

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

**EXTENSÃO DA FERRAMENTA PLONE
PARA PUBLICAÇÃO DE NOTÍCIAS
EXTRAÍDAS DE BANCOS DE DADOS**

TRABALHO DE GRADUAÇÃO

Valdir Stumm Júnior

Santa Maria, RS, Brasil

2008

EXTENSÃO DA FERRAMENTA PLONE PARA PUBLICAÇÃO DE NOTÍCIAS EXTRAÍDAS DE BANCOS DE DADOS

por

Valdir Stumm Júnior

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^a Andrea Schwertner Charão

Co-orientador: Giana Lucca Kroth

Trabalho de Graduação N. 261

Santa Maria, RS, Brasil

2008

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

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

**EXTENSÃO DA FERRAMENTA PLONE PARA PUBLICAÇÃO DE
NOTÍCIAS EXTRAÍDAS DE BANCOS DE DADOS**

elaborado por
Valdir Stumm Júnior

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

COMISSÃO EXAMINADORA:

Prof^a Andrea Schwertner Charão
(Presidente/Orientador)

Prof. Cláudio Rocha Lobato (UFSM)

Prof. João Carlos Damasceno Lima (UFSM)

Santa Maria, 01 de Fevereiro de 2008.

"Regression Testing? If it compiles, it's good, if it boots up, it's perfect."

— LINUS TORVALDS

AGRADECIMENTOS

Gostaria de agradecer, em primeiro lugar, à minha família e à minha namorada Cristiana, que sempre me deram forças e me apoiaram, fazendo de tudo para que eu me motivasse para os estudos, principalmente em tempos difíceis, quando tudo parecia estar dando errado, me dando um motivo maior para seguir em frente, apesar das dificuldades.

Gostaria de agradecer também aos meus colegas de faculdade, principalmente àqueles que foram colegas de verdade não somente em períodos de entrega de trabalhos, mas sempre. Quero ressaltar o valor da amizade principalmente com relação aos colegas Elias, Dewes, Douglas, Marília, João Lima e Diego Friedrich, que durante todo esse tempo de graduação me mostraram que além de ser meus colegas, são amigos de verdade. Ao Elias agradeço pela parceria e pelas incontáveis risadas, ao Dewes e ao Douglas pelo companheirismo durante os cansativos finais de semana pré-provas e durante as madrugadas em claro fazendo compiladores e jogando videogame. À Marília, pelas palavras de apoio e força e pelo exemplo de pessoa que é. Ao Lima pela ajuda que sempre deu e pelas risadas no caminho da janta no RU. E ao Diego, pela parceria e lealdade com a qual sempre me tratou.

Agradeço também à minha orientadora, a Professora Andrea, pela ajuda, pelas correções, pelo apoio que me foi dado para a realização deste trabalho, pelo exemplo profissional que pretendo seguir e pela sua constante prestatividade para ajudar.

Também não posso esquecer de agradecer a minha chefe e co-orientadora Giana, pela oportunidade que me foi dada de desenvolver este trabalho como parte de meu estágio. Sou muito grato também pelos ensinamentos que recebi, pela ajuda que me foi dada e pelo exemplo de profissionalismo.

Aos membros da banca, pelas correções e sugestões que contribuíram para aumentar a qualidade deste trabalho.

A todos que citei aqui, muito obrigado, pois de uma forma ou de outra, me ajudaram a alcançar meus objetivos e sonhos.

RESUMO

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

EXTENSÃO DA FERRAMENTA PLONE PARA PUBLICAÇÃO DE NOTÍCIAS EXTRAÍDAS DE BANCOS DE DADOS

Autor: Valdir Stumm Júnior

Orientador: Prof^a Andrea Schwertner Charão

Co-orientador: Giana Lucca Kroth

Local e data da defesa: Santa Maria, 01 de Fevereiro de 2008.

O papel da informação torna-se a cada dia mais importante, principalmente no contexto de grandes organizações. É necessário, diariamente, que sejam publicadas informações relativas a processos executados e outras notícias relevantes. Para suprir a falta de um meio eficaz e acessível para publicação e armazenamento de conteúdo, surgiram os Sistemas de Gerenciamento de Conteúdo. Sistemas desse tipo buscam minimizar os esforços na parte de desenvolvimento de portais Web, deixando para os usuários apenas a tarefa de criar o conteúdo e publicá-lo. Neste trabalho, apresenta-se o desenvolvimento de um mecanismo de comunicação entre o Sistema de Gerenciamento de Conteúdo Plone e os Sistemas de Gerenciamento de Bancos de Dados utilizados por Sistemas Integrados de Gestão, os quais são amplamente utilizados para gerenciamento administrativo das grandes corporações. Através deste mecanismo, torna-se possível a construção de portais Web integrados a Sistemas de Gestão, tornando viável a publicação automática de informações extraídas dos bancos de dados destes sistemas.

Palavras-chave: Gerenciamento de conteúdo web; plone; zope; ODBC; portal web; notícia; CMS; python; sistema integrado de gestão.

ABSTRACT

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

PLONE EXTENSION FOR PUBLISHING NEWS EXTRACTED FROM DATABASES

Author: Valdir Stumm Júnior
Advisor: Prof^a Andrea Schwertner Charão
Coadvisor: Giana Lucca Kroth

The role of information inside large organizations is increasingly important. Business process status and other relevant information have to flow on a daily basis. Content Management Systems came up to supply the demand for an efficient and accessible mean for publishing and storing content. This kind of systems try to minimize the efforts in the task of developing Web portals, leaving to their users just the task of creating the content and publishing it. In this work, we present the development of a communication mechanism for connecting the Plone Content Management System and the Database Management Systems used by Enterprise Resource Planning systems, which are largely used for administrative management of large corporations. Using this mechanism, it is possible to build Web portals integrated to Enterprise Resource Planning Systems and to easily automate the publishing of information extracted from their databases.

Keywords: CMS, Plone, Web, information, ERP.

LISTA DE FIGURAS

Figura 2.1 – Ciclo de vida de um SIG em uma instituição.....	16
Figura 2.2 – Tela inicial padrão do Plone	19
Figura 2.3 – Portal Plone com alto nível de personalização visual.....	19
Figura 2.4 – Produto simples gerado com a utilização do ArchGenXml.	22
Figura 3.1 – Funcionamento básico da ferramenta a ser implementada.....	25
Figura 3.2 – Ciclo de execução de uma consulta utilizando módulo baseado na DB API	26
Figura 3.3 – Código Python para conexão a um banco de dados utilizando ODBC .	26
Figura 3.4 – Passos para criação do conteúdo NoticiaPlone e do <i>News Item</i> relaci- onado	30
Figura 3.5 – Formulário gerado pelo Archetypes para a classe NoticiaPlone	31
Figura 3.6 – Funcionamento da ferramenta com suporte a listas	33
Figura 3.7 – Tela de visualização de item NoticiaPlone e <i>News Item</i>	34
Figura 4.1 – Tela de inserção de um item NoticiaPlone	36
Figura 4.2 – Código SQL que retorna o número de alunos formados na UFSM.....	37
Figura 4.3 – Item NoticiaPlone sendo criado, pronto para ser submetido	38
Figura 4.4 – Visualização do item NoticiaPlone recém criado	38
Figura 4.5 – Visualização da notícia gerada, após sua publicação.....	39
Figura 4.6 – Notícia (<i>News Item</i>) gerada com o item NoticiaPlone alterado	39
Figura 4.7 – Código SQL alterado que retorna o número de alunos formados na UFSM no ano de 2007	40

LISTA DE TABELAS

Tabela 3.1 – Comparativo entre os módulos Python para acesso a Bancos de Dados. 29

LISTA DE ABREVIATURAS E SIGLAS

CMS	Content Management System
GPL	General Public License
SGBD	Sistema de Gerenciamento de Banco de Dados
SGC	Sistema de Gestão de Conteúdo
SIG	Sistema Integrado de Gestão
SQL	Structured Query Language
UML	Unified Modeling Language
WCMS	Web Content Management System
ZMI	Zope Management Interface
Zope	Z Object Publishing Environment
ZPL	Zope Public License

SUMÁRIO

1	INTRODUÇÃO	12
2	CONCEITOS E TECNOLOGIAS	14
2.1	Gestão de Conteúdo	14
2.2	Sistemas Integrados de Gestão	15
2.3	Plone	17
2.3.1	Arquitetura	18
2.3.2	Extensibilidade	20
2.3.3	Produtos Plone	20
3	DESENVOLVIMENTO DA FERRAMENTA	23
3.1	Projeto da Ferramenta	23
3.2	Ferramentas adicionais	25
3.2.1	Módulos Python para Acesso a Bancos de Dados	27
3.2.2	Avaliação das Alternativas	29
3.3	Implementação da Ferramenta	29
3.3.1	Definições Básicas	29
3.3.2	Implementação	30
4	RESULTADOS E AVALIAÇÃO	35
4.1	Funcionamento da Ferramenta	35
4.2	Ambiente de Testes	36
4.3	Resultado Final	37
4.4	Análise do Produto Implementado	38
5	CONCLUSÃO	41
	REFERÊNCIAS	43
	APÊNDICE A MANUAL DE INSTALAÇÃO	46
A.1	Dependências e Requisitos	46
A.2	Microsoft Windows	46
A.3	Ubuntu GNU/Linux	49
A.4	Adicionando o Produto ao Portal	50

1 INTRODUÇÃO

A quantidade e diversidade de informações geradas diariamente em uma instituição de grande porte requerem a utilização de Sistemas Integrados de Gestão (SIG) para controle e manutenção eficientes destes dados. Além de facilitar a gerência das informações, os SIGs diminuem consideravelmente os custos administrativos de uma instituição (LOZINSKY, 1996).

A implantação de um sistema de gestão de grande porte é um processo que compreende diversas etapas, tais como: preparação do ambiente para o cumprimento dos requisitos básicos previamente determinados, treinamento para os futuros usuários, instalação e configuração dos módulos que compõem o sistema, auxílio ao usuário, correção e inserção de novas funcionalidades e constantes manutenções para garantir que o sistema funcione corretamente.

Durante o desenrolar deste processo, gera-se várias informações, incluindo dados relevantes sobre a própria utilização do sistema, como por exemplo, números relativos à utilização de determinados módulos, número de usuários utilizando o sistema e demais dados que podem ser considerados de importância para o processo. Porém, estas informações não são publicadas de forma distribuída, o que dificulta a visualização dos resultados obtidos ao longo de cada etapa.

Com o advento da Internet, uma opção para suprir esta carência seria a utilização de portais Web, visando promover a comunicação entre o implantador e os usuários deste novo sistema, de modo que informações estratégicas relacionadas ao processo de implantação e de utilização do sistema possam ser tornadas públicas.

Mais recentemente, ferramentas para geração de portais têm se difundido como uma opção para o desenvolvimento rápido e padronizado de portais Web voltados ao conteúdo. Os Sistemas de Gerenciamento de Conteúdo (SGC) formam um conjunto de ferramentas

que possibilitam a criação, edição e publicação de conteúdo, envolvendo esforço mínimo por parte do administrador do portal. Algumas ferramentas dessa categoria apresentam uma característica muito importante para que o portal desenvolvido possa ter sua gama de funcionalidades ampliada: a extensibilidade. A ferramenta Plone (PLONE FOUNDATION, 2007) apresenta esta característica como um dos seus pontos mais fortes, possibilitando o desenvolvimento de novas funcionalidades de modo ágil e fácil.

O objetivo geral deste trabalho é a construção de uma ferramenta que possibilite a execução de consultas em um banco de dados relacional, a extração dos dados provenientes destas consultas para a montagem de notícias e, por fim, a publicação destas notícias em um Portal Web. Para isso, desenvolveu-se um sistema para cadastro de notícias, o qual envolve também o cadastro de consultas SQL para obtenção de dados que fazem parte do texto da notícia. Dado o contexto da implantação de SIGs, esta ferramenta pode ser aplicada para fornecer uma forma de publicação de notícias com dados relacionados ao sistema.

Este trabalho está estruturado da seguinte forma: o capítulo 2 faz uma revisão sobre conceitos e tecnologias úteis para o entendimento do trabalho em geral; o capítulo 3 trata do desenvolvimento da ferramenta que é o objetivo deste trabalho e o capítulo 4 realiza a avaliação dos resultados. Por fim, o capítulo 5 conclui o trabalho, realizando uma análise sobre o que foi desenvolvido.

