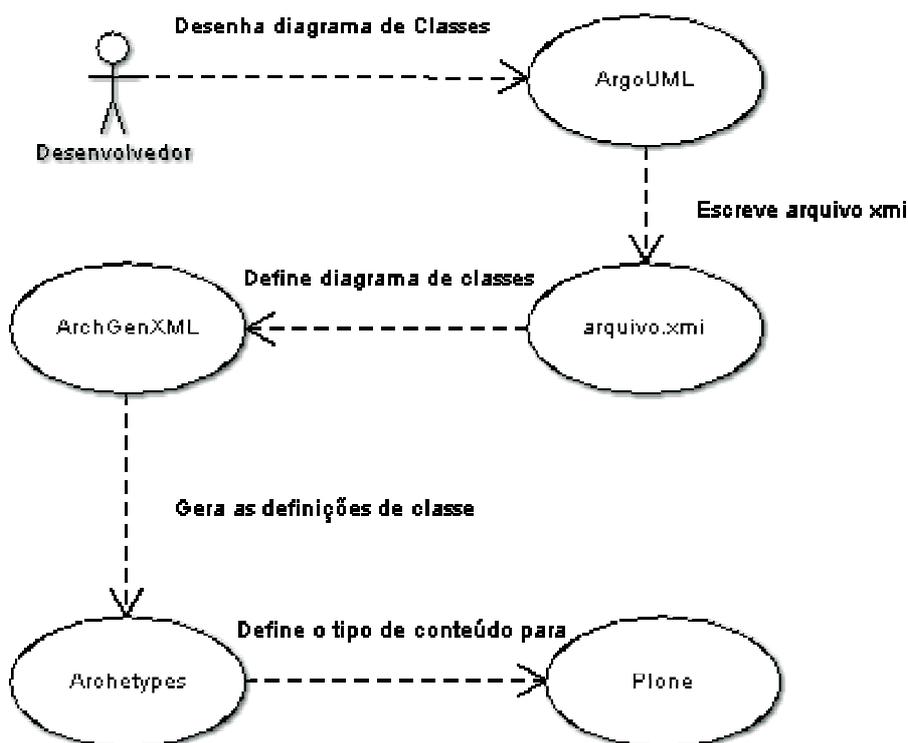


Figura 3.7: Processo de desenvolvimento de aplicações para o Plone

usando ArchGenXML, gera-se o produto (aplicação) que deverá ser adicionado ao *site* Plone. Para isso, a aplicação deve ser colocada dentro da pasta onde se encontram os demais produtos Zope. Em seguida, deve-se reiniciar o Zope e acessar a seção de configuração do *site* como administrador (opção de adição/remoção de produtos) e adicionar a aplicação. Com esses procedimentos é possível criar aplicações integradas ao Plone sem escrever nenhuma linha de código. Para personalização da aplicação, pode-se alterar o código gerado em linguagem Python. As seções seguintes mostram exemplos práticos de aplicações desenvolvidas.

### 3.3.2 Exemplo: gerenciador de informações sobre computadores da rede

Nesta seção ilustra-se o desenvolvimento de uma aplicação simples usando os recursos descritos na seção anterior. O exemplo escolhido é um gerenciador de informações sobre os computadores da rede, inspirado em necessidades reais na administração da rede da Informática UFSM. Essa aplicação permite gerenciar informações sobre computadores através de uma interface Web de administração simples e única. É possível armazenar várias informações sobre um computador, como seu nome e endereço na Internet (*hostname*), descrição de processador, memória, placa de rede, etc., assim como a sala ou local



submeter a instalação já é possível adicionar os novos tipos de conteúdo criados, em qualquer parte do sítio Plone, e fazer uso da nova aplicação.

A Figura 3.9 mostra a interface de cadastro de computadores. Nela são informados os campos obrigatórios, Hostname e IP, seguidos da Descrição que detalha o computador. Como essas informações de computadores de uma rede são geralmente restritas, pode-se tornar privado o contêiner (coleção) que abriga os computadores, permitindo apenas o acesso ao pessoal autorizado. Na seqüência, a Figura 3.10 exibe a interface de cadastro de locais. No formulário aparecem os campos Nome (campo obrigatório), Descrição do local, e por fim, permite marcar os computadores que fazem parte desse local/sala. A edição de computadores e locais funciona da mesma forma que o cadastro. A interface de pesquisa funciona da mesma maneira que para outros tipos de conteúdo (notícias, fotos, eventos), conforme já demonstrado anteriormente.

