

EZEQUIEL CONTE

**Intranet na CERTI:
Um modelo baseado em recursos de Enterprise 2.0**

Florianópolis, 2007

EZEQUIEL CONTE

Intranet na CERTI:

Um modelo baseado em recursos de Enterprise 2.0

Trabalho de conclusão de curso apresentado como parte dos requisitos para obtenção do grau de Bacharel em Ciências da Computação na Universidade Federal de Santa Catarina.

Orientador: Prof. Vitório Bruno Mazzola

Florianópolis, 2007

EZEQUIEL CONTE

**Intranet na CERTI:
Um modelo baseado em recursos de Enterprise 2.0**

Trabalho de conclusão de curso apresentado como parte dos requisitos para obtenção do grau de Bacharel em Ciências da Computação na Universidade Federal de Santa Catarina.

Orientador: Prof. Vitorio Bruno Mazzola

Banca examinadora

Prof. Mário Antônio Ribeiro Dantas

Prof. Roberto Willrich

“The old computing was about what computers could do; the new computing is about what users can do.”

Ben Shneiderman

RESUMO

As intranets surgiram a partir da Internet. Com o atual surgimento de plataformas sociais na Internet é natural que essas também sejam aplicadas às intranets. Wikis, blogs e web feeds, são alguns dos elementos que serão analisados neste trabalho, visando à concepção de um modelo e implementação da intranet da Fundação CERTI. O objetivo é fazer com que os sistemas sejam uma plataforma de colaboração e comunicação, fornecendo aos usuários a possibilidade de não apenas buscar informações, mas também compartilhar seus conhecimentos com os colegas de trabalho de uma maneira eficiente e democrática.

Palavras-chave: Intranet. Enterprise 2.0. Software Social. Wikis. Blogs. Sistemas Colaborativos.

ABSTRACT

The Intranets derived from the Internet. Due to the arise of social platforms in the Internet it is natural that these platforms were being used in the intranets. Wikis, blogs and web feeds will be studied in this paper, aiming to the creation of the model and implementation of CERTI Foundation's intranet. The goal is to turn the systems into a platform for collaboration and communication, giving to the users the ability of finding information and sharing their knowledge in an efficient and democratic way.

Keywords: Intranet. Enterprise 2.0. Social Software. Wikis. Blogs. Colaborative Systems

LISTA DE FIGURAS

Figura 1 Organograma da Fundação CERTI.....	12
Figura 2 Pilha de Tecnologias – fonte: Calvin Andrus	32
Figura 3 Modelo da Intranet.....	33
Figura 4 - Página inicial padrão do TWiki.....	38
Figura 5 - Página inicial customizada.....	38
Figura 6 - Editor Simple versão original	40
Figura 7 - Novo Editor Simple	40
Figura 8 - Funcionalidade Visualização do Editor Simple.....	41
Figura 9 - Modo Avançado através de AJAX	41
Figura 10 - Página Inicial do ambiente de blogs	43
Figura 11 - Estatísticas: Total de Tópicos	46
Figura 12 - Estatísticas: Tópicos Acessados no Mês	46
Figura 13 - Estatísticas: Modificações no Mês	47
Figura 14 - Estatísticas: Total de Usuários Cadastrados no Sistema	48

SUMÁRIO

1 INTRODUÇÃO	10
1.1 OBJETIVO GERAL.....	11
1.2 OBJETIVOS ESPECÍFICOS	11
1.3 DIAGNÓSTICO DA EMPRESA	11
1.4 JUSTIFICATIVA	13
1.4.1 CENÁRIO DA CERTI.....	13
1.4.2 CENÁRIO GLOBAL.....	14
2 FUNDAMENTAÇÃO TEÓRICA	16
2.1 INTRANET	16
2.2 WEB 2.0 E ENTERPRISE 2.0	18
2.3 SOFTWARES SOCIAIS.....	19
2.3.1 WIKIS	20
2.3.2 BLOGS	21
2.3.3 SOCIAL BOOKMARKS.....	22
2.3.4 WEB FEEDS.....	23
3 CONCEPÇÃO DO MODELO	24
3.1 AVALIAÇÃO DA PESQUISA	25
3.1.1 ANÁLISE DOS DADOS	26
3.2 MODELO PROPOSTO	31
3.2.1 WIKI	33
3.2.2 BLOGS	34

	9
3.2.3 SOCIAL BOOKMARKS.....	35
3.2.4 WEB FEEDS.....	35
4 IMPLEMENTAÇÃO E RESULTADOS	37
4.1 ESTATÍSTICAS DE USO DO SISTEMA.....	44
5 CONCLUSÕES E TRABALHOS FUTUROS	49
REFERÊNCIAS BIBLIOGRÁFICAS	51
ANEXO I - QUESTIONÁRIO SOBRE FERRAMENTAS DE TECNOLOGIA NO TRABALHO	54
ANEXO II - DADOS COLETADOS DO QUESTIONÁRIO SOBRE FERRAMENTAS DE TECNOLOGIA NO TRABALHO	58
ANEXO III - ARTIGO.....	64
ANEXO IV - CÓDIGO FONTE	73

1 INTRODUÇÃO

Para enfrentar este mundo cada vez mais dinâmico, é vital para as empresas o emprego de ferramentas que estimulem a comunicação interna e auxiliem na gestão do conhecimento. Fontes de informação cada vez mais amplas e mudanças ocorrendo a passos acelerados são alguns dos obstáculos que uma empresa deve transpor para manter-se sólida no mercado.

Diante deste cenário, uma solução amplamente utilizada é a adoção de intranets. Uma intranet fornece um mecanismo seguro, simples e facilmente acessível para a comunicação interna da organização.

As intranets surgiram a partir da utilização das ferramentas da Internet no contexto organizacional. Com a evolução da Internet, novas ferramentas foram surgindo, como é o caso dos wikis, blogs, redes sociais e outros.

A massiva utilização de blogs e wikis nos mostram o quanto estas ferramentas facilitam o trabalho colaborativo, a troca de informações e a geração de conhecimento. O exemplo que melhor ilustra estas afirmações é a Wikipedia, uma enciclopédia on-line escrita totalmente pelos seus próprios usuários. Além disso, são inúmeros os blogs que se tornam referência em temas específicos, tornando-se ponto de encontro para os interessados no assunto.

Como era possível prever, todas essas novas ferramentas surgidas na Internet também estão aos poucos sendo implantadas nas intranets. Isso torna a intranet muito mais do que um mecanismo de comunicação: transforma-a em uma plataforma de colaboração, auxiliando nos processos criativos e na gestão do conhecimento da organização.

Este novo conceito de intranet é cunhado pelo termo Enterprise 2.0. Ou, como McAfee (2006) define, Enterprise 2.0 é o uso das emergentes plataformas de

software social dentro das companhias, ou entre companhias e seus parceiros ou clientes.

1.1 OBJETIVO GERAL

Este trabalho tem por objetivo geral consolidar a base da intranet na Fundação CERTI usando os conceitos de Enterprise 2.0.

1.2 OBJETIVOS ESPECÍFICOS

- Aprofundar conhecimentos sobre o tema Enterprise 2.0.
- Pesquisar necessidades dos usuários e formas de supri-las na intranet.
- Propor modelo da intranet com base nos conceitos de Enterprise 2.0.
- Desenvolver a intranet, através da implantação de sistemas de wiki, blog e outros que se fizerem necessários, promovendo a integração entre os mesmos.
- Personalização dos sistemas, criando uma identidade visual e adaptando-os às necessidade da CERTI.
- Estudar e promover maneiras de incentivar o uso da intranet.

1.3 DIAGNÓSTICO DA EMPRESA

Criada em 1984 por iniciativa de algumas empresas, da Universidade Federal de Santa Catarina e dos Governos Federal e Estadual, a Fundação CERTI é uma instituição independente e sem fins lucrativos, de pesquisa e desenvolvimento tecnológico.

Com sua sede localizada no Campus da Universidade Federal de Santa Catarina, a Fundação CERTI tem como foco a inovação em negócios, produtos e serviços no segmento de tecnologia da informação.

A CERTI congrega centros de referências, laboratórios e institutos, que funcionam de forma integrada.

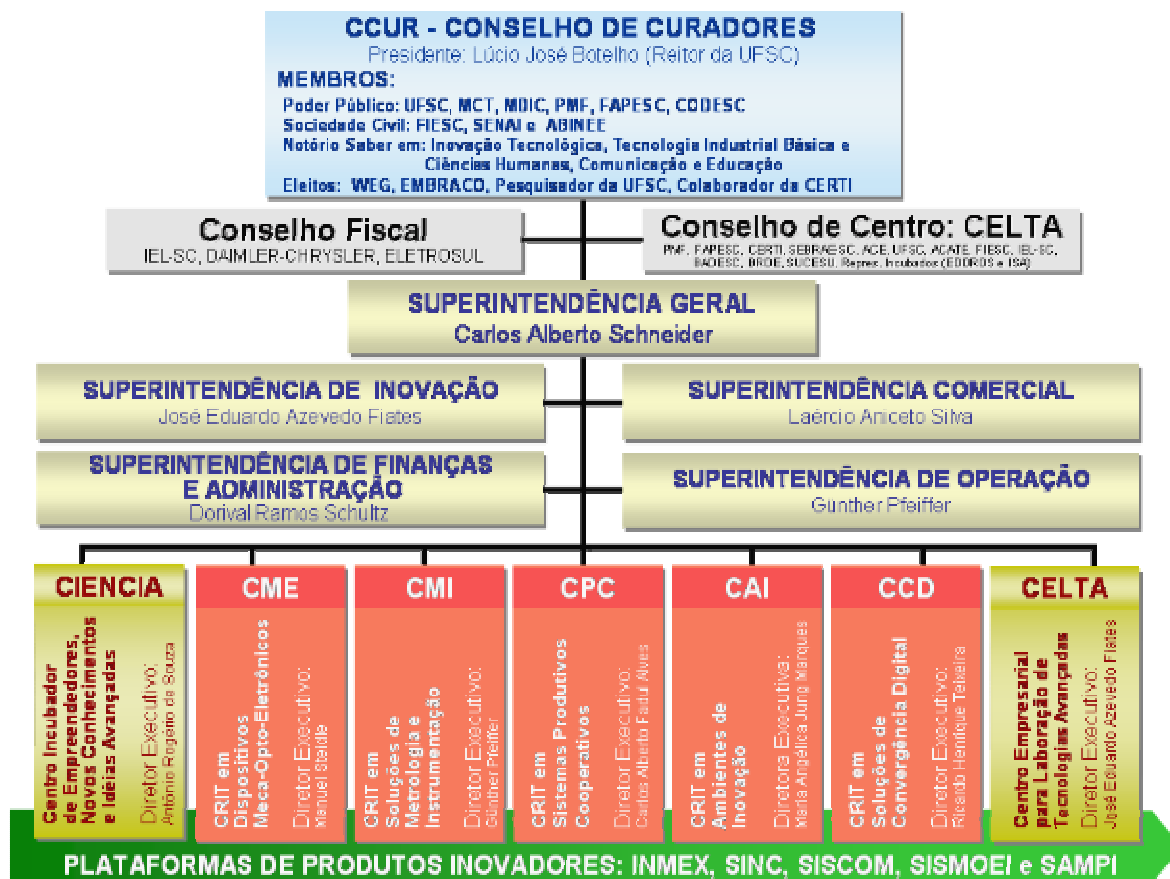


Figura 1 Organograma da Fundação CERTI

A Fundação possui filiais em Brasília e Manaus, contabilizando no total uma equipe com aproximadamente 200 colaboradores.

As principais áreas de destaque da CERTI são as de tecnologia da informação, sistemas mecatrônicos e metrologia, sendo alguns de seus projetos a Urna Eletrônica, o Terminal Web de Acesso Público e o Sapiens Parque.

1.4 JUSTIFICATIVA

1.4.1 CENÁRIO DA CERTI

A Fundação CERTI provê vários serviços a seus colaboradores através da rede interna. Alguns exemplos são os serviços de webmail, gerenciamento de projetos, sistema de gestão de equipes e serviços de acesso à informação, como o wiki do setor de informação da Fundação.

Porém, esses recursos encontram-se isolados em seus respectivos sites, sem um ponto de acesso central a todos. Para acessar qualquer um destes serviços, o colaborador deve saber da existência do serviço e ter o endereço específico do mesmo. Ocorre com isso que muitos sistemas são muito pouco utilizados e até mesmo desconhecidos pela maioria dos colaboradores da CERTI. Um exemplo é o fórum de discussões, que apresenta um número pouco expressivo de tópicos.

Outro ponto relevante é a falta de divulgação interna dos projetos. Os colaboradores não tomam conhecimento de quais projetos estão sendo desenvolvidos pela CERTI, sabendo apenas sobre o qual está participando ou o de algum colega próximo. Há situações em que projetos trabalham em áreas relacionadas, ou usando as mesmas tecnologias e não há uma troca de experiências entre as equipes.

No âmbito da gestão do conhecimento, existe na CERTI um sistema de acesso à arquivos através do uso de diretórios compartilhados. Este sistema é utilizado para troca de arquivos e como depósito de documentos com informações sobre cada projeto. Nele, existe uma hierarquia bem definida, com padrões de nomenclatura para organizar dos dados. Porém, existe uma grande dificuldade para se realizar buscas sobre estas informações. Da necessidade de lidar com este tipo de situação foi que surgiu o conceito de hyperlinks e a própria web.

Uma outra carência é referente à necessidade de um espaço para a comunicação interna, estimulando a troca de informações entre os colaboradores. Nesta área, os colaboradores poderiam divulgar suas habilidades e áreas de interesse, discutir sobre temas, além de ser um espaço para as informações institucionais, como o manual do colaborador e estrutura organizacional.

1.4.2 CENÁRIO GLOBAL

Os ganhos provindos da adoção de uma intranet são bastante conhecidos. Através de estudos realizados em 2001 (WHITE, 2003) pela Melcrum Publishing notou-se que entre os vários benefícios para a empresa estão:

- Melhor comunicação interna;
- Processos melhorados;
- Compartilhamento de boas práticas;
- Redução de trabalho em papel;
- Troca de informações em tempo real;

Outro estudo, feito pela Chartered Institute of Marketing, mostrou que 93% dos entrevistados acreditam que as intranets facilitam a inovação.

Somando-se a isso, já existem muitos casos de sucesso de aplicação de Enterprise 2.0 nas intranets. Esses casos vão desde empresas de pequeno e médio porte à empresas grandes, como Siemens e IBM.

Na Siemens, é ampla a utilização de RSS, como por exemplo, na notificação de manutenção de servidores, algo que anteriormente era feito através de e-mail, muitas vezes atingindo a não interessados.

A IBM foi uma das empresas pioneiras na utilização de blogs internos, primeiramente como uma pesquisa. Hoje, a iniciativa é um sucesso e conta com

milhares de blogs de empregados. Também na IBM vemos a utilização de social bookmarking e wikis como forma de compartilhamento de informações.

"A IBM percebeu que a criação de grandes volumes de conversações paralelas ajuda as pessoas a navegar pelas complexidades de um mundo dinâmico"

Redefining manager interaction at IBM (DORSETT 2002)

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo serão descritos alguns estudos que formam o embasamento teórico para o trabalho.

2.1 INTRANET

Conforme Hills, 1997, uma intranet é uma rede que usa as mesmas tecnologias empregadas na Internet, principalmente a Web, porém usada num contexto interno da empresa. As intranets têm finalidades específicas e são desenvolvidas e adaptadas à realidade da empresa e às necessidades de seus usuários.

Uma intranet é um ambiente para troca de informações muito amplo, permitindo a interação da empresa com filiais, clientes e fornecedores, a colaboração entre profissionais e a divulgação de idéias e conhecimentos.

As origens das intranets nos remetem às origens da Internet e da Web. A Internet foi criada pelo governo norte-americano na década de 1970, visando proteger as comunicações militares caso sofresse um ataque nuclear. O objetivo era criar uma rede em malha, impedindo que a destruição de um nó afetasse a comunicação de toda a rede.

Alguns anos depois, a rede foi expandida para as universidades, a fim de fornecer o acesso aos supercomputadores e facilitar a pesquisa. Já na década de 1990, foi permitido o acesso comercial, causando o gigantesco crescimento da Internet, que se estende até hoje.

A World Wide Web foi criada em 1989 no Laboratório Europeu de Física de Partículas – CERN pelo físico Tim Berners-Lee. A idéia básica era facilitar o acesso

às informações entre os pesquisadores do CERN através da utilização de hipertextos, que é um vínculo entre as informações. Acessando um hipertexto, o usuário recebia informações adicionais ou era guiado a outras páginas.

Pouco tempo depois, para facilitar o acesso à Web, uma interface gráfica chamada Mosaic, o precursor dos navegadores gráficos Web, foi criada por Marc Andreessen e alguns colegas. Este aplicativo foi disponibilizado gratuitamente pela Internet e a partir daí a utilização da Web tomou proporções gigantescas.

A partir do lançamento do Mosaic, em 1993, algumas empresas começaram a enxergar o potencial da Web e experimentar a Intranet. Perceberam também o valor dessas ferramentas para acesso às informações internas. Começaram a colocar informações privadas nos servidores, protegendo-os do acesso externo. Assim surgiram as intranets.

As aplicações das intranets são muito variáveis, dependendo principalmente das particularidades de cada organização. Uma intranet pode ser constituída apenas por uma página HTML estática ou possuir todo o controle logístico e financeiro da empresa. Alguns exemplos de serviços que as intranets podem oferecer:

- Informações institucionais
- Correio Eletrônico
- Diretórios
- Veiculação de notícias
- Manuais
- Relatórios financeiros
- Treinamento
- Grupos de discussão

Como na Internet, historicamente o conteúdo Web das intranets era um sistema 1-n (BUFFA,2006), onde uma pessoa ou um grupo pequeno era responsável pela publicação, dando aos outros usuários apenas a opção de leitura das informações disponibilizadas.

Mas esta situação vem mudando gradativamente, à medida que sistemas colaborativos, como wikis, são implantados nas intranets. Isso dá uma liberdade de criação maior para todos os usuários, facilitando a troca de conhecimento entre os colegas de trabalho. Esta mudança está ligada aos conceitos de Enterprise 2.0.

2.2 WEB 2.0 E ENTERPRISE 2.0

Como se pode notar, a Web está atualmente em processo de evolução, onde a colaboração entre os usuários é o fator chave. Esta mudança está sendo cunhada pelo termo Web 2.0.

Segundo Tim O'Reilly, "Web 2.0 é a mudança para uma Internet como plataforma, e um entendimento das regras para obter sucesso nesta nova plataforma. Entre outras, a regra mais importante é desenvolver aplicativos que aproveitem os efeitos de rede pra se tornarem melhores quanto mais são usados pelas pessoas, aproveitando a inteligência coletiva".

Alguns exemplos que ilustram bem este novo momento são a Wikipedia, uma enciclopédia virtual, totalmente escrita pelos seus usuários e o crescente uso de Weblogs (blogs).

O tema Web 2.0 é motivo de muitas discussões na Internet, sendo que alguns a consideram uma "buzzword", uma palavra sem significado claro, uma vez que as definições do termo são vagas e, além disso, alegam que nada de novo foi criado, sendo que a Web 2.0 seria apenas um processo natural de melhoria da Web.

Porém, as melhorias dessa nova fase podem ser percebidas e a área corporativa, aproveitando-se dos benefícios dessas novas tendências, começou a utilização destes mesmos sistemas sociais em suas redes internas, surgindo o conceito de Enterprise 2.0:

"Enterprise 2.0 é o uso das emergentes plataformas de software social dentro das companhias, ou entre companhias e seus parceiros ou clientes".

(Andrew McAfee ,2006)

Já são muitos os exemplos de empresas seguindo esta tendência. Empresas que adotam sistemas wiki para gestão do conhecimento, blogs para discussão, web feeds para notificações são exemplos empresas que estão seguindo os conceitos de Enterprise 2.0. O Google é um exemplo de empresa que usa um sistema wiki, o TWiki, desde o início de suas atividades (FINCK, 2005).

A base desse paradigma é a utilização de softwares sociais, que serão analisados a seguir.

2.3 SOFTWARES SOCIAIS

O termo Software Social foi cunhado em 2002 pelo escritor e professor Clay Shirky. Ele define software social simplesmente como "software que suporta interação em grupo" (OWEN, 2007).

Outra definição é a de que software social se refere à vários tipos de aplicações conectadas, que permitem aos indivíduos se comunicarem com os demais e acompanhar discussões através da Web enquanto elas ocorrem (GREEN, 2005). Esta definição é um pouco mais restritiva, afirmando que a interação entre os indivíduos ocorre através da Web, o que nem sempre é o caso, como por exemplo, em uma rede interna.

Outros estudiosos restringem o termo à gêneros de softwares mais recentes como blogs e wikis (WIKIPEDIA, 2007), caracterizando software social pelas suas novas dinâmicas de uso.

Nesse caso, a principal característica é que o conteúdo emerge a partir dos interesses dos participantes. São os próprios usuários dos sistemas que definem o que é importante e o sistema evolui a partir das características dos seus usuários.

Geralmente, softwares sociais são vistos como uma extensão da comunicação face à face.

A seguir serão apresentadas as principais ferramentas de Software Social.

2.3.1 WIKIS

O primeiro sistema wiki surgiu em 1995, desenvolvido por Ward Cunningham (BUFFA,2006). A palavra wiki vem da havaiana *wiki-wiki*, que significa rápido (TWIKI.ORG, 2006), uma alusão à simplicidade de edição das páginas wiki.

Diferentemente do sistema de edição centralizada de uma página Web, um wiki é um site da Web onde os usuários têm poder para criar, editar textos e hiperlinks entre páginas, de maneira simples, utilizando o próprio navegador Web.

Essa característica traz vários benefícios, sendo um deles o auxílio na elaboração de conteúdo coletivamente, criando um ponto de acesso centralizado onde todos editam, sem a necessidade de troca de mensagens entre os colaboradores para repassar suas contribuições aos demais.

Diferentemente do HTML, o wiki utiliza uma sintaxe simples para marcar títulos e destacar elementos, inspirada em como os textos puros eram destacados nos primórdios da Web. Isso facilita a utilização dos wikis, pois possibilita a

participação por usuários não-técnicos, tirando a necessidade de aprender uma linguagem de marcação complexa.

Outra contribuição de Cunningham foi a introdução do conceito de *WikiWords* (palavras wiki), como forma de criar hiperlinks para novas páginas. Uma WikiWord é basicamente a junção de duas palavras iniciadas por letras maiúsculas, que ao ser escrita no texto automaticamente se transforma em hiperlink para uma página.

Atualmente vários sistemas wikis foram desenvolvidos, sendo que há sites especializados em comparações de sistemas wikis, como o WikiMatrix (<http://www.wikimatrix.org/>). Os mais conhecidos são os softwares livres TWiki, MediaWiki e SocialText.

2.3.2 BLOGS

Tecnicamente, blog ou weblog é um tipo de website, bastante simples de se atualizar, bastando preencher as informações que serão adicionadas em um formulário (GILROY, 2006). Geralmente, as informações entradas são exibidas em ordem cronológica e permitem aos visitantes deixarem comentários. Um blog pode ser de autoria coletiva ou mantido por um único autor.

Os blogs vêm sendo usados para as mais diversas finalidades. Algumas pessoas usam blogs como um “diário virtual”, contando seu dia-a-dia, suas emoções e opiniões. Outros têm um enfoque mais profissional, tratando apenas de assuntos ligados ao seu trabalho; tendo o blog como um “cartão de visitas”.

Nas organizações, os blogs estão cada vez mais sendo usados, tanto na divulgação externa quanto interna. Usado no interior da empresa, o blog pode ser usado para discussões sobre projetos ou divulgação de estudos realizados pelos seus trabalhadores.

Externamente, os blogs organizacionais servem para divulgar notícias e produtos da empresa para os consumidores. Como os consumidores podem expressar suas opiniões, o blog se torna uma excelente ferramenta para a empresa verificar a repercussão de seus produtos no mercado.

2.3.3 SOCIAL BOOKMARKS

Social Bookmarks é um serviço Web para compartilhamento de bookmarks (endereços eletrônicos, também conhecidos como *favoritos*) (WIKIPÉDIA, 2007). Nele, os usuários podem armazenar, classificar, compartilhar e procurar endereços eletrônicos.

Ao cadastrar um endereço no sistema, o usuário associa-o à uma lista de palavras-chave. Essas informações, classificadas pelas palavras-chave, podem então ser acessadas por outros usuários do sistema. Essa forma de classificação é conhecida como folksonomia.

Um usuário pode também indicar endereços à outro, criando uma rede de compartilhamento muito útil em tarefas em equipes. A equipe pode montar uma lista coletiva de endereços pertinentes aos seus trabalhos.

Efetuar uma busca em sistemas de Social Bookmarking geralmente traz resultados mais satisfatórios que em uma busca normal, pois quanto mais relevante um endereço é para um determinado tema, mais pessoas o terão cadastrado.

2.3.4 WEB FEEDS

Web Feed (do inglês *alimentar*), ou fonte, é um mecanismo de distribuição de conteúdo, que contém uma lista de atualizações de um determinado site. RSS e Atom são os principais padrões utilizados para esta função.

Essas fontes são utilizadas para que um usuário seja notificado sobre a atualização de seus sites prediletos. Para isso, o usuário cadastra as fontes em um software *agregador*. O agregador se encarrega de automatizar o processo de verificação de atualizações e as exibe para o usuário.

Algumas das notificações que podem ser feitas através de Web feeds são notícias de um portal, novidades de um blog, novos livros em uma biblioteca, novos produtos em um portal de comércio eletrônico, novos endereços em um serviço de bookmarking social e alterações em um wiki.

As fontes também podem ser cadastradas em websites, criando portais dinâmicos, personalizados para os usuários. Alguns exemplos são Netvibes (<http://www.netvibes.com/>) e o Google IG (<http://www.google.com/ig>).

3 CONCEPÇÃO DO MODELO

Para formação das idéias iniciais do desenvolvimento da intranet, foram feitas algumas reuniões com membros da Fundação, que atuam em áreas distintas. Isso rendeu a idéia básica do sistema, bem como algumas idéias para melhorias, como a realização de uma pesquisa com os colaboradores para avaliar suas necessidades e o estudo de sistemas semelhantes.

O foco principal da intranet era possibilitar uma estrutura emergente, dando aos usuários a possibilidade de criar e publicar informações. Assim a intranet estaria se beneficiando da construção através de um grande grupo de colaboradores ao invés de um grupo pequeno.

Os sistemas da intranet foram implantados gradualmente, permitindo que pudessem ser feitas análise das prioridades durante todo o processo de desenvolvimento.