2 CONCEITOS E TECNOLOGIAS

A organização da informação em instituições é um ponto relevante para o bom funcionamento destas de uma forma global. Para garantir que os dados sejam armazenados e analisados de forma consistente, as organizações recorrem à utilização dos Sistemas Integrados de Gestão, que representam uma solução bastante eficaz para a sua administração, pois além de facilitar a gestão, também diminuem os custos administrativos.

A Gestão de Conteúdo é um conceito que pode ser utilizado para a expansão da capacidade dos SIGs, pois podem possibilitar a publicação, de modo simples, de dados relativos a estes.

Desse modo, esta seção aborda os conceitos e ferramentas relacionadas com a Gestão de Conteúdo, bem como aspectos relacionados a Sistemas Integrados de Gestão.

2.1 Gestão de Conteúdo

Dado o contexto de uma organização, conteúdo é toda e qualquer informação que se mostre relevante para a instituição.

Gerir conteúdo significa utilizar um conjunto de conceitos e ferramentas que visam amenizar ou resolver os problemas de produção de conteúdo (BAX; PARREIRAS, 2003).

A Gestão de Conteúdo (GC), além da simples manipulação de conteúdo, engloba o ciclo de vida completo de edição, armazenagem, disseminação e controle de versões de conteúdos, possuindo algumas características consideradas fundamentais, tais como (BAX; PARREIRAS, 2003):

- Gestão de usuários e de seus direitos;
- Criação, edição e armazenamento de informações em diversos formatos de dados;
- Utilização de meta-dados;

- Controle do conteúdo (através de sistemas de controle de *workflow*);
- Classificação, indexação e busca das informações;
- Gestão da interface com o usuário;
- Gestão de configuração.

Estendendo este conceito, surgiu a Gestão de Conteúdo Web, que trata-se de uma especialização da Gestão de Conteúdo tradicional, somada às facilidades e praticidades que um portal Web oferece. Assim, o conteúdo é apresentado através de páginas Web, as quais são disponibilizadas e organizadas de forma a permitir fácil acesso por parte dos interessados (MARI, 2005).

Um dos problemas para a Gestão de Conteúdo, há um tempo não muito distante, era que a única forma de se realizar o gerenciamento de informações em portais Web era através de um *webmaster* que realizava as alterações manualmente sempre que necessário (BAX; PEREIRA, 2002). Porém, com o acelerado crescimento dos portais, esta tarefa tornou-se bastante demorada e trabalhosa. Neste contexto, visando reduzir a quantidade de trabalho manual exigido para realizar a tarefa de manter um *site* Web, foram criados os Sistemas de Gerenciamento de Conteúdo (SGC).

O objetivo básico dos SGCs é separar o gerenciamento do conteúdo do *design* gráfico das páginas que apresentam o conteúdo (BAX; PEREIRA, 2002). Os SGCs devem permitir o desenvolvimento colaborativo de conteúdo, além de suportar algumas funcionalidades básicas, tais como: segurança, indexação, busca, *workflow* e gestão de usuários (BAX; PEREIRA, 2002).

2.2 Sistemas Integrados de Gestão

Sistemas Integrados de Gestão (SIG) são sistemas que possuem como finalidade dar suporte e agilizar a realização de tarefas necessárias para o funcionamento de uma instituição (SOUZA, 2000). Normalmente, sistemas desse tipo são divididos em vários módulos que atuam sobre uma mesma base de dados central, cada qual sendo responsável pela cobertura de um determinado setor da instituição, de forma que as informações geradas em um módulo possam ser utilizadas de modo instantâneo em outro módulo, realizando assim uma integração geral dos diversos setores que compõem a instituição que utiliza o sistema (PADILHA; MARTINS, 2005).

Sistemas que se enquadram nesta categoria são comumente distribuídos sob a forma de um pacote de *software* comercial (SOUZA, 2000). A concepção atual de Sistemas Integrados de Gestão geralmente não contempla sistemas desenvolvidos com um fim específico para uma determinada instituição. De fato, a principal característica desses sistemas é a de serem apresentados sob a forma de um vasto pacote de programas que visa centralizar e integrar a atividade de gerência de instituições dos mais variados portes, podendo ser adaptados de acordo com as reais necessidades da organização.

A principal característica que diferencia os SIGs de outras categorias de sistemas é a adaptabilidade que é necessária para que estes sistemas conquistem espaço nas instituições, devido ao universo heterogêneo ao qual estas pertencem. Para atingir a desejada capacidade de adaptação a diversos ambientes e realidades de trabalho, os módulos do sistema devem ser genéricos o suficiente para permitir um certo grau de flexibilidade e personalização (PADILHA; MARTINS, 2005). Desse modo, a fase de projeto de um sistema desse tipo deve levar em conta necessidades de personalização de determinadas seções deste.

Tendo em mente essas características, pode-se entender que um Sistema Integrado de Gestão é um *software* composto de diversos módulos concebidos juntamente, que, de forma independente uns dos outros, compartilham um banco de informações através do qual realizam operações que resultam em um controle centralizado da instituição na qual o sistema é utilizado.

Por se tratarem, de modo geral, de sistemas de grande porte, o ciclo de vida dos SIGs não pode ficar restrito às etapas que compõem o processo de desenvolvimento de *software* de propósito geral. Pode-se dividir o ciclo de vida de um SIG em uma instituição em várias fases. Uma forma possível é dividi-lo em quatro partes: decisão, seleção, implantação e utilização (BERVIAN, 2004). A figura 2.1 mostra, de forma resumida, a seqüência de etapas do ciclo de vida de um SIG.

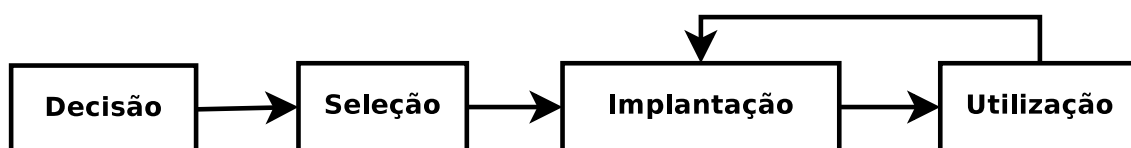


Figura 2.1: Ciclo de vida de um SIG em uma instituição

A etapa de **decisão** é caracterizada pela escolha de manter a informação atual em prol do custo que ela representa, contra melhorar a qualidade da informação, arcando com as

despesas decorrentes deste processo (BERVIAN, 2004). Ou seja, define-se a base para a utilização de um novo sistema.

A etapa seguinte é a de **seleção** do SIG por parte da instituição, a qual se dá através de uma análise detalhada sobre vários fatores que são considerados relevantes para esta (BERVIAN, 2004). Inicialmente é realizada uma pesquisa sobre alternativas existentes e, após concluída essa fase, são definidos critérios para que seja realizada uma avaliação para determinar a alternativa que melhor se adequa às necessidades da instituição.

Após concluída a etapa de seleção, é iniciada a fase de **implantação**. Nesta etapa, os módulos componentes do sistema são colocados em funcionamento. Essa fase é bastante crítica, pois devido a requisitos e à organização do novo sistema, mudanças organizacionais podem ser requeridas na instituição. Além das mudanças na organização, é necessária a realização de customizações no sistema, visando aumentar a aderência deste na instituição.

Juntamente com a etapa de implantação, inicia-se a fase de **utilização**, a qual se estende durante todo o período no qual o sistema permanecer em uso. É importante perceber que esta é uma etapa contínua do processo, pois enquanto o SIG estiver sendo utilizado, serão necessárias correções de problemas do *software*, inserção de novas funcionalidades, além da instalação de novas versões do sistema completo. Dependendo da complexidade das alterações entre uma versão e outra, essa atualização pode ser considerada uma nova implantação (SOUZA, 2000), ou seja, o sistema entra em um laço de repetição que só cessa quando o *software* deixar de ser utilizado.

Os Sistemas Integrados de Gestão formam um conceito útil para o entendimento deste trabalho por serem uma das principais aplicações da ferramenta desenvolvida.

2.3 Plone

O Plone é um poderoso *framework* para Gerenciamento de Conteúdo, desenvolvido em linguagem Python e executado sobre o servidor de aplicações Zope (PLONE FOUNDATION, 2007). Devido a sua **facilidade de utilização, personalização** e a sua **extensibilidade**, o Plone é bastante difundido como produto na área de gerenciamento de conteúdo Web.

O Plone foi o SGC escolhido para o desenvolvimento deste trabalho, por apresentar diversas características importantes, como será descrito nesta seção, e, principal-

mente, por ter sido utilizado para o desenvolvimento de Trabalhos de Graduação anteriores a este (ADIERS, 2007; SCHOLZ, 2008), assim como em uma Dissertação de Mestrado (LUCCA, 2007) nesta instituição. Tem-se assim uma base de conhecimentos pré-estabelecida e uma cadeia de trabalhos relacionados que podem ser utilizados como base para futuros desenvolvimentos e aplicações.

A utilização básica de um portal Plone para gerir conteúdo é bastante simples. O Plone é um *software* também voltado ao público não-técnico, mas mesmo assim, apresenta diversas características importantes e que acrescentam robustez aos portais que o utilizam. Essas características são: controle de versões do conteúdo, sistema de *work-flow*, gerenciamento de autorização e autenticação, dentre outros, o que dispensa grandes conhecimentos em desenvolvimento Web para a criação de um portal utilizando este SGC.

Outra característica bastante atrativa neste SGC é a facilidade e flexibilidade existente para a personalização visual e funcional dos portais nele desenvolvidos.

O visual padrão de um portal Plone é mostrado na figura 2.2. Este é o estilo com o qual um portal Plone, recém instalado e não-personalizado, é exibido no navegador. Os estilos visuais no Plone são chamados de *skins* e é possível a personalização total destes. Um *skin* engloba configurações tais como a organização na qual os elementos do portal serão exibidos no navegador, cores dos elementos, imagens a serem exibidas, *scripts*, entre outros (ENFOLDSYSTEMS, 2005).

Na figura 2.3 pode ser visto o portal do SERPRO (SERVIÇO FEDERAL DE PROCESSAMENTO DE DADOS, 2007), o qual é desenvolvido utilizando-se o SGC Plone e possui um nível de personalização bem alto. Além do SERPRO, diversos outros portais de grande porte utilizam o Plone como SGC, tal como o Portal do Software Livre (COMITÊ TÉCNICO DE IMPLEMENTAÇÃO DE SOFTWARE LIVRE, 2007).

2.3.1 Arquitetura

O Plone é uma aplicação Web que executa sobre o servidor de aplicações Zope (*Z Object Publishing Environment*). O Zope é um servidor de aplicações voltado para a Web, desenvolvido em Python (PELLETIER; LATTEIER, 2005).

O ponto de vista utilizado para o desenvolvimento do Zope considera a Web como um sistema orientado a objetos, no qual as páginas são consideradas objetos dispostos em uma estrutura hierárquica que pode ser representada sob a forma de uma URL (PELLETIER;

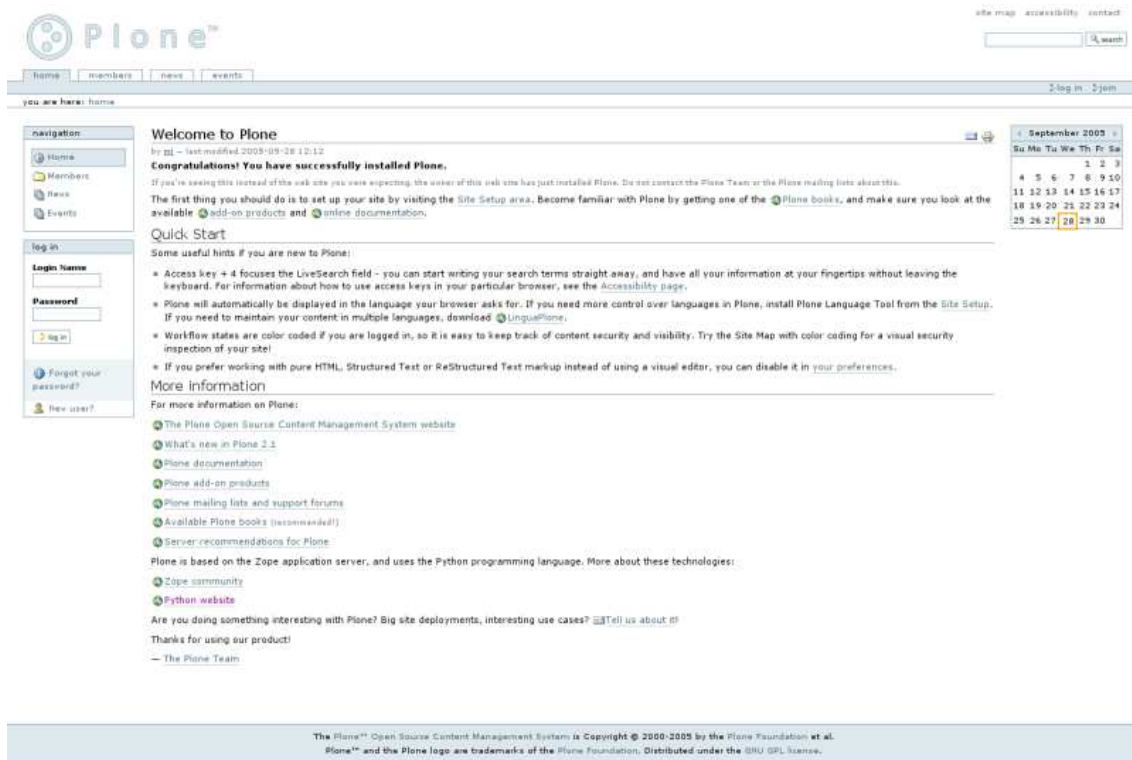


Figura 2.2: Tela inicial padrão do Plone



Figura 2.3: Portal Plone com alto nível de personalização visual.