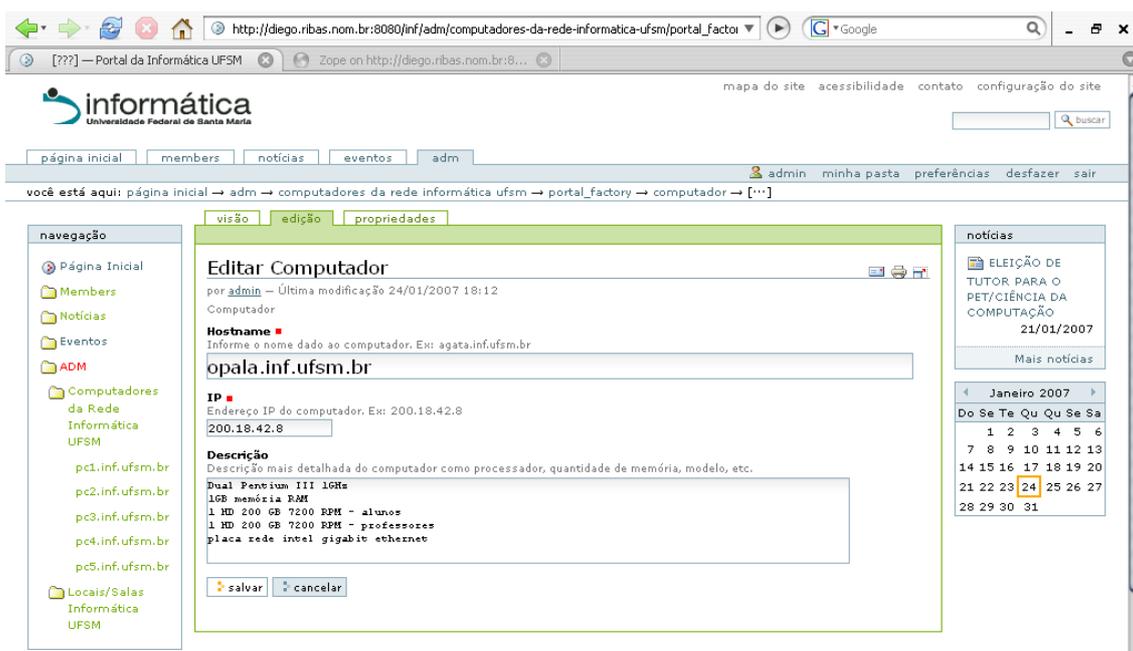


Figura 3.9: Interface de cadastro de computadores da aplicação exemplo

Esse primeiro exemplo de aplicação demonstra a facilidade da criação de produtos para o Plone. Trata-se de uma aplicação simples, em que bastou criar-se o modelo UML e identificar as informações necessárias dos campos do formulário. Não foi necessário ao desenvolvedor escrever uma linha de código, ao contrário do que geralmente se espera no desenvolvimento de aplicações. A seção seguinte detalha outro exemplo de aplicação que acessa um banco de dados MySQL (MYSQL AB, 2007), que é muito popular dentro das instituições públicas de ensino.

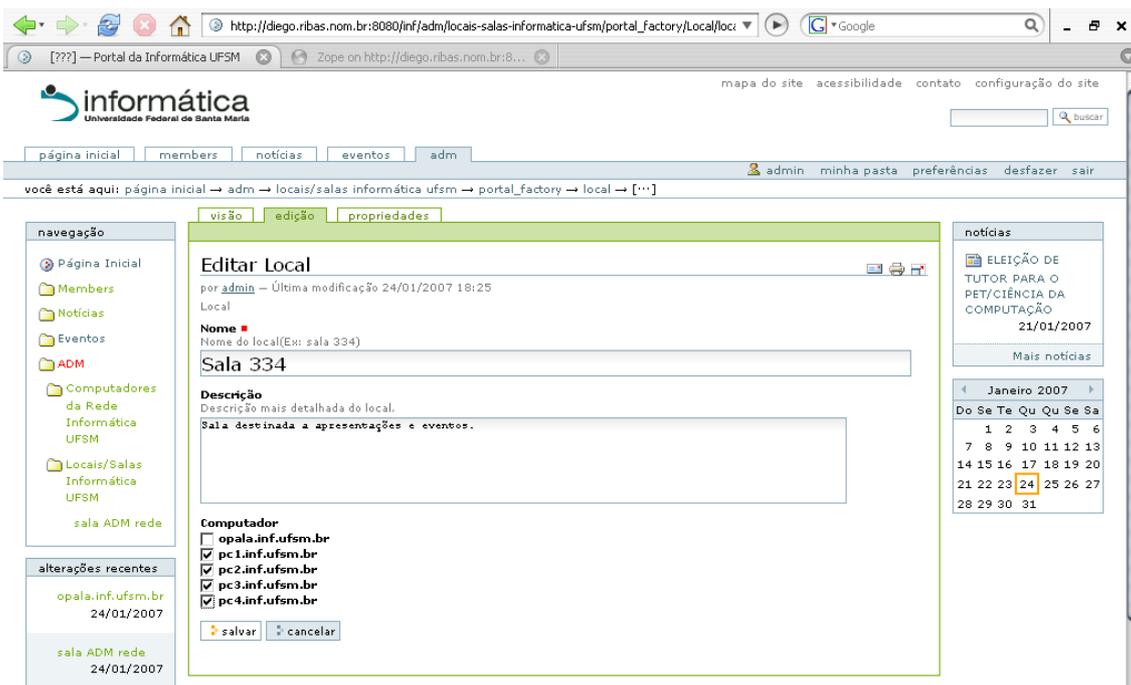


Figura 3.10: Interface de cadastro de locais/salas da aplicação exemplo

### 3.3.3 Exemplo: gerência de alunos usando MySQL

Considerando que muitas instituições de ensino possuem seus dados espalhados em diversos bancos de dados distintos, foi criada uma aplicação que mostra a interação entre o Plone/Zope com o banco de dados MySQL (MYSQL AB, 2007). Escolheu-se o MySQL por ser um dos bancos de dados de código aberto mais populares atualmente, e muito usado em instituições de ensino, inclusive na Informática UFSM. Na escolha dessa aplicação, houve a preocupação em mantê-la simples, pois o intuito principal dessa aplicação é demonstrar a flexibilidade da plataforma Plone/Zope.

A aplicação tem como objetivo inserir e buscar dados de uma tabela de alunos armazenada num banco de dados MySQL. Nessa aplicação não foi usado o processo de desenvolvimento de produtos Plone com Archetypes descrito anteriormente. Ao invés disso, a aplicação aprofunda os conceitos de ZPT (*Zope Page Templates*) e *scripts* em linguagem python no processamento de formulários. ZPT é uma linguagem processada ao lado do servidor, usada para criar páginas HTML dinâmicas, de forma semelhante a outras linguagens como PHP (THE PHP GROUP, 2007), ASP (MICROSOFT CORPORATION, 2007) e JSP (SUN MICROSYSTEMS, 2007).

Nos parágrafos seguintes descreve-se alguns detalhes da implementação dessa aplicação. A instalação do MySQL foge do escopo desse trabalho e não será detalhada aqui.

Para começar, foi necessário criar a tabela Aluno no MySQL. A tabela criada pode ser visualizada na Figura 3.11.