Periodicamente eram analisadas as necessidades principais dos usuários, focando em implementar as melhorias que trouxessem maior eficiência no trabalho e melhoria de processos.

Durante o processo de desenvolvimento, foi realizada a pesquisa, que consistia em um questionário on-line, aproveitando para divulgar as primeiras etapas do desenvolvimento da intranet. A avaliação da pesquisa será analisada no próximo capítulo.

Prevaleceu a idéia de que os sistemas fossem software livre, pois dentre as vantagens teríamos diminuição de custos, sistemas amplamente utilizados e seria possível retornar as melhorias à comunidade, podendo as mesmas serem testadas e evoluídas pelas própria comunidade.

Os sistemas de wiki e blog obedecem à Lei de Metcalfe, que diz que o valor de um sistema de comunicação cresce aproximadamente ao quadrado do número de nós do sistema (ANDRUS, 2005). Ou seja, eles só terão valor se tiverem uma grande quantidade de usuários contribuindo.

Então, para garantir a participação, foram adotadas algumas estratégias. A primeira foi a aplicação do questionário, visando conhecer o perfil dos usuários e os elementos que serão mais úteis, criando uma intranet mais adequada às necessidades de Fundação CERTI.

Outra estratégia foi disponibilizar serviços que aumentariam a frequência de acesso dos usuários ao sistema. Um serviço é o guia de ramais, onde foram listados os colaboradores, seus ramais e funções. Outro serviço é a lista de aniversariantes do mês. E por fim, notícias publicadas nos blogs pelos próprios colaboradores tentaram promover o acesso diário.

3.1 AVALIAÇÃO DA PESQUISA

Para a intranet se tornar um ambiente de uso efetivo, é fundamental que os sistemas sejam adaptados ao nível de experiência dos usuários. Não sendo complexo para a maioria nem sendo ineficiente para outros.

Com o objetivo principal de entender a dinâmica de trabalho dos colaboradores da CERTI e captar possíveis padrões de comportamento, foi aplicado um questionário on-line, abordando a experiência do usuário com as tecnologias de informação.

Como objetivos secundários estão:

- Descobrir quais as principais ferramentas tecnológicas ou não que são utilizadas para auxiliar no trabalho do dia-a-dia

- Identificar possíveis gargalos de produtividade existentes
- Identificar as ferramentas de comunicação e troca de informação mais utilizadas e de maior utilidade, bem como suas deficiências
- Identificar possíveis demandas de ferramentas não atendidas e captar sugestões de melhorias em sistemas já existentes

A pesquisa visava também divulgar à intranet aos colaboradores. Por isso, para acessar o questionário era necessário acessar a intranet. Para garantir uma participação significativa, todos os colaboradores foram notificados sobre a pesquisa na intranet via correio eletrônico.

O questionário, bem como os dados coletados estão incluídos nos anexos do trabalho.

3.1.1 ANÁLISE DOS DADOS

Foram realizadas duas aplicações do questionário. A primeira foi apenas no Centro de Convergência Digital - CCD e no Instituto Sapientia - IS, servindo como pré-avaliação do próprio questionário, obtendo um total de 21 respostas. Estas duas unidades estão próximas fisicamente e são as que se espera um maior nível de experiência com as tecnologias analisadas, uma vez que as equipes são compostas por profissionais da área de software.

Após algumas pequenas modificações no questionário, este foi aplicado no contexto geral da CERTI, obtendo mais 39 respostas, totalizando 60 respostas.

As respostas foram analisadas levando-se em conta as duas aplicações do questionário.

1. Primeiras tarefas do dia

Podemos ver logo pela primeira questão que há uma grande dependência do e-mail, tornando-se este uma das primeiras atividades do dia para 96% dos respondentes. Outras atividades realizadas com frequência são a verificação da agenda para o dia (50%) e leitura de notícias na Internet(43%). Através desses dados, podemos ter um perfil das necessidades dos colaboradores e tentar facilitá-las com auxílio da intranet, como por exemplo, adicionando notícias de portais da internet nas páginas da intranet.

2. Importância das atividades básicas

Como a maioria dos itens foram marcados como importante, podemos ter uma análise mais significativa agrupando as respostas em essenciais/importantes e pouco importante/sem importância.

Atividade	Essencial / Importante	Pouco Importante / Sem Importância
Compartilhar arquivos	0,93	0,07
Receber informações atualizadas diariamente	0,88	0,11
Acompanhar as atividades de projeto	0,86	0,14
Estudar assuntos em equipe/grupo	0,86	0,14
Acessar rapidamente os ramais dos colegas	0,82	0,18
Receber informações sobre a CERTI	0,79	0,21
Registrar as atividades do dia-a-dia	0,77	0,23
Comunicar-se de forma instantânea (MSN,etc.)	0,75	0,25
Compartilhar informações/notícias	0,75	0,25
Escrever textos em parceria (ex:propostas)	0,75	0,25
Informações das agendas de viagem e reuniões	0,55	0,45

Receber informações sobre Editais em geral	0,46	0,54
Controlar as alocações de pessoal	0,41	0,59

As grandes demandas são compartilhar arquivos, receber informações (reforçando o item anterior), estudar assuntos em grupo e acompanhar atividades de projeto. O objetivo dessa questão é analisar as principais necessidades e verificar se estão e como estão sendo sanadas. Os itens menos relevantes para o grupo também merecem atenção, uma vez que podem representar nichos não atendidos pelas ferramentas atuais.

3. Meios de Comunicação

Após contato pessoal, vemos que o e-mail lidera como sendo a ferramenta de comunicação mais utilizada. Em seguida temos contato por telefone e mensagens instantâneas. Apesar da praticidade, devemos notar que muitas discussões são realizadas e decisões são tomadas através desses meios de comunicação. O grande problema é a falta de registros, de históricos dessas discussões.

4. Fontes de Informação

Há um certo equilíbrio, conforme o esperado, sendo que a Internet é utilizada como principal ferramenta de pesquisa por todos.

5. Armazenamento de Informação

Vemos que 72% dos colaboradores mantêm as informações importantes em arquivos em seus computadores e que 65% também usam o e-mail como forma de armazenar informações. Para uma melhor conclusão, seria necessário saber o tipo das informações que estão sendo guardadas e se poderiam ser compartilhadas ou não. Um problema em utilizar e-mail para guardar informações é que é difícil reaver esses dados, e, além disso, as informações ficam restritas aos destinatários do e-mail, não gerando um conhecimento comum para a CERTI.

6. Compartilhamento de Endereços Eletrônicos

As duas maneiras mais utilizadas para trocar endereços são e-mail e mensagens instantâneas. São formas de alertar diretamente os interessados (quem envia o link espera que seja útil ao destinatário), porém restringe-se esse conhecimento apenas para o grupo, ou, no caso das mensagens instantâneas, apenas aos usuários que estiverem disponíveis no momento. Ferramentas de social bookmarking surgiram para resolver os problemas apontados, uma vez que os links ficam disponíveis permanentemente.

7. Acesso aos Ramais

Na CERTI, guias de ramais são impressos e distribuídos aos colaboradores, por isso temos 70% de colaboradores utilizando este mecanismo. Apesar de ter uma certa praticidade, é comum o guia impresso ficar rapidamente desatualizado. A idéia do guia na Intranet é justamente manter os dados atualizados e agilizar a busca.

Apesar da pouca divulgação da Intranet na CERTI no momento da pesquisa, em geral, 27% dos colaboradores usam também a intranet para buscar ramais.

8. Compartilhamento de Arquivos

A forma mais comum de se compartilhar arquivos é através do e-mail, superando a utilização dos diretórios da CERTI. Apesar de certa praticidade essa prática causa alguns problemas já citados anteriormente decorrente do uso de e-mail, como a restrição do acesso ao conteúdo.

9. Grau de Satisfação com a Estrutura de Diretórios

A estrutura de diretórios, apesar de ser considerada estável e razoavelmente fácil de acessar, foi descrita como confusa e burocrática por boa parte dos usuários, o que dificulta a produtividade. Isso pode ser o motivo pelo qual os colaboradores preferem utilizar e-mail para trocar arquivos conforme visto na questão anterior.

10. Autoria Colaborativa

Para escrever textos em parceria, vemos que os colaboradores estão utilizando a estrutura de diretórios da CERTI. A próxima questão responde como estão sendo controladas as versões, para o usuário saber qual é a versão mais recente do arquivo, a anterior, etc.

11. Controle de Versão

A maioria dos colaboradores utiliza o sistema de nomenclatura padrão da CERTI para identificar as versões dos arquivos. Essa padronização é importante pois torna a localização das versões um pouco mais fácil. Porém, esse sistema possui algumas desvantagens em relação aos wikis, como edição através da web, facilidades de busca e sistema de controle de versões automático. Como podemos ver, no CCD e IS, a adoção do wiki foi grande, tendo praticamente igualdade de usuários usando wiki e nomenclatura padrão da CERTI.

12. Experiência com Tecnologias de Informação e Softwares Sociais

Podemos perceber que a grande parte dos colaboradores desconhece algumas ferramentas que pretende-se implantar, como é o caso do Social Bookmarking desconhecido por 47% dos colaboradores e os web feeds RSS por 39%. Isso significa que para proliferar o uso dessas ferramentas, devemos desenvolver mecanismos para divulgar os benefícios e a forma de utilização das mesmas.

Blogs e wikis são conhecidos pela grande maioria, porém poucos utilizam com frequência, sendo a maioria apenas leitores e não participantes ativos. Devemos estar preparados para lidar com possíveis barreiras culturais que possam ocorrer devido à essa característica.

3.2 MODELO PROPOSTO

Paralelamente à aplicação do questionário e das reuniões com a equipe, foi realizado um estudo buscando experiências de desenvolvimento de sistemas

semelhantes, sendo que o artigo *“The Wiki and the Blog: Toward a Complex Adaptive Intelligence Community”* (ANDRUS, 2005) foi o mais inspirador.

O artigo baseia-se em seu estudo sobre Sistemas Adaptativos Complexos para propor um novo ambiente para o compartilhamento de informações entre os agentes da Agencia Central de Inteligência dos Estados Unidos - CIA. O sistema é inspirado nos princípios de auto-organização, emergência, realimentação, relacionamento, adaptabilidade e não-linearidade.

O autor baseou-se na junção de cinco tecnologias para atingir os objetivos: repositório de arquivos, wiki, blog, sistema de busca e mecanismo de notificação.

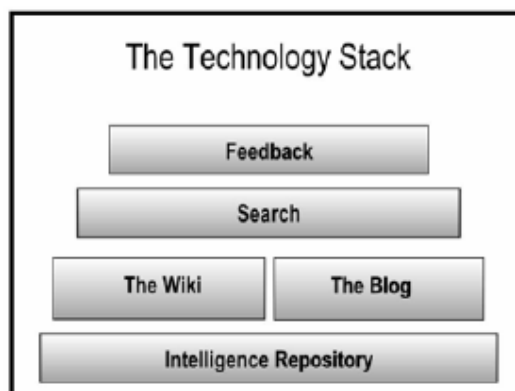


Figura 2 Pilha de Tecnologias – fonte: Calvin Andrus

Este foi o modelo que mais se aproximou do almejado para a intranet e por isso foi escolhido como base, sendo que o modelo proposto neste trabalho adiciona alguns elementos à pilha, focando atender as necessidades apontadas pela pesquisa e pelos acompanhamentos realizados.

Os sistemas que compõe o modelo da intranet são os seguintes:

- Serviço de Diretório
- Repositório de Arquivos
- Wikis
- Blogs

- Sistema de Busca
- Mecanismos de Classificação
- Mecanismos de Notificação

Modelo da Intranet

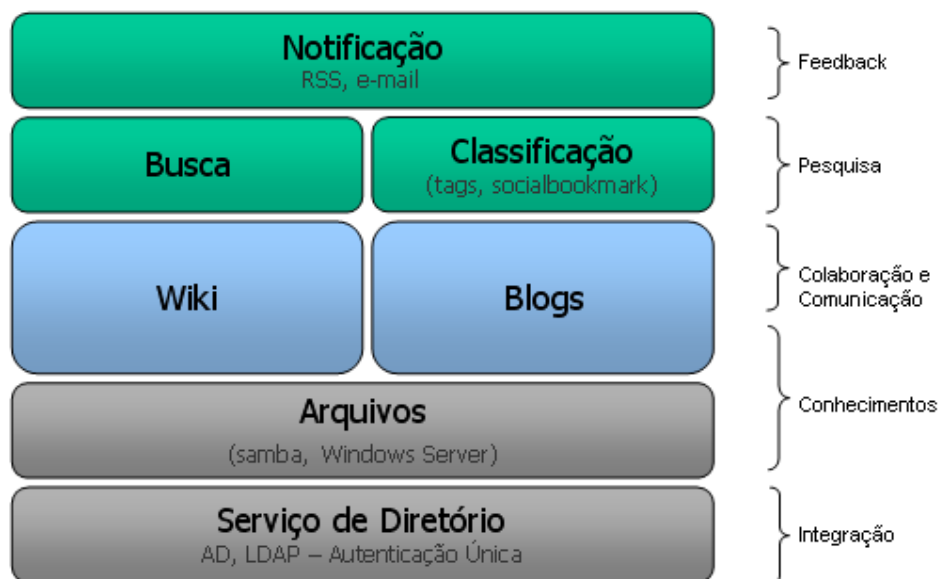


Figura 3 Modelo da Intranet

A seguir serão descritas as motivações que levaram a decisão de implantar os principais sistemas citados.

3.2.1 WIKI

Como foi descrito na justificativa para o trabalho, a Fundação CERTI disponibiliza aos seus colaboradores vários serviços na rede interna. Porém, a rede carece de um ponto central, que descreva e possibilite o acesso a todos estes serviços. Um portal se faz necessário para sanar esta situação.

Um fato que ocorre na CERTI é que vários projetos são desenvolvidos em paralelos, e muitas equipes desconhecem o que as outras estão fazendo. Isso faz com que muitas vezes trabalhos semelhantes sejam desenvolvidos por equipes distintas sem a troca de conhecimento entre ambas. Um sistema de cadastro de projetos, permitindo o compartilhamento informações ajudaria a contornar esse problema.

Outro ponto relevante, é que devido a grande quantia de colaboradores, é praticamente impossível conhecer as habilidades e interesses de todos os colegas de trabalho. Uma rede social auxiliaria a encontrar colegas com interesses em comum e a encontrar um parceiro para tirar dúvidas sobre um determinado tema.

Essas três situações levam a sugestão de implantação de um sistema wiki. Nele é possível centralizar o acesso aos serviços da CERTI, servindo como um portal, onde novos serviços seriam facilmente incluídos por qualquer colaborador.

O wiki também permite criar áreas para cadastro de projetos. Nesse espaço, é possível adicionar informações relevantes ao projeto, como descrever tecnologias e produzir a documentação.

Por fim, cada usuário do wiki possui uma página pessoal, onde define seu perfil, compartilha seus gostos e conhecimentos, criando assim o sistema de rede social.

3.2.2 BLOGS

No contexto de um projeto, o wiki por si só não é completo. Falta uma ferramenta que permita uma flexibilidade maior, uma ferramenta que permita a troca de notícias relevantes ao projeto e que auxilie na exposição de decisões tomadas e no debate de caminhos a serem seguidos.

A ferramenta que mais se encaixa nesse modelo proposto é o blog. Nele é possível divulgar notícias e decisões, tendo uma área de discussão (os comentários) em que os colegas podem trocar idéias.

A princípio serão criados blogs individuais, onde o colaborador pode expor seus interesses e blogs coletivos, sobre temas e projetos, onde serão feitas discussões e decisões.

3.2.3 SOCIAL BOOKMARKS

Em uma empresa de inovação como a CERTI, é fundamental a pesquisa e a troca de informações entre os colaboradores. Provavelmente o meio mais comum de pesquisa é a Internet. A troca de endereços eletrônicos, referências para pesquisa, é uma forma bastante comum de compartilhar informações.

Além disso, é importante classificar e quantificar a importância de um determinado endereço. Algo que não é possível através dos sistemas convencionais usados no compartilhamento de endereços, como a troca informal através de mensagens instantâneas e e-mails.

Para suprir essas necessidades é que o sistema de Social Bookmarking será implantado. Ele permitirá uma forma organizada e persistente para a troca de endereços eletrônicos.

3.2.4 WEB FEEDS

Como a intranet será bastante dinâmica, com atualizações freqüentes de páginas do wiki, novas entradas nos blogs, novos endereços cadastrados no social

bookmarking, é imprescindível a utilização de um sistema de notificações das alterações.

Para esse fim, todos os serviços disponibilizarão várias opções de Web feeds. Para acompanhamento dos projetos, o wiki deve fornecer feeds que possibilitem a notificação de alterações em um projeto específico, além de o blog do projeto também notificar suas mudanças.

Para uma visão mais geral, deve haver feeds globais, uma que notifique qualquer alteração no wiki e outra que agregue as novas entradas de todos os weblogs.

Para promover uma integração maior entre os sistemas, o wiki exibirá dinamicamente informações extraídas dos Web feeds dos outros serviços. Assim, acessando apenas o wiki, o usuário já descobrirá quais as novidades nos outros sistemas, podendo acessar apenas as informações que achar relevante.

4 IMPLEMENTAÇÃO E RESULTADOS

A implementação do modelo da intranet consistiu na instalação dos sistemas propostos e adaptação dos mesmos às necessidades da CERTI. Além da customização de todos os ambientes, foram desenvolvidos sistemas de cadastro de unidades, projetos e pessoas.

A rede da CERTI já possuía um serviço de diretório, fornecendo um mecanismo de autenticação centralizado. Esse recurso foi integrado à todos os sistemas, para facilitar o acesso dos usuários e evitar a necessidade do usuário ter que se cadastrar em cada sistema.

Outro recurso que também já estava implantado na Fundação CERTI era o servidor de arquivos. O sistema consiste em um servidor com uma estrutura de diretórios compartilhada acessíveis aos colaboradores, com controle de acesso.

O primeiro sistema a ser implantado foi o wiki, uma vez que seria o sistema central e, além disso, o que proporcionaria mais benefícios à equipe.

Após a análise de alguns sistemas wikis, o TWiki foi selecionado como sistema wiki por melhor atender aos requisitos, como controle de acesso bastante flexível, suporte à plugins, comunidade ativa, hierarquia (webs e subwebs). Outra característica de destaque do TWiki foi a facilidade de criar aplicações no próprio ambiente do TWiki, como formulários, cadastros e pesquisas.

Após instalado o TWiki, foi desenvolvido um tema customizado, visando facilitar a utilização por usuários leigos e também adicionar a identidade visual da CERTI.

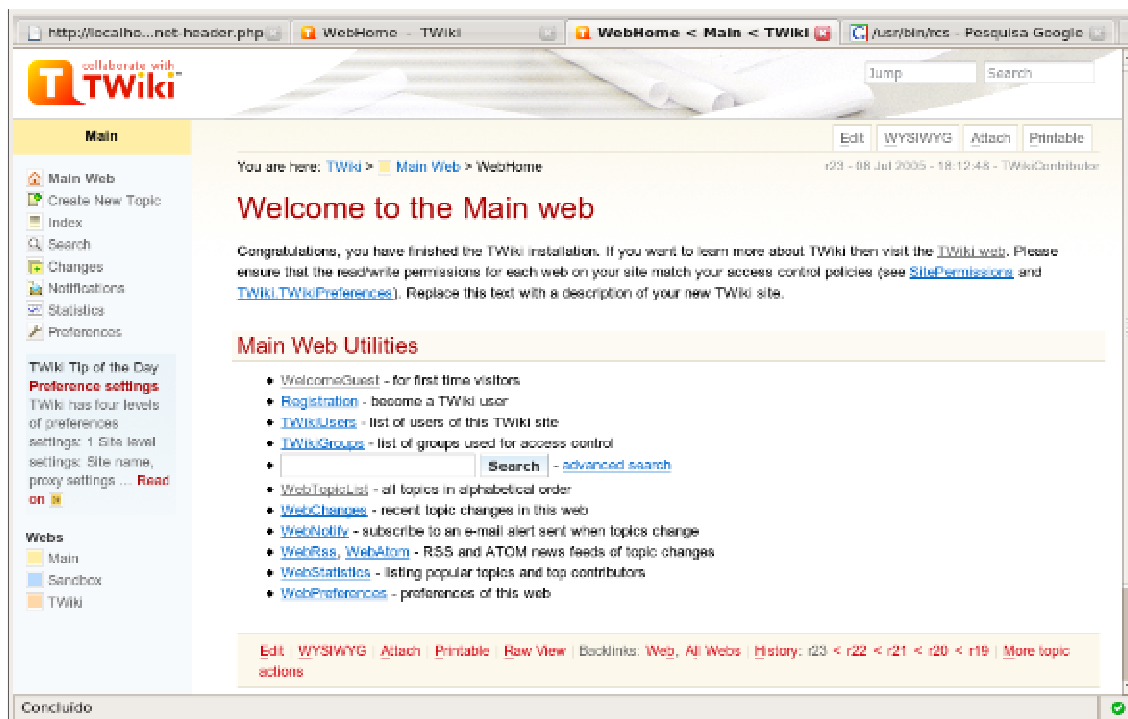


Figura 4 - Página inicial padrão do TWiki



Figura 5 - Página inicial customizada

Em seguida, foram desenvolvidas aplicações TWiki para cadastro e pesquisa de projetos, criados os espaços das unidades e a listagem dos colaboradores.

O TWiki também possui um sistema de notificação através de web feeds RSS, porém o mesmo se mostrou ineficiente, pois não permitia flexibilidade na escolha do conteúdo dos web feeds e não guardava o histórico completo das mudanças, como a inclusão de anexos e alterações subsequentes em um tópico. Para sanar esta necessidade, foi desenvolvido o plugin TimeLineDataPlugin para garantir um registro mais preciso das mudanças e maior flexibilidade do conteúdo dos web feeds.

A modificação mais trabalhosa foi a implantação e melhoria do Editor Simples (WYSIWYG) do TWiki. A forma edição mais comum de tópicos do TWiki é através de texto puro com a sintaxe de marcação do TWiki, mas também é possível utilizar um plugin de editor simples (WYSIWYG), para facilitar a edição por usuário inexperientes, evitando a memorização da linguagem de marcação e permitindo um acompanhamento visual de como ficará o texto. Porém, este editor apresentava vários problemas de usabilidade, falta de ferramentas básicas e inconsistências na transformação para a linguagem de marcação do TWiki.