LATTEIER, 2005). Tendo essa visão "orientada a objetos", um *site* Zope é basicamente um conjunto de objetos que contém outros objetos, seguindo a estrutura hierárquica já

descrita acima.

Outra característica importante é que o Zope é um servidor de aplicações multi-plataforma e distribuído sob a Licença ZPL (*Zope Public License*), a qual concorda com os termos da GPL (*General Public License*).

O Zope, além de ser um servidor de aplicações, pode ser considerado como um *framework* para desenvolvimento, pois oferece diversos mecanismos que agilizam e facilitam o desenvolvimento de sistemas Web.

2.3.2 Extensibilidade

Além da capacidade de customização visual do Plone, é possível a realização de customizações funcionais, visto que o Plone é um sistema bastante extensível, sendo esta uma de suas características que o diferenciam dos outros SGCs e que foi decisiva para a sua escolha na implementação deste trabalho.

Estender um portal Plone significa adicionar funcionalidades não disponíveis na instalação padrão e desenvolver sua própria funcionalidade customizada, o que amplia as áreas de aplicação de portais Plone. Esta nova funcionalidade criada aparece sob a forma de um novo tipo de conteúdo.

No Plone, especificamente, um novo tipo de conteúdo é chamado de **produto** e pode ser desenvolvido com a utilização de *frameworks*, como será descrito na próxima subseção.

2.3.3 Produtos Plone

Um Produto Plone é, basicamente, um módulo que provê, em geral, novas funcionalidades para um portal Plone (ENFOLDSYSTEMS, 2005). Esses produtos podem tanto ser obtidos via Internet quanto ser desenvolvidos de acordo com necessidades específicas. O desenvolvimento de Produtos Plone é bastante facilitado pela utilização do *framework* Archetypes, um sistema que provê diversos recursos para o desenvolvimento de novos tipos de conteúdo para o Plone e que possui várias características importantes, tais como (FOUNDATION, 2007a):

- Geração automática de formulários;
- Biblioteca alto-nível de tipos de campos, *widgets* e validadores de campos;
- Permite definição de customizações nestes;

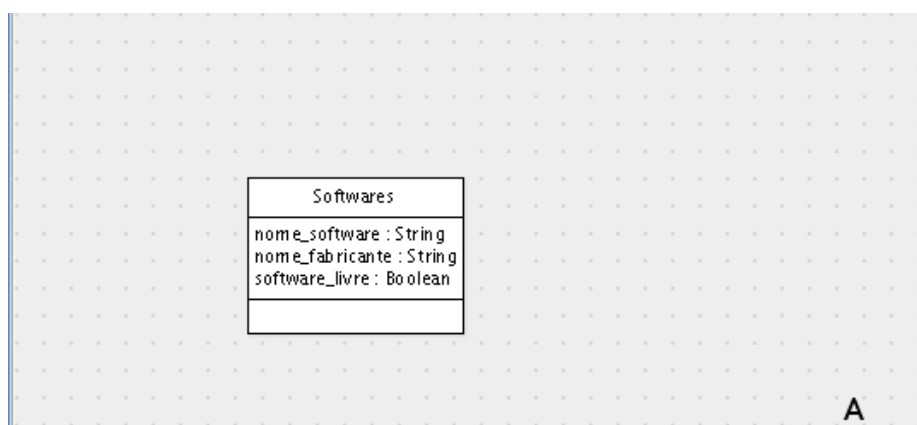
- Motor de ligação entre objetos.

Através da utilização do Archetypes, é possível a criação de novos tipos de conteúdo, utilizando-se de toda a estrutura de classes disponibilizada por este *framework*, o que garante que os novos tipos de conteúdo desenvolvidos apresentarão as mesmas características que os tipos padrão de conteúdo do Plone possuem, pois os novos tipos de conteúdo são extensões dos tipos base.

O funcionamento do Archetypes se baseia na definição de *Schemas*. Um *Schema* é como é denominado um conjunto de campos de determinado objeto. Esses campos possuem diversos atributos, como o tipo, permissões, métodos, além de outras propriedades úteis ao desenvolvimento de um produto. Para definir um objeto de um Produto, define-se um *Schema* que contenha os atributos que, juntos, representam este objeto.

Devido à direta relação existente entre os campos de um produto Plone e os atributos de uma classe, foi criada uma ferramenta que automatiza o processo de criação de Produtos Plone, o **ArchGenXml** (FOUNDATION, 2007b). Este *software* realiza a transformação de diagramas UML em classes que utilizam o Archetypes para geração de campos. Assim, o ArchGenXml realiza a leitura de arquivos UML e converte as classes em novos formulários em um portal Plone e os atributos destas classes em campos de entrada de dados nesses formulários.

Para manipulação do novo tipo de conteúdo, o Archetypes gera os formulários para inserção, edição e visualização do conteúdo relacionado à nova classe implementada. Analisando a figura 2.4, é possível verificar a relação existente entre uma classe em um diagrama UML com um produto gerado pelo ArchGenXml.



Adicionar Softwares

Padrão ■ Metadados

Título ■


Nome_software
Nome_fabricante 
 Software_livre

Figura 2.4: Produto simples gerado com a utilização do ArchGenXml.

3 DESENVOLVIMENTO DA FERRAMENTA

Neste capítulo será detalhado o processo de desenvolvimento da ferramenta de comunicação entre o Plone e os SGBDs utilizados no desenvolvimento de Sistemas Integrados de Gestão. O *software* desenvolvido visa fornecer uma ferramenta para a geração de notícias, com base em dados provenientes de uma base de dados, para publicação em um portal Web.

Primeiramente será descrito o projeto da ferramenta desenvolvida, em seguida serão detalhadas as ferramentas utilizadas no desenvolvimento e, para concluir o capítulo, será descrita a implementação em si.

3.1 Projeto da Ferramenta

O escopo do projeto desta ferramenta limita-se ao cadastro de notícias, visto que questões de segurança, tais como autorização e autenticação, e questões funcionais, tais como fluxo de documentos, controle de versões, pesquisa, localização e armazenamento não precisaram ser incluídas no processo, pois estas características já estão presentes na ferramenta de Gestão de Conteúdo escolhida, o Plone.

De acordo com avaliações feitas sobre o funcionamento real de uma ferramenta como o *software* desenvolvido neste trabalho, foram determinadas as principais características que o sistema deve possuir:

1. Permitir que sejam geradas notícias estáticas, do tipo *News Item* (padrão no Plone), com base em informações extraídas da base de dados, obtidas através da execução de uma consulta SQL cadastrada;
2. Permitir que uma notícia cadastrada possa ser re-publicada no portal, com base em informações mais atualizadas;

3. Permitir a exclusão e alteração de uma consulta geradora, sem que isso afete as notícias geradas a partir dela;
4. Permitir que vários campos do texto da notícia possam ser substituídos por dados retornados pela consulta executada;
5. Portabilidade com relação ao SGBD, devido à grande diversidade de SGBDs;
6. Prover uma interface Web para cadastro de notícias com base em informações vindas do banco de dados;

Para cumprir os requisitos definidos para a ferramenta, definiu-se o funcionamento básico desta. Para a implementação da ferramenta, decidiu-se criar um novo tipo de conteúdo, chamado de *NoticiaPlone*.

O item *NoticiaPlone* a ser cadastrado pelo usuário do sistema é composto por um título, uma consulta SQL, que será executada sobre a base de dados e que terá seus dados usados como valores dinâmicos na geração do conteúdo *News Item* a ser publicado no portal. Também faz parte do item *NoticiaPlone* um texto fixo, que representa a notícia em si, e que poderá possuir campos especiais de formatação, nos quais será descrito o nome da coluna retornada pela execução da consulta SQL que irá substituir o campo formatado.

A figura 3.1 exemplifica o funcionamento do *software*, mostrando a situação do cadastro de uma notícia, envolvendo o cadastro do texto desta e uma consulta SQL para buscar os dados. Abaixo destes campos é mostrado um exemplo de como ficaria uma notícia cadastrada com a utilização da ferramenta desenvolvida.

Na figura 3.1, o campo **[NUM_ALUNOS]** no texto da notícia será substituído pelo conteúdo retornado na coluna correspondente na execução da consulta no banco de dados no momento da geração da notícia. Desse modo é gerado o texto que irá compor o conteúdo a ser publicado no portal. No caso de a consulta executada retornar mais de uma linha, os resultados são apresentados sob forma de uma lista.

Deve ficar claro que, para a utilização da ferramenta, é necessário que o usuário possua sólidos conhecimentos sobre SQL e, principalmente, sobre o modelo de dados utilizado pelo SIG.

Texto

No ano de 2007, [NUM_ALUNOS] foram formados pela UFSM

Consulta SQL

```
select count(*) as NUM_ALUNOS
from CURSOS_ALUNOS A
where A.ANO_EVASAO=2007 and A.FORMA_EVASAO=4
```

Resultado

No ano de 2007, 360 alunos foram formados pela UFSM.

Figura 3.1: Funcionamento básico da ferramenta a ser implementada

3.2 Ferramentas adicionais

O Plone é desenvolvido na linguagem de programação Python, a qual é uma linguagem interpretada, extensível e com fortes características de orientação a objetos (The Python Language, 2007). Devido à sua natureza interpretada, Python é amplamente utilizada para o desenvolvimento de *scripts* e também para a programação de aplicações de grande porte. Aplicações escritas em Python podem ser desenvolvidas em várias plataformas, bastando que estas possuam um interpretador Python disponível (ROSSUM, 2006).

Como consequência direta de sua estrutura organizada e simples, o desenvolvimento de módulos para Python deve respeitar uma série de padrões pré-definidos. Os módulos para acesso a banco de dados em Python, por exemplo, devem seguir o padrão determinado pela **Python DB API** a qual fornece especificações a serem seguidas, como nomes e assinatura de métodos, dentre outras.

Um programa Python que utiliza um módulo compatível com a DB API pode seguir um procedimento padrão para executar uma consulta, seja qual for o módulo e o banco de dados utilizado. A figura 3.2 mostra o diagrama de fluxo que representa o procedimento.

O fluxograma apresentado na figura 3.2 pode ser representado em um *script* Python

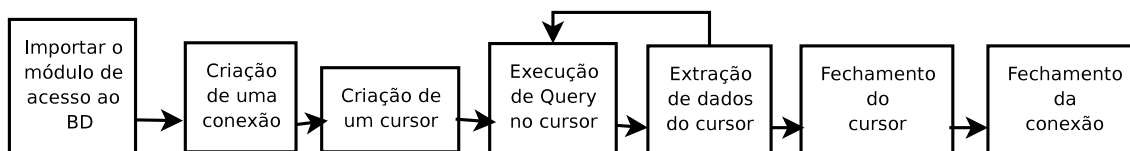


Figura 3.2: Ciclo de execução de uma consulta utilizando módulo baseado na DB API

semelhante ao código 3.3.

```
#!/usr/bin/env python
```

```
import pyodbc
```

```
conn = pyodbc.connect('DSN=sample;UID=nome;PWD=*****')
```

```
cursor = conn.cursor()
```

```
cursor.execute('SELECT * FROM nome_tabela')
```

```
rows = cursor.fetchall()
```

```
for item in rows:
```

```
    print item
```

```
cursor.close()
```

```
conn.close()
```

Figura 3.3: Código Python para conexão a um banco de dados utilizando ODBC

Para migrar o *script* citado para obtenção de dados de outro SGBD, basta substituir a linha onde é realizada a importação do módulo, no caso de este seguir as especificações definidas pela Python DB API.