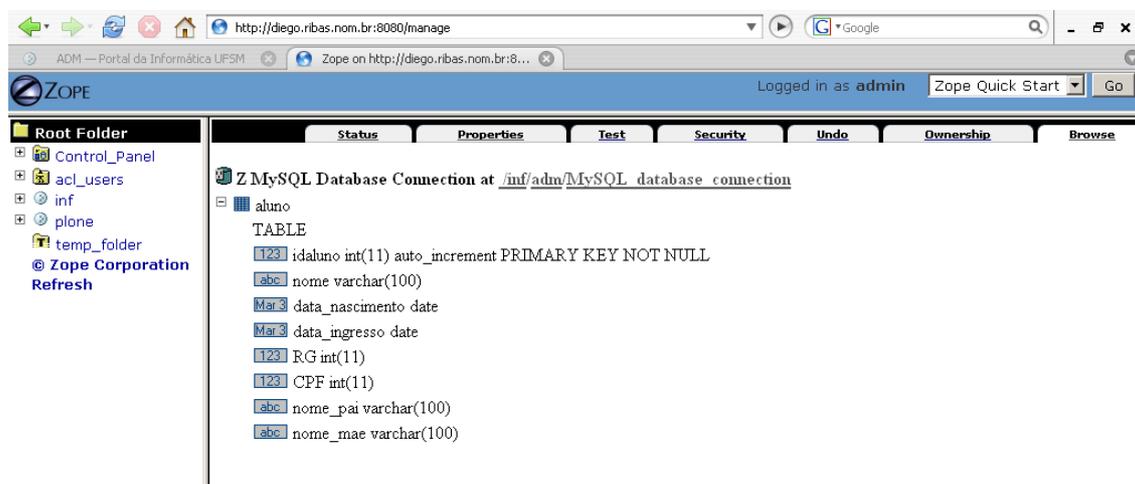


Figura 3.11: Visualização da tabela Aluno criada no MySQL dentro da ZMI

Para que o Zope seja capaz de se comunicar com o banco de dados MySQL, é necessário instalar alguns programas adicionais. Um deles é Python-MySQL (DUSTMAN, 2007a), que é uma interface para que programas Python acessem o MySQL. O outro é o ZMySQLDA (DUSTMAN, 2007b), um adaptador para o Zope acessar o MySQL. Foram usadas as versões 1.2.1\_p2 do Python-MySQL e a 2.0.8 do ZMySQLDA. Após a instalação dos mesmos, a ZMI (*Zope Management Interface*) disponibiliza o novo tipo de objeto Z MySQL Database Connection, que é responsável por criar conexões com o MySQL. Via ZMI, deve-se criar um objeto desse tipo, informando-se dados sobre usuário e senha do seu banco de dados. Depois da conexão ter sido configurada, o próximo passo é criar métodos SQL para manipulação de dados. No caso em questão, criou-se um método para consulta por nome do aluno, e outro para inserção de dados na tabela de alunos. Para criação de métodos de acesso, adiciona-se o objeto Z SQL Method via ZMI e preenche-se os dados requisitados e o código SQL, conforme ilustrado na Figura 3.12.

A Figura 3.12 apresenta o método SQL `sql_inserere_aluno` criado para realizar a inserção de um aluno no banco de dados. O código SQL que consulta alunos por nome (`sql_consulta_aluno`) é apresentado na Figura 3.13. Nesse código SQL, apenas o nome do aluno é passado como argumento na invocação do método. O resultado da execução deste método são todos os alunos cujos nomes contêm a sequência de letras procurada, ordenados pelos seus nomes.

Após a criação das consultas e inserções em SQL no Zope, é necessário criar inter-

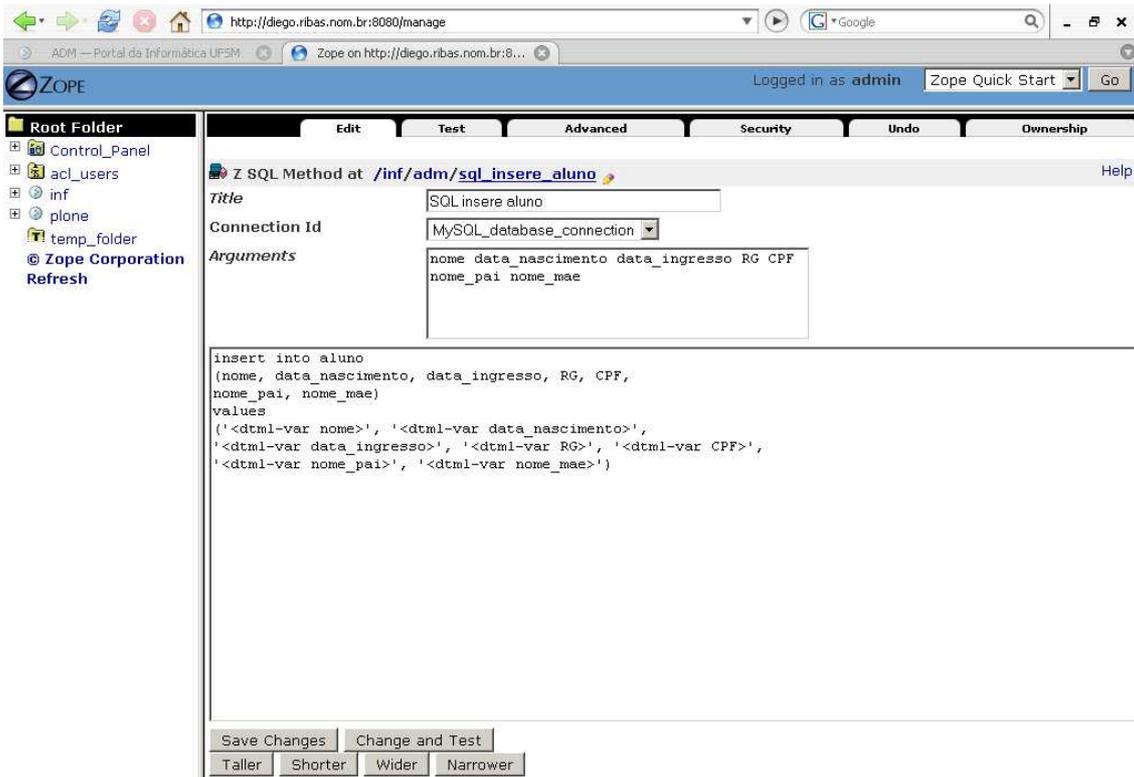


Figura 3.12: Criação de um Z SQL Method dentro da ZMI

```
select * from aluno
where nome like <dtml-sqlvar "'%' + nome + '%'" type=string>
order by nome
```

