

**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ  
DEPARTAMENTO ACADÊMICO DE INFORMÁTICA  
ANÁLISE E DESENVOLVIMENTO DE SISTEMAS**

**ALCIDES PEDROSA DE SOUZA JÚNIOR**

**ELIAS SCHEIFER**

**WELLINGTON WILCZAK PEREIRA**

**IMPLEMENTAÇÃO DE UMA INTERFACE WEB PARA LIBERAÇÃO  
DE SITES ATRAVÉS DO *DANSGUARDIAN* NO AMBIENTE DA  
UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ-CAMPUS  
PONTA GROSSA**

**TRABALHO DE CONCLUSÃO DE CURSO**

**PONTA GROSSA**

**2015**

**ALCIDES PEDROSA DE SOUZA JÚNIOR**

**ELIAS SCHEIFER**

**WELLINGTON WILCZAK PEREIRA**

**IMPLEMENTAÇÃO DE UMA INTERFACE WEB PARA LIBERAÇÃO  
DE SITES ATRAVÉS DO *DANSGUARDIAN* NO AMBIENTE DA  
UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ-CÂMPUS  
PONTA GROSSA**

Trabalho de Conclusão de Curso apresentado como requisito parcial à obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas, do Departamento Acadêmico de Informática, da Universidade Tecnológica Federal do Paraná.

Orientador: Prof. Msc. Rogério Ranthum

Coorientador: Fabian Malheiros

**PONTA GROSSA**

**2015**



Ministério da Educação  
**Universidade Tecnológica Federal do Paraná**  
Campus Ponta Grossa

Diretoria de Pesquisa e Pós-Graduação  
Coordenação de informática  
Curso de Tecnologia em Análise e Desenvolvimento de Sistemas



---

## TERMO DE APROVAÇÃO

IMPLEMENTAÇÃO DE UMA INTERFACE WEB PARA LIBERAÇÃO DE SITES  
ATRAVÉS DO *DANSGUARDIAN* NO AMBIENTE DA UNIVERSIDADE  
TECNOLÓGICA FEDERAL DO PARANÁ - CÂMPUS PONTA GROSSA

por

ALCIDES PEDROSA DE SOUZA JÚNIOR  
ELIAS SCHEIFER  
WELLINGTON WILCZAK PEREIRA

Este Trabalho de Conclusão de Curso (TCC) foi apresentado em 20 de novembro de 2015 como requisito parcial para a obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas. Os candidatos foram arguidos pela Banca Examinadora composta pelos professores abaixo assinados. Após deliberação, a Banca Examinadora considerou o trabalho aprovado.

---

Rogério Ranthum  
Prof. Orientador

---

Geraldo Ranthum  
Membro titular

---

Fabian Maurice Malheiros Franco  
Membro titular

## **AGRADECIMENTOS**

Alcides Pedrosa de Souza Júnior:

Gostaria de agradecer primeiramente a Deus, que permitiu, que tudo isso acontecesse ao longo de minha vida e não somente nestes anos como universitário, mas que em todos os momentos é o maior mestre que alguém pode conhecer.

Agradeço ao Prof. Rogério Ranthum, pela sabedoria com que nos guiou nesta trajetória, também a Fabian Malheiros pela dedicação em nos assessorar, bem como repassar seu valioso conhecimento.

Gostaria de agradecer aos meus familiares pelo apoio mesmo que distantes.

Agradecer aos meus amigos, principalmente os que dividiram esse trabalho comigo, Elias e Wellington.

Também agradecer a minha namorada e companheira Júlia Carla de Almeida que sempre acreditou em mim e me incentivou a prosseguir mesmo nos momentos mais difíceis.

Por fim à Universidade Tecnológica Federal do Paraná, na figura de seus professores e servidores, por me acolherem nesse universo de respeito e aprendizado.

Elias Scheifer :

Primeiramente agradeço a Deus por ter me guiado por todo esse caminho e me dado saúde para conseguir completar esta etapa.

Ao Professor Rogério Ranthum pela orientação e amizade.

Agradeço a meu irmão Abrahão Scheifer em especial, não conseguiria chegar até aqui sem o seu apoio e irmandade.

Agradeço a minha mãezinha querida, Anair Scheifer que, esta desde o começo do meu curso me dando força e me cuidando ao lado de Deus.

Agradeço ao meu pai pela garra e honestidade que me passou durante a vida.

Agradeço a toda minha família por sempre estarem me apoiando e presente em minha vida.

Agradeço a minha noiva Walnisa Leoncio pelo amor e cuidado que tem tido para que eu chegasse até essa etapa, sem você tudo seria muito mais difícil.

Agradeço aos meus amigos que me ajudaram nessa etapa, em especial Fabian Malheiros, Jean Carlos Rodrigues, que com o vasto conhecimento e suas valorosas amizades me encaminharam até o término do curso.

Agradeço aos meus parceiros de TCC Wellington Wilczak e ao Alcides Pedrosa de Souza Junior pelas discussões pelos acertos, mas acima de tudo pelo nosso trabalho finalizado.

Wellington Wilczak Pereira:

A minha amada esposa Francieli que com muito carinho e apoio não mediu esforços para que eu alcançasse essa meta em minha vida. Obrigado meu amor pelas palavras de encorajamento, pelos abraços confortantes quando precisava, por ter compreendido quando por algumas vezes estava nervoso e estressado.

Ao meu filho Gustavo, que apesar da pouca idade me encorajou a cada dia buscar mais conhecimento, sendo ele o meu maior motivador nessa longa caminhada. Se muitas vezes pensei em desistir foi seu beijo de bom dia ou sua alegria ao ver a minha chegada que me davam força para enfrentar o dia a dia.