Essa padronização torna Python uma linguagem muito atrativa para desenvolvimento de aplicações que devem realizar acesso a bancos de dados, pois, uma vez implementado um *software* que realiza operações sobre uma base de dados, basta realizar a troca do módulo para que o programa passe a funcionar acessando o novo SGBD desejado, não acarretando em grandes alterações no código-fonte do programa. Este fato, aliado ao fato de o Plone ser escrito em Python, determinaram a escolha desta linguagem de programação para a implementação deste trabalho.

Seguindo os padrões definidos pela especificação da DB API, foram desenvolvidos diversos módulos para acesso aos mais variados bancos de dados relacionais. Buscando maximizar a compatibilidade da ferramenta com os SIGs, foram realizados estudos sobre os possíveis módulos para acesso a SGBDs que poderiam ser utilizados na implementação deste trabalho.

3.2.1 Módulos Python para Acesso a Bancos de Dados

Dentre vários módulos para acesso aos mais diversos SGBDs relacionais existentes, foram analisadas as possibilidades de utilização de quatro módulos: dois mais específicos, PyDb2 (O'BRIEN; TURGEMAN, 2007) e `ibm_db` (IBM OPEN SOURCE APPLICATION ENABLEMENT TEAM, 2007), e dois mais genéricos, PyODBC (KLEEHAMMER; KLEEHAMMER, 2007) e mxODBC (EGENIX.COM, 2007). As próximas subseções descrevem os módulos testados e as escolhas realizadas.

3.2.1.1 IBM DB2

O DB2 é um SGBD relacional comercial desenvolvido pela IBM (IBM, 2007). Por ser um sistema bastante robusto, ele é largamente utilizado por SIGs de diversos portes para o armazenamento e gerenciamento dos dados. Devido à sua grande utilização nos SIGs disponíveis no mercado, este trabalho, inicialmente, se deteve a realizar testes e implementações utilizando-se módulos para acesso exclusivamente ao SGBD DB2, que são os módulos **PyDB2** e `ibm_db`.

PyDb2 é um módulo que provê uma interface compatível com a especificação DB API, para conexão com o SGBD IBM DB2. Dentre as alternativas existentes, esta foi a primeira a ser testada e avaliada. Utilizando este módulo, é possível realizar consultas, extrair resultados, além de manipular transações (O'BRIEN; TURGEMAN, 2007).

Porém, durante os testes foi constatada a existência de um problema no gerenciamento de memória na utilização do módulo PyDb2 em conjunto com a versão 2.5 da linguagem Python, o que acabou por comprometer a sua utilização neste trabalho, pois, apesar de o *software* a ser desenvolvido utilizar a versão 2.4 de Python, o uso deste módulo comprometeria futuras migrações para versões mais recentes da linguagem. Sendo assim, decidiu-se testar outro módulo para acesso ao SGBD DB2, o `ibm_db`.

O módulo `ibm_db` é o pacote oficial da IBM para conexão de programas Python ao SGBD DB2. Após o lançamento do módulo `ibm_db`, foram realizados os testes para determinar a viabilidade da utilização deste módulo. Após uma série de testes, foi constatado que, apesar de estar em uma versão *beta* e de não seguir as especificações da DB API, o módulo é estável e não apresenta defeitos que comprometam a sua utilização no desenvolvimento deste trabalho.

3.2.1.2 ODBC

A utilização de módulos para acesso a um SGBD específico contraria um dos requisitos da ferramenta proposta neste trabalho, que é o funcionamento na maior diversidade de SGBDs possível. Sendo assim, partiu-se para alternativas mais genéricas, baseadas no padrão *Open Database Connectivity (ODBC)*. ODBC é uma interface para acesso a dados em SGBDs heterogêneos relacionais ou não-relacionais (Microsoft Support, 2007). Esta interface provê um conjunto padrão de métodos para acesso aos SGBDs, fornecendo independência de linguagem de programação, SGBD e sistema operacional (SO).

Devido a diversas implementações do padrão, é possível utilizar ODBC em uma grande variedade de sistemas operacionais. Para SOs da família Microsoft Windows, existe a implementação da Microsoft. Para sistemas baseados em UNIX, existem duas alternativas bastante utilizadas: o iODBC (OPENLINK, 2007) e o unixODBC (EASY-SOFT, 2006), ambas distribuídas sob licença GPL (GNU, 2007).

Para o desenvolvimento deste trabalho, o padrão ODBC aparece como uma importante alternativa, visto que a independência de SO e de SGBD é desejada, pois aumenta a abrangência da ferramenta desenvolvida, possibilitando a sua utilização em um maior número de SIGs do que no caso de desenvolver a ferramenta voltada para acesso a um SGBD específico. Além disso, a independência de sistema operacional é importante pelo fato de que existe uma variedade de SOs nos quais os SIGs são instalados. Devido a isso, foram analisados dois módulos Python para acesso a bancos de dados via ODBC, o **PyODBC** e o **mxODBC**

O **Pyodbc** é um módulo *open-source*, distribuído sob licença GPL e que segue as especificações da DB API. Possui um *software* instalador para a plataforma Windows e disponibiliza também o código-fonte para compilação em ambientes UNIX. Este módulo foi testado para acesso a bancos de dados DB2 e Oracle, nos quais se mostrou uma solução estável e de boa utilização.

Desenvolvido pela eGenix (EGENIX.COM, 2007), o **mxODBC** constitui uma alternativa comercial que também implementa a DB API, e tem por objetivo possibilitar a comunicação entre *scripts* Python e *drivers* ODBC para os mais diversos SGBDs. Porém, o fato de a solução ser proprietária e comercial impede sua utilização no presente trabalho.

3.2.2 Avaliação das Alternativas

Após instalar e testar os módulos descritos anteriormente, foi realizada uma avaliação sobre estes para determinar qual deles melhor se encaixa nos propósitos da ferramenta. A tabela 3.1 mostra os fatores que foram levados em consideração para a realização da escolha pelo módulo a ser utilizado para a implementação da ferramenta.

Tabela 3.1: Comparativo entre os módulos Python para acesso a Bancos de Dados.

Módulo	Licença	Multi-plataforma	SGBDs	Status do Projeto
PyDb2	GPL	Sim	IBM DB2	Parado desde 2005
ibm_db	Apache	Sim	IBM DB2	Ativo
pyODBC	MIT	Sim	Vários	Ativo
mxODBC	Proprietário	Sim	Vários	Ativo

Para a realização da escolha, o principal fator analisado foi a portabilidade, tanto na questão de número de SGBDs suportados, quanto na flexibilidade com relação ao sistema operacional. Neste ponto, o pyODBC se destacou, pois cumpre os dois requisitos. Além destes pontos, a licença de distribuição deste módulo também foi um fator decisivo, visto que o outro módulo disponível para acesso a dados via ODBC é um *software* comercial. Assim, foi realizada a escolha pelo módulo pyODBC, devido ao fato de que, através de sua utilização, é possível acessar diversos SGBDs, bastando que estes possuam *drivers* ODBC disponíveis, possibilitando assim uma maior abrangência da ferramenta a ser desenvolvida com relação aos SIGs existentes.

3.3 Implementação da Ferramenta

De posse do escopo da ferramenta, da lista de requisitos e das ferramentas a serem utilizadas, pode-se iniciar a fase de implementação, na qual os itens que foram definidos anteriormente são traduzidos para a linguagem de programação Python. Esta seção descreverá os procedimentos técnicos que foram realizados para a implementação do *software* que foi desenvolvido como resultado deste trabalho.

3.3.1 Definições Básicas

Antes de iniciar a implementação propriamente dita, foram definidas algumas restrições para o desenvolvimento da ferramenta. O ponto principal a ser discutido é a utilização dos recursos de conexão com os SGBDs.

Para evitar um número demasiado de acessos ao banco de dados, a conexão a estes é feita somente no momento de geração da notícia, evitando o grande número de conexões que seria gerado caso fosse utilizada uma abordagem de conexão ao banco de dados a cada acesso para visualização das notícias.

A figura 3.4 ilustra a sequência de passos a ser realizada quando for criado um conteúdo do tipo `NoticiaPlone`. Assim, como mostra a figura 3.4, é criado um conteúdo do tipo *"News Item"* com base nos dados do item `NoticiaPlone`. Sendo um conteúdo do tipo padrão de notícia do Plone, seu conteúdo permanecerá estático no banco de dados do Plone, não necessitando, no momento da visualização, acesso às informações do banco de dados relacional do qual foram extraídos os dados para a montagem da notícia, evitando o *overhead* causado pelo acesso a um SGBD.

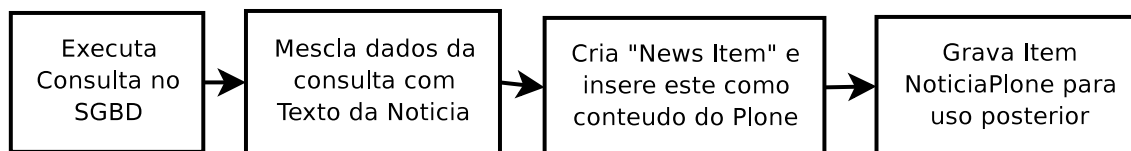


Figura 3.4: Passos para criação do conteúdo `NoticiaPlone` e do *News Item* relacionado

3.3.2 Implementação

Analisando-se a lista de requisitos projetada, foi possível definir a classe responsável pelo controle dos dados referentes ao Produto `NoticiaPlone`. Conforme já foi determinado anteriormente no projeto, na seção 3.1, a classe `NoticiaPlone` é composta pelos seguintes atributos:

- **Título:** *String*;
- **SQL:** *String*;
- **Texto:** *String*.

Além destes atributos, a classe `NoticiaPlone` possui alguns métodos que serão descritos mais adiante, visto que sua utilização se tornará mais clara durante o decorrer deste texto.

Tendo a classe a ser implementada definida em uma ferramenta de diagramação UML, pode-se usar a ferramenta ArchGenXml para gerar o Produto Plone que utilize o *framework* Archetypes. Após a geração do produto Plone, tem-se um componente pronto

para instalação e utilização em um portal. Assim, para a classe `NoticiaPlone` representada através de UML é gerado um novo tipo de conteúdo Plone.

A figura 3.5 mostra o formulário de edição gerado pela ferramenta Archetypes, para o cadastro de conteúdo do tipo `NoticiaPlone`.

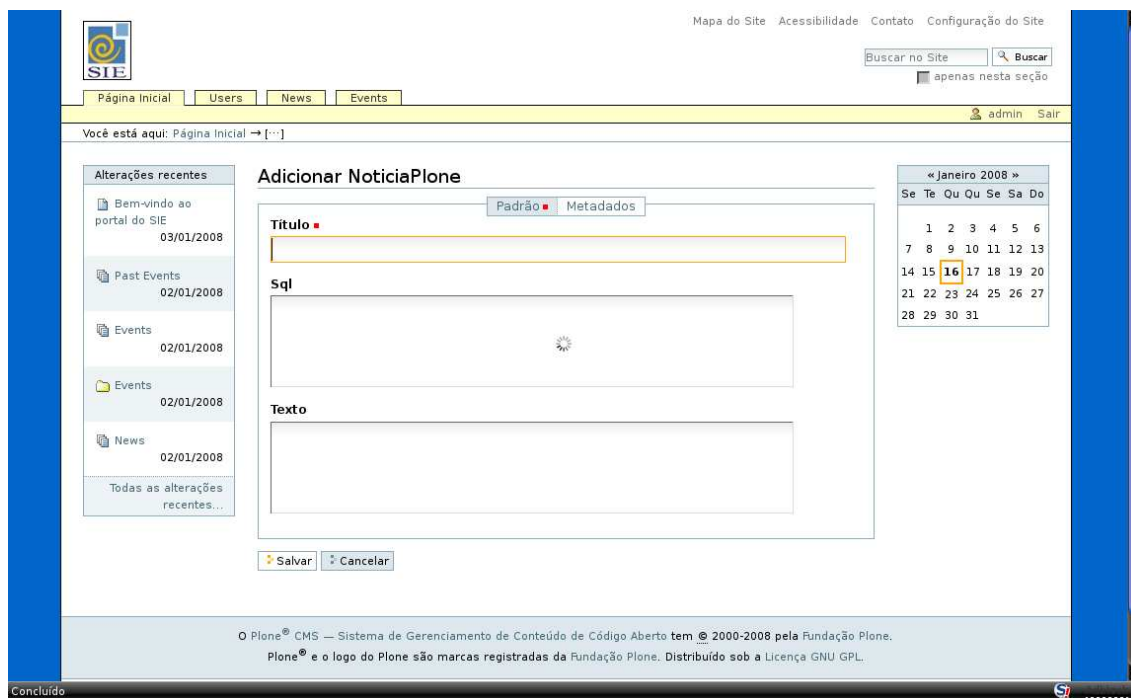


Figura 3.5: Formulário gerado pelo Archetypes para a classe `NoticiaPlone`

O produto gerado com o ArchGenXml é bastante simples e não realiza as ações que são desejadas para o Produto `NoticiaPlone`. Dessa forma, se faz necessária a customização do produto gerado para que este atenda aos requisitos previamente estipulados. Esta customização é realizada alterando-se o código gerado pelo ArchGenXml.