Figura 3.13: Código SQL que consulta alunos

faces que executam essas ações. Primeiramente descreve-se a criação do formulário de pesquisa, que é o mais simples. Basicamente, a criação de formulários é feita com o uso da ZPT. Na ZMI do Zope, adiciona-se um objeto do tipo Page Template. No caso em questão, esse objeto foi chamado de *zpt\_consulta\_alunos* e está representado na Figura 3.14. O código da Figura 3.14 é responsável por gerar o formulário de pesquisa apresentado na Figura 3.15.

É importante notar que o formulário de pesquisa usa outro *script* ZPT, denominado *resultado\_da\_pesquisa*, para mostrar e paginar os resultados obtidos da pesquisa. Esse *script* é apresentado na Figura 3.16.

O código na Figura 3.16 tem a função de invocar o SQL já criado anteriormente em *sql\_consulta\_aluno*, que retorna os alunos encontrados com determinado nome. Além disso, ele mostra e pagina em uma tabela cada um dos alunos retornados pela pesquisa.

```

<html metal:use-macro="here/main_template/macros/master">
  <head>
    <title tal:content="resultado_da_pesquisa/title">The title</title>
  </head>
  <body><div metal:fill-slot="main">
    <hr>
    <h3>Formulário de Pesquisa</h3>
    <hr>
    <form action="." method=get>
      <p>Digite o Nome do Aluno a pesquisar.</p>
      <table>
        <tr>
          <td>Nome:</td>
          <td><input name=nome width="30"></td>
          <td><input type=submit value="Pesquisar Aluno"
            name=resultado_da_pesquisa:method>
          </td>
        </tr>
      </table></form>
    </div></body>
</html>

```

Figura 3.14: Código ZPT que gera o formulário de pesquisa



Figura 3.15: Interface de pesquisa de alunos no MySQL

Para ilustrar melhor a função desse *script*, a Figura 3.17 mostra o resultado de uma pesquisa no banco de dados .

É importante ressaltar que, para implementar a pesquisa, foram necessários dois *scripts* ZPT: um para criar o formulário de pesquisa e outro para executar o código SQL e mostrar o resultado.

Descreve-se a seguir o processo de uma inserção de um aluno no banco de dados MySQL. Parte-se do pressuposto que existe uma conexão criada com o banco MySQL e o código SQL responsável pela inserção (Figura 3.12). Primeiramente, é preciso criar um formulário para que o usuário preencha os dados do aluno. O ZPT criado para gerar o

formulário encontra-se na Figura 3.18.

O código na Figura 3.18 gera um formulário simples, que irá invocar um *script* Python, ainda a ser criado, que será responsável por processar os dados do formulário e inseri-los no MySQL. Esse formulário pode ser visualizado na Figura 3.19.

O passo seguinte é a inserção, através da ZMI, de um *script* Python chamado *py\_inclui*, conforme foi especificado no formulário. Este *script* é representado na Figura 3.20 e já está comentado para melhor entendimento.

O *script* *py\_inse* na Figura 3.20 recebe os valores preenchidos no formulário e invoca o SQL *sql\_inse\_aluno*, criado anteriormente, que faz o trabalho de inserir o aluno no MySQL. Por fim, ele retorna um Page Template chamado *resposta\_insecao* que será responsável por mostrar a mensagem de retorno. É importante notar que não foi feita nenhuma validação dos dados nesse formulário, pois esses exemplos são meramente ilustrativos e visam apenas demonstrar de forma simples o uso conjunto de Plone/Zope e do banco de dados MySQL. O código *resposta\_insecao* é simples e está apresentado na Figura 3.21. Esse código simplesmente mostra a mensagem passada pelo *script* Python *py\_inse* e mostra um *link* para voltar ao formulário de inserção de alunos.

A aplicação que insere alunos no MySQL necessitou de dois ZPT e um *script* Python. Para finalizar, a Figura 3.22 mostra a visualização dos arquivos dentro da ZMI que foram necessários para a criação dessas aplicações que interagem com o MySQL. Através da ZMI, o Zope oferece um suporte ao desenvolvimento de aplicações totalmente via Web, abrangendo desde a criação de arquivos até sua edição. Apesar de não possuir todas as vantagens de um editor de programas (como marcação sintática e endentação automática), a ZMI permite ao desenvolvedor criar aplicações simples sem o uso de editores externos.