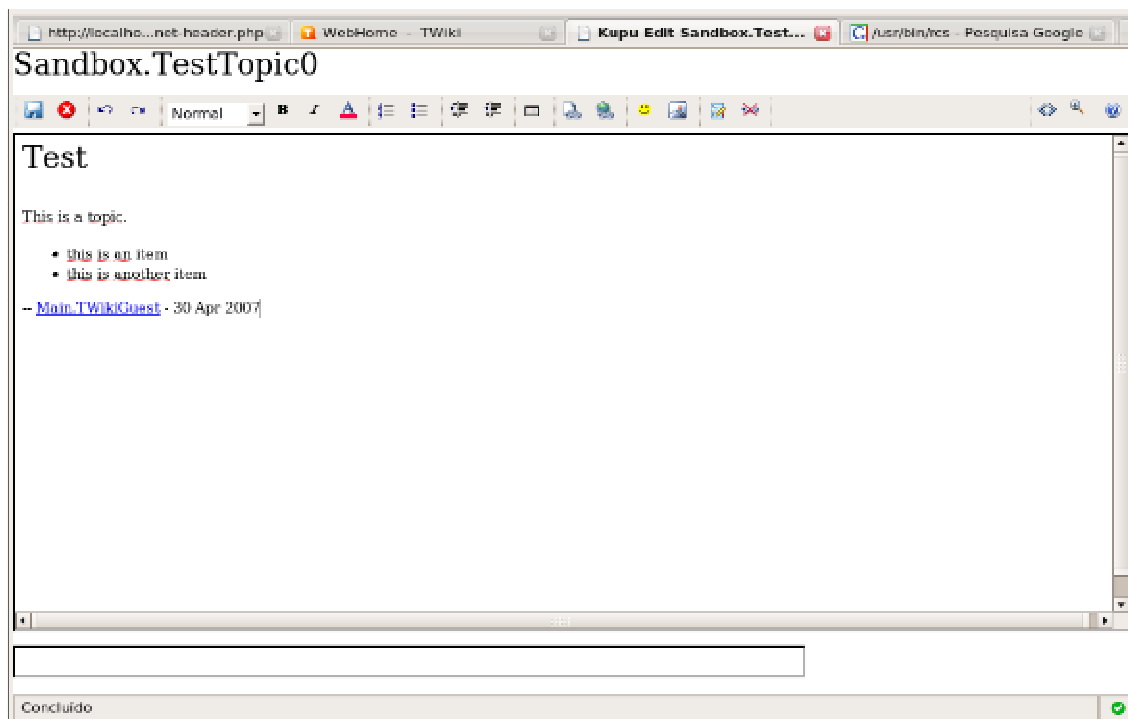


Figura 6 - Editor Simple versão original

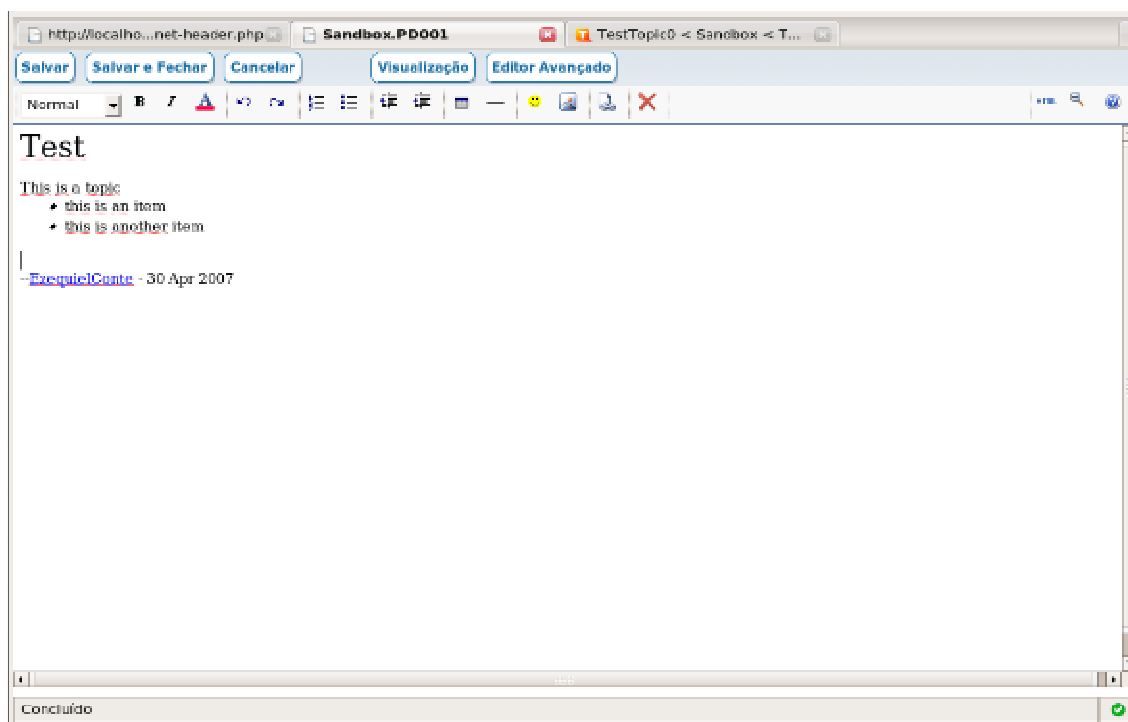


Figura 7 - Novo Editor Simple

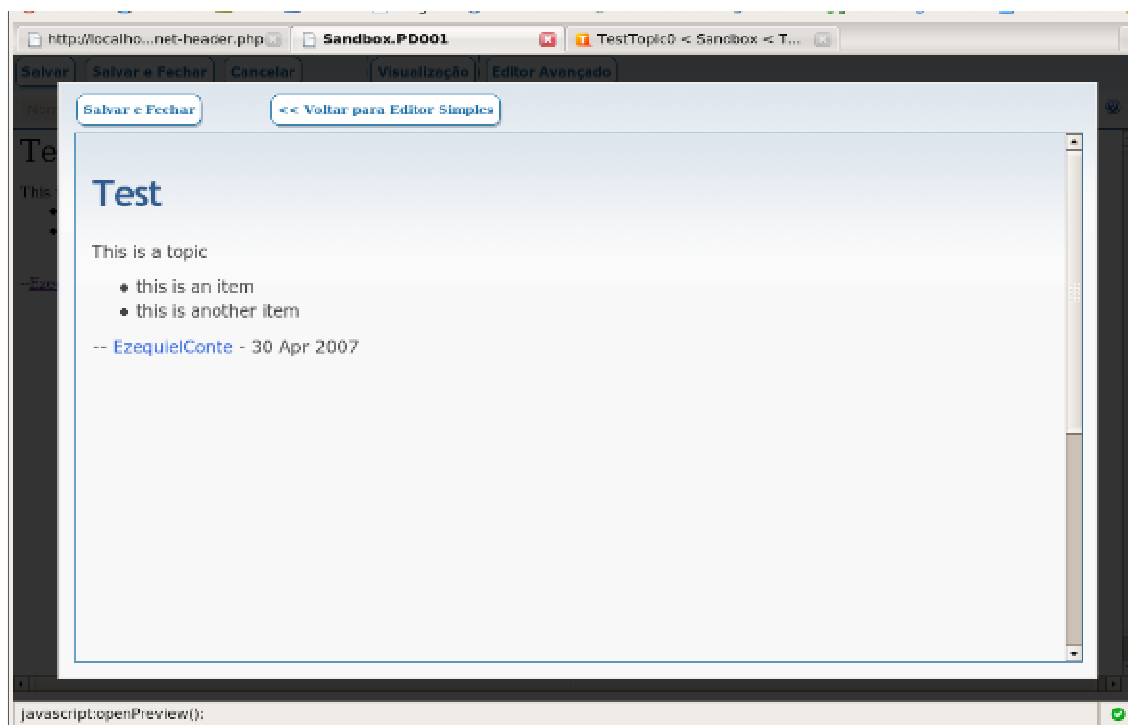


Figura 8 - Funcionalidade Visualização do Editor Simples

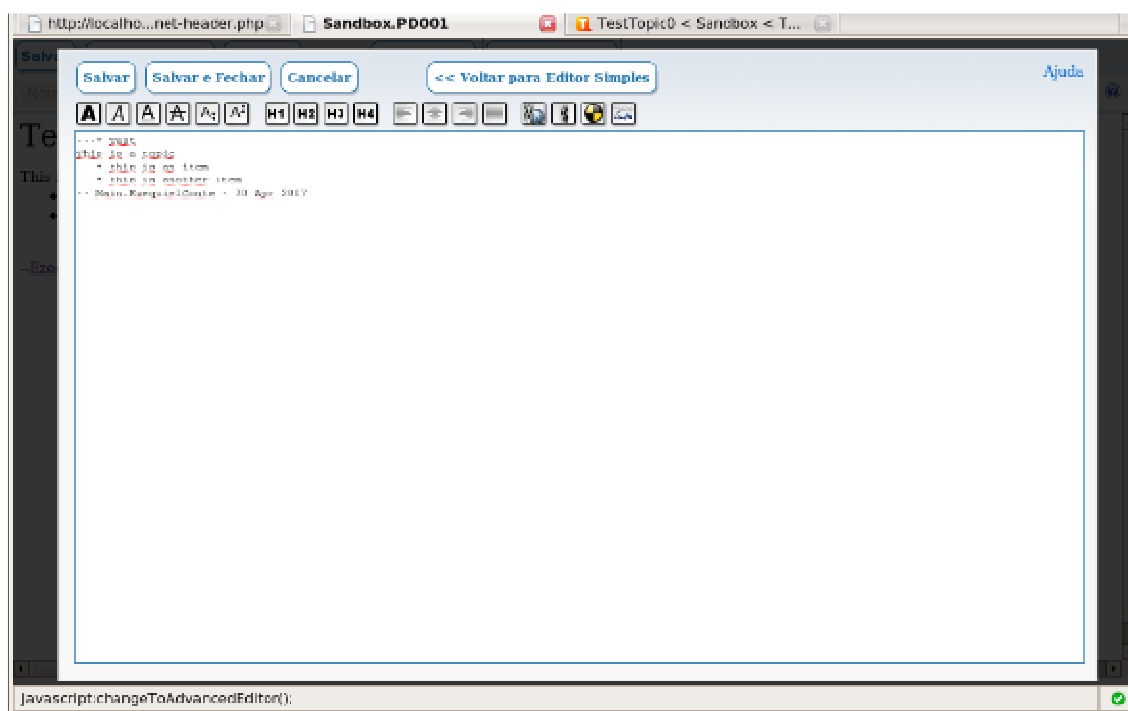


Figura 9 - Modo Avançado através de AJAX

Como esse editor era um elemento fundamental no âmbito da intranet, esses problemas citados foram sendo corrigidos e posteriormente revertidos para a comunidade TWiki no modelo de software livre.

Para o sistema de blogs, o software selecionado foi o WordPress MU, dado sua característica de gerenciar múltiplos blogs, possuir notificações através de web feeds, ser software livre e ser o sistema mais popular, contando com centenas de plugins e temas.

A primeira etapa da implantação do WordPress MU foi a instalação do mesmo e integração com o sistema de diretórios. Em seguida, foi desenvolvido um tema padrão para novos blogs com a identidade visual da Fundação CERTI. Além disso, a usabilidade foi melhorada através da adição de novos itens de menu para facilitar o acesso aos web feeds e às funções básicas, como a postagem de novo artigo.

O maior trabalho desenvolvido neste sistema foi a criação de uma página inicial para o ambiente de blogs. A página inicial padrão do WordPress MU conta apenas com uma lista dos novos blogs e a opção de criar um novo blog. O objetivo era criar um portal, onde seriam listados os blogs mais populares, últimos posts, os blogs do colaborador, notícias e a lista de blogs por categoria. Esses objetivos foram atingidos através da instalação de alguns plugins, do desenvolvimento de um tema e do plugin para categorização dos blogs, devido aos diferentes usos dos blogs na CERTI (blogs de projeto, pessoais, de assunto e de centros).

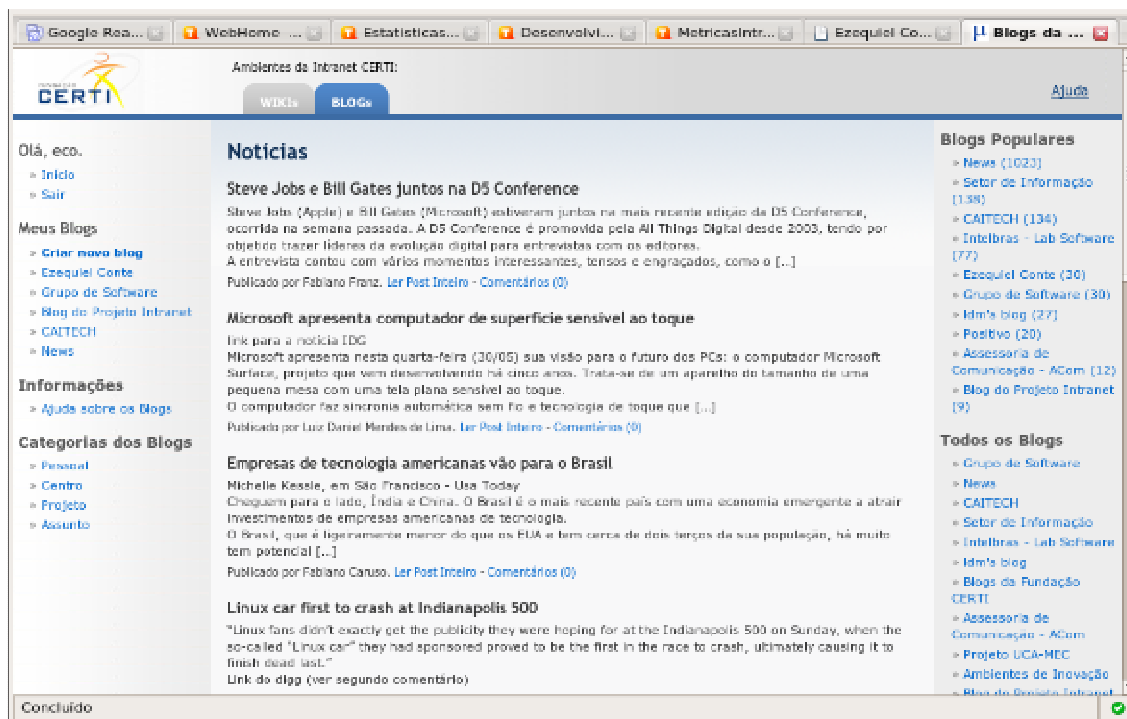


Figura 10 - Página Inicial do ambiente de blogs

O plugin para listar blogs por categoria era uma necessidade antiga do WordPress MU, sendo que o mesmo foi, a exemplo do editor do TWiki, compartilhado com a comunidade de software livre.

Durante o desenvolvimento já pode ser percebido alguns benefícios da participação da comunidade software livre. Sugestões levantadas por pessoas externas foram sendo implantadas nos plugins, melhorando o sistema desenvolvido.

Como o processo de desenvolvimento foi iterativo, lançando as implementações periodicamente, durante praticamente todo o período de desenvolvimento o sistema foi sendo utilizado, primeiramente contando com apenas o wiki e posteriormente com a inclusão dos blogs.

Isso possibilitou que métricas do sistema fossem monitorados durante todo o desenvolvimento. Assim, foi possível coletar estatísticas de acesso, número de usuários cadastrados, entre outros dados, que serão analisados posteriormente.

Um resultado positivo foi no projeto de análise do laptop do projeto One Laptop Per Child. O objetivo do projeto era fazer um relato do estado do laptop e verificar se alguns sistemas da empresa Positivo funcionavam corretamente. Com a utilização do wiki e blogs, foi possível elaborar textos coletivamente, manter todos informados com as notícias recentes e um acompanhamento em tempo real pelo cliente.

Os testes do laptop eram feitos pela equipe e anotados na página do projeto no sistema wiki. Cada membro poderia anotar suas conclusões e complementar a dos colegas. Inclusive o cliente poderia complementar os relatos, pois tinha acesso ao sistema e era notificado sobre novas mudanças através de web feeds.

No blog do projeto, notícias relacionadas ao projeto eram publicadas, tornando homogêneo o conhecimento da equipe.

Após este, outros projetos também ocorreram de maneira semelhante, também com resultados positivos.

Um lado negativo foi a dificuldade de alguns usuários adotarem os sistemas, principalmente, no caso do wiki, pela necessidade de aprender uma sintaxe para edição ou então desmotivados pelos problemas iniciais do editor simples, alguns corrigidos atualmente.

4.1 ESTATÍSTICAS DE USO DO SISTEMA

Uma boa maneira de acompanhar a utilidade de um sistema web é através estatísticas de acesso e uso. Através delas podemos perceber quais são elementos mais importantes do sistema e verificar se as divulgações surtiram o efeito esperado.

Desde o começo do projeto, o ambiente wiki foi monitorado, fornecendo estatísticas como o número de acessos, visitantes únicos, número de edições, colaboradores mais ativos, entre outros.

O ambiente de blogs passou a ser monitorado mais recentemente devido sua implantação tardia em relação ao wiki e, por isso, não está contido nos resultados a seguir. Abaixo está uma tabela com alguns dados coletados no decorrer do projeto.

Mês / Ano	Total de Tópicos	Tópicos Acessados no Mês	Modificações no Mês	Total de Usuários
Outubro 2006	660	4831	136	6
Novembro 2006	807	10653	528	14
Dezembro 2006	1093	15347	603	25
Janeiro 2007	1301	21958	856	34
Fevereiro 2007	1509	17308	965	43
Março 2007	1709	18946	1710	60
Abril 2007	2008	19222	923	65
Mai 2007	2156	23392	1837	70

Para facilitar a interpretação, serão listados a seguir os gráficos relativos a estes dados, bem como sua análise.

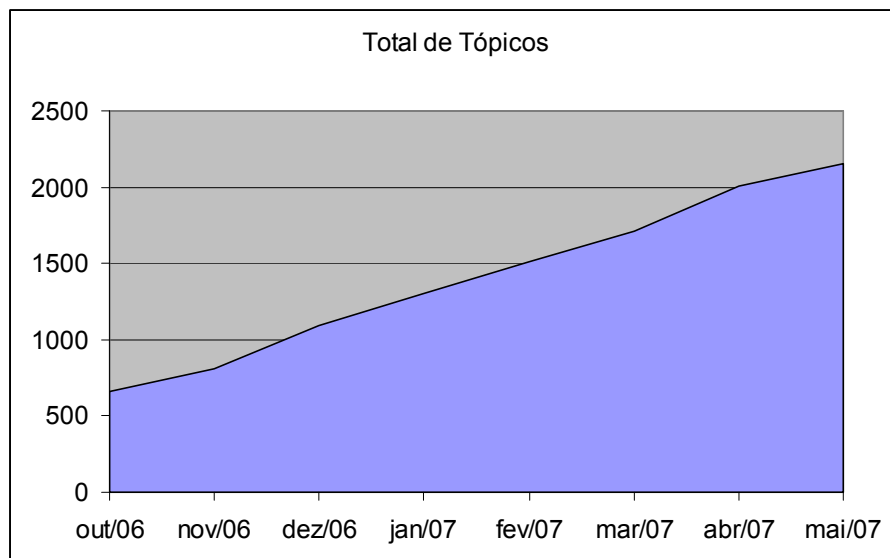


Figura 11 - Estatísticas: Total de Tópicos

O número total de tópicos na intranet representa a criação, a geração de novos conhecimentos. Pode-se perceber que possui um crescimento praticamente linear, principalmente devido a característica do wiki ser usado na documentação dos projetos. A média foi de 213,7 tópicos criados mensalmente.

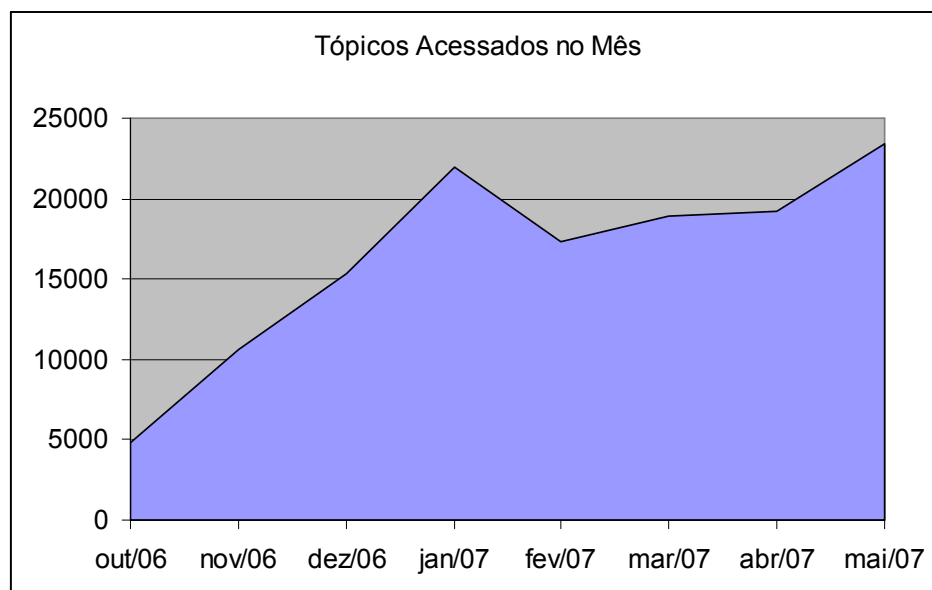


Figura 12 - Estatísticas: Tópicos Acessados no Mês

O número de tópicos acessados no mês representa a troca de conhecimento entre os membros dos projetos: alguns colaboradores geram conteúdo, outros consomem. Podemos perceber um pico em janeiro, ocorrido pelas divulgações da lista de ramais e da pesquisa. O gráfico apresenta um crescimento pois houve medições durante o decorrer do projeto, quando não havia muita divulgação da intranet. Provavelmente chegará a um valor relativamente constante, como apresentado nos últimos quatro meses. O número médio de tópicos visitados mensalmente ficou em 16457,1.

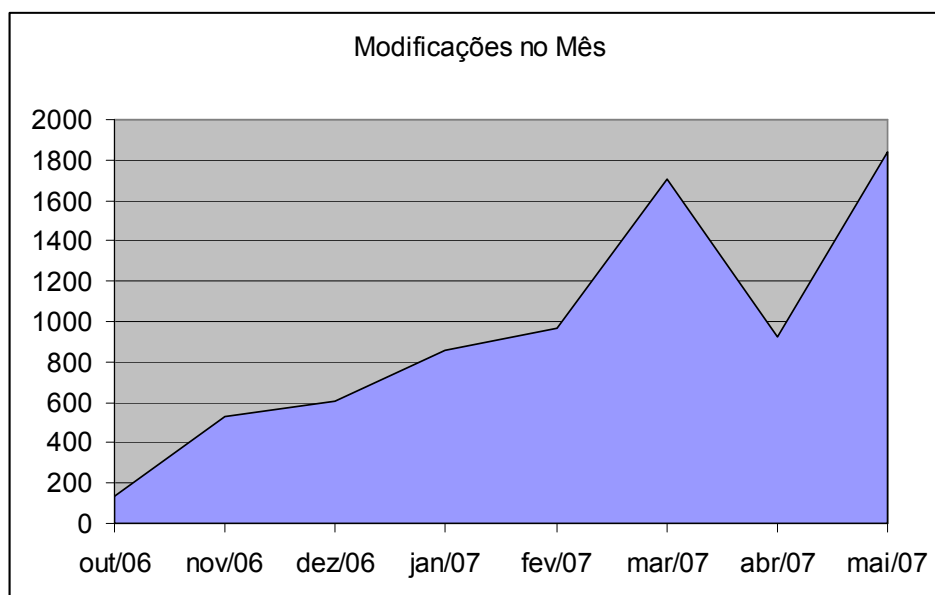


Figura 13 - Estatísticas: Modificações no Mês

As modificações realizadas sugerem a colaboração entre os membros da equipe, a complementação de idéias e melhoria da qualidade dos tópicos. Esse valor será bastante variável, dependendo principalmente das atividades de cada projeto. A média foi de 944,7 modificações mensais.

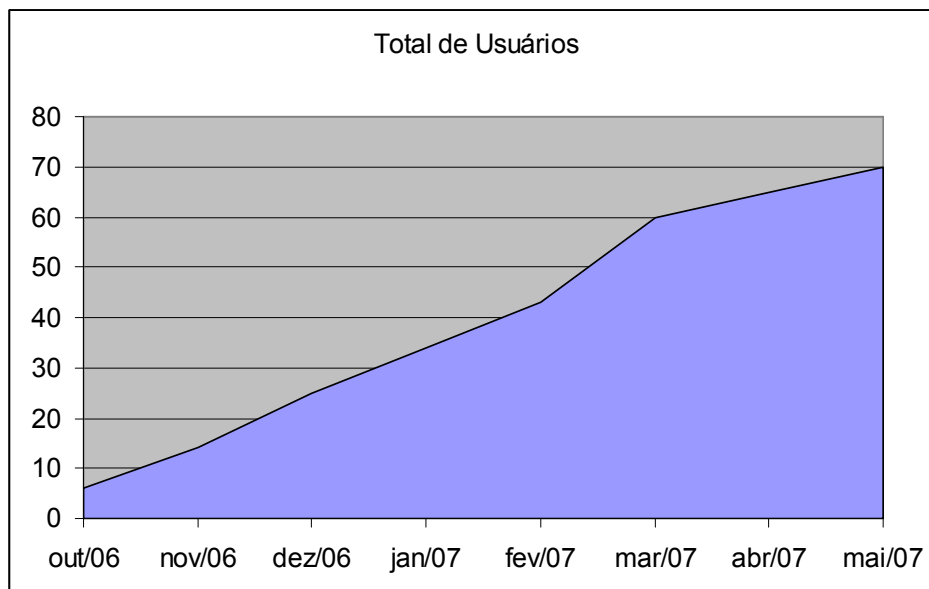


Figura 14 - Estatísticas: Total de Usuários Cadastrados no Sistema

O total de usuários cadastrados no sistema aparenta um crescimento linear devido ao pouco tempo de análise. Futuramente o comportamento esperado é o de uma curva logarítmica pois atingirá um valor próximo ao número máximo do sistema que é o número de colaboradores da CERTI. Em média, 8,7 colaboradores se cadastraram no sistema por mês.

Faz-se necessário esclarecer que o total de usuários cadastrados no sistema representa os usuários que criaram uma página pessoal. Os tópicos do wiki são acessíveis e modificáveis por todos os colaboradores da Fundação CERTI. O cadastro é incentivado no intuito de facilitar a interação entre os colaboradores e facilitar o acesso à algumas informações pessoais.

Através da análise pode-se perceber que o sistema está sendo efetivamente utilizado e apresenta um bom crescimento. Porém, seria interessante uma análise futura para conseguir uma verificação melhor dos efeitos da utilização do sistema.

5 CONCLUSÕES E TRABALHOS FUTUROS

O objetivo deste trabalho foi propor um modelo para a intranet da Fundação CERTI com base nos inovadores recursos de Enterprise 2.0. A meta era criar um sistema que atendesse às necessidades da CERTI, fornecendo um ambiente incentivador para as atividades de criação, colaboração e comunicação entre os colaboradores.

Para atender à proposta, foi necessário fazer um estudo sobre Enterprise 2.0, Web 2.0 e softwares sociais, além de intranets. Estes são temas que abrangem áreas variadas como a engenharia de software, usabilidade, gestão do conhecimento e sistemas colaborativos. Além disso, foi de grande valor a experiência de participar em projetos de software livre e conhecer suas características.

Na etapa de concepção do modelo, foi realizada uma pesquisa para analisar o comportamento dos colaboradores em relação às atividades diárias no computador e o grau de experiência com os softwares que se desejava implantar. Isso permitiu desenvolver um modelo voltado às necessidades dos colaboradores.

A implementação feita de forma iterativa com lançamentos frequentes foi de grande proveito, pois permitiu que o sistema fosse desenvolvido baseado nas necessidades reais apontadas pelos próprios usuários.

Pelos resultados, pode-se perceber que o sistema está de maneira geral sendo bem recebido, com bom nível de utilização, apesar do pouco tempo de uso.

Como sugestão de trabalhos futuros, poderia ser implantado o sistema em outras empresas, verificando se o modelo proposto é adequado à outras realidades e propondo melhorias ao mesmo.

Outra sugestão seria dar continuidade ao projeto, agregando novas funcionalidades ao sistema e analisando os benefícios que a intranet baseada em recursos de Enterprise 2.0 pode trazer.

REFERÊNCIAS BIBLIOGRÁFICAS

ANDRUS, D. C. The Wiki and the Blog: Toward a Complex Adaptive Intelligence Community. *Studies in Intelligence*, Washington, v. 49, n. 3, 2005.

BUFFA M. Intranet Wikis. *IntraWeb Workshop WWW2006*. Edimburgo, 2006.

DORSETT, L.; FONTAINE, M. A.; DRISCOLL, T. Redefining Manager Interaction. *Knowledge Management Review*, Melcrum Publishing, 2002.

GREEN, D. T.; PEARSON, J. M.; Social Software and Cyber Networks: Ties That Bind or Weak Associations within the Political Organization?. *38th Hawaii International Conference on System Sciences*, 2005.

HILLS, M. *Intranet como Groupware*. São Paulo: Berkeley. 1997. 252 p.

FINCK, N.; HODDER, M.; STONE, B. Enhancing Internal Communications with Blogs, Wikis, and More. Seattle, 2005 Disponível em: <<http://www.nickfinck.com/presentations/bbs2005/01.html>>. Acesso em: 23 jan. 2007

GILROY, K.; IVES, B. Preparing for Intranet 2.0: how to integrate new communication technology into your intranet. In: SUMMER, J. *Transforming Your Intranet*. Melcrum Publishing, 2006. p. 65-81.

HINCHCLIFFE, D. Enterprise Web 2.0. Alexandria, Virginia. Disponível em: <<http://blogs.zdnet.com/Hinchcliffe/>>. Acesso em: 15 jan. 2007.

MAJCHRZAK, A.; WAGNER, C.; YATES, D. Corporate wiki users: results of a survey. International Symposium On Wikis, Odense, p. 99-104, 2006.

MCAFEE, A. P. Andrew McAfee's Blog. Cambridge, 2006. Disponível em: <<http://blog.hbs.edu/faculty/amcafee/index.php>>. Acesso em: 20 jan. 2007.

MCAFEE, A. P. Enterprise 2.0: The Dawn of Emergent Collaboration. MIT Sloan Management Review, Cambridge, v. 47, n. 3, p.21-28, 2006.