Python, assim como outras linguagens de programação, define um padrão para o desenvolvimento de métodos para acesso e edição dos dados. Assim, é definido que o método que realiza o acesso de leitura nos dados é um método *accessor*, enquanto que o método que realiza a escrita nos dados é um método *mutator*. Assim sendo, o *framework* Archetypes, em sua definição de *Schemas*, possibilita que se registre um método para ser chamado como *accessor*, outro método para ser chamado como *mutator* e ainda métodos para validação, os chamados *validators*. Cada atributo criado com o Archetypes pode ter esses métodos cadastrados e a utilização destes deve ser de acordo com a necessidade:

- *Accessor*: utiliza-se este método quando é necessário que se realize uma transformação nos dados antes que eles sejam mostrados na tela em modo de visualização;

- *Mutator*: utiliza-se este método quando se deseja que os dados sofram alguma transformação antes de serem enviados para armazenamento;
- *Validators*: para o campo *validators*, o Archetypes fornece um conjunto de métodos pré-implementados para realizar a validação dos dados fornecidos pelo usuário ao editar informações no formulário. Também é possível o desenvolvimento de métodos customizados.

Para a implementação deste trabalho, é de especial importância a utilização de um método que realize uma transformação nos dados após eles serem submetidos. Para isso, no produto gerado com o ArchGenXml foi configurado um método *mutator* para o campo *Texto*, ou seja, após o item *NoticiaPlone* ter sido submetido, o método será chamado para realizar a transformação nas informações, antes de inseri-las na base de dados.

Após terem sido realizadas as operações para transformação do texto, é adicionado um conteúdo do tipo *News Item* padrão para itens de notícia no Plone. O conteúdo dessa notícia é composto pelo mesmo título dado para o item *NoticiaPlone* cadastrado e o texto da notícia é composto pelo texto fixo cadastrado pelo usuário, com os campos dinâmicos substituídos pelos resultados da execução da consulta no momento do cadastro.

Além do suporte a valores únicos extraídos do SGBD, com a sintaxe **[NOME_CAMPO]**, foi implementado o suporte a listas de dados, possibilitando assim a geração de notícias com listagens de valores. Para a utilização de listas de valores, foi definida a sintaxe **{NOME_CAMPO}**. A figura 3.6 ilustra o funcionamento desta funcionalidade.

Com a implementação do suporte a listas de valores, surge a possibilidade de criação de notícias com listas de dados. Alguns exemplos de aplicações desta funcionalidade são: exibição da lista de classificados em um concurso, exibição dos nomes dos alunos matriculados em um determinado curso, dentre outros.

É importante salientar a diferença existente entre um conteúdo do tipo *NoticiaPlone* e um conteúdo do tipo *News Item*. O item *NoticiaPlone* é cadastrado pelo usuário da ferramenta, e como já descrito, utiliza a consulta SQL cadastrada para buscar informações na base de dados e gerar um conteúdo do tipo *News Item*, para que este fique visível no portal sob a forma de uma notícia padrão de portais Plone. Então, o item *NoticiaPlone* não fica visível aos usuários do portal, mas apenas aos administradores deste, para que gerem *News Item* quando for conveniente. A figura 3.7 enfatiza a diferença existente entre os dois itens, mostrando que os dois conteúdos são persistidos como objetos separados.

Texto

Os seguintes alunos foram selecionados: {NOME_ALUNO}

Consulta SQL

```
select NOME_ALUNO  
from ALUNOS  
where MEDIA_ALUNO > 7
```

Resultado

Os seguintes alunos foram selecionados:
João José
Mario Marcos
Marta Maria
Pedro José
Roberto Sergio

Figura 3.6: Funcionamento da ferramenta com suporte a listas

Visão
Edição
Propriedades
Compartilhamento

Ações ▼
Estado: Privado ▼

novo

por [admin](#) — última modificação 31/01/2008 14:21

Sql:
select count(*) as NUM_ALUNOS from dbsm.cursos_alunos a where a.forma_evasao_item = 4 and a.ano_evasao = 2006

Texto:
No ano de 2006, [NUM_ALUNOS] alunos foram formados pela UFSM.

[Enviar isso](#) — [Imprimir isso](#) —

Visão
Edição
Compartilhamento
Histórico

Ações ▼
Estado: Publicado ▼

novo

por [admin](#) — última modificação 31/01/2008 14:22

No ano de 2006, 2954 alunos foram formados pela UFSM.

[Enviar isso](#) — [Imprimir isso](#) —

⊕ **Histórico**

Figura 3.7: Tela de visualização de item NoticiaPlone e *News Item*

4 RESULTADOS E AVALIAÇÃO

Todo projeto de *software* passa por uma fase de validação, na qual avalia-se se o que foi implementado está de acordo com os requisitos levantados.

Este capítulo irá apresentar a ferramenta implementada, bem como exemplos de sua utilização e uma análise prática do processo de instalação, configuração e utilização da ferramenta.

4.1 Funcionamento da Ferramenta

De acordo com os requisitos levantados e seguindo o projeto previamente definido, foi realizada a implementação da ferramenta NoticiaPlone.

Conforme descrito no capítulo 3, a ferramenta desenvolvida deve possuir três atributos editáveis via formulário em uma página Web: título, consulta SQL e texto. A figura 4.1 representa a tela de inserção de um item NoticiaPlone ao portal.

Conforme pode ser visualizado na figura 4.1, os três campos acima descritos estão disponíveis para edição por parte do usuário, tornando possível para este a inserção de um item NoticiaPlone ao seu portal Plone.

Após o preenchimento dos campos na interface, deve-se submeter os dados para que estes sejam gravados na base de dados do Plone, podendo, a partir daí, ser transformados em um conteúdo do tipo *News Item*, tornando possível a publicação da notícia gerada.

No momento em que é realizado o envio das informações para o banco de dados utilizado pelo Plone, é também criado um conteúdo do tipo *News Item*, que contém o mesmo título do item NoticiaPlone, e um campo texto, o qual é composto pela mescla dos dados que foram obtidos do banco de dados do SIG com as porções de texto fixo cadastrados pelo usuário. Esta notícia é gerada com o *status* **privado**, possibilitando assim ao administrador do portal revisar a notícia antes que esta seja tornada pública.

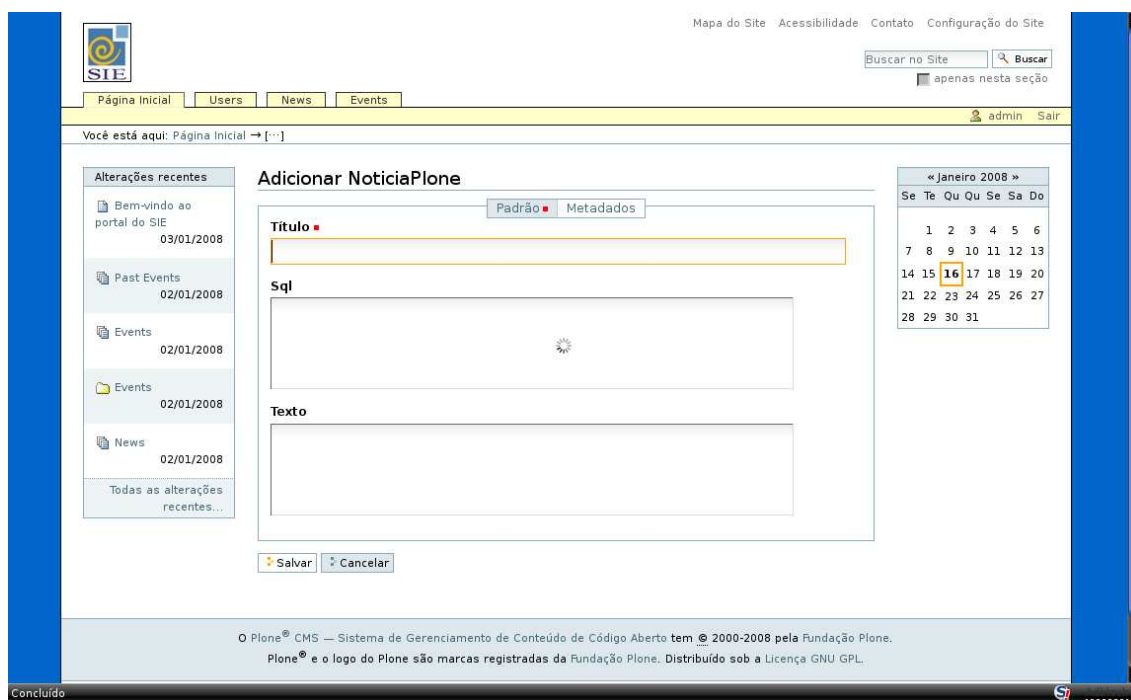


Figura 4.1: Tela de inserção de um item NoticiaPlone

4.2 Ambiente de Testes

Visando demonstrar o funcionamento prático da ferramenta, foi criado um exemplo de utilização do *software*. Para a execução deste exemplo, foi realizada a instalação da ferramenta em um portal Plone executando sobre um servidor Zope em uma máquina servidora com as seguintes especificações:

- **Processador:** Intel Pentium IV 1.8 Ghz;
- **Memória Principal:** 512 MB;
- **Sistema Operacional:** GNU/Linux versão 2.6.22-14.

Para realizar a instalação do sistema, que compreende desde a fase de configuração dos requisitos do ambiente até a instalação do produto Plone implementado, foram seguidos os passos descritos no manual de instalação/utilização que também foi gerado como produto deste trabalho. Este manual pode ser encontrado no apêndice A.

Após instalado o sistema, o produto Plone foi configurado para realizar acesso ao banco de dados de réplica do SIG utilizado na Universidade Federal de Santa Maria (UFSM), o Sistema de Informações para o Ensino (SIE). Sendo assim, tornou-se possível a criação da notícia de exemplo.

4.3 Resultado Final

Tendo um ambiente de testes pronto, o passo seguinte foi a criação de um exemplo de utilização prática da ferramenta, relacionada ao contexto do ambiente.

Como exemplo de utilização, resolveu-se criar uma notícia baseada em dados relacionados à UFSM e que apresentasse um caráter temporal. Assim, foi criada uma notícia que mostra o número de alunos formados pela UFSM no ano de 2006. Tendo isso em mente, foi criado o texto da notícia, o qual foi definido como o trecho abaixo:

"No ano de 2006, [NUM_ALUNOS] alunos foram formados pela UFSM."

O campo delimitado por colchetes representa o trecho de conteúdo dinâmico, o qual será buscado na base de dados do SIE, através da consulta mostrada no código 4.2, que foi cadastrada como a consulta que irá fornecer os dados para preenchimento dos campos no texto da notícia.

```
SELECT
  COUNT(*) AS num_alunos
FROM
  cursos_alunos a
WHERE
  a.ano_evasao = 2006
  AND a.forma_evasao_item = 4
```

Figura 4.2: Código SQL que retorna o número de alunos formados na UFSM

As figuras 4.3, 4.4 e 4.5 mostram a seqüência de passos para a geração da notícia que mostre o número de alunos formados no ano de 2006 na Universidade Federal de Santa Maria (UFSM). A figura 4.3 mostra a situação de um item do tipo *NoticiaPlone* sendo cadastrado, com seus campos corretamente preenchidos. Na figura 4.4 pode ser vista a tela seguinte à submissão dos dados, a qual mostra o item *NoticiaPlone* recém cadastrado. Na seqüência, a figura 4.5 mostra a notícia do tipo *News Item*, gerada a partir do item de conteúdo *NoticiaPlone*, após sua publicação no portal, já contendo os dados extraídos do banco de dados do SIG.