Nesta seção demonstrou-se como construir aplicações que se comunicam com um banco de dados não nativo ao Zope, no caso o MySQL. Também é possível fazer essa plataforma se comunicar com outros tipos de banco de dados como LDAP (OPENLDAP FOUNDATION, 2007), PostgreSQL (POSTGRESQL.ORG, 2007) e Oracle (ORACLE, 2007). As aplicações apresentadas neste capítulo servem para alavancar o desenvolvimento de aplicações mais complexas e robustas em torno da plataforma Zope/Plone. Elas servem, ainda, para mostrar a flexibilidade e poder dessa plataforma que pode ser expandida e alterada de acordo com as necessidades da Informática UFSM.

```

<html metal:use-macro="here/main_template/macros/master">
<body metal:fill-slot="main" tal:define="results
  here/sql_consulta_aluno;
  start request/start|python:0;
  batch python:modules['ZTUtils'].Batch(results, size=20, start=start);
  previous python:batch.previous;
  next python:batch.next"><br>
<a href="index.html">Voltar</a>
<hr><h3>Resultado da Pesquisa</h3><hr><p>
<a tal:condition="previous"
tal:attributes="href string:${request/URL0}?
                start:int=${previous/first}"
href="previous_url">previous <span tal:replace="previous/length">20
</span> results</a>
<a tal:condition="next"
tal:attributes="href string:${request/URL0}?start:int=${next/first}"
href="next_url">next <span tal:replace="next/length">20</span>
results</a>
</p><table border><tr>
  <th>Idaluno</th>
  <th>Nome</th>
  <th>Data nascimento</th>
  <th>Data ingresso</th>
  <th>RG</th>
  <th>CPF</th>
  <th>Nome pai</th>
  <th>Nome mae</th>
</tr>
<tal:x repeat="result batch" >
<tr>
  <td><span tal:replace="result/idaluno">idaluno goes here
    </span></td>
  <td><span tal:replace="result/nome">nome goes here</span></td>
  <td><span tal:replace="result/data_nascimento">data_nascimento
    goes here</span></td>
  <td><span tal:replace="result/data_ingresso">data_ingresso
    goes here</span></td>
  <td><span tal:replace="result/RG">RG goes here</span></td>
  <td><span tal:replace="result/CPF">CPF goes here</span></td>
  <td><span tal:replace="result/nome_pai">nome_pai goes here
    </span></td>
  <td><span tal:replace="result/nome_mae">nome_mae goes here
    </span></td>
</tr></tal:x>
</table><p>
<a tal:condition="previous"
  tal:attributes="href string:${request/URL0}?start:int=
    ${previous/first}" href="previous_url">previous
<span tal:replace="previous/length">20</span> results</a>
<a tal:condition="next"
  tal:attributes="href string:${request/URL0}?start:int=${next/first}"
  href="next_url">next <span tal:replace="next/length">20</span>
results</a>
</p></body></html>

```

Figura 3.16: Código ZPT que mostra o resultado da pesquisa

The screenshot shows a web browser window displaying a search results page. The browser's address bar shows the URL: `http://diego.ribas.nom.br:8080/inf/adm/?nome=&resultado_da_pesquisa%3Amethod=Pesqui...`. The page header includes the logo for 'informática Universidade Federal de Santa Maria' and navigation links like 'página inicial', 'members', 'notícias', 'eventos', and 'adm'. Below the header, there are user options: 'admin', 'minha pasta', 'preferências', 'desfazer', and 'sair'. The main content area is titled 'Resultado da Pesquisa' and contains a table with the following data:

Idaluno	Nome	Data nascimento	Data ingresso	RG	CPF	Nome pai	Nome mae
13	Bibas Lojeft	1980/09/10	2002/05/20	2147483647	230948234	Marcos da Silva	Janete Junior Benske
5	Diego Ribas Adiers	1982/08/29	2002/05/20	1087334545	985345345	Rog?rio Camargo Adiers	Salete Ribas Adiers
3	Gabriela Costa	1985/05/16	2006/03/25	3435345	34534535	Jorge Costa	Rosemeri Costa
10	Joana Maria Corote	1980/09/10	2002/01/20	1898734899	2147483647	Marcos da Silva	Maria Junior Benske
1	Jose Silva	1985/02/28	2004/03/22	1098373648	293837665	Joao Silva	Maria Jose
12	Lil Lucca	1980/09/10	2002/05/20	2147483647	230948234	Marcos da Silva	Janete Junior Benske
7	Luco Marciano Junior	1985/10/20	2006/10/20	2147483647	2147483647	Luco Maior	Mariluze Marciano
9	Marciano Da Silva Junior	1980/09/10	2002/01/20	1898734899	2147483647	Marcos da Silva	Maria Junior Benske
8	Marciano Da Silva Junior	1980/09/10	2002/01/20	1898734899	2147483647	Marcos da Silva	Maria Junior Benske
11	Marconildo Bekzjel	1980/09/10	2002/05/20	2147483647	230948234	Marcos da Silva	Maria Junior Benske
2	Marquiel Clo	1982/06/14	2006/03/25	983837455	983876658	Rodrigo Pessoa Clo	Dorildes Adiers
6	teste	2006/10/01	2006/10/01	24243	4324324	teste	teste

The interface also features a left sidebar with navigation options like 'Página Inicial', 'Members', 'Notícias', 'Eventos', and 'ADM'. A right sidebar contains a 'notícias' section with a link to 'ELEIÇÃO DE TUTOR PARA O PET/CIÊNCIA DA COMPUTAÇÃO' dated 21/01/2007, and a calendar for February 2007.

Figura 3.17: Resultado de uma pesquisa por alunos no MySQL

```

<html metal:use-macro="here/main_template/macros/master">
  <head>
    <title tal:content="template/title">The title</title>
  </head>
  <body> <div metal:fill-slot="main">
    <form action="." method=post>
      <h1>Coloque as informações do Aluno a ser cadastrado</h1><br>
      <table>
        <tbody>
          <tr>
            <td>Nome</td>
            <td><input type="text" name="nome" maxlength="100"
              size="50" value=""></td>
          </tr>
          <tr>
            <td>Data Nascimento</td>
            <td>
              <input type="text" name="data_nascimento" maxlength="10"
                size="13" value="">&nbsp;Formato: aaaa-mm-dd</td>
          </tr>
          <tr>
            <td>Data Ingresso</td>
            <td><input type="text" name="data_ingresso" maxlength="10"
              size="13" value="">&nbsp;Formato: aaaa-mm-dd</td>
          </tr>
          <tr>
            <td>RG</td>
            <td><input type="text" name="RG" maxlength="11"
              size="14" value=""></td>
          </tr>
          <tr>
            <td>CPF</td>
            <td><input type="text" name="CPF" maxlength="11"
              size="14" value=""></td>
          </tr>
          <tr>
            <td>Nome Pai</td>
            <td><input type="text" name="nome_pai" maxlength="100"
              size="50" value=""></td>
          </tr>
          <tr>
            <td>Nome Mãe</td>
            <td><input type="text" name="nome_mae" maxlength="100"
              size="50" value=""></td>
          </tr>
          <tr>
            <td></td>
            <td><input type=submit value="inserir"
              name="py_inclui:method"></td>
          </tr>
        </tbody>
      </table>
    </form>
  </div></body>
</html>