OWEN, M. et al. Social software and learning: An Opening Education report from Futurelab. Futurelab, 2007. Disponível em: <http://www.futurelab.org.uk/research/opening_education/social_software_01.htm>. Acesso em: 14 maio 2007

SOCIAL Bookmarking. WIKIPÉDIA: a enciclopédia livre. Wikimedia, 2007. Disponível em: <http://en.wikipedia.org/wiki/Social_bookmarking>. Acesso em: 21 jan. 2007.

TAPSCOTT, D. Winning with the Enterprise 2.0. New Paradigm, IT&CA Big Idea Series, 2006.

TAPSCOTT, D.; WILLIAMS, A. D. *Wikinomics*. Washington : Portfolio. 2006. 320 p.

TWIKISITE. TWiki.org, 2006. Disponível em: <<http://twiki.org/cgi-bin/view/TWiki/TWikiSite>>. Acesso em: 21 jan. 2007.

WEB 2.0. WIKIPÉDIA: a enciclopédia livre. Wikimedia, 2007. Disponível em: <http://pt.wikipedia.org/wiki/Web_2.0>. Acesso em: 21 jan. 2007.

WHITE, M. Creating an Effective Intranet. Horsham, 2003. Disponível em: <<http://www.intranetfocus.com/information/effectiveintranets.pdf>>. Acesso em: 17 jan. 2007.

ANEXO I - QUESTIONÁRIO SOBRE FERRAMENTAS DE TECNOLOGIA NO TRABALHO

Sigla: __ Idade: __ anos Centro: _____

1) Quais as primeiras coisas que você faz diariamente na CERTI, no seu computador?

- Lê notícias na Internet (Folha On-line, etc.)
 Lê notícias em RSS
 Verifica o e-mail
 Manda notícias para colegas
 Trabalha nos arquivos do dia anterior
 Lê BLOGs
 Vê páginas de Internet em geral
 Verifica sua agenda para o dia
 Vê sua página pessoal na Internet (Netvibes, etc.) Qual? _____

 Outras: _____

2) Utilizando a escala abaixo mencionada, defina a importância das seguintes questões para o seu trabalho/função na CERTI?

	Sem importância	Pouco importante	Importante	Essencial
Acompanhar as atividades de projeto				
Controlar as alocações de pessoal				
Informações das agendas de viagem e reuniões				
Receber informações atualizadas diariamente				
Receber informações sobre a CERTI				
Receber informações sobre Editais em geral				
Acessar rapidamente os ramais dos colegas				
Registrar as atividades do dia-a-dia				
Comunicar-se de forma instantânea (MSN, etc.)				
Compartilhar informações/notícias				
Compartilhar arquivos				

Estudar assuntos em equipe/grupo				
Escrever textos em parceria (ex:propostas, documentos)				

Outras atividades importantes que queira mencionar :

3) O que você utiliza para se comunicar com os seus colegas de trabalho?

- Conversa pessoalmente
- Recados em papel, texto impresso, post-its
- E-mail
- Telefone , VoIP ou Skype
- Mensagens Instantâneas (MSN, ICQ, Google Talk)
- BLOGs
- Wikis
- Através da secretária
- Não preciso
- Outros: _____

4) O que você utiliza para obter informação ou estudar para suas atividades na CERTI?

- Páginas na Internet (bookmarks, busca,etc.)
- Wikis
- BLOGs
- Email
- INF (CERTI)
- Notícias em RSS
- Revistas e Jornais online
- Revistas e Jornais impressos
- Livros em PDF
- Livros impressos
- Outros: _____

5) O que você utiliza para registrar/guardar informações importantes (agenda, endereços, notícias, links,etc.) ?

- Agenda/calendário impressos, folhas de papel, etc.
- Agenda/calendário ou arquivos no seu computador (.txt, .doc, etc.)
- Arquivos na rede da CERTI (.txt, .doc, etc.)
- Calendário ou Agenda na Internet (online)
- Wikis
- Blogs
- E-mail
- Registra com a Secretária (envia por e-mail, entrega impresso, etc.)
- Não preciso/não registro
- Sistema da CERTI: _____
- Outros: _____

6) O que você utiliza para trocar links (endereços eletrônicos) importantes com seus colegas?

- E-mail
 Mensagens Instantâneas (MSN, ICQ, Google Talk)
 Wikis
 BLOGs
 Listas de Discussão
 Outros: _____

7) Como você faz para saber um ramal ou os dados de alguém da CERTI ou empresa?

- Liga pra secretária
 Caderno impresso dos ramais
 Agenda (física) ou caderno
 Arquivos locais (no seu PC)
 Arquivos na rede da CERTI
 Não preciso
 Agenda celular
 Sistema da CERTI: _____ []
 Outros: _____

8) Quando precisa compartilhar um arquivo com o colega, você:

- Disponibiliza em um diretório compartilhado no seu computador ou no PC dele
 Copia no diretório compartilhado de um terceiro (secretária, etc.)
 Manda por e-mail em anexo ("atachado")
 Grava em CD/DVD/pen-drive (dispositivo móvel de armazenagem) e entrega pra ele
 Grava em um diretório padrão na rede (servidor CERTI)
 Usa um serviço de compartilhamento na Internet (Yousendit, etc.)
 Envia através de mensagem instantânea (MSN, Skype, Google Talk)
 Outros: _____

9) A estrutura atual de diretórios , no servidor da CERTI, para você é:

Clara	5	4	3	2	1	Confusa
Fácil de acessar	5	4	3	2	1	Difícil de acessar
Flexível	5	4	3	2	1	Burocrática
Estável	5	4	3	2	1	Instável
Produtiva	5	4	3	2	1	Improdutiva

10) Ao precisar criar um arquivo de texto junto com outros colegas, você:

- Troca e-mails com o texto em questão
 Trabalha num arquivo local e troca as diferentes versões com ele por e-mail
 Trabalha no arquivo na rede (servidor CERTI) e informa o caminho para o colega
 Utiliza uma ferramenta na Internet (Wikis, BLOGs etc.)
 Outros: _____

11) Como você faz para controlar as versões dos arquivos de trabalho:

- Utiliza o padrão da CERTI de nomenclatura ou semelhante (V00, V01.. no final do nome)
 Vai alterando o nome para indicar as mudanças (revisado, final, etc.)
 Trabalha numa única versão e utiliza as ferramentas de revisão do software (ex.:Word)
 Utiliza uma ferramenta na Internet (Wikis, BLOGs etc.)
 Não utiliza/não sente necessidade
 Outros: _____

12) Qual sua experiência com as seguintes ferramentas?

	Utilizo com Frequência	Já Utilizei	Conheço mas nunca utilizei	Não conheço
E-mail				
Mensagens Instantâneas (MSN, Skype, Google Talk, etc.)				
RSS				
Blogs				
Wikis (wikipedia, etc.)				
Bookmark Social (del.icio.us, etc.)				
Edição On-line Colaborativa (google docs, etc.)				
Redes Sociais (orkut, myspace, etc.)				
Agenda Compartilhada (google calendar, etc.)				
Página Pessoal Customizável (Netvibes, google ig, etc.)				

- Experiências positivas e negativas com as ferramentas que queira mencionar:

13) Seus comentários sobre uso do e-mail e outras ferramentas de tecnologia no trabalho:

ANEXO II - DADOS COLETADOS DO QUESTIONÁRIO SOBRE FERRAMENTAS DE TECNOLOGIA NO TRABALHO

	CERTI	CCD/IS	Total
Total de respostas	39	21	60

Quais as primeiras coisas que você faz profissionalmente na CERTI em seu computador?			
	CERTI	CCD/IS	Total (%)
Lê notícias na Internet (Folha On-line)	14	11	43,1%
Lê notícias em RSS	5	12	29,3%
Verifica o e-mail	37	19	96,6%
Manda notícias para colegas	5	6	19,0%
Trabalha nos arquivos do dia anterior	17	5	37,9%
Lê BLOGs	3	2	8,6%
Vê páginas de Internet em geral	8	7	25,9%
Verifica sua agenda para o dia	22	7	50,0%
Vê sua página pessoal na Internet	5	6	19,0%
Outras (especifique)	2	4	10,3%
Total de respostas	37	21	
Não responderam	2		

Utilizando a escala abaixo mencionada defina a importância das seguintes questões para o seu trabalho/função na CERTI?		
	Essencial/Importante	Pouco/Sem Importancia
Acompanhar as atividades de projeto	87,5%	14,3%
Controlar as alocações de pessoal	42,9%	58,9%
Informações das agendas de viagem e reuniões	57,1%	44,6%
Receber informações atualizadas diariamente	89,3%	10,7%
Receber informações sobre a CERTI	80,4%	21,4%
Receber informações sobre Editais em geral	48,2%	53,6%
Acessar rapidamente os ramais dos colegas	83,9%	17,9%
Registrar as atividades do dia-a-dia	78,6%	23,2%
Comunicar-se de forma instantânea (MSNetc.)	76,8%	25,0%
Compartilhar informações/notícias	76,8%	25,0%
Compartilhar arquivos	94,6%	7,1%
Estudar assuntos em equipe/grupo	87,5%	14,3%
Escrever textos em parceria (ex:propostas)	76,8%	25,0%

Outras atividades importantes para seu trabalho que queira mencionar:

- Elaborar propostas e enviar para clientes - Divulgar os produtos do CMI - Obter informações do mercado
- Receber informações das Visitas que a Fundação CERTI agenda com a Sapiens Parque e não informa a Secretaria. Receber informações sobre as mudanças que ocorrem na

Fundação CERTI. Receber informações do ARH sobre as novas contratações e a função a ser exercida pelo novo Colaborador. Receber do ARH as atribuições do Pessoal da SFA e do Pessoal da GLI é essencial p/ o bom desenvolvimento do meu trabalho. Receber informações institucionais atualizadas (Relatório de Atividades, Folderes, Portfólios ...)

- Receber informações relacionadas a atividade principal da unidade.
- Conhecer novas TI que possam ajudar no desenvolvimento do meu trabalho
- Comunicação com clientes; Projeto colaborativo (Compartilhamento de documentos, análise conjunta, criar áreas de projeto independentes, etc..)
- Melhorar a facilidade de acesso aos diretórios para agilizar o trabalho que está sendo executado.
- Ter sistema de pesquisa compartilhável, que permita análise e resumos por diversos colaboradores.
- Os ramais poderiam ser listados no Wiki ou em outra página na Intranet. Uso o caderno de endereços do Thunderbird, mas ele tem de ser atualizado manualmente. -
- Cursos, treinamento e certificações
- desenvolvimento colaborativo, acesso a informações que são de interesse geral
- Seminários

O que você utiliza para se comunicar com os seus colegas de trabalho?			
	CERTI	CCD/IS	Total (%)
Conversa pessoalmente	36	21	100,0%
Recados em papel	8	3	19,3%
E-mail	36	19	96,5%
Telefone	32	17	86,0%
Mensagens Instantâneas (MSN)	26	20	80,7%
BLOGs	1	0	1,8%
Wikis	4	5	15,8%
Através da secretária	8	1	15,8%
Não preciso	0	0	0,0%
Outros	1	1	3,5%
Total de respostas	36	21	
Não responderam	3		

O que você utiliza para obter informação ou estudar para suas atividades na CERTI?			
	CERTI	CCD/IS	Total (%)
Páginas na Internet	36	21	100%
Wikis	9	13	38,6%
BLOGs	9	13	38,6%
Email	26	9	61,4%
INF (CERTI)	10	3	22,8%
Notícias em RSS	8	9	29,8%
Revistas e Jornais online	16	12	49,1%
Revistas e Jornais impressos	14	12	45,6%
Livros em PDF	20	17	64,9%
Livros impressos	20	15	61,4%
Outros	5	1	10,5%
Total de respostas	36	21	
Não responderam	3		

O que você utiliza para registrar/guardar informações importantes (agenda endereços notícias linksetc.) ?			
	CERTI	CCD/IS	Total (%)
Agenda/calendário impressos	20	12	56,1%
Agenda/calendário ou arquivos no seu computador (.txt	25	16	71,9%
Arquivos na rede da CERTI (.txt	21	8	50,9%
Calendário ou Agenda na Internet (online)	12	9	36,8%
Wikis	2	0	3,5%
Blogs	2	1	5,3%
E-mail	28	9	64,9%
Registra com a Secretária (envia por e-mail	4	0	7,0%
Não preciso/não registro	0	0	0,0%
Sistema da CERTI ou Outros	5	2	12,3%
Total de respostas	36	21	
Não responderam	3		

O que você utiliza para trocar links (endereços eletrônicos) importantes com seus colegas?			
	CERTI	CCD/IS	Total (%)
E-mail	37	19	98,2%
MSN	14	13	47,4%
Skype	9	6	26,3%
Wikis	4	7	19,3%
BLOGs	4	0	7,0%
Listas de Discussão	1	3	7,0%
Outros	3	5	14,0%
Total de respostas	36	21	
Não responderam	3		

Como você faz para saber um ramal ou os dados de alguém da CERTI ou empresa?			
	CERTI	CCD/IS	Total (%)
Liga pra secretária	13	6	33,9%
Caderno impresso dos ramais	28	11	69,6%
Agenda (física) ou caderno	5	3	14,3%
Arquivos locais (no seu PC)	7	1	14,3%
Arquivos na rede da CERTI	3	2	8,9%
Não preciso	0	0	0,0%
Agenda celular	5	1	10,7%
Sistema da CERTI (Intranet)	8	7	26,8%
Outros	5	0	8,9%
Total de respostas	37	19	
Não responderam	2		

Quando precisa compartilhar um arquivo com o colega você:			
	CERTI	CCD/IS	Total (%)
Disponibiliza em um diretório compartilhado no seu computador ou no PC dele	20	11	54,4%
Copia no diretório compartilhado de um terceiro (secretária	2	0	3,5%

Manda por e-mail em anexo (atachado)	31	17	84,2%
Grava em CD/DVD/pen-drive (dispositivo móvel de armazenagem) e entrega pra ele	10	7	29,8%
Grava em um diretório padrão na rede (servidor CERTI)	29	9	66,7%
Usa um serviço de compartilhamento na Internet (Yousendit)	0	0	0,0%
Envia através de mensagem instantânea (MSN)	13	1	24,6%
Outros	0	0	0,0%
Total de respostas	37	20	
Não responderam	2		

A estrutura atual de diretórios no servidor da CERTI para você é:						
Clara	5	14	13	6	17	Confusa
Fácil de acessar	7	13	16	16	4	Difícil de acessar
Flexível	5	9	14	18	7	Burocrática
Estável	9	19	11	11	4	Instável
Produtiva	4	14	18	10	8	Improdutiva

Ao precisar criar um arquivo de texto junto com outros colegas você:			
	CERTI	CCD/IS	Total (%)
Troca e-mails com o texto em questão	10	4	25,0%
Trabalha num arquivo local e troca as diferentes versões com ele por e-mail	9	8	30,4%
Trabalha no arquivo na rede (servidor CERTI) e informa o caminho para o colega	29	10	69,6%
Utiliza uma ferramenta na Internet	5	10	26,8%
Outros	2	0	3,6%
Total de respostas	36	20	
Não responderam	2		

Como você faz para controlar as versões dos arquivos de trabalho:			
	CERTI	CCD/IS	Total (%)
Utiliza o padrão da CERTI de nomenclatura ou semelhante	28	11	68,4%
Vai alterando o nome para indicar as mudanças	10	6	28,1%
Trabalha numa única versão e utiliza as ferramentas de revisão do software (ex.:Word)	4	4	14,0%
Utiliza uma ferramenta na Internet	4	10	24,6%
Não utiliza/não sente necessidade	1	2	5,3%
Outros	4	2	10,5%
Total de respostas	37	20	
Não responderam	2		

Qual sua experiência com as seguintes ferramentas?				
	Utilizo com Freqüência	Já Utilizei	Conheço mas nunca utilizei	Não conheço

E-mail	37	0	0	0
Mensagens Instantâneas (MSN Skype Google Talk etc.)	26	9	2	0
RSS	8	5	9	14
Blogs	6	8	20	1
Wikis (wikipedia etc.)	10	14	8	3
Bookmark Social (del.icio.us etc.)	7	2	7	17
Edição On-line Colaborativa (google docs etc.)	6	9	8	13
Redes Sociais (orkut myspace etc.)	7	16	9	3
Agenda Compartilhada (google calendar etc.)	5	13	10	7
Página Pessoal Customizável (Netvibes google ig etc.)	6	5	12	12
Total de respostas	37			
Não responderam	2			

Experiências positivas e negativas com as ferramentas acima que queira mencionar:

- Utilizo computador como instrumento de trabalho para elaboração de textos e planilhas, armazenamento de arquivos, transmissão/recepção de informações, comunicação com os diretores, a equipe da ACOM e demais colaboradores. No entanto, prefiro sempre falar pessoalmente ou por telefone, quando preciso me comunicar. Além do email, utilizei o orkut e MSN, mas já não tenho interesse, nem paciência...
- Acho que falta algumas orientações para o uso do MSN e SKYPE.
- As redes sociais são um saco. O Google Calendar é muito interessante. Poderíamos adotar algo semelhante! Todos sabem os calendários de todos. Blogs legais, para mim, são os colaborativos, como o Overmundo.
- MSN- Não existe limite de troca de informação, as pessoas não possuem bom senso e a disciplina não existe. Observo que as pessoas perdem muito tempo com assuntos banais e partiuculares que não deveriam interferir no trabalho profissional
- Tentei utilizar com a equipe o google docs mas houve resistência.
- O uso de Wikis é muito útil porém ainda vai demorar para que as pessoas incorporem essa ferramenta no dia-a-dia. Acho que muitas pessoas ainda não conhecem a ferramenta ou acham muito complicada.
- Acho que as ferramentas de comunicação on-line tomam muito tempo e, na verdade, nunca se sabe se a outra pessoa não tinha algo mais urgente para fazer. Prefiro utilizar e-mail, mesmo que não seja instantâneo (depende de quando o outro vai abrir a mensagem) e, para coisas urgentes, o telefone (mais eficiente).
- Positiva: Skype - provê agilidade na comunicação Negativa: MS messenger - varias pessoas fazem mal uso, algumas jah com LER!
- Redes sociais devem ser restringidas para horários de intervalo ou extintas do ambiente de trabalho
- Pessoalmente não tive nenhuma experiência ruim com nenhuma destas ferramentas. Com relação a questão, algumas delas nunca utilizei, o que não significa que mão as conheço
- Apesar de conter muitas informações que podem acelerar o trabalho sites como wikipedia e sites pessoais contem dados falsos, já encontrei dados errados que causaram atrazos no meu trabalho. Não utilizo muito mas já consegui realizar bons trabalhos cooperativos utilizando o google docs and spreadsheets.
- O único comentário é que ao utilizar serviços de edição on-line (google spreadsheet, google doc), sempre estaremos sujeitos a ter um arquivo extraviado ou divulgado publicamente sem a nossa autorização, o que não é uma boa idéia.
- Não utilizo mais IM (MSN). Acho que as conversas com amigos atrapalham um pouco o andamento das atividades de trabalho.
- Wikis: demora um pouco para se acostumar com a sintaxe de edição, principalmente criação de links Blogs: ainda não fiz o meu, mas quero fazer! RSS: torna-se cada vez mais necessário, pois há muita fonte de informação !

Seus comentários sobre uso do e-mail e outras ferramentas de tecnologia no trabalho:

- O e-mail é uma ferramenta que pode gerar muito "lixo" simplesmente pelo fato de algo não ser bem compreendido, o que gera uma avalanche de emails em respostas. Isso aconteceu comigo em uma lista de discussão no qual existem mais de 2.500 pessoas e quase todas resolveram responder "o mal entendido"
- Quanto ao email, é imprescindível para nossa atividade, especialmente porque os consultores da ACOM são externos.
- Hoje em dia o e-mail é uma ferramenta fundamental para nosso trabalho, pois agiliza a troca de informações e diminui gastos com papel. A Intranet melhorou muito na CERTI nos últimos anos, mas acho que os colaboradores ainda não estão tendo o hábito de usa-la, talvez uma divulgação maior seria importante p/ que a Intranet se torne uma importante ferramenta de comunicação interna.
- -Sinto falta de ferramentas mais corporativas ou unificadas. O outlook por exemplo apresenta diversas possibilidades desde convocação de reuniões como acompanhamento de atividades mas não são utilizados
- O MSN não é a ferramenta ideal para todos os colaboradores, pois, devido à sua altíssima difusão, ele é uma constante fonte de desatenção de quem usa computadores. Eu estou usando, porém acredito que poderíamos adotar o uso do Skype, por exemplo, como padrão. É mais para o ambiente profissional. Bom, claro que isso pode provocar uma revolução...
- e-mail é para ser utilizado como forma de correspondência externa e para assuntos do tipo reuniões, notícias tem q ser usado blog, wiks, etc.
- Acho FUNDAMENTAL utilizar ferramentas colaborativas na CERTI, especificamente as ferramentas: Google Calendar. Google Documents and Spreadsheets. Goolge Talk. GMAIL. Hoje a Google está disponibilizando de graça para empresas sem fins lucrativos "Google para seu domínio". Eles fornecem contas de email, calendar, documents, etc. todo de graça e ainda o email fica com o domínio da CERTI: email@certi.org.br. Eu acho que esse seria um bom momento para migrar o sistema de email da CERTI para esse sistema.
- São ferramentas necessárias para agilizar a troca de informações e para realizar o trabalho em equipe que existe na Certi, onde nem sempre é possível estar no mesmo local com todos os envolvidos.
- Com relação ao e-mail, estou bastante satisfeito. Já com relação aos arquivos depositados na rede, estes são muito difíceis de serem localizados (principalmente os realizados em projetos anteriores). Inclusive já localizei na internet materiais produzidos pela CERTI e publicado por nossos clientes com uma maior facilidade que em nossa própria rede.
- As ferramentas são essenciais para a agilidade de comunicação entre o pessoal da empresa.
- - Email: imprescindível - Wiki: demora a se acostumar, mas é útil - Blogue: difícil de a equipe criar hábito de contribuição. Acaba virando depósito de notícias 'copia e cola' e se ficar aberto ao público externo, acaba passando má impressão. - Ferramentas de edição colaborativa (writeboard, google docs): muito útil para documentos não-formatados e até determinado tamanho. Para documentos com formatação específica e muito extensos, não funciona direito. - Calendário colaborativo: ainda não vejo necessidade.
- Imprescindíveis
- Considero muito importante principalmente pelos tipos de projetos realizados pela CERTI
- Precisam ser fáceis de serem aprendidas, pois no corre corre do dia a dia não existe tempo para aprender a utilizar ferramentas complexas
- Uso email apenas para comunicação interna. Listas de discussão e emails pessoais não confiro no trabalho. Uso apenas RSSs para fins de informação. Não acesso sites de notícias nem recebo emails. No trabalho, confiro apenas os relacionados a tecnologia.

ANEXO III – ARTIGO

Intranet na CERTI: Um modelo baseado em recursos de Enterprise 2.0

Ezequiel Conte

Departamento de Informática e Estatística
Universidade Federal de Santa Catarina (UFSC) – Florianópolis, SC – Brasil

ezequiel@inf.ufsc.br

***Abstract.** The Intranets derived from the Internet. Due to the arise of social platforms in the Internet it is natural that these platforms were being used in the intranets. Wikis, blogs and web feeds will be studied in this paper, aiming to the creation of the model and implementation of CERTI Foundation's intranet. The goal is to turn the systems into a platform for collaboration and communication, giving to the users the ability of finding information and sharing their knowledge in an efficient and democratic way.*

***Resumo.** As intranets surgiram a partir da Internet. Com o atual surgimento de plataformas sociais na Internet é natural que essas também sejam aplicadas às intranets. Wikis, blogs e web feeds, são alguns dos elementos que serão analisados neste trabalho, visando à concepção de um modelo e implementação da intranet da Fundação CERTI. O objetivo é fazer com que os sistemas sejam uma plataforma de colaboração e comunicação, fornecendo aos usuários a possibilidade de não apenas buscar informações, mas também compartilhar seus conhecimentos com os colegas de trabalho de uma maneira eficiente e democrática.*

1. Introdução

Para enfrentar este mundo cada vez mais dinâmico, é vital para as empresas o emprego de ferramentas que estimulem a comunicação interna e auxiliem na gestão do conhecimento. Fontes de informação cada vez mais amplas e mudanças ocorrendo a passos acelerados são alguns dos obstáculos que uma empresa deve transpor para manter-se sólida no mercado.

Diante deste cenário, uma solução amplamente utilizada é a adoção de intranets. Uma intranet fornece um mecanismo seguro, simples e facilmente acessível para a comunicação interna da organização.

As intranets surgiram a partir da utilização das ferramentas da Internet no contexto organizacional. Com a evolução da Internet, novas ferramentas foram surgindo, como é o caso dos wikis, blogs, redes sociais e outros.

Como era possível prever, todas essas novas ferramentas surgidas na Internet também estão aos poucos sendo implantadas nas intranets. Isso torna a intranet muito mais do que um mecanismo de comunicação: transforma-a em uma plataforma de colaboração, auxiliando nos processos criativos e na gestão do conhecimento da organização.

Este novo conceito de intranet é cunhado pelo termo Enterprise 2.0. Ou, como McAfee (2006) define, Enterprise 2.0 é o uso das emergentes plataformas de software social dentro das companhias, ou entre companhias e seus parceiros ou clientes.

2. Fundamentação Teórica

Neste capítulo serão descritos alguns estudos que formam o embasamento teórico para o trabalho.

2.1. Intranet

Conforme Hills (1997), uma intranet é uma rede que usa as mesmas tecnologias empregadas na Internet, principalmente a Web, porém usada num contexto interno da empresa. As intranets têm finalidades específicas e são desenvolvidas e adaptadas à realidade da empresa e às necessidades de seus usuários.

Uma intranet é um ambiente para troca de informações muito amplo, permitindo a interação da empresa com filiais, clientes e fornecedores, a colaboração entre profissionais e a divulgação de idéias e conhecimentos.

Como na Internet, historicamente o conteúdo Web das intranets era um sistema 1-n [BUFFA 2006], onde uma pessoa ou um grupo pequeno era responsável pela publicação, dando aos outros usuários apenas a opção de leitura das informações disponibilizadas.

Mas esta situação vem mudando gradativamente, à medida que sistemas colaborativos, como wikis, são implantados nas intranets. Isso dá uma liberdade de criação maior para todos os usuários, facilitando a troca de conhecimento entre os colegas de trabalho. Esta mudança está ligada aos conceitos de Enterprise 2.0.

2.2. Web 2.0 e Enterprise 2.0

Como se pode notar, a Web está atualmente em processo de evolução, onde a colaboração entre os usuários é o fator chave. Esta mudança está sendo cunhada pelo termo Web 2.0.