Cumprindo um dos requisitos básicos exigidos pela projeto da ferramenta, foi realizado o teste da alteração do item *NoticiaPlone* previamente criado, objetivando a geração de um novo conteúdo do tipo *News Item*, porém baseado em uma consulta SQL modificada. A figura 4.6 mostra a Notícia gerada através da alteração do item *NoticiaPlone* anteriormente cadastrado.

Adicionar NoticiaPlone

Padrão Metadados

Título

Alunos Formados 2006

Sql

```
SELECT COUNT(*) AS NUM_ALUNOS
FROM DBSM.CURSOS_ALUNOS A
WHERE A.ANO_EVASAO = 2006 AND A.FORMA_EVASAO_ITEM = 4
```

Texto

No ano de 2006, [NUM_ALUNOS] alunos foram formados pela UFSM.

Salvar Cancelar

Figura 4.3: Item NoticiaPlone sendo criado, pronto para ser submetido

Visão Edição Propriedades Compartilhamento

Ações Estado: Privado

Alunos Formados 2006

por [admin](#) — última modificação 23/01/2008 15:42

Sql:

```
SELECT COUNT(*) AS NUM_ALUNOS FROM DBSM.CURSOS_ALUNOS A WHERE A.ANO_EVASAO = 2006
AND A.FORMA_EVASAO_ITEM = 4
```

Texto:

No ano de 2006, [NUM_ALUNOS] alunos foram formados pela UFSM.

Enviar isso — Imprimir isso —

Figura 4.4: Visualização do item NoticiaPlone recém criado

O trecho de código 4.7 mostra a alteração realizada no código SQL que faz a busca dos dados no banco.

4.4 Análise do Produto Implementado

Através do exemplo realizado pôde-se verificar que a ferramenta desenvolvida cumpre os requisitos determinados na seção 3.1.

É importante ressaltar o fato de o produto Plone desenvolvido dar suporte a uma

The screenshot shows a web portal interface. At the top right, there are links for 'Mapa do Site', 'Acessibilidade', and 'Contato'. Below these is a search bar with the text 'Buscar no Site' and a 'Buscar' button. A navigation menu includes 'Página Inicial', 'Users', 'News', 'Events', 'Urgente', and 'Alunos Formados 2006'. The main content area displays a news item titled 'Alunos Formados 2006' with the text 'No ano de 2006, 2954 alunos foram formados pela UFISM.' and options to 'Enviar Isso' and 'Imprimir Isso'. A calendar for January 2008 is visible on the right, with the 23rd highlighted. On the left, there is a sidebar with 'Notícias' (Alunos Formados 2006, 23/01/2008), 'Urgente' (18/01/2008), and an 'Acessar' section with a login form for 'Nome do Usuário' and 'Senha'.

Figura 4.5: Visualização da notícia gerada, após sua publicação

The screenshot shows the same web portal interface but with a different news item titled 'Alunos Formados 2007' and the text 'No ano de 2007, 1459 alunos foram formados pela UFISM.' The navigation menu now includes 'Alunos Formados 2007'. The sidebar 'Notícias' shows 'Alunos Formados 2007' (23/01/2008) and 'Alunos Formados 2006' (23/01/2008). The 'Acessar' section shows the login form with 'Nome do Usuário' set to 'admin' and 'Senha' masked with asterisks. The calendar on the right remains the same, with the 23rd highlighted.

Figura 4.6: Notícia (*News Item*) gerada com o item *NoticiaPlone* alterado

grande diversidade de SGBDs e também poder ser utilizado nas plataformas Windows e Linux. Essa característica é de extrema importância a ferramenta, visto que amplia bastante sua área de atuação, podendo ser utilizado em um maior número de SIGs.

Nesse momento, faz-se necessária também uma análise sobre os aspectos que podem ser melhorados na ferramenta. Um dos pontos importantes é a verificação que é realizada

```
SELECT
    COUNT(*) AS num_alunos
FROM
    cursos_alunos a
WHERE
    a.ano_evasao = 2007
    AND a.forma_evasao_item = 4
```

Figura 4.7: Código SQL alterado que retorna o número de alunos formados na UFSM no ano de 2007

sobre a consulta SQL preenchida pelo usuário. No momento, é realizada uma varredura no texto da consulta, buscando identificar a ocorrência de um comando que possa alterar a situação dos dados no banco de dados. Assim somente consultas podem ser executadas no banco de dados, garantindo a integridade das informações armazenadas. Caso o comando SQL contenha alguma instrução de alteração, a operação não é realizada. A busca por alguma ferramenta que realize essa atividade é uma das atividades que ainda devem ser realizadas. Além desta, pode-se expandir as capacidades da ferramenta, possibilitando a utilização de texto em formato *Rich Text*, utilização de imagens extraídas do SGBD, entre outras possibilidades.

5 CONCLUSÃO

Neste trabalho, desenvolveu-se uma ferramenta Web para geração de notícias com informações oriundas de SGBDs relacionais utilizando o Sistema de Gerenciamento de Conteúdo Plone. O *software* desenvolvido tem como principal objetivo prover uma forma de publicação de informações relevantes extraídas de Bancos de Dados de um sistema qualquer e servir como uma ferramenta de apoio a este sistema. Uma possível aplicação desta ferramenta é no apoio ao processo de implantação de Sistemas Integrados de Gestão (SIG), possibilitando ao responsável pelo SIG cadastrar uma notícia que, através da execução de uma consulta SQL no banco de dados deste, busque dados relativos aos processos desenvolvidos e publique-a em um portal com as informações que foram obtidas através da consulta, mescladas ao texto fixo da notícia, também previamente cadastrado.

Para o desenvolvimento deste trabalho foram pesquisadas e testadas diversas ferramentas que tornassem possível a implementação do sistema, tais como o SGC Plone, módulos em Python para acesso a bancos de dados, além de ferramentas genéricas para acesso a bancos de dados heterogêneos.

A utilização de SGCs para a criação de portais Web voltados ao conteúdo também foi confirmada na realização deste trabalho, pois o SGC utilizado, o Plone, apresentou-se como uma excelente opção para o desenvolvimento do trabalho, tendo a sua extensibilidade como ponto mais importante, pois permite o desenvolvimento rápido e padronizado de aplicações Web adequadas às necessidades da instituição que a utiliza.

Pôde-se verificar na prática o funcionamento da ferramenta desenvolvida, através de testes sobre um Banco de Dados do SIG utilizado na UFSM, nos quais a ferramenta mostrou cumprir com os requisitos determinados na fase de projeto.

A criação do produto Plone gerado por este trabalho é de grande importância, pois além de poder ser usado como ferramenta de auxílio a SIGs de um modo geral, possibi-

lita futuras expansões das funcionalidades desenvolvidas para este produto. Alguns itens propostos como trabalhos futuros são:

- Inserção de novas possibilidades de formatação dos campos dinâmicos no texto da notícia, dando suporte a imagens e expressões básicas;
- Desenvolvimento de um *software* para a realização automatizada da instalação e configuração de um portal Plone customizado, de acordo com a instituição;
- Possibilidade de cadastro de consultas SQL independentemente, podendo então realizar a vinculação destas a itens NoticiaPlone;
- Aplicação da ferramenta ao novo portal do Curso de Ciência da Computação da UFSM, o qual foi desenvolvido utilizando-se o SGC Plone.

REFERÊNCIAS

ADIERS, D. R. **Gerenciamento de Conteúdo na Web usando o Plone**: aplicação ao portal da informática da ufsm. Santa Maria: Curso de Ciência da Computação. Universidade Federal de Santa Maria, 2007.

BAX, M. P.; PARREIRAS, F. S. **Gestão de Conteúdos com Softwares Livres**. Belo Horizonte: Programa de Pós-Graduação em Ciência da Informação, Escola de Ciência da Informação, Universidade Federal de Minas Gerais (UFMG), 2003.

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.

BERVIAN, A. E. **Critérios para a decisão de personalização de processos na implantação de sistemas ERP**. São Leopoldo: Universidade do Vale dos Sinos, 2004.

COMITÊ TÉCNICO DE IMPLEMENTAÇÃO DE SOFTWARE LIVRE. **Portal Software Livre**. Disponível em: <http://www.softwarelivre.gov.br> . Acesso em: 30 de novembro de 2007.

EASYSOFT. **unixODBC**. Disponível em: <http://www.unixodbc.org> . Acesso em: 21 de janeiro de 2008.

EGENIX.COM. **mxODBC**. Disponível em: <http://www.egenix.com> . Acesso em: 22 de novembro de 2007.

ENFOLDSYSTEMS. **Definitive Guide to Plone**. [S.l.: s.n.], 2005. Disponível em: <<http://plone.org/documentation/manual/definitive-guide>>. Acesso em: 22 de setembro 2007.

FOUNDATION, P. **Archetypes Developer Manual**. Disponível em: <http://plone.org/documentation> . Acesso em: 30 de novembro de 2007.

FOUNDATION, P. **ArchGenXml**. Disponível em: <http://plone.org/products/archgenxml> . Acesso em: 18 de janeiro de 2008.

GNU. **General Public License**. Disponível em: <http://www.gnu.org> . Acesso em: 30 de novembro de 2007.

IBM. **IBM**. Disponível em: <http://www.ibm.com> . Acesso em: 30 de novembro de 2007.

IBM OPEN SOURCE APPLICATION ENABLEMENT TEAM. **ibm_db**. Disponível em: http://cheeseshop.python.org/pypi/ibm_db/0.1.0 . Acesso em 27 de Novembro de 2007.

KLEEHAMMER, D.; KLEEHAMMER, M. **PyODBC**. Disponível em: <http://www.sourceforge.net/projects/pyodbc> . Acesso em: 22 de novembro de 2007.

LOZINSKY, S. Software: tecnologia de negócio. In: KM BRASIL, 1996, Rio de Janeiro. **Anais...** [S.l.: s.n.], 1996.

LUCCA, G. **PLONARQ**: gerenciamento eletrônico de documentos arquivísticos baseado em software livre. Santa Maria: Programa de Pós-Graduação em Engenharia de Produção, Universidade Federal de Santa Maria, 2007.

MARI, F. de. Obtendo produtividade e conhecimento através do investimento em tecnologia da informação. **Portal KMOL**, [S.l.], 2005. Disponível em: http://www.kmol.online.pt/artigos/200504/mar05_1.html . Acesso em 28 de novembro de 2007.

Microsoft Support. **ODBC Overview**. Disponível em: <http://support.microsoft.com/kb/110093/en-us> . Acesso em: 21 de janeiro de 2008.

O'BRIEN, P. K.; TURGEMAN, Y. **PyDB2**. Disponível em: <http://www.sourceforge.net/projects/pydb2> . Acesso em: 22 de novembro de 2007.

OPENLINK. **iODBC**. Disponível em: <http://www.iodbc.org> . Acesso em: 21 de janeiro de 2008.

PADILHA, T. C. C.; MARTINS, F. A. S. Sistemas ERP: características, custos e tendências. **Revista Produção**, [S.l.], v.15, n.1, 2005.

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/>. Acesso em: 22 de novembro de 2007.

PLONE FOUNDATION. **Plone CMS – Open Source Content Management System**. Disponível em: <<http://www.plone.org/>>. Acesso em: 22 de novembro de 2007.

ROSSUM, G. V. **Python Tutorial**. [S.l.: s.n.], 2006. Disponível em: <<http://docs.python.org/tut/>>. Acesso em: 22 de novembro de 2007.

SCHOLZ, R. A. **Modernização do Portal de Informática da UFSM Utilizando um Sistema de Gestão de Conteúdo Web**. Santa Maria: Curso de Ciência da Computação. Universidade Federal de Santa Maria, 2008.

SERVIÇO FEDERAL DE PROCESSAMENTO DE DADOS. **Portal SERPRO**. Disponível em: <http://www.serpro.gov.br> . Acesso em: 30 de novembro de 2007.

SOUZA, C. A. de. Sistemas integrados de gestão empresarial: estudos de caso de implementação de sistemas erp. **FEA/USP**, [S.l.], 2000.

The Python Language. 2007.

APÊNDICE A MANUAL DE INSTALAÇÃO

Este documento visa descrever as atividades necessárias para a instalação/configuração da ferramenta **NoticiaPlone** versão 1.0. Inicialmente serão listados os *softwares* dos quais a ferramenta NoticiaPlone é dependente; em seguida serão listados os passos necessários para instalação dos *softwares* necessários em ambiente Windows e Linux; para finalizar, será descrito o processo de instalação do Produto a um portal Plone.