```

Figura 3.18: Código ZPT que gera o formulário de cadastro de aluno



Figura 3.19: Interface de cadastro de alunos no MySQL

```
#####
# Pega as variáveis do Ambiente
#####
request = contadiner.REQUEST
RESPONSE = request.RESPONSE

#####
# Recebe os valores da Página
#####
nome = request.get('nome')
data_nascimento = request.get('data_nascimento')
data_ingresso = request.get('data_ingresso')
RG = request.get('RG')
CPF = request.get('CPF')
nome_pai = request.get('nome_pai')
nome_mae = request.get('nome_mae')

#####
# Insere as informações no banco
#####
context.sql_inser_e_aluno(nome=nome, data_nascimento=data_nascimento,
data_ingresso=data_ingresso, RG=RG, CPF=CPF, nome_pai=nome_pai,
nome_mae=nome_mae)

#####
# Retorna para o template as informações
#####
mensagem = "Aluno Cadastrado com sucesso!"
return context.resposta_insercao(mensagem=mensagem)
```

Figura 3.20: Script Python que processa o formulário de cadastro de aluno

```

<html metal:use-macro="here/main_template/macros/master">
  <head>
    <title tal:content="template/title">The title</title>
  </head>
  <body>
    <div metal:fill-slot="main">
      <h3><span tal:replace="options/mensagem | default"></span></h3>
      <br>
      <a href="zpt_inserere_aluno">Voltar</a>
    </div>
  </body>
</html>

```

Figura 3.21: Código ZPT que mostra a resposta da inserção

The screenshot shows a Zope web interface with a file browser view of the /inf/adm directory. The interface includes a navigation pane on the left, a top menu bar, and a main content area displaying a list of files with columns for Type, Name, Size, and Last Modified. The files listed include workflow configurations, database connections, and various scripts related to student management.

Type	Name	Size	Last Modified
	.wf_policy_config (Workflow policy configuration)	1 Kb	2007-01-23 17:19
	MySQL_database_connection (Z MySQL Database Connection)	1 Kb	2007-01-30 17:44
	Pesquisar Alunos (Pesquisar Alunos)	1 Kb	2007-01-31 23:03
	computadores-da-rede-informatica-ufsm (Computadores da Rede Informática UFSM)	1 Kb	2007-01-24 18:38
	inserir-aluno (Inserir Aluno)	1 Kb	2007-01-31 23:03
	locais-salas-informatica-ufsm (Locais/Salas Informática UFSM)	1 Kb	2007-01-24 18:38
	py_inclui (Script que faz a insercao do aluno)	1 Kb	2007-01-31 23:39
	resposta_insercao (Resposta da Insercao)	1 Kb	2007-02-01 01:55
	resultado_da_pesquisa (Resultado da Pesquisa)	2 Kb	2007-02-01 01:53
	sql_consulta_aluno (SQL consulta alunos pelo nome)	1 Kb	2007-02-08 19:34
	sql_inserere_aluno (SQL inserere aluno)	1 Kb	2007-01-31 17:05
	zpt_consulta_alunos (Consulta Alunos)	1 Kb	2007-02-01 01:52
	zpt_inserere_aluno (Formulário de Inclusão do Aluno)	1 Kb	2007-02-01 01:50

Figura 3.22: Visualização dos arquivos dentro da ZMI

## 4 CONCLUSÃO

O volume de informações digitais disponíveis em *sites* Web vem aumentando consideravelmente, sendo que a popularização da Internet tem contribuído para esse acontecimento. Neste cenário, o modelo de gestão da informação baseado na figura única do *webmaster* começou a ficar inadequado, devido à grande quantidade de informações que precisam estar sempre atualizadas. Isso impulsionou o surgimento de sistemas que visam suprir essas necessidades de rápido e fácil gerenciamento da informação. Tais necessidades são observadas hoje no atual portal da Informática UFSM. Logo, esse trabalho teve como objetivo mostrar uma solução de gestão de conteúdo viável ao portal da Informática, usando o CMS Plone.

O ambiente Plone/Zope mostrou-se apto a suprir as necessidades básicas e imediatas do portal da Informática, conforme demonstrado no capítulo 3. O processo de publicação e gestão de toda a informação do portal pode ser feito de maneira distribuída e simples com o uso do Plone, ao contrário do que acontece no portal atual. Por ser uma ferramenta flexível e de código aberto, o Plone permite uma personalização completa, desde a modificação da apresentação até integração com outras ferramentas, que são recursos necessários à instituição. Além disso, o Plone vem sendo implantado com sucesso em diversas instituições públicas, o que reforça ainda mais as vantagens de seu uso.

Outro aspecto analisado no trabalho foi o desenvolvimento de aplicações baseadas na plataforma Plone/Zope, pois é de grande interesse da Informática UFSM agregar ao portal aplicações que sejam úteis aos alunos e professores. O processo de desenvolvimento de aplicações mostrou-se simples e rápido. Foram criadas duas aplicações exemplo, dentro da realidade da instituição, que exploram diferentes formas de desenvolvimento. A primeira aplicação, um gerenciador de informações sobre computadores da rede, mostra na prática o funcionamento do processo de desenvolvimento de novos produtos Plone.

A segunda aplicação, gerência de alunos usando MySQL, demonstra a integração da plataforma com outros bancos de dados. Assim, a plataforma Plone/Zope mostrou-se capaz de agregar aplicações que possam ser necessárias ao portal da Informática.

Com base na experiência obtida neste trabalho, pode-se dizer que o Plone é uma boa alternativa de gestão de conteúdo para o portal Informática UFSM. Entretanto, assim como outros sistemas de gestão de conteúdo, o Plone precisa ser moldado de acordo com as necessidades de cada instituição, sendo que ele oferece uma arquitetura que facilita essa adaptação. Além disso, Plone permite o desenvolvimento de aplicações para a instituição de forma rápida e simples.