Segundo Tim O'Reilly, *"Web 2.0 é a mudança para uma Internet como plataforma, e um entendimento das regras para obter sucesso nesta nova plataforma. Entre outras, a regra mais importante é desenvolver aplicativos que aproveitem os efeitos de rede pra se tornarem melhores quanto mais são usados pelas pessoas, aproveitando a inteligência coletiva"*.

Alguns exemplos que ilustram bem este novo momento são a Wikipedia, uma enciclopédia virtual, totalmente escrita pelos seus usuários e o crescente uso de Weblogs (blogs).

Porém, as melhorias dessa nova fase podem ser percebidas e a área corporativa, aproveitando-se dos benefícios dessas novas tendências, começou a utilização destes mesmos sistemas sociais em suas redes internas, surgindo o conceito de Enterprise 2.0:

"Enterprise 2.0 é o uso das emergentes plataformas de software social dentro das companhias, ou entre companhias e seus parceiros ou clientes". [Andrew McAfee 2006]

A base desse paradigma é a utilização de softwares sociais, que serão analisados a seguir.

2.3. Softwares Sociais

Software social se refere à vários tipos de aplicações conectadas, que permitem aos indivíduos se comunicarem com os demais e acompanhar discussões através da Web enquanto elas ocorrem [GREEN 2005]. Outros estudiosos restringem o termo à gêneros de softwares mais recentes como blogs e wikis [WIKIPEDIA 2007], caracterizando software social pelas suas novas dinâmicas de uso.

Nesse caso, a principal característica é que o conteúdo emerge a partir dos interesses dos participantes. São os próprios usuários dos sistemas que definem o que é importante e o sistema evolui a partir das características dos seus usuários.

A seguir serão apresentadas as principais ferramentas de Software Social.

2.3.1. Wikis

Diferentemente do sistema de edição centralizada de uma página Web, um wiki é um site da Web onde os usuários têm poder para criar, editar textos e *hyperlinks* entre páginas, através de uma marcação simples, utilizando o próprio navegador Web.

Essa característica traz vários benefícios, sendo um deles o auxílio na elaboração de conteúdo coletivamente, criando um ponto de acesso centralizado onde todos editam, sem a necessidade de troca de mensagens entre os colaboradores para repassar suas contribuições aos demais.

2.3.2. Blogs

Tecnicamente, blog ou weblog é um tipo de website, bastante simples de se atualizar, bastando preencher as informações que serão adicionadas em um formulário [GILROY 2006]. Geralmente, as informações entradas são exibidas em ordem cronológica e permitem aos visitantes deixarem comentários. Um blog pode ser de autoria coletiva ou mantido por um único autor.

Os blogs vêm sendo usados para as mais diversas finalidades. Algumas pessoas usam blogs como um “diário virtual”, contando seu dia-a-dia, suas emoções e opiniões. Outros têm um enfoque mais profissional, tratando apenas de assuntos ligados ao seu trabalho; tendo o blog como um “cartão de visitas”.

Nas organizações, os blogs estão cada vez mais sendo usados, tanto na divulgação externa quanto interna e também como ferramenta de comunicação nos projetos.

2.3.3. Social Bookmarking

Social Bookmarking é um serviço Web para compartilhamento de *bookmarks* (endereços eletrônicos, também conhecidos como favoritos) [WIKIPÉDIA 2007]. Nele, os usuários podem armazenar, classificar, compartilhar e procurar endereços eletrônicos.

Ao cadastrar um endereço no sistema, o usuário associa-o à uma lista de palavras-chave. Essas informações, classificadas pelas palavras-chave, podem então ser acessadas por outros usuários do sistema. Essa forma de classificação é conhecida como *folksonomia*.

2.3.4. Web Feeds

Web Feed (do inglês alimentar), ou fonte, é um mecanismo de distribuição de conteúdo, que contém uma lista de atualizações de um determinado site. RSS e Atom são os principais padrões utilizados para esta função.

Essas fontes são utilizadas para que um usuário seja notificado sobre a atualização de seus sites prediletos. Para isso, o usuário cadastra as fontes em um software agregador. O agregador se encarrega de automatizar o processo de verificação de atualizações e as exibe para o usuário.

3. Concepção do Modelo

Para formação das idéias iniciais do desenvolvimento da intranet, foram feitas algumas reuniões com membros da Fundação, que atuam em áreas distintas. Isso rendeu a idéia básica do sistema, bem como algumas idéias para evoluções, como a realização de uma pesquisa com os colaboradores para avaliar suas necessidades e o estudo de sistemas semelhantes.

O foco principal da intranet era possibilitar uma estrutura emergente, dando aos usuários a possibilidade de criar e publicar informações. Assim a intranet estaria se beneficiando da construção através de um grande grupo de colaboradores ao invés de um grupo pequeno.

3.1. Modelo Proposto

O modelo proposto foi inspirado pelo modelo proposto no artigo “*The Wiki and the Blog: Toward a Complex Adaptive Intelligence Community*” [ANDRUS 2005]. O artigo baseia-se em seu estudo sobre Sistemas Adaptativos Complexos para propor um novo ambiente para o compartilhamento de informações entre os agentes da Agencia Central de Inteligência dos Estados Unidos - CIA. O sistema é composto pela junção de cinco componentes: repositório de arquivos, wiki, blog, sistema de busca e mecanismo de notificação.

Este foi o modelo que mais se aproximou do almejado para a intranet e por isso foi escolhido como base, sendo que o modelo proposto neste trabalho adiciona alguns elementos à pilha, focando atender as necessidades apontadas pela pesquisa e pelos acompanhamentos realizados.

Os sistemas que compõem o modelo da intranet são os seguintes: serviço de diretório, repositório de arquivos, wikis, blogs, sistema de busca, mecanismos de classificação, e mecanismos de notificação.



Figura 15 Modelo da Intranet

No wiki é possível centralizar o acesso aos serviços da CERTI, servindo como um portal, onde novos serviços seriam facilmente incluídos por qualquer colaborador. O wiki também permite criar áreas para cadastro de projetos. Nesse espaço, é possível adicionar informações relevantes ao projeto, como descrever tecnologias e produzir a documentação. E por fim, cada usuário do wiki possui uma página pessoal, onde define seu perfil, compartilha seus gostos e conhecimentos, criando assim o sistema de rede social.

Com os blogs é possível divulgar notícias e decisões de projetos, tendo uma área de discussão (os comentários) em que a equipe pode trocar idéias. A princípio foram criados blogs individuais, onde o colaborador pode expor seus interesses e blogs coletivos, sobre temas e projetos, onde serão realizadas discussões e decisões.

Devido à dinamicidade esperada da intranet, todos os serviços disponibilizarão várias opções de Web feeds. Para acompanhamento dos projetos, o wiki deve fornecer feeds que

possibilitem a notificação de alterações em um projeto específico, além dos blogs de projetos também notificarem suas mudanças.

4 Implementação e Resultados

A implementação do modelo da intranet consistiu na instalação dos sistemas propostos e adaptação dos mesmos às necessidades da CERTI. Além da customização de todos os ambientes, foram desenvolvidos sistemas de cadastro de unidades, projetos e pessoas.

A rede da CERTI já possuía um serviço de diretório, fornecendo um mecanismo de autenticação centralizado. Esse recurso foi integrado à todos os sistemas, para facilitar o acesso dos usuários e evitar a necessidade do usuário ter que se cadastrar em cada sistema.

Outro recurso que também já estava implantado na Fundação CERTI era o servidor de arquivos. O sistema consiste em um servidor com uma estrutura de diretórios compartilhada acessíveis aos colaboradores, com controle de acesso.

O primeiro sistema a ser implantado foi o wiki, uma vez que seria o sistema central e, além disso, o que proporcionaria mais benefícios à equipe.

Após a análise de alguns sistemas wikis, o TWiki foi selecionado como sistema wiki por melhor atender aos requisitos, como controle de acesso bastante flexível, suporte à plugins, comunidade ativa, hierarquia (webs e subwebs). Outra característica de destaque do TWiki foi a facilidade de criar aplicações no próprio ambiente do TWiki, como formulários, cadastros e pesquisas.

Depois de instalado o TWiki, foi desenvolvido um tema customizado, visando facilitar a utilização por usuários leigos e também adicionar a identidade visual da CERTI. Em seguida, foram desenvolvidos aplicações TWiki para cadastro e pesquisa de projetos, criados os espaços das unidades e a listagem dos colaboradores. O sistema de notificação através de web feeds RSS do TWiki se mostrou ineficiente e foi melhorado para atender às necessidades.

A modificação mais trabalhosa foi a implantação e melhoria do Editor Simples –*What You See Is What You Get* (WYSIWYG) do TWiki. Este editor apresentava vários problemas de usabilidade, falta de ferramentas básicas e inconsistências na transformação para a linguagem de marcação do TWiki. As melhorias realizadas foram revertidas para a comunidade TWiki no modelo de software livre.

Para o sistema de blogs, o software selecionado foi o WordPress MU, dado sua característica de gerenciar múltiplos blogs, possuir notificações através de web feeds, ser software livre e ser o sistema mais popular, contando com centenas de plugins e temas.

A primeira etapa da implantação do WordPress MU foi a instalação do mesmo e integração com o sistema de diretórios. Em seguida, foi desenvolvido um tema padrão para novos blogs com a identidade visual da Fundação CERTI. Além disso, a usabilidade foi melhorada através da adição de novos itens de menu para facilitar o acesso aos web feeds e às funções básicas, como a postagem de novo artigo. Outra contribuição foi o desenvolvimento do plugin para categorização dos blogs, que também foi revertido para a comunidade do WordPress MU.

Durante o desenvolvimento já pode ser percebido alguns benefícios da participação das comunidades de software livre. Sugestões levantadas por pessoas externas foram sendo implantadas nos plugins desenvolvidos, melhorando o sistema da CERTI.

As métricas do sistema foram monitoradas durante todo o desenvolvimento. Assim, foi possível coletar estatísticas de acesso, número de usuários cadastrados, entre outros dados, que serão analisados a seguir.

Um lado negativo foi a dificuldade de alguns usuários adotarem os sistemas, principalmente, no caso do wiki, pela necessidade de aprender uma sintaxe para edição ou então desmotivados pelos problemas iniciais do editor simples, alguns corrigidos atualmente.

4.1 Estatísticas de Uso do Sistema

Para demonstrar a utilização do sistema serão analisados os dados referentes ao número total de tópicos no wiki e o número de tópicos acessados mensalmente.

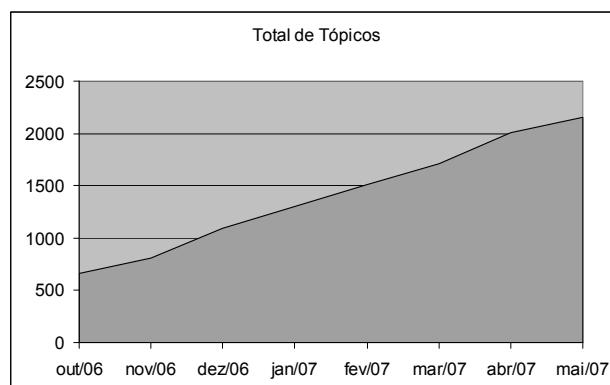


Figura 16 - Estatísticas: Total de Tópicos

O número total de tópicos na intranet representa a criação, a geração de novos conhecimentos. Pode-se perceber que possui um crescimento praticamente linear, principalmente devido a característica do wiki ser usado na documentação dos projetos. A média foi de 213,7 tópicos criados mensalmente.

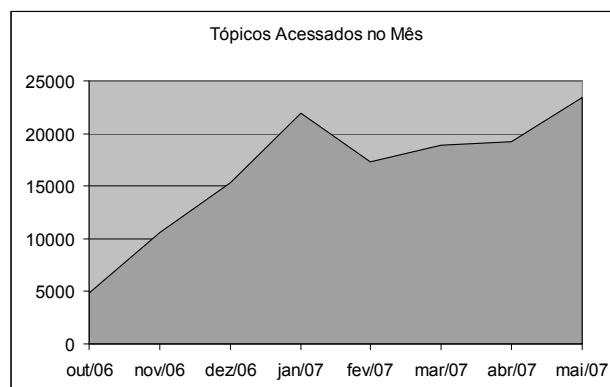


Figura 17 - Estatísticas: Tópicos Acessados no Mês

O número de tópicos acessados no mês representa a troca de conhecimento entre os membros dos projetos: alguns colaboradores geram conteúdo, outros consomem. Podemos perceber um pico em janeiro, ocorrido pelas divulgações da lista de ramais e da pesquisa. O gráfico apresenta um crescimento pois houve medições durante o decorrer do projeto, quando não havia muita divulgação da intranet. Provavelmente chegará a um valor relativamente constante, como apresentado nos últimos quatro meses. O número médio de tópicos visitados mensalmente ficou em 16457,1.

5 Conclusões e Trabalhos Futuros

O objetivo deste trabalho foi propor um modelo para a intranet da Fundação CERTI com base nos inovadores recursos de Enterprise 2.0. A meta era criar um sistema que atendesse às necessidades da CERTI, fornecendo um ambiente incentivador para as atividades de criação, colaboração e comunicação entre os colaboradores.

Para atender à proposta, foi necessário fazer um estudo sobre Enterprise 2.0, Web 2.0 e softwares sociais, além de intranets. Estes são temas que abrangem áreas variadas como a engenharia de software, usabilidade, gestão do conhecimento e sistemas colaborativos. Além disso, foi de grande valor a experiência de participar em projetos de software livre e conhecer suas características.

Na etapa de concepção do modelo, foi realizada uma pesquisa para analisar o comportamento dos colaboradores em relação às atividades diárias no computador e o grau de experiência com os softwares que se desejava implantar. Isso permitiu desenvolver um modelo voltado às necessidades dos colaboradores.

A implementação feita de forma iterativa com lançamentos frequentes foi de grande proveito, pois permitiu que o sistema fosse desenvolvido baseado nas necessidades reais apontadas pelos próprios usuários.

Pelos resultados, pode-se perceber que o sistema está de maneira geral sendo bem recebido, com bom nível de utilização, apesar do pouco tempo de uso.

Como sugestão de trabalhos futuros, poderia ser implantado o sistema em outras empresas, verificando se o modelo proposto é adequado à outras realidades e propondo melhorias ao mesmo.

Outra sugestão seria dar continuidade ao projeto, agregando novas funcionalidades ao sistema e analisando os benefícios que a intranet baseada em recursos de Enterprise 2.0 pode trazer.

Referências Bibliográficas

- Andrus, D. C. “The Wiki and the Blog: Toward a Complex Adaptive Intelligence Community”. *Studies in Intelligence*, Washington, v. 49, n. 3, 2005.
- Buffa M. “Intranet Wikis”. *IntraWeb Workshop WWW2006*. Edimburgo, 2006.
- Dorsett, L.; Fontaine, M. A.; Driscoll, T. “Redefining Manager Interaction”. *Knowledge Management Review*, Melcrum Publishing, 2002.
- Green, D. T.; Pearson, J. M.; “Social Software and Cyber Networks: Ties That Bind or Weak Associations within the Political Organization?”. *38th Hawaii International Conference on System Sciences*, 2005.
- Hills, M. “Intranet como Groupware”. São Paulo: Berkeley. 1997. 252 p.
- Finck, N.; Hodder, M.; Stone, B. “Enhancing Internal Communications with Blogs, Wikis, and More”. Seattle, 2005 Disponível em: <<http://www.nickfinck.com/presentations/bbs2005/01.html>>. Acesso em: 23 jan. 2007
- Gilroy, K.; Ives, B. “Preparing for Intranet 2.0: how to integrate new communication technology into your intranet”. In: SUMMER, J. *Transforming Your Intranet*. Melcrum Publishing, 2006. p. 65-81.

- Hinchcliffe, D. "Enterprise Web 2.0". Alexandria, Virginia. Disponível em: <<http://blogs.zdnet.com/Hinchcliffe/>>. Acesso em: 15 jan. 2007.
- Majchrzak, A.; Wagner, C.; Yates, D. "Corporate wiki users: results of a survey". International Symposium On Wikis, Odense, p. 99-104, 2006.
- McAfee, A. P. Andrew McAfee's Blog. Cambridge, 2006. Disponível em: <<http://blog.hbs.edu/faculty/amcafee/index.php>>. Acesso em: 20 jan. 2007.
- McAfee, A. P. "Enterprise 2.0: The Dawn of Emergent Collaboration". MIT Sloan Management Review, Cambridge, v. 47, n. 3, p.21-28, 2006.
- Owen, M. et al. "Social software and learning: An Opening Education report from Futurelab". Futurelab, 2007. Disponível em: <http://www.futurelab.org.uk/research/opening_education/social_software_01.htm>. Acesso em: 14 maio 2007
- Social Bookmarking. Wikipédia: a enciclopédia livre. Wikimedia, 2007. Disponível em: <http://en.wikipedia.org/wiki/Social_bookmarking>. Acesso em: 21 jan. 2007.
- Tapscott, D. "Winning with the Enterprise 2.0". New Paradigm, IT&CA Big Idea Series, 2006.
- Tapscott, D.; Williams, A. D. "Wikinomics". Washington : Portfolio. 2006. 320 p.
- Twikisite. TWiki.org, 2006. Disponível em: <<http://twiki.org/cgi-bin/view/TWiki/TWikiSite>>. Acesso em: 21 jan. 2007.
- Web 2.0. Wikipédia: a enciclopédia livre. Wikimedia, 2007. Disponível em: <http://pt.wikipedia.org/wiki/Web_2.0>. Acesso em: 21 jan. 2007.
- White, M. "Creating an Effective Intranet". Horsham, 2003. Disponível em: <<http://www.intranetfocus.com/information/effectiveintranets.pdf>>. Acesso em: 17 jan. 2007.

ANEXO IV - CÓDIGO FONTE

=== WordPressMU/mu-plugins/wpmu-topposts/wpmu-topposts.php ===

```
<?php
/*****
*****
```

Plugin Name: zappo_wpmu_topposts

Plugin URI: <http://heftagaub.wordpress.com/2007/03/11/wpmu-top-posts-plugin/>

Description: Simple WordPressMU plugin for tracking top posts and top blogs. Inspired by WordPress.com's Top Posts feature, developed in the style of wp-shortstats.

Version: 0.42.3

Author: Brad Hefta-Gaub

Author URI: <http://heftagaub.wordpress.com/>

Copyright: (c) 2007 Brad Hefta-Gaub

License:

THE WORK IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

Attribution-NonCommercial-ShareAlike 3.0

Unported

You are free:

- * to Share it to copy, distribute and transmit the work
- * to Remix it to adapt the work

Under the following conditions:

* Attribution. You must attribute the work in the manner specified by the author or licensor (but not in any way that suggests that they endorse you or your use of the work).

* Noncommercial. You may not use this work for commercial purposes.

* Share Alike. If you alter, transform, or build upon this work, you may distribute the resulting work only under the same or similar license to this one.

* For any reuse or distribution, you must make clear to others the license terms of this work. The best way to do this is with a link to this web page.

* Any of the above conditions can be waived if you get permission from the copyright holder.

* Nothing in this license impairs or restricts the author's moral rights.

The complete license can be found here:
<http://creativecommons.org/licenses/by-nc-sa/3.0/legalcode>

My reasoning behind this license choice is as follows:

* I intend for these posts to be informative and educational. They are not intended to be fully functioning implementations. They are intended to represent reusable ideas. Therefore, you shouldn't need more than this license to learn from what I'm presenting here.

* If you are planning on using this as is, then I would like you to give me the appropriate level of credit. And if you see me on the street, please introduce yourself and say "Thanks Man!"

* If you are want to use this code as is for commercial purposes, then, well, contact me. I make my living as a technologist, and if you're making money off my hard work, then I want some control over that. That being said, I don't mind teaching you what I've learned, and so please feel free to read my code, learn from it, and go about your business as you see fit. But you can't use this code for commercial purposes without contacting me first.

If you like this code, and you run into me in person, then you should say, "Hey Zappo, that was cool code, I learned something from it, it helped me out. Thanks!", and offer to buy me a beer or a coffee depending on what we're both in the mood for at the time.

Please may not remove this copyright and license message.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

History:

0.42.3 - Finally actually fixed the setup behavior. In the past I was trying to use the built in 'maybe_create_table' function, but sometimes this function is available and sometimes it isn't so people were having trouble one way or the other. Now I've made a local copy of the function and call it in setup everytime. This way the tables are guaranteed to be created.

0.42.2 - Released under Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported License.

0.42.1 - Small bug fix to 'maybe_create_table' behavior. Namely, we used to try to load it if this function wasn't available, and now we simply check that the function is available.

Added blog_id to the posts returned by get_top_posts. This can be useful in forming correct permalinks to your blog posts.

Added function zap_setup_post_globals() which in some cases will setup the globals for other wordpressmu templates to work, but this doesn't always reliably work.

0.42 - Original Version

```

*****
*****/

/*****
* Class:   zappo_wpmu_topposts
* Purpose: This guy does all the work.
*
*****/
class zappo_wpmu_topposts
{
    /*****
    * Member Variables- Used throughout the class
    *****/
    var $table_hits;
    var $timezone;
    var $current_time;

    /*****
    * Method: Constructor - zappo_wpmu_topposts()
    *
    *****/
    function zappo_wpmu_topposts()
    {
        global $table_prefix,$wpdb;

        /*
        * our hit tracking table name, this is kinda sleazy, but
        * since we want our hits to be tracked across all of the blogs
        (site wide)

```

```

        * we don't want to use $table_prefix, because wpmu rewrites
that prefix
        * to include the blog_id. So we will use the "blogs" table
name as our prefix.
        * This is the same for all blogs.
        */
        $this->table_hits = $wpdb->blogs . "_zap_hits";

    /*
    * Time related details
    */
    $this->timezone = get_settings('gmt_offset') * 3600;
    $this->current_time = strtotime(gmtime('Y-m-d g:i:s a'))+$this-
>timezone;

    }

    /*****
    * Method: maybe_create_table()
    * Purpose: This function comes from mu's admin functions, but
    *           I couldn't find a way to reliably ensure that it was
    *           available, so I've included it here. What it basically
    *           does is check for the existence of a table, and if the
    *           table doesn't exist then it creates it using the create
    *           query string passed in.
    *****/
function maybe_create_table($table_name, $create_ddl)
{
    //echo "<!-- wpmu-topposts.php maybe_create_table() -->";
    global $wpdb;
    foreach ($wpdb->get_col("SHOW TABLES",0) as $table )
    {
        if ($table == $table_name)
        {
            //echo "<!-- maybe_create_table() found table-->";
            return true;
        }
    }
    //didn't find it try to create it.
    $q = $wpdb->query($create_ddl);
    // we cannot directly tell that whether this succeeded!
    foreach ($wpdb->get_col("SHOW TABLES",0) as $table )
    {
        if ($table == $table_name)
        {
            //echo "<!-- maybe_create_table() created table-->";
            return true;
        }
    }
    //echo "<!-- maybe_create_table() no table?!? -->";
    return false;
}

    /*****
    * Method: setup()
    * Purpose: creates our tracking table if needed.
    *****/
function setup()
{
    //echo "<!-- wpmu-topposts.php setup() -->";

```

```

        $table_hits_query = "CREATE TABLE $this->table_hits (
                                id int(11) unsigned NOT NULL
auto_increment,
                                blog_id BIGINT(20) unsigned NOT
NULL,
                                post_id BIGINT(20) unsigned,
                                hit_time int(10) unsigned NOT NULL
default '0',
                                UNIQUE KEY id (id)
                                )";
        $this->maybe_create_table($this->table_hits,
$table_hits_query);
    }

    /*****
    * Method: recordhit()
    * Purpose: records a hit to a page/post
    *****/
    function recordhit()
    {
        global $wpdb;
        global $blog_id;
        $post_id = 0; /* in case of non-single post hits, we store a 0
for the post_id */
        $hit_time = $this->current_time; /* What time is it? */

        /*
        * Things we don't track: admin hits, 404's, previews, and login.
        */
        if (
            is_admin() ||
            is_404() ||
            is_preview() ||
            strpos($_SERVER['PHP_SELF'], 'wp-login.php')
        )
            return;

        /*
        * If this is a "single" post page, then we record it as a post
hit
        * otherwise we just record it as a blog hit.
        */
        if(is_single())
        {
            $post_id = get_the_ID();
        }

        /*
        * We are inserting these hits even in non-single post cases,
because
        * we may eventually implement support for "top blogs" based on
        * hits to the blog. Right now, we only support accessor functions
for
        * top_posts.
        */
        $query = "INSERT INTO $this->table_hits
(blog_id,post_id,hit_time)
                VALUES ('$blog_id','$post_id',$hit_time)";

        $wpdb->query($query);
    }

```

```

}

/*****
*****
*
* Method : get_top_posts()
* Purpose :
* Similar to WP's standard get_posts() function, it is used to
return a list
* of posts, that qualify as "top posts".
* Parameters:
* numberposts
* (integer) Number of posts to return. Defaults to 5.
* offset
* (integer) Offset from the "top most" post. Defaults to 0.
* orderby
* ("string") Sort Posts by one of various values, including:
* 'post_hits' - Sort by number of hits to the post
(Default).
* 'post_id' - Sort by numeric Post ID.
* 'blog_id' - Sort by numeric Blog ID.
*
* order
* (string) Sort order for options. Valid values:
* 'DESC' - Sort from highest to lowest (Default).
* 'ASC' - Sort from lowest to highest.
*
* Note: Unlike get_posts(), the default order is DESC so
that the "top_posts"
* has a "top" behavior where the post with the "most
hits" is listed first.
*
* include/exclude
* same behavior as get_posts()
*/
function get_top_posts($args = '')
{
    global $wpdb, $wpmuBaseTablePrefix;

    /*
    * Same behavior as get_posts(), this function accepts an array of
args or
    * a URL encoded list of args.
    */

    if ( is_array($args) )
        $r = &$args;
    else
        parse_str($args, $r);

    /*
    * default vaules array here.
    *
    * Note: This is different from get_posts() defaults as noted in
the comment
    * of this function.
    */
    $defaults = array('numberposts' => 5, 'offset' => 0, 'category'
=> '',