A.1 Dependências e Requisitos

Para instalação e utilização do sistema NoticiaPlone, é necessário que os seguintes *softwares* estejam instalados e corretamente configurados no sistema operacional alvo:

1. **Interpretador Python versão 2.4:** <http://www.python.org/>;
2. **Servidor de Aplicações Zope versão 2.10.5:** <http://www.zope.org/>;
3. **Plone versão 3:** <http://www.plone.org>;
4. **Módulo Python pyODBC:** <http://pyodbc.sourceforge.net/>;
5. **Gerenciador de Drivers ODBC;**

As próximas seções especificarão a instalação dos *softwares* acima listado para os sistemas operacionais Microsoft Windows e Ubuntu GNU/Linux, respectivamente.

A.2 Microsoft Windows

Caso os itens 1, 2 e 3 da lista ainda não estejam instalados no sistema, uma boa alternativa para instalação destes é o pacote, disponibilizado no portal <http://www.plone.org>, o qual contém os três *softwares* empacotados em um programa instalador de fácil utilização.

Este instalador realiza a instalação e configuração do interpretador Python, do Servidor de Aplicações Zope e já configura um portal Plone no sistema. Basta seguir os passos descritos no passo-a-passo do instalador, que este configura tudo corretamente.

Para testar se o portal Plone foi instalado corretamente, abra seu navegador Web e digite o endereço: `http://hostname:8080/Plone`, onde **hostname** é o nome do servidor no qual o pacote Plone foi instalado.

Caso a instalação tenha ocorrido com sucesso, você visualizará uma página semelhante à da figura A.1.

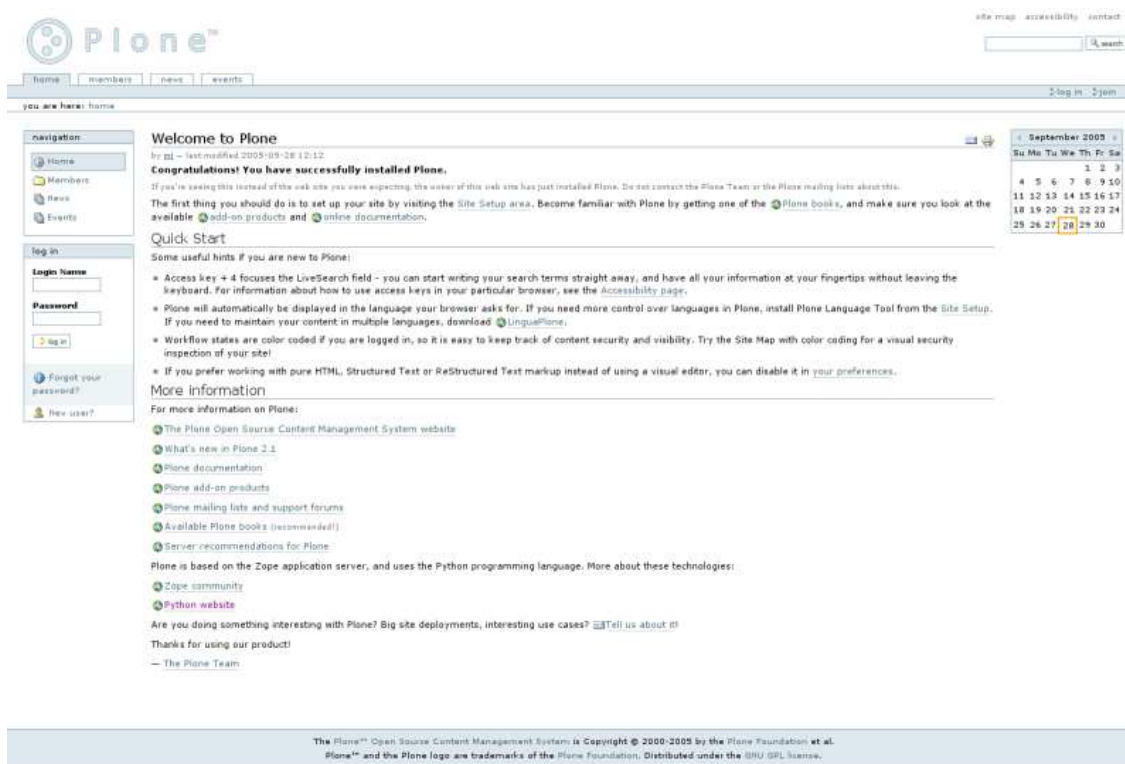


Figura A.1: Tela inicial do Plone

Caso a página não seja exibida, verifique se o processo servidor Zope está rodando em seu sistema. Caso não esteja, é possível iniciá-lo através de um *software* de gerenciamento deste, que pode ser acessado através do menu:

Iniciar->Programas->Plone->Plone

Tendo acessado o *software*, basta clicar no botão **Iniciar Plone**. Tente novamente acessar o endereço citado acima no texto para visualizar o portal Plone recém instalado.

Caso queira instalar os três *softwares*, um a um, pode-se fazê-lo seguindo as instruções de instalação de cada produto. Porém, este manual desencoraja a instalação em separado

dos componentes 1, 2 e 3, visto que a instalação do pacote fornece uma organização mais estruturada do sistema.

Após ter os componentes 1, 2 e 3 instalados e configurados corretamente, deve-se instalar o módulo Python **pyODBC**. A instalação deste módulo é bastante simplificada em ambientes Windows, visto que existe um **software** instalador que realiza esta tarefa.

Novamente, siga os passos descritos no instalador e conclua esta. Para testar se o módulo foi mesmo instalado em sua versão Python, abra um *prompt* interativo Python pertencente ao interpretador Python com o qual foi realizada a instalação do Plone. Se a instalação foi feita através do pacote, basta acessar o menu

Iniciar->Programas->Plone->Python

que o *prompt* lhe será exibido.

No *prompt*, digite:

» **import pyodbc**

Caso tenha recebido uma mensagem de erro como a mensagem abaixo, significa que o módulo não foi instalado corretamente. Procure ler o manual do produto para verificar possíveis problemas que podem ter acontecido.

Traceback (most recent call last):

File «stdin>', line 1, in ?

ImportError: No module named pyodbc

Caso a mensagem acima não tenha aparecido em seu *prompt*, significa que o módulo está instalado corretamente para o seu interpretador Python. Pode-se seguir para o próximo passo.

O passo seguinte é a configuração de uma fonte de dados ODBC no seu sistema. Esta tarefa pode ser realizada utilizando-se o gerenciador de fontes de dados que acompanha o Microsoft Windows. Este pode ser acessado através do **Painel de Controle** do Windows, na seção **Ferramentas Administrativas**. No gerenciador ODBC, configure uma fonte de dados apropriada ao banco de dados que deseja utilizar. Caso este banco de dados não esteja listado nos *drivers* disponíveis, pode-se realizar o *download* do *driver* apropriado no portal do desenvolvedor do Sistema de Gerenciamento de Banco de Dados desejado.

Você pode testar a fonte de dados no seu gerenciador de *drivers* ODBC.

Após configurar uma fonte de dados corretamente, os próximos passos a serem executados são relacionados à configurações do Plone.

Tendo a fonte de dados corretamente configurada, o próximo passo é copiar a pasta `NoticiaPlone` para o diretório de Produtos do Plone. Este diretório fica em seu sistema de arquivos, dentro da pasta do Plone, onde está configurada sua instância Plone. Se instalado com o pacote de instalação do Plone, esta pasta é:

"Arquivos de Programas->Plone->Zope->skel->Products"

Ou seja, a pasta `NoticiaPlone` deverá ser copiada para dentro do diretório acima.

Tendo feito isso, o próximo passo está na seção A.4.

A.3 Ubuntu GNU/Linux

Caso os itens 1, 2 e 3 da lista ainda não estejam instalados no sistema, uma boa alternativa para instalação destes é o pacote *PloneUnifiedInstaller*, disponibilizado no portal <http://www.plone.org>, o qual contém os três *softwares* empacotados em um programa instalador de fácil utilização.

Este instalador realiza a instalação e configuração do interpretador Python, do Servidor de Aplicações Zope e já configura um portal Plone no sistema. Basta seguir os passos descritos no passo-a-passo do instalador, que este configura tudo corretamente.

Após instalado, o diretório do Plone será: `/opt/Plone` .

Para instalar o módulo PyODBC, baixe os fontes desse módulo, descompacte-os em um diretório qualquer e dentro deste diretório, digite em um shell:

```
$ sudo /opt/Plone/Python2.4/bin/python ./setup.py install
```

Isto irá instalar o módulo PyODBC para o interpretador Python utilizado pelo Plone.

Para testar se a instalação ocorreu corretamente, abra um shell Python e digite:

```
» import pyodbc
```

Caso tenha recebido uma mensagem de erro como a mensagem abaixo, significa que o módulo não foi instalado corretamente. Procure ler o manual do produto para verificar possíveis problemas que podem ter acontecido.

Traceback (most recent call last):

File «stdin», line 1, in ?

ImportError: No module named pyodbc

Caso a mensagem acima não tenha aparecido em seu *prompt*, significa que o módulo está instalado corretamente para o seu interpretador Python. Pode-se seguir para o próximo passo.

O passo seguinte é a configuração da fonte de dados ODBC, que pode ser feita de posse do *driver* para acesso ao SGBD desejado e do software unixODBC, que trata-se de um gerenciador de fontes de dados ODBC.

Tendo a fonte de dados configurada, o próximo passo é copiar a pasta NoticiaPlone para o diretório `/opt/Plone/zinstance/Products`.

Tendo feito isso, pode-se passar para a seção A.4.

A.4 Adicionando o Produto ao Portal

Para adicionar o Produto NoticiaPlone ao seu portal, após tê-lo copiado para a pasta de produtos, deve-se iniciar o Plone, conforme já foi explicado neste manual. Deve-se então acessar o portal, efetuar login com conta de administrador e ir para a seção de **configurações do portal**, através do link no canto superior direito da tela. A figura A.2 mostra onde deve-se clicar para acessar a área de configurações administrativas do Plone.

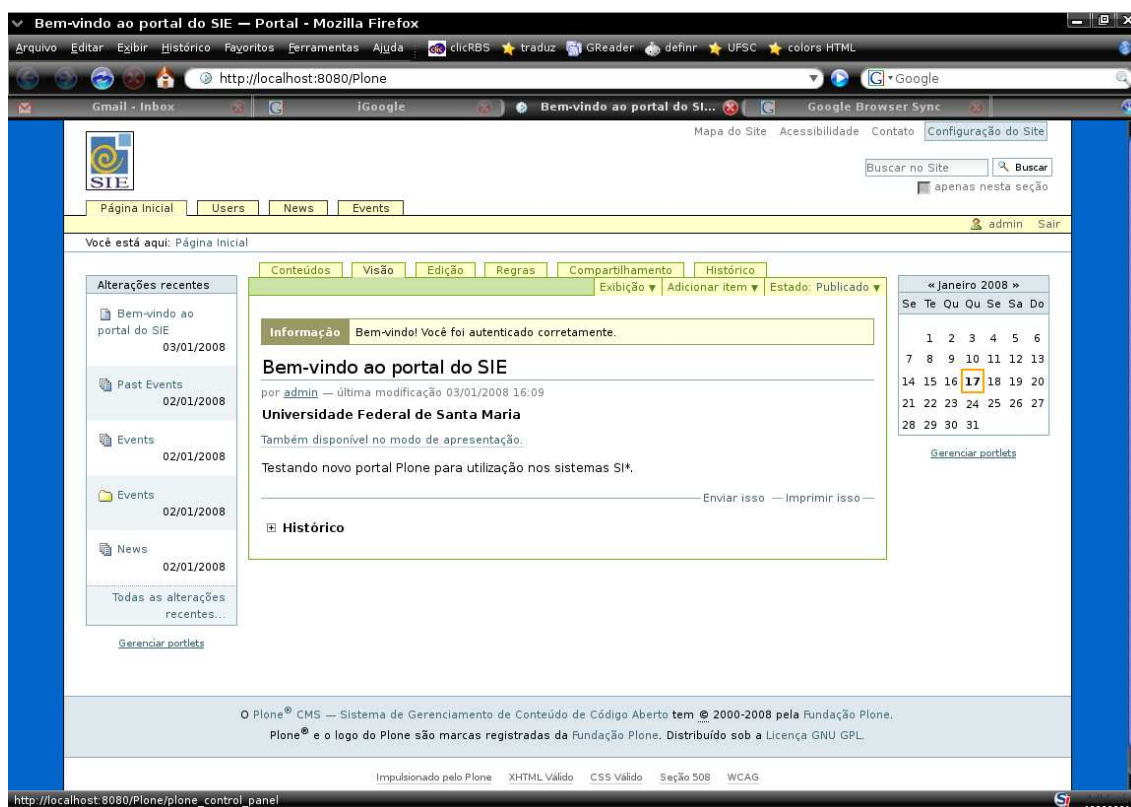


Figura A.2: Acesso às configurações do portal

Ao acessar as configurações, deve-se clicar em **Produtos Adicionais**. A figura A.3 mostra a tela de configurações que deve ser acessada.