## REFERÊNCIAS

ATWAL, S. **Building Websites with XOOBS**: a step-by-step tutorial. Birmingham: Packt Publishing, 2006.

BALASUBRAMANIAN, V. et al. Document management and Web technologies: alice marries the mad hatter. **Communications of the ACM**, [S.l.], v.41, n.7, 1998.

BAX, M. P.; PEREIRA, J. C. L. Introdução à Gestão de Conteúdos. In: WORKSHOP BRASILEIRO DE INTELIGÊNCIA COMPETITIVA E GESTÃO DO CONHECIMENTO, 2002, São Paulo. **Anais...** KM Brasil, 2002. n.3.

COELHO, E. A. **Gestão de Conteúdos na Web com Plone**. Belo Horizonte: Monografia (Especialização em Gestão Estratégica da Informação) — Escola de Ciência da Informação, Universidade Federal de Minas Gerais, 2004.

DIAS, C. A. Portal corporativo: conceitos e características. **Ciência da Informação**, [S.l.], v.30, n.1, p.50–60, 2001.

DUSTMAN, A. **MySQL for Python**. Disponível em: <<http://sourceforge.net/projects/mysql-python>>. Acesso em: 15 março 2007.

DUSTMAN, A. **Zope MySQL Database Adapter**. Disponível em: <<http://www.zope.org/Members/adustman/Products/ZMySQLDA>>. Acesso em: 15 março 2007.

ENFOLDSYSTEMS. **Definitive Guide to Plone**. [S.l.: s.n.], 2005. Disponível em: <<http://plone.org/documentation/manual/definitive-guide>>. Acesso em: 15 março 2006.

GENTLEWARE AG. **Poseidon for UML**. Disponível em: <<http://www.gentleware.com>>. Acesso em: 15 março 2007.

GRAF, H. **Building Websites with Joomla! 1.5 Beta 1**. Birmingham: Packt Publishing, 2007.

HAZRA, T. K. Building enterprise portals: principles to practice. In: ACM INTERNATIONAL CONFERENCE ON SOFTWARE ENGINEERING, 2002, Orlando, Florida. **Anais...** [S.l.: s.n.], 2002.

JOOMLA TEAM. **Joomla! CMS**. Disponível em: <<http://www.joomla.org/>>. Acesso em: 15 março 2007.

LERNER, R. M. At the forge: content management. **Linux Journal**, [S.l.], v.2003, n.108, 2003.

MAMBO FOUNDATION. **Mambo CMS**. Disponível em: <<http://mambo-foundation.org/>>. Acesso em: 15 março 2007.

MICROSOFT CORPORATION. **The Official Microsoft ASP.NET 2.0 Site**. Disponível em: <<http://asp.net/>>. Acesso em: 15 março 2007.

MYSQL AB. **MySQL AB**: the world's most popular open source database. Disponível em: <<http://mysql.com/>>. Acesso em: 15 março 2007.

OPENLDAP FOUNDATION. **OpenLDAP**: community developed ldap software. Disponível em: <<http://www.openldap.org/>>. Acesso em: 15 março 2007.

ORACLE. **Oracle Database**. Disponível em: <<http://oracle.com/>>. Acesso em: 15 março 2007.

PELLETIER, M.; LATTEIER, A. **The Zope Book**. 2.6.ed. [S.l.]: ZOPE CORPORATION, 2005. Disponível em: <[http://zope.org/Documentation/Books/ZopeBook/2\\_6Edition/](http://zope.org/Documentation/Books/ZopeBook/2_6Edition/)>. Acesso em: 15 março 2007.

PHPNUKE.ORG. **PHP-Nuke CMS**. Disponível em: <<http://www.phpnuke.org/>>. Acesso em: 15 março 2007.

PLONE FOUNDATION. **Plone CMS – Open Source Content Management System**. Disponível em: <<http://www.plone.org/>>. Acesso em: 15 março 2007.

PLONE.ORG. **Archetypes**. Disponível em: <<http://plone.org/products/archetypes>>. Acesso em: 15 março 2007.

POSTGRESQL.ORG. **PostgreSQL**: the world's most advanced open source database. Disponível em: <<http://postgresql.org>>. Acesso em: 15 março 2007.

RAMIREZ, A.; VANPEPERSTRAETE, P.; RUECKERT, A.; ODUTOLA, K.; BENNET, J.; TOLKE, L.; WULP, M. van der. **ArgoUML User Manual**. [S.l.: s.n.], 2007. Disponível em: <<http://argouml.tigris.org/documentation/manual-0.24/>>. Acesso em: 15 março 2006.

SUN MICROSYSTEMS. **JavaServer Pages Technology**. Disponível em: <<http://java.sun.com/products/jsp/>>. Acesso em: 15 março 2007.

TCHEZOPE. **Comunidade Zope e Plone Brasileira**. Disponível em: <<http://www.tchezope.org/>>. Acesso em: 15 março 2007.

THE PHP GROUP. **PHP**: hypertext preprocessor. Disponível em: <<http://php.net>>. Acesso em: 15 março 2007.

TIGRIS.ORG. **ArgoUML**. Disponível em: <<http://argouml.tigris.org>>. Acesso em: 15 março 2007.

US GOVERNMENT. **Section 508 - The Road of Accessibility**. Disponível em: <<http://www.section508.gov/>>. Acesso em: 15 março 2006.

W3C. **Web Content Accessibility Guidelines 1.0**. [S.l.: s.n.], 1999. Disponível em: <<http://www.w3.org/TR/WAI-WEBCONTENT/>>. Acesso em: 15 março 2006.

WEITZMAN, L. et al. Transforming the content management process at IBM.com. In: ACM CONFERENCE ON HUMAN FACTORS AND COMPUTING SYSTEMS, 2002, Minneapolis, Minnesota. **Anais...** [S.l.: s.n.], 2002.

XOOPS PROJECT. **Official XOOPS Website**. Disponível em: <<http://www.xoops.org/>>. Acesso em: 15 março 2007.