```

```

        'orderby' => 'post_hits', 'order' => 'DESC', 'include' =>
'', 'exclude' => '', 'meta_key' => '', 'meta_value' => '');
        $r = array_merge($defaults, $r);
        extract($r);

        /*
        * Ok, like get_posts() we will attempt to support include='' and
exclude=''
        */

        $inclusions = '';
        if ( !empty($include) ) {
            $offset = 0; //ignore offset, category, exclude, meta_key,
and meta_value params if using include
            $category = '';
            $exclude = '';
            $meta_key = '';
            $meta_value = '';
            $incposts = preg_split('/[\\s,]+/', $include);
            $numberposts = count($incposts); // only the number of
posts included
            if ( count($incposts) ) {
                foreach ( $incposts as $incpost ) {
                    if (empty($inclusions))
                        $inclusions = ' AND ( ID = ' .
intval($incpost) . ' ' ;
                    else
                        $inclusions .= ' OR ID = ' .
intval($incpost) . ' ' ;
                }
            }
            if (!empty($inclusions))
                $inclusions .= ')';
        }

        /*
        * Like get_posts() we support exclude=''
        */
        $exclusions = '';
        if ( !empty($exclude) ) {
            $exposts = preg_split('/[\\s,]+/', $exclude);
            if ( count($exposts) ) {
                foreach ( $exposts as $expost ) {
                    if (empty($exclusions))
                        $exclusions = ' AND ( ID <> ' .
intval($expost) . ' ' ;
                    else
                        $exclusions .= ' AND ID <> ' .
intval($expost) . ' ' ;
                }
            }
            if (!empty($exclusions))
                $exclusions .= ')';
        }

        /*
        * Here's where we start to significantly diverge from get_posts(),
        * namely we are reading from a very different table, and so we
        * have to construct a different query.
        */
        WHERE $exclusions $inclusions

```

```

*/
$query = "SELECT blog_id, post_id, COUNT(post_id) AS 'post_hits'
        FROM $this->table_hits
        WHERE post_id <> '0' $exclusions $inclusions
        GROUP BY blog_id, post_id
        ORDER BY $orderby $order LIMIT $offset , $numberposts
";

/*****

From: get_posts() left here as a reference for later support of
more
        features.

$query = "SELECT DISTINCT * FROM $wpdb->posts " ;
$query .= ( empty( $category ) ? "" : ", $wpdb->post2cat " ) ;
$query .= ( empty( $meta_key ) ? "" : ", $wpdb->postmeta " ) ;
$query .= " WHERE (post_type = 'post' AND post_status =
'publish') $exclusions $inclusions " ;
$query .= ( empty( $category ) ? "" : "AND ($wpdb->posts.ID =
$wpdb->post2cat.post_id AND $wpdb->post2cat.category_id = " . $category. " )
" ) ;
$query .= ( empty( $meta_key ) | empty($meta_value) ? "" : " AND
($wpdb->posts.ID = $wpdb->postmeta.post_id AND $wpdb->postmeta.meta_key =
'$meta_key' AND $wpdb->postmeta.meta_value = '$meta_value' )" ) ;
$query .= " GROUP BY $wpdb->posts.ID ORDER BY " . $orderby . " "
. $order . " LIMIT " . $offset . ', ' . $numberposts ;

*****/

/*
* Note: this doesn't actually contain a nice "posts" style array.
* It only contains an array of blog_id, post_ids.Next we need to
* get the actual table details from the prefix_{blog_id}_posts
tables.
*/
$top_posts_mappings = $wpdb->get_results($query);

/*
* Start with a fresh array, fill it in below.
*/
$postes = array();

foreach($top_posts_mappings as $post_mapping)
{
    $table_blog_posts = $wpmuBaseTablePrefix.$post_mapping->
blog_id."_posts";
    $query = "SELECT *, '$post_mapping->post_hits' AS
'post_hits', '$post_mapping->blog_id' AS 'blog_id' FROM $table_blog_posts
WHERE ID = $post_mapping->post_id";
    $results = $wpdb->get_results($query);

    if ( !empty($results) )
    {
        /* sleazy - should really be certain this only returned 1
row... */
        $single_post = $results[0];
        $posts[] = $single_post;
    }
}

```



```

    }

    /*
    * This is a call to a standard WP caching function. It may be
counter productive to call
    * this for our little function, but since get_posts() does it, it
seems like a reasonable
    * thing to do.
    */
    update_post_caches($posts);

    return $posts;
}

/*
* format some HTML for a top posts list.
*/
function get_top_posts_html($args = '')
{
    global $post; /* needs to be global if you want get_permalink() and
get_the_title() to work. */

    $top_posts = $this->get_top_posts($args);

    $html="<ul>";

    foreach($top_posts as $post)
    {
        $html.="<li><a
href='".get_permalink()."'">".get_the_title()."[".$post-
>post_hits."]</a></li>";
    }
    $html.="</ul>";

    return $html;
}

function get_top_blogs($limit = 10, $days_back = 90)
{
    global $wpdb, $wpdbBaseTablePrefix;

    $query = " SELECT blog_id, COUNT(blog_id) AS 'blog_hits'
                FROM $this->table_hits
                WHERE blog_id <> '1'
                AND TO_DAYS(NOW()) -
TO_DAYS(FROM_UNIXTIME(hit_time)) <= $days_back
                GROUP BY blog_id
                ORDER BY blog_hits DESC LIMIT $limit;";

    $top_blogs_mappings = $wpdb->get_results($query);

    return $top_blogs_mappings;
}

function get_top_blogs_html($limit = 10, $days_back = 90)
{
    foreach($this->get_top_blogs($limit, $days_back) as $blog_mapping)
    {

```

```

        $details = get_blog_details($blog_mapping->blog_id);
        echo "<li><a href='http://'" . $details->domain .
$details->path . "'>" . $details->blogname . " (" . $blog_mapping-
>blog_hits .")</a></li>";
    }
}

};

// This will be our main "tracking object" we will keep things nice and
// object oriented by doing all of our work inside this object.
$zap_wpmutp = new zappo_wpmu_topposts();

// We should make this smarter so that we don't waste time attempting to
create
// tables over and over again, but in the meantime, this will do.
$zap_wpmutp->setup();

// This hook will be called for every page.
add_action('shutdown', array(&$zap_wpmutp, 'recordhit'));

/*
 * Function: zap_get_top_posts()
 * Purpose :
 *     Similar to WP's standard get_posts() function, it is used to return
a list
 *     of posts, that qualify as "top posts".
 * Parameters:
 *     See zappo_wpmu_topposts::get_top_posts()
 */
function zap_get_top_posts($args = '')
{
    global $zap_wpmutp;
    return $zap_wpmutp->get_top_posts($args);
}

/*
 * Function: zap_get_top_posts_html()
 * Purpose :
 *     Returns some HTML for get_top_posts().
 * Parameters:
 *     See zappo_wpmu_topposts::get_top_posts_html()
 */
function zap_get_top_posts_html($args = '')
{
    global $zap_wpmutp;
    return $zap_wpmutp->get_top_posts_html($args);
}

/*
 * Function: zap_top_posts_html()
 * Purpose :
 *     echos some HTML for get_top_posts().
 * Parameters:
 *     See zappo_wpmu_topposts::get_top_posts()
 */
function zap_top_posts_html($args = '')
{
    global $zap_wpmutp;
    echo $zap_wpmutp->get_top_posts_html($args);
}

```

```

}

/*
 * mimics the_title() except that it shows 'post_hits' as opposed to
 'title' for a post
 *
 * Assumes that some zap_..._top_posts() function has been called first, or
 else the post_hits
 * member will not be available for $post;
 */
function zap_the_post_hits($before = '', $after = '', $echo = true)
{
    $post_hits = zap_get_the_post_hits();
    if ( strlen($post_hits) > 0 ) {
        $post_hits = apply_filters('the_post_hits', $before .
$post_hits . $after, $before, $after);
        if ( $echo )
            echo $post_hits;
        else
            return $post_hits;
    }
}

/*
 * mimics get_the_title() except that it shows 'post_hits' as opposed to
 'title' for a post
 *
 * Assumes that some zap_..._top_posts() function has been called first, or
 else the post_hits
 * member will not be available for $post;
 */
function zap_get_the_post_hits()
{
    global $post;
    $post_hits = $post->post_hits;
    return $post_hits;
}

/*
 * call this for each post object returned by zap_get_top_posts() to set
 the global variables
 * so that the_permalink() and other template functions work properly.
 *
 * Assumes that some zap_..._top_posts() function has been called first, or
 else the post_hits,
 * and blog_id members will not be available for $post;
 */
function zap_setup_post_globals($post_in)
{
    global $post,$blog_id;
    $post = $post_in;
    $blog_id = $post_in->blog_id;
}

/*
 * get top blogs
 */
function zap_top_blogs_html($limit = 10, $days_back = 90)
{

```

```

    global $zap_wpmutp;
    echo $zap_wpmutp->get_top_blogs_html($limit, $days_back);
}
?>

=== WordPressMU/mu-plugins/wpmu-blogcategories/blog_categories.php ===

<?php
/* Short and sweet */
define('WP_USE_THEMES', true);
require( 'wp-config.php' );
?>

<?php get_header(); ?>

<div id="container">

    <div id="center" class="content">

        <h1>Blogs da Categoria <?php echo
eco_get_category_name($_REQUEST['idcat']) ?></h1>
        <ul>
            <?php eco_get_blogs_from_category_id_html($_REQUEST['idcat']); ?>
        </ul>

    </div>

<div id="left" class="menu">
    <?php include (TEMPLATEPATH . '/leftbar.php'); ?>
</div>

<?php get_footer(); ?>

=== WordPressMU/mu-plugins/wpmu-blogcategories/wp-content/mu-plugins/wpmu-
blogcategories.php ===

<?php
/*****
*****

Plugin Name: eco_wpmu_blogcategories

Plugin URI: http://eco.certi.org.br/

Description: WordPressMU plugin for classifying blogs into categories.
Developed in the style of wpmu-topposts.

Version: 0.1

Author: Ezequiel Conte

Author URI: http://eco.blogs.certi.org.br

```

Copyright: (c) 2007 Fundação CERTI

```

*****
*****/

class eco_blog_categories
{
    var $table_categories;

    function eco_blog_categories()
    {
        global $table_prefix,$wpdb;

        $this->table_categories = $wpdb->blogs . "_eco_categories";
    }

    function setup()
    {
        global $wpdb, $wpmuBaseTablePrefix;

        //check if table was already created
        if($wpdb->get_var("show tables like '$this->table_categories'"
!= $this->table_categories) {

            $table_categories_query = "CREATE TABLE $this->table_categories
(
                                id int(11) unsigned NOT NULL
auto_increment,                                category_name VARCHAR(55) NOT NULL
default '',                                UNIQUE KEY id (id)
                                );";

            $results = $wpdb->query($table_categories_query);

            //insert categories into database
            $categories = array('Projeto','Centro', 'Pessoal',
'Assunto');
            foreach ( $categories as $category ) {
                $insert = "INSERT INTO $this->table_categories
(category_name) VALUES ('$category');";
                $results = $wpdb->query( $insert );
            }

            //add option blog_category to every blog
            $blog_list = $wpdb->get_results( "SELECT blog_id FROM
$wpdb->blogs;");

            foreach ( $blog_list as $blog ) {
                //TODO check if option already exists
                $insert = "INSERT INTO " . $wpmuBaseTablePrefix .
$blog->blog_id . "_options
                                (option_name,option_value) VALUES
('blog_category','1');";
                $results = $wpdb->query( $insert );
            }
        }
    }
}

```

```

}

function get_blogs_from_category_id($category_id)
{
    global $wpdb, $wpmuBaseTablePrefix;

    $blog_list = $wpdb->get_results( "SELECT blog_id FROM $wpdb->blogs;");

    $blogs_from_category = array();
    foreach ( $blog_list as $blog ) {
        $category = $wpdb->get_var("SELECT option_value FROM " .
$wpmuBaseTablePrefix . $blog->blog_id . "_options
        WHERE option_name LIKE
'blog_category'");
        if ($category == $category_id) {
            array_push($blogs_from_category, $blog);
        }
    }

    return $blogs_from_category;
}

function get_blogs_from_category($category_name)
{
    global $wpdb, $wpmuBaseTablePrefix;

    $category_id = $wpdb->get_var("SELECT id FROM $this->table_categories
        WHERE category_name LIKE
'$category_name'");

    return $this->get_blogs_from_category_id($category_id);
}

function get_blogs_from_category_id_html($category_id)
{
    foreach ( $this->get_blogs_from_category_id($category_id) as $blog )
    {
        $details = get_blog_details($blog->blog_id);
        echo "<li><a href='http://'" . $details->domain .
$details->path . "'>" . $details->blogname . "</a></li>";
    }
}

function get_blogs_from_category_html($category_name)
{
    foreach ( $this->get_blogs_from_category($category_name) as $blog )
    {
        $details = get_blog_details($blog->blog_id);
        echo "<li><a href='http://'" . $details->domain .
$details->path . "'>" . $details->blogname . "</a></li>";
    }
}

function get_category_name($category_id)
{

```

```

        global $wpdb;
        return $wpdb->get_var("SELECT category_name FROM $this-
>table_categories
                                WHERE id = '$category_id'");
    }

    function get_categories()
    {
        global $wpdb;
        return $wpdb->get_results("SELECT id, category_name FROM $this-
>table_categories;");
    }

    function get_categories_html()
    {
        foreach ( $this->get_categories() as $category ) {
            echo "<li><a href='" . get_settings('siteurl') .
"/blog_categories.php?idcat=$category->id'>" . $category->category_name .
"</a></li>";
        }
    }
};

$eco_wpmubc = new eco_blog_categories();

function eco_get_blogs_from_category_html($category_name = 'Pessoal')
{
    global $eco_wpmubc;

    //find a better way to call setup.
    $eco_wpmubc->setup();
    return $eco_wpmubc->get_blogs_from_category_html($category_name);
}

function eco_get_blogs_from_category_id_html($category_id = '1')
{
    global $eco_wpmubc;

    //find a better way to call setup.
    $eco_wpmubc->setup();
    return $eco_wpmubc->get_blogs_from_category_id_html($category_id);
}

function eco_get_category_name($category_id = '1')
{
    global $eco_wpmubc;

    //find a better way to call setup.
    $eco_wpmubc->setup();
    return $eco_wpmubc->get_category_name($category_id);
}

function eco_get_categories_html()
{
    global $eco_wpmubc;

    //find a better way to call setup.
    $eco_wpmubc->setup();
    return $eco_wpmubc->get_categories_html();
}

```

?>

```
=== WordPressMU/mu-plugins/mu-intranet-header/wp-content/mu-plugins/mu-
intranet-header.php ===
```

```
<?php
```

```
/* <WP Plugin Data>
 * Plugin Name: MU Intranet Header
 * Version: 1.0
 * Plugin URI: http://blogs.certi.org.br
 * Description: Adds a header to every page
 * Author: Ezequiel Conte
 * Author URI: http://eco.blogs.certi.org.br/
 */
```

```
function wp_intranet_header() {
    global $user_identity, $posts;

    /* Added because user_identiyt wasn't being displayed. */
    get_currentuserinfo();

    $_GET['intranet_blog'] = '1';
    #include($_SERVER['DOCUMENT_ROOT'] . "intranet_header.php");
    include("intranet-header.php");
}

```

```
function wp_intranet_header_style () {
global $user_level;

//if ( isset($user_level) ) { // Uncomment to only add style if user logged
in.
```

```
    ob_start();
```

```
    ?>
    <style type="text/css">
        /*  */</pre>
</div>
<div data-bbox="168 677 697 717" data-label="Text">
<pre>body {
    padding-top: 0px; /* this may cause problems */
}
</pre>
</div>
<div data-bbox="222 730 347 771" data-label="Text">
<pre>    /* ]]&gt; */
    &lt;/style&gt;
    &lt;?php</pre>
</div>
<div data-bbox="168 784 716 851" data-label="Text">
<pre>        $css = ob_get_contents();
        ob_end_clean();
        print($css);
    //} // Uncomment to only add style if user logged in.
}
</pre>
</div>
<div data-bbox="168 864 868 906" data-label="Text">
<pre>add_action('wp_footer', 'wp_intranet_header'); /* part of hack to make this
show at the top of each blog */
add_action('wp_head', 'wp_intranet_header_style');</pre>
</div>
```



```

?>

=== WordPressMU/mu-plugins/mu-intranet-header/intranet-header.php ===

<?php

    $imagesPath = "http://".$_SERVER['HTTP_HOST']."/wpmu/wp-
includes/images";
    $blogsPath = "http://blogs.certi.org.br";
    $wikisPath = "http://intranet.certi.org.br";
    $helpPath = "http://intranet.certi.org.br/bin/view/Main/IntranetFaq";

?>
<div id="intranet-header">

    <div id="div-help">
        <a href="<?php echo $helpPath; ?>">Ajuda</a>
    </div>
    <div id="div-logo">
        <a href="<?php echo $wikisPath; ?>"></a>
    </div>
    <div id="intranet-content">
        <div id="description">Ambientes da Intranet CERTI:</div>
        <div id="systems-buttons">
            <ul>
                <li class="<?php if ($_GET{intranet_wiki}) echo 'intranet-
active' ?>"><a href="<?php echo $wikisPath; ?>">WIKIs</a></li>
                <li class="<?php if ($_GET{intranet_blog}) echo 'intranet-
active' ?>"><a href="<?php echo $blogsPath; ?>">BLOGs</a></li>
            </ul>
        </div>
    </div>
</div>
<style type="text/css">
#intranet-header {
    position:absolute;
    top:0;
    left:0;
    width:100%;
    height:60px;
    background-color:#D9D9D9;
    background-image:url('<?php echo $imagesPath; ?>/topo_certi.png');
    background-position:left top;
    background-repeat:repeat-y;
    border:0px;
    border-bottom:5px solid #3C6bA4;
    text-align:left;
    font-family:'Lucida Grande',Verdana,Arial,Sans-Serif;
    font-size:10px;
}
#intranet-header #div-logo {
    padding-left:20px;
    display:inline;
}
#intranet-header #intranet-content {
    padding-left:200px;
    color:white;
}

```

```

}
#intranet-header #description {
    color:black;
    position:absolute;
    top:10px;
}
#intranet-header #systems-buttons ul {
    margin:0;
    padding:10px 10px 0;
    list-style:none;
}
#intranet-header #systems-buttons li {
    float:left;
    background:url("<?php echo $imagesPath; ?>/left.gif") no-repeat left top;
    margin:0;
    padding:0 0 0 9px;
}
#intranet-header #systems-buttons a {
    float:left;
    display:block;
    background:url("<?php echo $imagesPath; ?>/right.gif") no-repeat right
top;
    padding:10px 15px 4px 6px;
    text-decoration:none;
    font-weight:bold;
    color:white;
}
/* Commented Backslash Hack
   hides rule from IE5-Mac */
#intranet-header #systems-buttons a {float:none;}
/* End IE5-Mac hack */
#intranet-header #systems-buttons a:hover {
    text-decoration:underline;
    color:white;
}
#intranet-header #systems-buttons .intranet-active {
    background-image:url("<?php echo $imagesPath; ?>/left_on.gif");
}
#intranet-header #systems-buttons .intranet-active a {
    background-image:url("<?php echo $imagesPath; ?>/right_on.gif");
    padding-bottom:4px;
    color:white;
}
#intranet-header #systems-buttons {
    position:absolute;
    bottom:0px;
}
#intranet-header a, #intranet-header a:hover{
    background-color:transparent;
    text-decoration:none;
    color:white;
}
#intranet-header a:hover{
    text-decoration:underline;
    color:white;
}
#intranet-header * {
    margin:0;
    padding:0;
}
#intranet-header #div-help{

```

```

        float:right;
        margin-top:30px;
        margin-right:30px;
        font-size:1.2em;
    }
    #intranet-header #div-help a{
        color:#003777;
        text-decoration:underline;
    }
    body {
        padding-top:62px;
    }
</style>

```

```

=== TWiki/Plugins/AutoNotifyPlugin/AutoNotifyPlugin.pm ===

```

```

# Plugin for TWiki Enterprise Collaboration Platform, http://TWiki.org/
#
# Copyright (C) 2000-2003 Andrea Sterbini, a.sterbini@flashnet.it
# Copyright (C) 2001-2006 Peter Thoeny, peter@thoeny.org
# and TWiki Contributors. All Rights Reserved. TWiki Contributors
# are listed in the AUTHORS file in the root of this distribution.
# NOTE: Please extend that file, not this notice.
#
# This program is free software; you can redistribute it and/or
# modify it under the terms of the GNU General Public License
# as published by the Free Software Foundation; either version 2
# of the License, or (at your option) any later version. For
# more details read LICENSE in the root of this distribution.
#
# This program is distributed in the hope that it will be useful,
# but WITHOUT ANY WARRANTY; without even the implied warranty of
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
#
# For licensing info read LICENSE file in the TWiki root.

# change the package name and $pluginName!!!
package TWiki::Plugins::AutoNotifyPlugin;

# Always use strict to enforce variable scoping
use strict;

# $VERSION is referred to by TWiki, and is the only global variable that
# *must* exist in this package.
use vars qw( $VERSION $RELEASE $SHORTDESCRIPTION $debug $pluginName
$NO_PREFS_IN_TOPIC );

# This should always be $Rev: 12445$ so that TWiki can determine the
checked-in
# status of the plugin. It is used by the build automation tools, so
# you should leave it alone.
$VERSION = '$Rev: 12445$';

# This is a free-form string you can use to "name" your own plugin version.
# It is *not* used by the build automation tools, but is reported as part
# of the version number in PLUGINDESCRPTIONS.
$RELEASE = 'Dakar';

```

```

# Short description of this plugin
# One line description, is shown in the %TWIKIWEB%.TextFormattingRules
topic:
$SHORTDESCRIPTION = 'AutoNotify Plugin adds a just registered user to
WebNotify List';

# You must set $NO_PREFS_IN_TOPIC to 0 if you want your plugin to use
preferences
# stored in the plugin topic. This default is required for compatibility
with
# older plugins, but imposes a significant performance penalty, and
# is not recommended. Instead, use $TWiki::cfg entries set in
LocalSite.cfg, or
# if you want the users to be able to change settings, then use standard
TWiki
# preferences that can be defined in your Main.TWikiPreferences and
overridden
# at the web and topic level.
$NO_PREFS_IN_TOPIC = 1;

# Name of this Plugin, only used in this module
$pluginName = 'AutoNotifyPlugin';

sub initPlugin {
    my( $topic, $web, $user, $installWeb ) = @_;

    # check for Plugins.pm versions
    if( $TWiki::Plugins::VERSION < 1.026 ) {
        TWiki::Func::writeWarning( "Version mismatch between $pluginName
and Plugins.pm" );
        return 0;
    }

    $debug = $TWiki::cfg{Plugins}{EmptyPlugin}{Debug} || 0;

    # Plugin correctly initialized
    return 1;
}

sub registrationHandler {
    # do not uncomment, use $_[0], $_[1]... instead
    ### my ( $web, $wikiName, $loginName ) = @_;
    TWiki::Func::writeDebug( "- ${pluginName}::registrationHandler( $_[0],
$_[1] )" ) if $debug;

    my $theWeb = $TWiki::cfg{UsersWebName} || "Main";
    my $theTopic = $TWiki::cfg{NotifyTopicName} || "WebNotify";
    my $quiet = "1";
    my $text = TWiki::Func::readTopicText( $theWeb, $theTopic );

    my $new = "    * $theWeb.$_[1]: $_[1] \n";
    if ( $text =~ /(.*)\s\s\s*(.*)/s ) {
        my $result = "$1$new    *$2";
        TWiki::Func::saveTopicText( $theWeb, $theTopic, $result, '',
$quiet );
    } else {
        my $result = "$text\n$new";
        TWiki::Func::saveTopicText( $theWeb, $theTopic, $result, '',
$quiet );
    }
}

```

```

    }
}
1;

=== TWiki/Plugins/AutoNotifyPlugin/AutoNotifyPlugin.txt ===

%META:TOPICINFO{author="TWikiContributor" date="1111929255" format="1.0"
version="9"%
---+ AutoNotify TWiki Plugin

When a user completes the registration process, this Plugin adds the user
to the %MAINWEB%.WebNotify list to notify about changes in the user topic
home.

---+ Syntax Rules

(none)

---+ Plugin Installation Instructions

---+ Plugin Info

| Plugin Author: | TWiki:Main.EzequielConte |
| Copyright: | &copy; 2001-2006, %TWIKIWEB%.TWikiContributor |
| License: | GPL ([[http://www.gnu.org/copyleft/gpl.html][GNU General
Public License]]) |
| Plugin Version: | 01 Feb 2006 |
| Change History: | <!-- versions below in reverse order -->&nbsp; |
| TWiki Dependency: | $TWiki::Plugins::VERSION 1.1 |
| CPAN Dependencies: | none |
| Other Dependencies: | none |
| Perl Version: | 5.005 |

Related Topics: %TWIKIWEB%.TWikiPlugins,
%TWIKIWEB%.DeveloperDocumentationCategory,
%TWIKIWEB%.AdminDocumentationCategory, %TWIKIWEB%.TWikiPreferences

-- %TWIKIWEB%.TWikiContributor - 01 Feb 2006

=== TWiki/Plugins/HeadlinesPlugin/HeadlinesPlugin/Core.pm ===

# Plugin for TWiki Enterprise Collaboration Platform, http://TWiki.org/
#
# Copyright (C) 2002-2006 Peter Thoeny, peter@thoeny.org
# Copyright (C) 2005-2006 Michael Daum <micha@nats.informatik.uni-
hamburg.de>
# Copyright (C) 2005 TWiki Contributors
#
# This program is free software; you can redistribute it and/or
# modify it under the terms of the GNU General Public License
# as published by the Free Software Foundation; either version 2
# of the License, or (at your option) any later version. For

```

```

# more details read LICENSE in the root of this distribution.
#
# This program is distributed in the hope that it will be useful,
# but WITHOUT ANY WARRANTY; without even the implied warranty of
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
#
# As per the GPL, removal of this notice is prohibited.
#
# =====
#
# This is the HeadlinesPlugin used to show RSS news feeds.
# Plugin home: http://TWiki.org/cgi-bin/view/Plugins/HeadlinesPlugin
#

# =====
package TWiki::Plugins::HeadlinesPlugin::Core;

use strict;
use Digest::MD5 qw(md5_hex);

# =====
use vars qw(
    $web $topic $debug
    $defaultRefresh $defaultLimit $defaultHeader $defaultFormat
    $useLWPUserAgent $isInitialized %entityHash
    $userAgent $userAgentTimeout $doneHeader
);

$debug = 0; # toggle me

%entityHash = (
    160 => 'nbsp',
    161 => 'iexcl',
    162 => 'cent',
    163 => 'pound',
    164 => 'curren',
    165 => 'yen',
    166 => 'brvbar',
    167 => 'sect',
    168 => 'uml',
    169 => 'copy',
    170 => 'ordf',
    171 => 'laquo',
    172 => 'not',
    173 => 'shy',
    174 => 'reg',
    175 => 'macr',
    176 => 'deg',
    177 => 'plusmn',
    178 => 'sup2',
    179 => 'sup3',
    180 => 'acute',
    181 => 'micro',
    182 => 'para',
    183 => 'middot',
    184 => 'cedil',
    185 => 'sup1',
    186 => 'ordm',
    187 => 'raquo',
    188 => 'frac14',