Em **Produtos Adicionais** o produto NoticiaPlone deverá ser listado na caixa **Produtos**

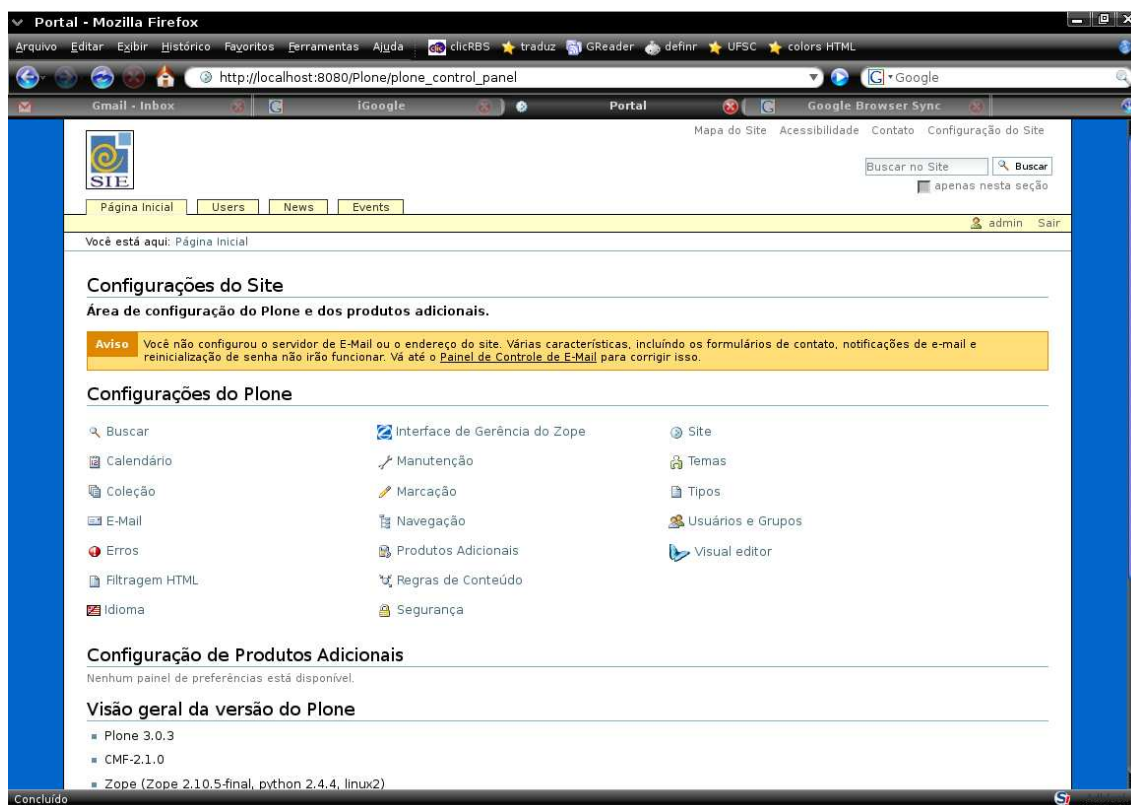


Figura A.3: Área de configuração de um portal Plone

Disponíveis para Instalação. Para instalá-lo, basta selecioná-lo e clicar em **instalar**. As figuras A.4 e A.5 mostram o produto sendo adicionado e após ter sido adicionado, respectivamente.

Após isso, o produto estará adicionado ao portal. Para criar um item NoticiaPlone, vá até a página inicial do portal e clique em **Adicionar->NoticiaPlone**, como mostra a figura A.6.

Assim será exibida a tela de cadastro de um item NoticiaPlone, conforme mostra a figura A.7.

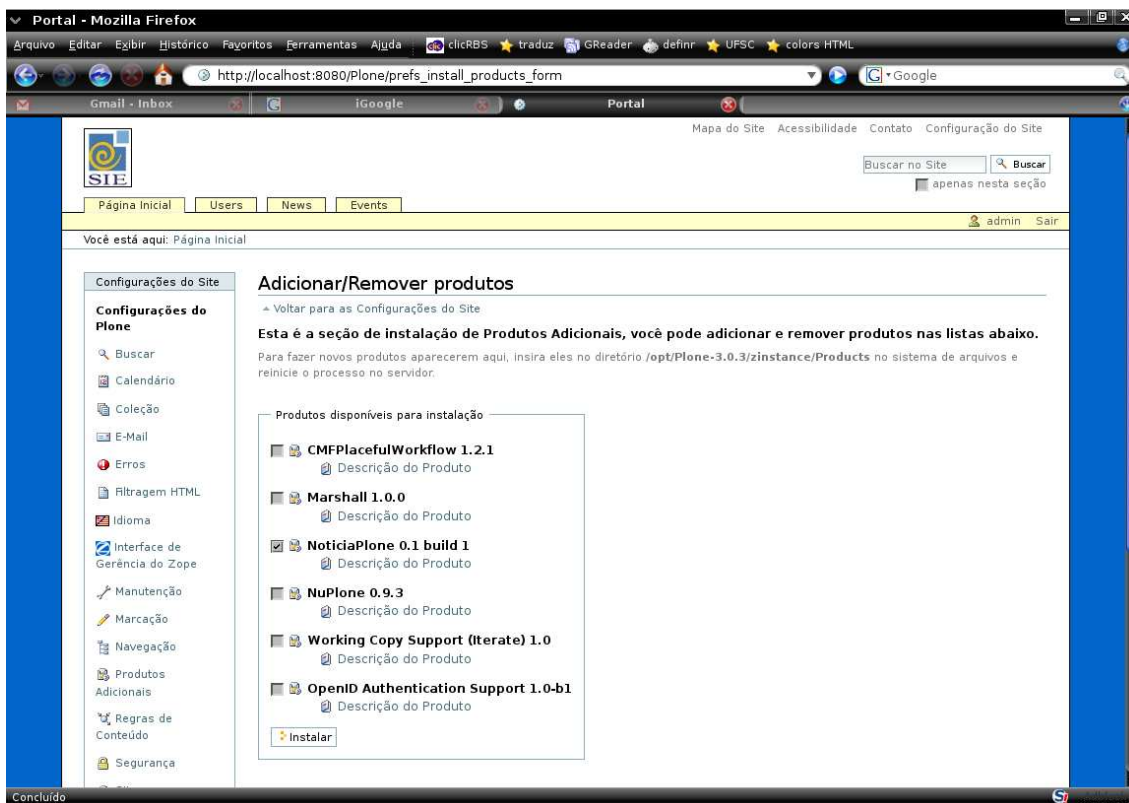


Figura A.4: Produto NoticiaPlone sendo adicionado

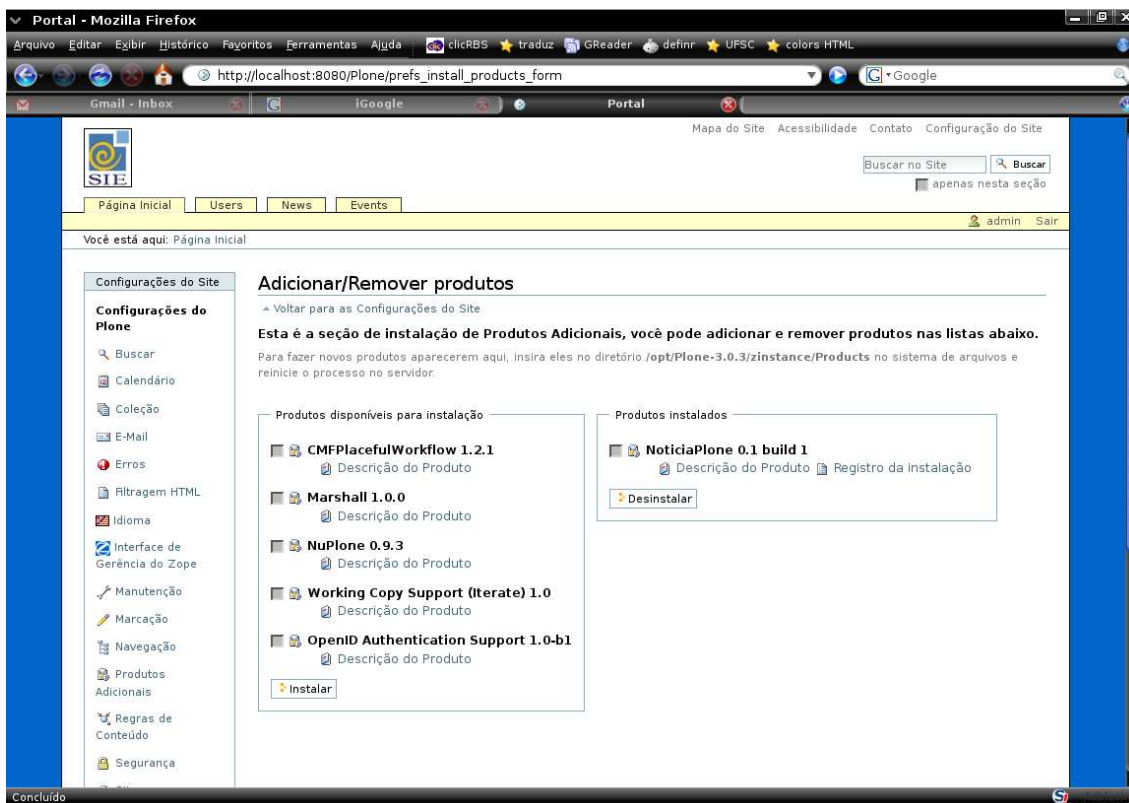


Figura A.5: Produto NoticiaPlone após ter sido adicionado



Figura A.6: Acesso à interface de edição de um item NoticiaPlone

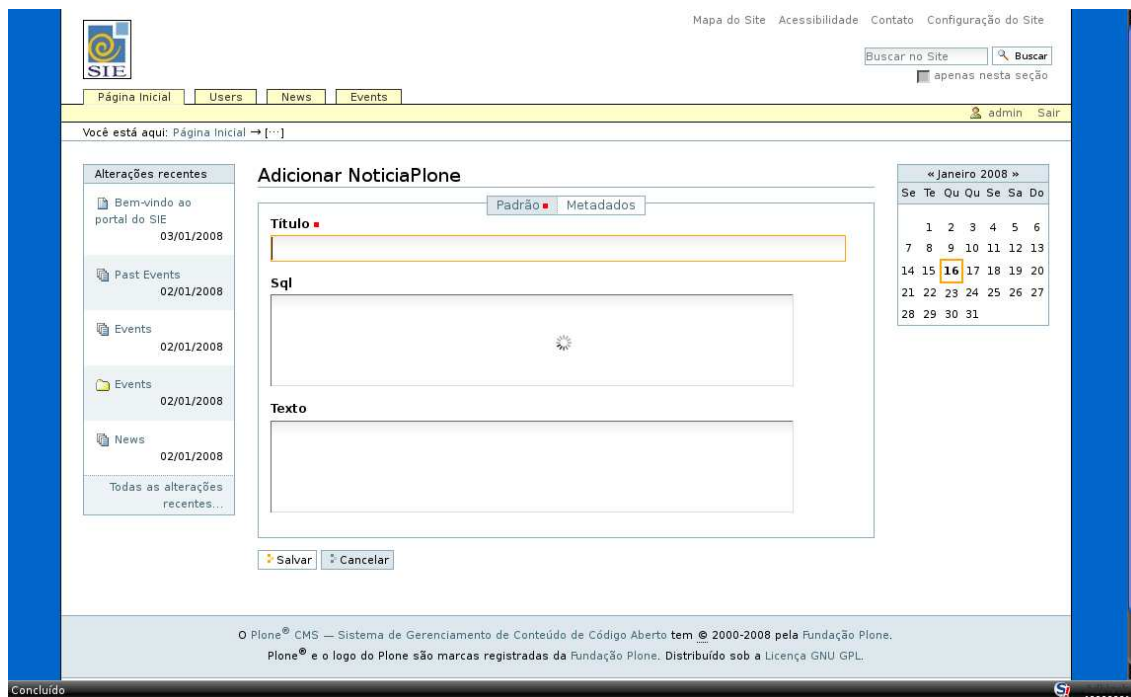


Figura A.7: Criação de um novo item NoticiaPlone