ZOPE CORPORATION. **Zope Community**. Disponível em: <<http://www.zope.org/>>. Acesso em: 15 março 2007.

## APÊNDICE A PERGUNTAS FREQUENTES SOBRE PLONE

Este apêndice tem como objetivo esclarecer dúvidas frequentes sobre o Plone, bem como indicar alguns endereços na Web onde existem informações adicionais sobre este CMS.

### 1. **Que *sites* usam Plone atualmente?**

Uma lista de *sites* que usam Plone se encontra no endereço `http://plone.org/about/sites`.

### 2. **Que tipo de computador servidor é recomendado para usar o Plone?**

O Plone usa muita memória e processamento de um servidor. Para pequenas a médias organizações é necessário em torno de 200 a 300 MB de memória e processador de 1 a 2 GHz. Para grandes organizações com tráfego intenso, o ideal é ter acima de 1 GB de memória e um ou mais processadores acima de 2 GHz.

### 3. **É difícil instalar o Plone?**

Não, é fácil instalar o Plone, pois ele já conta com instaladores para a maioria das plataformas. O processo de instalação é muito bem documentado e disponível nos endereços abaixo:

`http://docs.neuroinf.de/PloneBook/ch2.rst`

`http://www.tchezope.org/traducoes/gdp/ch2.rst/document_view`

### 4. **O Plone é escalável? Ele pode ser usado para *sites* com grande quantidade de acessos?**

O Plone é muito escalável. Algumas técnicas podem ser usadas para melhorar o desempenho e a escalabilidade, como balanceamento de carga, *caching* (Squid),

ZEO (*Zope Enterprise Object*). O Plone vem sendo usado sem problemas em *sites* com grande volume de acessos como por exemplo <http://www.brasil.gov.br>. Mais informações nos endereços abaixo:

<http://plone.org/documentation/tutorial/optimizing-plone>

[http://www.agentelinux.com.br/artigos/zope-howto\\_new/](http://www.agentelinux.com.br/artigos/zope-howto_new/)

## 5. Como fazer um *backup* de todo Plone site?

Existem basicamente duas formas de se fazer um *backup* de um *site* Plone. Uma delas é simplesmente copiar o arquivo `Data.fs`. Um *shell script* exemplo dessa forma de *backup* é apresentado na Figura A.1.

```
#!/bin/bash
# script simples para full backup do Plone site

#local onde vai ser armazenado o backup
DESTINO=/backup

# Onde o arquivo Data.fs está
DADOS=/opt/Plone-version/zeocluster/var/Data.fs

DATA=`date +%Y-%m-%d`

tar czf $DESTINO/$DATA.full.backup.tgz $DADOS
```

Figura A.1: Shell script para backup do Plone site

A outra forma é usar o *script* `repozo.py` do ZODB, disponível no endereço <http://cvs.zope.org/ZODB3/Tools/repozo.py>. Esse *script* é mais completo e permite fazer *backups* completos, incrementais e restauração da base de dados. Alguns exemplos de sua utilização são listados no código da Figura A.2.

## 6. Onde ficam armazenados os arquivos de *log* do Plone?

Existem dois arquivos de *log*: o `Z2.log`, que serve para registra os *logs* de tráfego para as ferramentas que geram os relatórios de acesso, e o `event.log`, onde ficam armazenados erros e mensagens do sistema. Esses arquivos geralmente ficam localizados em `/opt/Plone-version/zeocluster/client1/log/`, mas essa localização pode variar conforme o sistema operacional e a instalação.

## 7. Como instalo um produto (*add-on*) no meu Plone *site*?

```

#local onde vai ser armazenado o backup
DESTINO=/backup

# Onde o arquivo Data.fs está
DADOS=/opt/Plone-version/zeocluster/var/Data.fs

# exemplo de full backup
$ python repozo.py -B -F -v -r $DESTINO -f $DADOS

#exemplo de backup incremental
$ python repozo.py -B -v -r $DESTINO -f $DADOS

#exemplo de restauração de backups
$ python repozo.py -R -v -r $DESTINO -o $DADOS

#mais informações sobre o uso do repozo.py
$ python repozo.py -h

```

Figura A.2: Exemplos de *backup* e restauração com *repozo.py*

No endereço <http://plone.org/products> há uma lista de produtos Plone. Geralmente, para instalar um produto Plone deve-se seguir os passos:

- Descarregue e descompacte o produto no diretório de produtos Plone, geralmente localizado em `/opt/Plone-version/zeocluster/Products/`.
- Reinicialize o Zope.
- Entre como administrador do Plone *site* em “Configuração do site” → “adicionar/remover produtos”.
- Marque o novo produto a ser instalado e instale-o.
- O novo produto está instalado e pronto para uso. Em caso de erros, o registro de instalação do produto pode ser consultado no mesmo local.

## 8. Como faço para alterar o logotipo padrão do meu Plone *site*?

A maioria das tarefas de customização do Plone são feitas através da ZMI. Para alterar a imagem padrão do Plone *site* basta seguir os passos:

- Entre na ZMI (geralmente em <http://localhost:8080/manage>) e selecione o seu Plone *site*.
- Acesse `portal_skins` → `plone_images`.
- Ache `logo.jpg` na lista e clique nele.

- Clique no botão “Customize” e faça o envio da nova imagem.
- Verifique a alteração da imagem na tela inicial do seu Plone *site*.

## 9. Como faço para personalizar o meu Plone *site*?

Para uma personalização de interface, pode-se instalar e/ou desenvolver novas *skins*. Essas *skins* basicamente alteram arquivos CSS (*Cascading Style Sheets*), mudando a forma e cores do *site*. Uma personalização mais completa exige um conhecimento maior da arquitetura do Plone e das ZPT. Mais informações sobre isso estão disponíveis em:

<http://www.tchezope.org/traducoes/lp/6>

<http://plone.org/documentation/tutorial/creating-custom-style>

<http://plone.org/documentation/how-to/creating-custom-skins>