```

189 => 'frac12',
190 => 'frac34',
191 => 'iquest',
192 => 'Agrave',
193 => 'Aacute',
194 => 'Acirc',
195 => 'Atilde',
196 => 'Auml',
197 => 'Aring',
198 => 'AElig',
199 => 'Ccedil',
200 => 'Egrave',
201 => 'Eacute',
202 => 'Ecirc',
203 => 'Euml',
204 => 'Igrave',
205 => 'Iacute',
206 => 'Icirc',
207 => 'Iuml',
208 => 'ETH',
209 => 'Ntilde',
210 => 'Ograve',
211 => 'Oacute',
212 => 'Ocirc',
213 => 'Otilde',
214 => 'Ouml',
215 => 'times',
216 => 'Oslash',
217 => 'Ugrave',
218 => 'Uacute',
219 => 'Ucirc',
220 => 'Uuml',
221 => 'Yacute',
222 => 'THORN',
223 => 'szlig',
224 => 'agrave',
225 => 'aacute',
226 => 'acirc',
227 => 'atilde',
228 => 'auml',
229 => 'aring',
230 => 'aelig',
231 => 'ccedil',
232 => 'egrave',
233 => 'eacute',
234 => 'ecirc',
235 => 'euml',
236 => 'igrave',
237 => 'iacute',
238 => 'icirc',
239 => 'iuml',
240 => 'eth',
241 => 'ntilde',
242 => 'ograve',
243 => 'oacute',
244 => 'ocirc',
245 => 'otilde',
246 => 'ouml',
247 => 'divide',
248 => 'oslash',
249 => 'ugrave',

250 => 'uacute',
251 => 'ucirc',
252 => 'uuml',
253 => 'yacute',
254 => 'thorn',
255 => 'yuml',
338 => 'OElig',
339 => 'oelig',
352 => 'Scaron',
353 => 'scaron',
376 => 'Yuml',
710 => 'circ',
732 => 'tilde',
402 => 'fnof',
913 => 'Alpha',
914 => 'Beta',
915 => 'Gamma',
916 => 'Delta',
917 => 'Epsilon',
918 => 'Zeta',
919 => 'Eta',
920 => 'Theta',
921 => 'Iota',
922 => 'Kappa',
923 => 'Lambda',
924 => 'Mu',
925 => 'Nu',
926 => 'Xi',
927 => 'Omicron',
928 => 'Pi',
929 => 'Rho',
931 => 'Sigma',
932 => 'Tau',
933 => 'Upsilon',
934 => 'Phi',
935 => 'Chi',
936 => 'Psi',
937 => 'Omega',
945 => 'alpha',
946 => 'beta',
947 => 'gamma',
948 => 'delta',
949 => 'epsilon',
950 => 'zeta',
951 => 'eta',
952 => 'theta',
953 => 'iota',
954 => 'kappa',
955 => 'lambda',
956 => 'mu',
957 => 'nu',
958 => 'xi',
959 => 'omicron',
960 => 'pi',
961 => 'rho',
962 => 'sigmaf',
963 => 'sigma',
964 => 'tau',
965 => 'upsilon',
966 => 'phi',
967 => 'chi',

968 => 'psi',
969 => 'omega',
977 => 'thetasym',
978 => 'upsih',
982 => 'piv',
8194 => 'ensp',
8195 => 'emsp',
8201 => 'thinsp',
8204 => 'zwnj',
8205 => 'zwj',
8206 => 'lrm',
8207 => 'rlm',
8211 => 'ndash',
8212 => 'mdash',
8216 => 'lsquo',
8217 => 'rsquo',
8218 => 'sbquo',
8220 => 'ldquo',
8221 => 'rdquo',
8222 => 'bdquo',
8224 => 'dagger',
8225 => 'Dagger',
8240 => 'permil',
8249 => 'lsaquo',
8250 => 'rsaquo',
8364 => 'euro',
8226 => 'bull',
8230 => 'hellip',
8242 => 'prime',
8243 => 'Prime',
8254 => 'oline',
8260 => 'frasl',
8472 => 'weierp',
8465 => 'image',
8476 => 'real',
8482 => 'trade',
8501 => 'alefsym',
8592 => 'larr',
8593 => 'uarr',
8594 => 'rarr',
8595 => 'darr',
8596 => 'harr',
8629 => 'crarr',
8656 => 'lArr',
8657 => 'uArr',
8658 => 'rArr',
8659 => 'dArr',
8660 => 'hArr',
8704 => 'forall',
8706 => 'part',
8707 => 'exist',
8709 => 'empty',
8711 => 'nabla',
8712 => 'isin',
8713 => 'notin',
8715 => 'ni',
8719 => 'prod',
8721 => 'sum',
8722 => 'minus',
8727 => 'lowast',
8730 => 'radic',

```

8733 => 'prop',
8734 => 'infin',
8736 => 'ang',
8743 => 'and',
8744 => 'or',
8745 => 'cap',
8746 => 'cup',
8747 => 'int',
8756 => 'there4',
8764 => 'sim',
8773 => 'cong',
8776 => 'asymp',
8800 => 'ne',
8801 => 'equiv',
8804 => 'le',
8805 => 'ge',
8834 => 'sub',
8835 => 'sup',
8836 => 'nsub',
8838 => 'sube',
8839 => 'supe',
8853 => 'oplus',
8855 => 'otimes',
8869 => 'perp',
8901 => 'sdot',
8968 => 'lceil',
8969 => 'rceil',
8970 => 'lfloor',
8971 => 'rfloor',
9001 => 'lang',
9002 => 'rang',
9674 => 'loz',
9824 => 'spades',
9827 => 'clubs',
9829 => 'hearts',
9830 => 'diams',
);

# =====
sub writeDebug {
    &TWiki::Func::writeDebug('HeadlinesPlugin - ' . $_[0]) if $debug;
    #print STDERR 'HeadlinesPlugin - ' . $_[0] . "\n" if $debug;
}

# =====
sub doInit {

    return if $isInitialized;
    $isInitialized = 1;

    # Get plugin preferences
    $defaultRefresh =
TWiki::Func::getPreferencesValue('HEADLINESPLUGIN_REFRESH') || 60;
    $defaultLimit =
TWiki::Func::getPreferencesValue('HEADLINESPLUGIN_LIMIT') || 100;
    $defaultHeader =
TWiki::Func::getPreferencesValue('HEADLINESPLUGIN_HEADER') ||
    '| *[[${link}]${title}]* |';
    $defaultFormat =
TWiki::Func::getPreferencesValue('HEADLINESPLUGIN_FORMAT') ||

```

```

    '| [[${link}][${title}] |';
    $useLWPUserAgent =
TWiki::Func::getPreferencesValue('HEADLINESPLUGIN_USELWPUSERAGENT')
    || 'on';
    $userAgentTimeout =
TWiki::Func::getPreferencesValue("HEADLINESPLUGIN_USERAGENTTIMEOUT")
    || 20;

    $useLWPUserAgent =~ s/^\s*(.*?)\s*$/$1/go;
    $useLWPUserAgent = ($useLWPUserAgent =~ /on|yes|1/) ? 1:0;

}

# =====
sub errorMsg {
    return
        $_[0] .
        '<span class="twikiAlert">' .
        '<noautolink>' . "\n" .
        'HeadlinesPlugin ' .
        $_[1] . "\n" .
        '</noautolink>' . "\n" .
        '</span>';
}

# =====
sub readRssFeed
{
    my ($theUrl, $theRefresh) = @_ ;

    #writeDebug("readRssFeed($theUrl, $theRefresh)");

    my $cacheDir = '';
    my $cacheFile = '';
    if ($theRefresh) {
        if (defined &TWiki::Func::getWorkArea) {
            $cacheDir = TWiki::Func::getWorkArea('HeadlinesPlugin');
        } else {
            my $twikiWeb = &TWiki::Func::getTwikiWebname();
            $cacheDir = TWiki::Func::getPubDir() . '/' . $twikiWeb .
'/HeadlinesPlugin';
            $cacheDir =~ /(.*)/;
            $cacheDir = $1; # untaint (save because only internal variables)
        }
        $cacheFile = $cacheDir . '/_rss-' . md5_hex($theUrl);
        $cacheFile =~ /(.*)/; $cacheFile = $1; # untaint
        if ((-e $cacheFile) && ((time() - (stat(_))[9]) <= ($theRefresh * 60)))
        {
            # return cached version if it exists and isn't too old. 1440 = 24h *
60min
            return TWiki::Func::readFile($cacheFile);
        }
    }

    unless ($theUrl =~ /^https?:\/\//) { # internal
        my ($thisWeb, $thisTopic) = TWiki::Func::normalizeWebTopicName($web,
$theUrl);
        $theUrl = TWiki::Func::getViewUrl($thisWeb, $thisTopic);
        if ($theUrl =~ /^\/\//) {
            $theUrl = TWiki::Func::getUrlHost().$theUrl;
        }
    }
}

```

```

}
#writeDebug("url=$theUrl");

my ($text, $errorMsg) =
useLWPUserAgent?&getUrlLWP($theUrl):&getUrl($theUrl);
return (undef, "ERROR: $errorMsg") if $errorMsg;

if ($theRefresh) {
    unless(-e $cacheDir) {
        # create the cache directory in the pub dir of the HeadlinesPlugin
        umask(002);
        mkdir($cacheDir, 0775);
    }
    # save text in cache file before returning it
    TWiki::Func::saveFile($cacheFile, $text);
}

return ($text, undef);
}

# =====
sub handleHeadlinesTag {
    my ($thePre, $theArgs);
    ($web, $topic, $thePre, $theArgs) = @_;

    &doInit();

    my $href      = TWiki::Func::extractNameValuePair($theArgs) ||
                    TWiki::Func::extractNameValuePair($theArgs, 'href');
    my $refresh   = TWiki::Func::extractNameValuePair($theArgs, 'refresh') ||
                    $defaultRefresh;
    my $limit     = TWiki::Func::extractNameValuePair($theArgs, 'limit')   ||
                    $defaultLimit;
    my $header    = TWiki::Func::extractNameValuePair($theArgs, 'header')  ||
                    $defaultHeader;
    my $format    = TWiki::Func::extractNameValuePair($theArgs, 'format')  ||
                    $defaultFormat;
    my $summary   = TWiki::Func::extractNameValuePair($theArgs, 'summary')  ||
                    -1;

    $header =~ s/\$n([\^a-zA-Z])/\n$1/gos; # expand "$n" to new line
    $header =~ s/([\^n])$/\n/os;          # append new line if needed
    $header =~ s/\$percent/\%/gos;
    $format =~ s/\$n([\^a-zA-Z])/\n$1/gos; # expand "$n" to new line
    $format =~ s/([\^n])$/\n/os;          # append new line if needed
    $format =~ s/\$t\b/\t/go;
    $format =~ s/\$percent/\%/gos;

    unless($href) {
        return errorMsg($thePre, "href parameter (news source) is missing");
    }

    my ($raw, $msg) = readRssFeed($href, $refresh);
    return errorMsg($thePre, $msg) if $msg;

    my $text = $thePre."<div class=\"\`headlinesRss\`\"><noautolink>\n";

    # distinguish rss from atom
    if ($raw =~ /<feed xmlns=[^>]+?>/) {
        $text .= parseAtomFeed($raw, $header, $format, $limit, $summary);
    } else {

```

```

    $text .= parseRssFeed($raw, $header, $format, $limit, $summary);
}
$text .= "</noautolink></div>\n";

return $text;
}

# =====
sub parseRssFeed {
    my ($raw, $header, $format, $limit, $summary) = @_;

    #writeDebug("(1) raw=$raw");

    my $sub = '';
    my $val = '';
    my $text = '';
    my $baseRef = '';

    if ($raw =~ /<channel[^\>]*>(.*?)</channel>/) {
        $sub = $1;
        if ($sub =~ /(.*?)<item[^\>]*>/g) {
            $sub = $1; # cut stuff above all <item>s
        }
        if ($sub =~ /<title[^\>]*>(.*?)</title>/) {
            $val = &recode($1);
            $header =~ s/\$(channel)?title/$val/gos;
        }
        if ($sub =~ /<link[^\>]*>\s*(.*?)\s*</link>/) {
            $val = $1;
            $baseRef = $val;
            $baseRef =~ s/^(https?\.\/\./.*?)\/\.*$/$1/go;
            $header =~ s/\$(channel)?link/$val/gos;
        }
        if ($sub =~ /<description[^\>]*>(.*?)</description>/) {
            $val = &recode($1);
            $header =~ s/\$(channel)?description/$val/gos;
        }
        if ($sub =~ /<lastBuildDate[^\>]*>(.*?)</lastBuildDate>/) {
            $val = $1;
            $header =~ s/\$(channel)?date/$val/gos;
        }
        if ($sub =~ /<pubDate[^\>]*>(.*?)</pubDate>/) {
            $val = $1;
            $header =~ s/\$(channel)?date/$val/gos;
        }
        if ($sub =~ /<copyright[^\>]*>(.*?)</copyright>/) {
            $val = &recode($1);
            $header =~ s/\$rights/$val/gos;
        }
        &parseDC(\$sub, \$header);
    }
    if ($raw =~ /<image[^\>]*>(.*?)</image>/) {
        $sub = $1;
        if ($sub =~ /<title[^\>]*>(.*?)</title>/) {
            $val = &recode($1);
            $header =~ s/\$imagetitle/$val/gos;
        }
        if ($sub =~ /<url[^\>]*>(.*?)</url>/) {
            $val = $1;
            if ($baseRef && $val =~ /^\/\//o) {
                $val = $baseRef.$val; # fix relative image url
            }
        }
    }
}

```

```

    }
    $header =~ s/\$imageurl/$val/gos;
  }
  if ($sub =~ /<link[>]*>\s*(.*?)\s*</link>/) {
    $val = $1;
    $header =~ s/\$imagelink/$val/gos;
  }
  if ($sub =~ /<description[>]*>(.*?)</description>/) {
    $val = &recode($1);
    $header =~ s/\$imagedescription/$val/gos;
  }
}
$header =~
s/\$((channel|image)(title|link|description|url|date|rights))//go;
$header =~
s/\$(rights|coverage|relation|language|source|identifier|format|date|contri
butor|creator|title|subject|description)//go;
$text .= $header;

$raw =~ s/<\/?items>//go;
$raw =~ s/^.*(?<item[>]*>)/$1/gos; # cut stuff above all <item>s

my $line = "";
my $ok = 0;
my $count = 0;
foreach (split(/<item[>]*>/, $raw)) {
  next unless $_;
  #writeDebug("item='$_'");
  $line = $format;
  $ok = 0;
  if (/<title[>]*>(.*?)</title>/) {
    $val = &recode($1) || 'Untitled';
    $line =~ s/\$(item)?title/$val/gos;
    $ok = 1;
  }
  if (/<link[>]*>\s*(.*?)\s*</link>/) {
    $val = $1;
    $val =~ s/^http:\\\/\\.*\*(http:\\\/\\.*$)/$1/gos; # yahoo fix
    $line =~ s/\$(item)?link/$val/gos;
    $ok = 1;
  }
  if (&parseCONTENT(\$_, \$line, $summary)) {
    $ok = 1;
  }
  if (/<description[>]*>(.*?)</description>/) {
    $val = &recode($1);
    if ($summary > -1) {
      $val =~ s/<\/?[>]+>//gois;
      $val = substr($val,0, $summary);
    }
    $line =~ s/\$(item)?description/$val/gos;
    $ok = 1;
  }
  if (/<pubDate[>]*>(.*?)</pubDate>/) {
    $val = $1;
    $line =~ s/\$(item)?date/$val/gos;
    $ok = 1;
  }
  if (/<date[>]*>(.*?)</date>/) {
    $val = $1;
    $line =~ s/\$(item)?date/$val/gos;
  }
}

```

```

    $ok = 1;
}
if (/<category[^>]*>(.*?)</category>/) {
    $val = $1;
    $line =~ s/\$(item)?category/$val/gos;
    $ok = 1;
}
if (&parseDC(\$_, \$line)) {
    $ok = 1;
}
if (&parseIMAGE(\$_, \$line)) {
    $ok = 1;
}
$line =~ s/\$title/Untitled/go;
$line =~ s/\$(item)?(link|description|date|category)//go;
$line =~
s/\$(rights|coverage|relation|language|source|identifier|format|date|contri
butor|creator|subject|description)//go;
$text .= $line if ($ok);
$count++;
last if $count >= $limit;
}

# fix relative img urls
$text =~ s/(<img .*?src=['"])\:\/\/$1$baseRef\/\//go if $baseRef;

return $text;
}

# =====
sub parseAtomFeed {
    my ($raw, $header, $format, $limit, $summary) = @_;

    my $sub = '';
    my $val = '';
    my $text = '';
    my $baseRef = '';

    if ($raw =~ /<feed.*?>(.*?)</feed>/) {
        $sub = $1;
        if ($sub =~ /<title[^>]*>(.*?)</title>/) {
            $val = &recode($1);
            $header =~ s/\$(channel)?title/$val/gos;
        }
        if ($sub =~ /<link[^>]*href="(["]*)" [^>]*type="text\/html"[^>]*\/>/ ||
            $sub =~ /<link[^>]*type="text\/html"[^>]*href="(["]*)" [^>]*\/>/) {
            $val = $1;
            $baseRef = $val;
            $baseRef =~ s/^(https?\.\/\|.*)\/\.*\/$1/go;
            $header =~ s/\$(channel)?link/$val/gos;
        }
        if ($sub =~ /<updated[^>]*>(.*?)</updated>/) {
            $val = $1;
            $header =~ s/\$(channel)?date/$val/gos;
        }
        if ($sub =~ /<modified[^>]*>(.*?)</modified>/) {
            $val = $1;
            $header =~ s/\$(channel)?date/$val/gos;
        }
    }
    # if ($sub =~ /<link rel="icon" href="(["]*)" [^>]*\/>/) {
    #     $val = $1;

```

```

#   $header =~ s/\$imageurl/$val/gos;
#   }
if ($sub =~ /<logo[>]*>(.*?)<\logo>/) {
    $val = $1;
    if ($baseRef && $val =~ /^\/o) {
        $val = $baseRef.$val; # fix relative image url
    }
    $header =~ s/\$imageurl/$val/gos;
}
if ($sub =~ /<rights[>]*>(.*?)<\rights>/) {
    $val = &recode($1);
    $header =~ s/\$rights/$val/gos;
}
if ($sub =~ /<subtitle[>]*>(.*?)<\subtitle>/) {
    $val = &recode($1);
    $header =~ s/\$(channel)?description/$val/gos;
}
if ($sub =~ /<tagline[>]*>(.*?)<\tagline>/) {
    $val = &recode($1);
    $header =~ s/\$(channel)?description/$val/gos;
}
$header =~ s/\$(image)?(title|link|description|url|date|rights)//go;
$header =~
s/\$(rights|coverage|relation|language|source|identifier|format|contributor
|creator|subject)//go;
$text .= $header;

$raw =~ s/.?*(<entry[>]*>)/$1/os; # cut stuff above all entries
my $line = "";
my $ok = 0;
my $count = 0;
foreach (split(/<entry[>]*>/, $raw)) {
    next unless $_;
    $line = $format;
    $ok = 0;
    if (/<title[>]*>(.*?)<\title>/) {
        $val = &recode($1) || 'Untitled';
        $line =~ s/\$(item)?title/$val/gos;
        $ok = 1;
    }
    if (/<link[>]*href="(["]*)" [^>]*type="text\/html"[>]*\//> ||
        /<link[>]*type="text\/html"[>]*href="(["]*)" [^>]*\//>) {
        $val = $1;
        $val =~ s/^http:\/\/\.*\*(http:\/\/\.*\*)/$1/gos; # yahoo fix
        $line =~ s/\$(item)?link/$val/gos;
        $ok = 1;
    }
    if (/<updated[>]*>(.*?)<\updated>/) {
        $val = $1;
        $line =~ s/\$(item)?date/$val/gos;
        $ok = 1;
    }
    if (/<author[>]*>(.*?)<\author>/) {
        $sub = $1;
        if ($sub =~ /<name[>]*>(.*?)<\name>/) {
            $val = &recode($1);
            $line =~ s/\$contributor/$val/gos;
            $ok = 1;
        }
    }
}
if (/<created[>]*>(.*?)<\created>/) {

```



```

$val = $1;
$line =~ s/\$(item)?date/$val/gos;
$ok = 1;
}
if (/<modified[^\>]*>(.*?)</modified>/) {
$val = $1;
$line =~ s/\$(item)?date/$val/gos;
$ok = 1;
}
if (/<content[^\>]*>(.*?)</content>/) {
$val = &recode($1);
    if ($summary > -1) {
        $val =~ s/<\/?[^>]+>/gois;
        $val = substr($val,0, $summary);
    }
$line =~ s/\$(item)?description/$val/gos;
$ok = 1;
}
if (/<summary[^\>]*>(.*?)</summary>/) {
$val = &recode($1);
    if ($summary > -1) {
        $val =~ s/<\/?[^>]+>/gois;
        $val = substr($val,0, $summary);
    }
    $line =~ s/\$(item)?description/$val/gos;
$ok = 1;
}
if (/<category[^\>]*>(.*?)</category>/) {
$val = $1;
$line =~ s/\$(item)?category/$val/gos;
$ok = 1;
}

$line =~ s/\$(item)?title/Untitled/go;
$line =~ s/\$(item)?(link|description|date|category)//go;
$line =~
s/\$(rights|coverage|relation|language|source|identifier|format|contributor
|creator|subject)//go;
$text .= $line if ($ok);
$count++;
    last if $count >= $limit;
}
}

# fix relative img urls
$text =~ s/(<img .*?src=['"])\:\/\/$1$baseRef\/go if $baseRef;

return $text;
}

# =====
# TODO: externalize this function to a place of more general use
sub recode {
    my $text = shift;

    unless ($text =~ s/<!\[CDATA\[([.*?)]\]>/go) {
        $text =~ s/&lt;/go;
        $text =~ s/&gt;/go;
        $text =~ s/&amp;/go;
        $text =~ s/&quot;/go;
    }
}

```

```

$text =~ s/&#xD;/\n/go;

# TODO: partial utf8 support
$text =~ s/\xc2\xae/&reg;/go;
$text =~ s/\xc2\xab/&laquo;/go;
$text =~ s/\xc2\xbb/&raquo;/go;

$text =~ s/\xc3\xa8/&egrave;/go;
$text =~ s/\xc3\xa9/&eacute;/go;
$text =~ s/\xc3\xb1/&ntilde;/go;

$text =~ s/\xe2\x80\xa2/&bull;/go;
$text =~ s/\xe2\x80\xa6/&hellip;/go;
$text =~ s/\xe2\x80[\x93\x94]/-/go; # SMELL: use matching entities
$text =~ s/\xe2\x80[\x98\x99]'/go;
$text =~ s/\xe2\x80[\x9c\x9d]'/go;

$text =~ s/([\xc2\xc3])([\x80-
\xbf])/chr(ord($1)<<6&0xC0|ord($2)&0x3F)/eg;

# map integer representations to html entities
$text =~ s/&#(\d+)/($1<160)?chr($1):'&'.(SentityHash{$1}||"#$1").'/ge;

return $text;
}

# =====
sub formatDCdate {
    my $date = shift;

    if ($date =~ /(\d\d\d\d)-(\d\d)-(\d\d)T(\d\d):(\d\d).*/ ) {
        my $year = $1;
        my $month = $2;
        my $day = $3;
        my $hour = $4;
        my $minute = $5;
        return "$year-$month-$day $hour\:$minute";
    }

    return $date;
}

# =====
sub parseDC {
    my ($input, $output) = @_ ;
    my $ok = 0;
    my $val;

    if (!$output) {
        $output = $input;
    }

    #writeDebug("parseDC called");
    #writeDebug("- input = $$input");

    if ($$input =~ <dc:title[^\>]*>(.*?)<\/dc:title>/) {
        $val = &recode($1);
        $$output =~ s/\$title/$val/gos;
        $ok = 1;
    }
}

```

```

}
if ($$input =~/<dc:creator[^>]*>(.*?)</dc:creator>/) {
    $val = &recode($1);
    $$output =~ s/\$creator/$val/gos;
    $ok = 1;
}
if ($$input =~/<dc:subject[^>]*>(.*?)</dc:subject>/) {
    $val = &recode($1);
    $$output =~ s/\$subject/$val/gos;
    $ok = 1;
}
if ($$input =~/<dc:description[^>]*>(.*?)</dc:description>/) {
    $val = &recode($1);
    $$output =~ s/\$description/$val/gos;
    $ok = 1;
}
if ($$input =~/<dc:publisher[^>]*>(.*?)</dc:publisher>/) {
    $val = &recode($1);
    $$output =~ s/\$publisher/$val/gos;
    $ok = 1;
}
if ($$input =~/<dc:contributor[^>]*>(.*?)</dc:contributor>/) {
    $val = &recode($1);
    $$output =~ s/\$contributor/$val/gos;
    $ok = 1;
}
if ($$input =~/<dc:date[^>]*>(.*?)</dc:date>/) {
    $val = &formatDCdate($1);
    $$output =~ s/\$date/$val/gos;
    $ok = 1;
}
if ($$input =~/<dc:type[^>]*>(.*?)</dc:type>/) {
    $val = $1;
    $$output =~ s/\$type/$val/gos;
    $ok = 1;
}
if ($$input =~/<dc:format[^>]*>(.*?)</dc:format>/) {
    $val = $1;
    $$output =~ s/\$format/$val/gos;
    $ok = 1;
}
if ($$input =~/<dc:identifier[^>]*>(.*?)</dc:identifier>/) {
    $val = $1;
    $$output =~ s/\$identifier/$val/gos;
    $ok = 1;
}
if ($$input =~/<dc:source[^>]*>(.*?)</dc:source>/) {
    $val = $1;
    $$output =~ s/\$source/$val/gos;
    $ok = 1;
}
if ($$input =~/<dc:language[^>]*>(.*?)</dc:language>/) {
    $val = $1;
    $$output =~ s/\$language/$val/gos;
    $ok = 1;
}
if ($$input =~/<dc:relation[^>]*>(.*?)</dc:relation>/) {
    $val = $1;
    $$output =~ s/\$relation/$val/gos;
    $ok = 1;
}
}

```

```

if ($$input =~ /<dc:coverage[^>]*>(.*?)<\/dc:coverage>/) {
    $val = $1;
    $$output =~ s/\$coverage/$val/gos;
    $ok = 1;
}
if ($$input =~ /<dc:rights[^>]*>(.*?)<\/dc:rights>/) {
    $val = &recode($1);
    $$output =~ s/\$rights/$val/gos;
    $ok = 1;
}

#writeDebug("- output = $$output") if ($debug && $ok);

return $ok;
}

# =====
sub parseIMAGE {
    my ($input, $output) = @_ ;
    my $ok = 0;
    my $val;

    #writeDebug("- HeadlinesPlugin::parseIMAGE called");
    #writeDebug("- input = $$input");

    if ($$input =~ /<image:item ([^>]*?)>(.*?)<\/image:item>/) {
        my $attrs = $1;
        my $boddy = $2;
        my $img = "<img ";
        if ($attrs =~ /rdf:about="([^\"]*?)"/) {
            $img .= "src=\"$1\" ";
            $ok = 1;
        }
        if ($boddy =~ /<.*?title>(.*?)<\/.*?title>/) {
            $img .= "alt=\"$1\" ";
        }
        if ($boddy =~ /<image:width>(.*?)<\/image:width>/) {
            $img .= "width=\"$1\" ";
        }
        if ($boddy =~ /<image:height>(.*?)<\/image:height>/) {
            $img .= "height=\"$1\" ";
        }

        if ($ok) {
            $img .= ">";
            $$output =~ s/\$image/$img/gos;
        }
    }

    $$output =~ s/\$image//gos;

    #writeDebug("- output = $$output");

    return $ok;
}

# =====
sub parseCONTENT {
    my ($input, $output, $summary) = @_ ;
    my $ok = 0;

```

```

my $val;

#writeDebug("- HeadlinesPlugin::parseCONTENT called");
#writeDebug("input=$$input");

if ($$input =~ /<content:encoded>(.*?)</content:encoded>/) {
    #writeDebug("found content:encoded");
    $val = &recode($1);

    if ($summary > -1) {
        $val =~ s/<\/?[^\>]+>/gois;
        $val = substr($val,0, $summary);
    }

    if ($val) {
        $ok = 1;
        $$output =~ s/\$content/$val/g;
        $$output =~ s/\$description/$val/g;
    }
}

$$output =~ s/\$content//g;

return $ok;
}

# =====
sub getUrlLWP {
    my $theUrl = shift;

    #writeDebug("called getUrlLWP($theUrl)");

    unless ($userAgent) {
        eval "use LWP::UserAgent";
        die $@ if $@;

        my $proxyHost = TWiki::Func::getPreferencesValue('PROXYHOST') || '';
        my $proxyPort = TWiki::Func::getPreferencesValue('PROXYPORT') || '';

        $userAgent = LWP::UserAgent->new();
        $userAgent->agent('TWiki HeadlinesPlugin');
        # don't leave the LWP default string there as
        # this is blocked by some sites, e.g. google news
        $userAgent->timeout($userAgentTimeout);
        $userAgent->proxy("http", "$proxyHost:$proxyPort/")
            if $proxyHost && $proxyPort;
    }

    my $request = HTTP::Request->new('GET', $theUrl);
    $request->referer(TWiki::Func::getViewUrl($web, $topic));
    my $response = $userAgent->request($request);
    if ($response->is_error) {
        return (undef, $response->status_line);
    } else {
        my $text = $response->content;
        $text =~ s/\r\n?/ /gos;
        $text =~ s/\n/ /gos;
        # $text =~ s/ +/ /gos;
        return ($text, undef);
    }
}

```

```

}

# =====
sub getUrl {
    my $theUrl = shift;

    my $host = '';
    my $port = 0;
    my $path = '';
    if ($theUrl =~ /https?:\:\/\/(?:.*?):([0-9]+)(\/.*)/) {
        $host = $1;
        $port = $2;
        $path = $3;
    } elsif($theUrl =~ /https?:\:\/\/(?:.*?)(\/.*)/) {
        $host = $1;
        $path = $2;
    }
    return (undef, "invalid format of the href parameter") unless $path;

    # figure out how to get to TWiki::Net which is wide open in Cairo and
    before,
    # but Dakar uses the session object.
    my $text = $TWiki::Plugins::SESSION->{net}
        ? $TWiki::Plugins::SESSION->{net}->getUrl( $host, $port, $path )
        : TWiki::Net::getUrl( $host, $port, $path );

    if ($text =~ /text\/plain\s*ERROR\: (.*)/s) {
        my $msg = $1;
        $msg =~ s/[\n\r]/ /gos;
        return (undef, "Can't read $theUrl ($msg)");
    }
    if ($text =~ /HTTP\/[0-9\.]+\s*([0-9]+)\s*([\^\n]*)/s) {
        return (undef, "Can't read $theUrl ($1 $2)")
            unless $1 == 200;
    }
    $text =~ s/^\.*?\n\n(.*)/$1/os; # strip header
    $text =~ s/\r\n?/ /gos;
    $text =~ s/\n/ /gos;
    # $text =~ s/ +/ /gos;

    return ($text, undef);
}

```

```
1;
```

```
=== TWiki/Plugins/HeadlinesPlugin/HeadlinesPlugin.pm ===
```

```

# Plugin for TWiki Enterprise Collaboration Platform, http://TWiki.org/
#
# Copyright (C) 2002-2006 Peter Thoeny, peter@thoeny.org
# Copyright (C) 2005-2006 Michael Daum <micha@nats.informatik.uni-
hamburg.de>
# Copyright (C) 2005 TWiki Contributors
#
# This program is free software; you can redistribute it and/or
# modify it under the terms of the GNU General Public License

```

```

# as published by the Free Software Foundation; either version 2
# of the License, or (at your option) any later version. For
# more details read LICENSE in the root of this distribution.
#
# This program is distributed in the hope that it will be useful,
# but WITHOUT ANY WARRANTY; without even the implied warranty of
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
#
# As per the GPL, removal of this notice is prohibited.
#
# =====
#
# This is the HeadlinesPlugin used to show RSS news feeds.
# Plugin home: http://TWiki.org/cgi-bin/view/Plugins/HeadlinesPlugin
#

# =====
package TWiki::Plugins::HeadlinesPlugin;
use strict;

# =====
use vars qw($VERSION $RELEASE $isInitialized $doneHeader);

$VERSION = '$Rev: 11134 $';
$RELEASE = '2.11';

# =====
sub initPlugin {

    $isInitialized = 0;
    $doneHeader = 0;

    return 1;
}

# =====
sub commonTagsHandler {

    $_[0] =~ s/([\t]*)%HEADLINES{(.*)}%/handleHeadlinesTag($_[2], $_[1],
    $1, $2)/geo;

    unless ($doneHeader) {
        my $link =
            '<link rel="stylesheet" ' .
            'href="%PUBURL%/%TWIKIWEB%/HeadlinesPlugin/style.css" ' .
            'type="text/css" media="all" />';
        if($_[0] =~ s/<head>(.*?[\r\n]+)/<head>$1$link\n/o) {
            $doneHeader = 1;
        }
    }
}

# =====
sub handleHeadlinesTag {

    unless ($isInitialized) {
        eval 'use TWiki::Plugins::HeadlinesPlugin::Core;';
        die $@ if $@;
        $isInitialized = 1;
    }
}

```

```

    return TWiki::Plugins::HeadlinesPlugin::Core::handleHeadlinesTag(@_);
}

1;

===
TWiki/Plugins/TimeLineDataPlugin/lib/TWiki/Plugins/TimeLineDataPlugin.pm
===

# Plugin for TWiki Enterprise Collaboration Platform, http://TWiki.org/
#
# Copyright (C) 2000-2003 Andrea Sterbini, a.sterbini@flashnet.it
# Copyright (C) 2001-2006 Peter Thoeny, peter@thoeny.org
# and TWiki Contributors. All Rights Reserved. TWiki Contributors
# are listed in the AUTHORS file in the root of this distribution.
# NOTE: Please extend that file, not this notice.
#
# This program is free software; you can redistribute it and/or
# modify it under the terms of the GNU General Public License
# as published by the Free Software Foundation; either version 2
# of the License, or (at your option) any later version. For
# more details read LICENSE in the root of this distribution.
#
# This program is distributed in the hope that it will be useful,
# but WITHOUT ANY WARRANTY; without even the implied warranty of
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
#
# For licensing info read LICENSE file in the TWiki root.

# change the package name and $pluginName!!!
package TWiki::Plugins::TimeLineDataPlugin;

# Always use strict to enforce variable scoping
use strict;

# $VERSION is referred to by TWiki, and is the only global variable that
# *must* exist in this package.
use vars qw( $VERSION $RELEASE $SHORTDESCRIPTION $debug $pluginName
$NO_PREFS_IN_TOPIC $timeLineTopic);

# This should always be $Rev: 12445$ so that TWiki can determine the
checked-in
# status of the plugin. It is used by the build automation tools, so
# you should leave it alone.
$VERSION = '$Rev: 12445$';

# This is a free-form string you can use to "name" your own plugin version.
# It is *not* used by the build automation tools, but is reported as part
# of the version number in PLUGINDESCRPTIONS.
$RELEASE = 'Dakar';

# Short description of this plugin
# One line description, is shown in the %TWIKIWEB%.TextFormattingRules
topic:
$SHORTDESCRIPTION = 'Generates Timelines for the webs';

# You must set $NO_PREFS_IN_TOPIC to 0 if you want your plugin to use
preferences

```



```

# stored in the plugin topic. This default is required for compatibility
with
# older plugins, but imposes a significant performance penalty, and
# is not recommended. Instead, use $TWiki::cfg entries set in
LocalSite.cfg, or
# if you want the users to be able to change settings, then use standard
TWiki
# preferences that can be defined in your Main.TWikiPreferences and
overridden
# at the web and topic level.
$NO_PREFS_IN_TOPIC = 1;

# Name of this Plugin, only used in this module
$pluginName = 'TimeLineDataPlugin';

$timeLineTopic =
$TWiki::cfg{Plugins}{TimeLinePlugin}{TimeLineDataTopicName} ||
"TimeLineData";

sub initPlugin {
    my( $topic, $web, $user, $installWeb ) = @_ ;

    # check for Plugins.pm versions
    if( $TWiki::Plugins::VERSION < 1.026 ) {
        TWiki::Func::writeWarning( "Version mismatch between $pluginName
and Plugins.pm" );
        return 0;
    }

    $debug = $TWiki::cfg{Plugins}{EmptyPlugin}{Debug} || 0;

    # Plugin correctly initialized
    return 1;
}

sub afterSaveHandler {
    my ( $theText, $theTopic, $theWeb, $error, $meta ) = @_ ;

    TWiki::Func::writeDebug( "- ${pluginName}::afterSaveHandler( $_[0],
$_[1] )" ) if $debug;

    return if ($theTopic eq $timeLineTopic);

    #check if its a new revision
    #TODO find a better way
    unless ( $theText =~ /%META:TOPICINFO{.*reprev.*}%/ ) {
        my ( $revDate, $author, $rev, $comment ) = $meta-
>getRevisionInfo();
        my $wikiName = $author->webDotWikiName();
        my $newEntry = "| $revDate | $rev | $theWeb | $theTopic |
$wikiName | $comment | |";

        appendNewEntry($newEntry, $theWeb);
    }
}

sub afterAttachmentSaveHandler {
    my ( $attachmentAttrHash, $topic, $web, $error ) = @_ ;

```

```

    TWiki::Func::writeDebug( "-
    ${pluginName}::afterAttachmentSaveHandler( $_[0], $_[1] )" ) if $debug;

    #TODO get attachment rev
    my $newEntry = "| ".time()." | 1 | $web | $topic |
    $$attachmentAttrHash{'user'} | $$attachmentAttrHash{'comment'} |
    $$attachmentAttrHash{'attachment'} |";

    appendNewEntry($newEntry, $web);
}

sub appendNewEntry() {
    my ( $newEntry, $theWeb ) = @_ ;

    my $text = TWiki::Func::readTopicText( $theWeb, $timeLineTopic);

    if ($text =~ /#NEW_ENTRY/) {
        $text =~ s/#NEW_ENTRY/#NEW_ENTRY\n$newEntry/;
    } else {
        #there is no entries
        $text = $text."\n\n#NEW_ENTRY\n$newEntry\n#END_ENTRY";
    }

    TWiki::Func::saveTopicText( $theWeb, $timeLineTopic, $text, 1, 1 );
}

1;

=== TWiki/Plugins/TimeLineDataPlugin/data/TWiki/TimeLineDataPlugin.txt ===

%META:TOPICINFO{author="EzequielConte" date="1178911441" format="1.0"
version="1"}%
---+ !TimeLineData TWiki Plugin

This plugin intends to generate a database for timelines.

This plugin includes:
* %TWIKIWEB%.TimeLine : to visualize the timeline. It accept the
following parameters:
* web: name of the web you want to see
* dia: stop day
* periodo: numbers of days before stop day
* recurse: can be set to *on*, to recurse into subwebs
* %TWIKIWEB%.TimeLineRSS : RSS feed of the timeline. It accept the same
paramters as above.

---++ Syntax Rules

(none)

---++ Plugin Installation Instructions

---++ Plugin Info

```

```
| Plugin Author: | TWiki:Main:EzequielConte |
| Copyright: | &copy; 2001-2006, %TWIKIWEB%.TWikiContributor |
| License: | GPL ([http://www.gnu.org/copyleft/gpl.html][GNU General
Public License]]) |
| Plugin Version: | 01 Feb 2006 |
| Change History: | <!-- versions below in reverse order -->&nbsp;|
| TWiki Dependency: | $TWiki::Plugins::VERSION 1.1 |
| CPAN Dependencies: | none |
| Other Dependencies: | none |
| Perl Version: | 5.005 |
```

```
__Related Topics:__ %TWIKIWEB%.TWikiPlugins,
%TWIKIWEB%.DeveloperDocumentationCategory,
%TWIKIWEB%.AdminDocumentationCategory, %TWIKIWEB%.TWikiPreferences
```

```
-- %TWIKIWEB%.TWikiContributor - 01 Feb 2006
```

```
=== TWiki/Plugins/TimeLineDataPlugin/data/TWiki/TimeLine.txt ===
```

```
---+ !TimeLine de %URLPARAM{"web" default="TWiki"}%
```

```
<script type="text/javascript"
src="/pub/TWiki/JSCalendarContrib/twiki.js"></script>
<style type="text/css" media="all">
@import url('/pub/TWiki/JSCalendarContrib/calendar-blue.css');
.calendar {z-index:2000;}
</style>
<script type="text/javascript"
src="/pub/TWiki/JSCalendarContrib/calendar.js"></script>
<script type="text/javascript"
src="/pub/TWiki/JSCalendarContrib/lang/calendar-pt.js"></script>

<form action="%SCRIPTURLPATH{"view"}%/%WEB%/%TOPIC%">
Ver mudan as de <input type="text" id="dia" name="dia" size="10"
value="%URLPARAM{"dia" default="%GMTIME{"$year/$mo/$day"}%" }%" /> <input
name="calendar" src="/pub/TWiki/JSCalendarContrib/img.gif" alt="Calendar"
onClick="return showCalendar('dia','Y/%m/%d')" align="middle"
type="image"> e <input type="text" name="periodo" size="2"
value="%URLPARAM{"periodo" default="7"}%" /> dias antes.
<input type="hidden" name="web" value="%URLPARAM{"web"}%" />
<input type="submit" class="twikiSubmit" value="Atualizar" />
<p><input type="checkbox" id="recurse" name="recurse" value="on"
%IF{"defined recurse" then="checked="checked\""}%> <label
for="recurse">verificar subwebs</label></p>
</form>
```

```
%TABLE{columnwidths="18%,32%,10%,20%, 20%" sort="on" initsort="1"
initdirection="up"}%
| *Data* | *T pico* | *Vers o / Arquivo* | *Colaborador* | *Coment rio*
|
%SEARCH{"^|" regex="on" web="%URLPARAM{"web"}%" separator=" "
recurse="%URLPARAM{"recurse" default="off"}%" topic="TimeLineData"
multiple="on" nosearch="on" nototal="on"
format="%$percentCALC{\ "$SET (DATA, $pattern(^| ([0-9]*) \|.*)
$SET (REV, $pattern(^| [^|]*| *([0-9]*) *\|.*)
$SET (WEB, $pattern(^| [^|]*| [^|]*| *([A-Za-z0-9/]+) *\|.*) )
$SET (TOPIC, $pattern(^| [^|]*| [^|]*| [^|]*| *(\w+) *\|.*)
$SET (AUTHOR, $pattern(^| [^|]*| [^|]*| [^|]*| [^|]*| *([A-Za-z0-9.]*)
*\|.*) )
```



```

$SET(DIA,$TIME(%URLPARAM{"dia" default="%GMTIME{"$year/$mo/$day"}"}%
23:59)) $SET(PERIODO,$TIMEADD($GET(DIA),-%URLPARAM{"periodo"
default="7"}%, day) $IF($GET(DATA) <= $GET(DIA), $IF($GET(DATA) >
$GET(PERIODO), <item
rdf:about="\$SET(LINK,$IF($EXACT($GET(FILE),), $GET(URL)/bin/view%SCRIPTSUFF
IX%/$GET(WEB)/$GET(TOPIC)?rev=$GET(REV),
$GET(URL)/bin/viewfile%SCRIPTSUFFIX%/$GET(WEB)/$GET(TOPIC)?filename=$GET(FI
LE))$GET(LINK)\">
<guid>$GET(WEB) $GET(TOPIC) $GET(FILE) $GET(REV) $GET(DATA)</guid>
<title>$GET(TOPIC) - $IF($EXACT($GET(FILE),), rev $GET(REV), attachment
!$GET(FILE))</title> <link>$GET(LINK)</link>
<description>$IF($EXACT($GET(FILE),), &lt;a
href="\%SCRIPTURL%/rdiff%SCRIPTSUFFIX%/$GET(WEB)/$GET(TOPIC)?type=last&rend
er=sequential&context=9000&_foo=$GET(REV)#diff001\"&gt;See the
changes&lt;/a>&lt;br/&gt; , $GET(FILE) $GET(COMMENT)) &lt;br/&gt; (last changed by
<nop>$GET(AUTHOR)</description> <dc:date>$FORMATTIME($GET(DATA), $year-
$month-$dayT$hour:$minute:$second-03:00)</dc:date> <dc:contributor>
<rdf:Description link="\$GET(URL)/bin/view/$GET(AUTHOR)\">
<rdf:value>$GET(AUTHOR)</rdf:value> </rdf:Description>
</dc:contributor></item>, ,) \}$percent \}$%
<!--
* Set SKIN = rss
-->

```

=== TWiki/Applications/CadastroDeProjetos.txt ===

=== InfoProjetoForm ===

```

| *Name:* | *Type:* | *Size:* | *Values:* | *Tooltip message:* |
*Attributes:* |
| Nome | text | 25 | | Nome de guerra do Projeto | M |
| CÃ³digo | text | 6 | | CÃ³digo da Atividade | |
| Gerente | select | 1 | %SEARCH{"%META:FORM.*[U]serForm" web="%MAINWEB%"
regex="on" casesensitive="on" nototal="on" nosummary="on" nosearch="on"
format="$web.$topic" separator=", " sort="topic" excludetopic="Test*,
TWiki*"}% | Gerente do Projeto |
| DescriÃ§Ã£o | textarea | 80x4 | | InformaÃ§Ãµes | |
| Links | textarea | 80x4 | | Links | |
| Equipe | select+multi+values | 10 | %SEARCH{"%META:FORM.*[U]serForm"
web="%MAINWEB%" regex="on" casesensitive="on" nototal="on" nosummary="on"
nosearch="on" format="$web.$topic" separator=", " sort="topic"
excludetopic="Test*, TWiki*"}% | Colaboradores envolvidos no projetos | |
| EmpresaContratante | text | 25 | | Empresa Contratante | |
| Centro | select | 1 | CAI, CCD, CEI, CELTA, CID, CIENCIA, CME, CMI, CPC,
CSD, ICA, IS, LMA, NEO, PET, SAPIENS, SC, SFA, SG, SI, SO | Centro | |
| DataInicio | date | 25 | | Data de inÃ¡cio | |
| DataFim | date | 25 | | Data de conclusÃ£o | |
| ProjetosRelacionados | select+multi+values | 10 |
%SEARCH{"%META:FORM.*[I]nfoProjetoForm" topic="WebHome" web="all,-Sandbox,-
Main,-Trash" regex="on" casesensitive="on" nototal="on" nosummary="on"
nosearch="on" format="$web.$topic" separator=", " sort="topic"
excludetopic="InfoProjetoTemplate"}% | Projetos Relacionados | |
| Componentes | checkbox | 2 | Wiki, Blog, TRAC, Wiki com Acesso Externo |
Sistemas disponÃ¡veis | |

```

* Set ALLOWTOPICCHANGE = TWikiAdminGroup

```

=== InfoProjetoViewTemplate ===
%Tmpl:INCLUDE{"view"}%

%Tmpl:DEF{"active_form"}%<a
href='%SCRIPTURLPATH{"edit"}%/%WEB%/TOPIC?action=form&t=%GMTIME{"$epoch"}
%' rel='nofollow' title='Editar Projeto'>Editar Projeto</a>%Tmpl:END%
%Tmpl:DEF{"inactive_form"}% %Tmpl:END%

%Tmpl:DEF{"inactive_edit"}%<strike>Editar</strike>%Tmpl:END%
%Tmpl:DEF{"create_topic"}%Cri<span
class="twikiAccessKey">e</span>%Tmpl:END%
%Tmpl:DEF{"edit_topic"}%<span class="twikiAccessKey">E</span>dite%Tmpl:END%

%Tmpl:DEF{"active_edit"}%<a
href='%SCRIPTURLPATH{"edit"}%/%WEB%/TOPIC?action=text&t=%GMTIME{"$epoch"}
%' rel='nofollow' title='Edite topico'
accesskey='e'><b>%Tmpl:P{context="new_topic" then="create_topic"
else="edit_topic"}%</b></a>%Tmpl:END%

%Tmpl:DEF{"content"}%
<div class="natBoxDireita" style="float:right; width:30%;">
%INCLUDE{%TWIKIWEB%.InfoProjetoHeader}%
</div>
%TEXT%
#TopicEnd
<div style="clear:both;"></div>

%Tmpl:P{"attachments"}%
%Tmpl:P{"topicinfo"}%%Tmpl:END%

=== PatternSkinInfoProjetoViewTemplate ===
%Tmpl:INCLUDE{"view"}%

%Tmpl:DEF{"active_form"}%<span class="patternButton"><a
href="%SCRIPTURL{"edit"}%/%WEB%/TOPIC?action=form&t=%GMTIME{"$epoch"}%"
rel="nofollow" title='Editar Projeto'>Editar Projeto</a></span>%Tmpl:END%
%Tmpl:DEF{"inactive_form"}% %Tmpl:END%

%Tmpl:DEF{"inactive_edit"}%<span class="patternButton"><a
href="%SCRIPTURL{"view"}%/%WEB%/TOPIC" rel="nofollow"
title='%MAKETEXT{"View topic"}%'>%MAKETEXT{"View
topic"}%</a></span>%Tmpl:P{"sep"}%<span
class="patternButton"><strike>%MAKETEXT{"Edit"}%</strike></span>%Tmpl:END%

%Tmpl:DEF{"edit_topic"}%%MAKETEXT{"&Edit text"}%%Tmpl:END%

%Tmpl:DEF{"active_edit"}%%Tmpl:P{"sep"}%<span class="patternButton"><a
href="%SCRIPTURL{"edit"}%/%WEB%/TOPIC?action=text&t=%GMTIME{"$epoch"}%"
rel="nofollow" title='%MAKETEXT{"Edit this topic text"}%'

```

```

accesskey="E">%TMPL:P{context="new_topic" then="create_topic"
else="edit_topic"}%</a></span>%TMPL:P{context="WysiwygPluginEnabled"
then="wysiwyg"}%TMPL:END%

%TMPL:DEF{"formattop"}%
<div style="float:right; width:30%;background-color:#FFFFFF; border-
color:#E2DCC8; border-width:1px; border-style: solid; padding: 5px; font-
size:8pt;">
%INCLUDE{%TWIKIWEB%.InfoProjetoHeader}%
</div>
%TMPL:END%

%TMPL:DEF{"formatbottom"}%
<div style="clear:both;"></div>
%TMPL:END%

%TMPL:DEF{FORM:display:footer:editlink}%
%TMPL:END%

=== InfoProjetoHeader ===
Este tÃ³pico serÃ¡ incluido em todas as pÃ¡ginas de projetos.

---
%STARTINCLUDE%
---+++ %FORMFIELD{"Nome" topic="%BASEWEB%.%BASETOPIC%"}%
---++++ InformaÃ§Ãµes
%FORMFIELD{"CÃ³digo" topic="%BASEWEB%.%BASETOPIC%" format="CÃ³digo: $value
<br/>"}% %FORMFIELD{"Centro" topic="%BASEWEB%.%BASETOPIC%" format="Centro:
$value <br/>"}% %FORMFIELD{"DataInicio" topic="%BASEWEB%.%BASETOPIC%"
format="InÃ¡cio: $value <br/>"}% %FORMFIELD{"DataFim"
topic="%BASEWEB%.%BASETOPIC%" format="Fim: $value <br/>"}%
%FORMFIELD{"EmpresaContratante" topic="%BASEWEB%.%BASETOPIC%"
format="Empresa Contratante: $value <br/>"}%

%FORMFIELD{"DescriÃ§Ã£o" topic="%BASEWEB%.%BASETOPIC%"}%
---++++ Equipe
%FORMFIELD{"Gerente" topic="%BASEWEB%.%BASETOPIC%" format="Gerente : $value
<br/>"}% %FORMFIELD{"Equipe" topic="%BASEWEB%.%BASETOPIC%"
format="Colaboradores: $value <br/>"}%

---++++ Links
%FORMFIELD{"Links" topic="%BASEWEB%.%BASETOPIC%"}%

---++++ Projetos Relacionados
%FORMFIELD{"ProjetosRelacionados" topic="%BASEWEB%.%BASETOPIC%"}%

---++++ AÃ§Ãµes
<div style="float:right;">[[Projetos.InfoProjetoBusca][Ver Projetos]]</div>
<div style="float:left;"> <a
href="%SCRIPTURLPATH{"edit"}%/%BASEWEB%//%BASETOPIC?action=form&t=%GMTIME{"
$epoch"}%">Editar Projeto</a> </div>
<div style="clear:both;"></div>
%STOPINCLUDE%
-----

```

```
=== InfoProjetoTemplate ===
---+ %FORMFIELD{"Nome"}%

%FORMFIELD{"Descrição"}%

---++ Equipe
%FORMFIELD{"Equipe"}%

---++ Informações
Insira aqui as informações para divulgação interna do projeto
```