Obrigado aos meus pais pelo incentivo e admiração, mas em especial a minha mãe pelos sacrifícios que fez em razão de minha educação. Apesar da distância que nos separa sempre esteve presente em meu coração, obrigado mamãe pelas orações que fez por mim.

Aos meus tios Monica e João (in memoriam), que foram muito importantes em meu crescimento pessoal, acadêmico e profissional.

Aos meus primos Fernando, Sonia e Elaine, obrigado por terem sido para mim os irmãos que não tive a oportunidade de ter, agradeço também pelo incentivo de todas as formas para que finalizasse esse curso.

Aos meus sogros e cunhados pelo incentivo e motivação.

A todos os meus colegas de curso, porém não poderia deixar de destacar meus amigos Alcides e Elias pela amizade e companheirismo na elaboração desse trabalho.

Ao professor Msc. Rogério Ranthum pela orientação e incentivo que tornaram possível a conclusão desse trabalho.

A todos os professores do curso que foram importantes nessa vida acadêmica e na elaboração desse trabalho.

A todos aqueles que fizeram parte da minha formação direta ou indiretamente o meu sincero agradecimento.

O aspecto mais triste da vida atual é que a ciência ganha em conhecimento mais rapidamente que a sociedade em sabedoria. (Asimov, Issac, 1979)

## RESUMO

PEDROSA, A. de S. Jr.; SCHEIFER, Elias.; WILCZAK, W. P.. **Implementação de uma Interface WEB para liberação de sites através do *Dansguardian* no ambiente da Universidade Tecnológica Federal do Paraná - Câmpus Ponta Grossa**. 2015. 54 páginas. Trabalho de Conclusão de Curso de Tecnologia em Análise e Desenvolvimento de Sistemas - Universidade Tecnológica Federal do Paraná. Ponta Grossa, 2015.

A utilização da Internet em ambiente acadêmico é fundamental para a pesquisa, produção e disseminação de conhecimento, e para facilitar a administração das instituições de ensino, porém pode trazer problemas aos responsáveis, além de perda de produtividade no ambiente administrativo e educacional. A fim de se precaver de tais problemas, um conjunto de medidas são necessárias. Tais medidas vão desde a conscientização dos usuários até monitoramento e possível punição. Não obstante a configuração de ferramentas de controle de conteúdos provenientes da Internet, algum conteúdo pode ser indevidamente bloqueado, gerando tempo e esforço do administrador de redes para liberação. Esse Trabalho de Conclusão de Curso (TCC) teve como objetivo descrever o desenvolvimento de um sistema que auxilie na liberação de conteúdo considerado pertinente ao ambiente educacional bloqueado indevidamente, utilizando o *Dansguardian* conjuntamente com o *Squid*.

**Palavras-chave:** *Dansguardian*. Interface Web. Controle de conteúdo. LDAP *Lightweight Directory Access Protocol*. *Web Filters*.

## ABSTRACT

PEDROSA, A. de S. Jr.; SCHEIFER, Elias.; WILCZAK, W. P.. **IMPLEMENTING A WEB INTERFACE FOR RELEASE OF SITES THROUGH *Dansguardian* AT UNIVERSITY OF TECHNOLOGY ENVIRONMENT FEDERAL Paraná - Campus TIP THICK**. 2015. 54 pages. Work Completion Course Technology Analysis and Systems Development - Federal Technological University of Paraná. Ponta Grossa, 2015.

The use of the internet in an academic environment is essential to the research, production and dissemination of knowledge, and to facilitate the administration of educational institutions, but can bring legal problems for that responsible and lost productivity in the administrative and educational environment. In order to avoid such problems, a number of measures are necessary. Such measures range from awareness of users to monitoring and punishment. Despite the configuration of content control tools from the internet, some content may be unduly blocked, generating administrator time and effort networks to release this material. This Work Course Conclusion (TCC) aims to describe the development of a system to assist in the release of content deemed relevant to unduly blocked educational environment, using *Dansguardian* conjunction with *Squid*.

**Keywords:** *Dansguardian*. Interface Web. Content Filter. LDAP *Lightweight Directory Access Protocol*. *Web Filters*.

## LISTA DE ILUSTRAÇÕES

Figura 1 – Tela de bloqueio.....	32
Figura 2 – Tela de login.....	33
Figura 3 – Tela de Solicitação.....	34
Figura 4 – Tela de Confirmação.....	34
Figura 5 – Tela de Histórico do Usuário.....	35
Figura 6 – Tela de Lista de Sites Pendentes.....	35
Figura 7 – Tela de Liberação de Site.....	36
Figura 8 – Tela de Logs.....	37
Figura 9 – Tela de Configuração do Sistema.....	38
Figura 10 – Visão geral.....	39
Figura 11 – Ações do Administrador.....	41
Figura 12 – Ações do Professor.....	42
Figura 13 – Login do Aluno.....	43
Figura 14 – Fluxograma.....	43

## LISTA DE SIGLAS

UTFPR-PG	Universidade Tecnológica Federal do Paraná – Campus Ponta Grossa
VM	<i>Virtual Machine</i>
LDAP	<i>Lightweight Directory Access Protocol</i>
PHP	<i>Hypertext Preprocessor</i>
CSS	<i>Cascading Style Sheet</i>
HTML	<i>HyperText Markup Language</i>
WWW	<i>World Wide Web</i>
OMG	<i>Object Management Group</i>
UML	<i>Unified Modeling Language</i>
CA	<i>Certificate Authority</i> - Autoridade de Certificação
TCP/IP	<i>Transmission Control Protocol / Internet Protocol</i> - Protocolo de Controle de Transmissão / Protocolo de Internet
TI	Tecnologia de Informação
OU	<i>Organization Unit</i> - Unidade Organizacional
OpenLDAP	<i>Open Lightweight Directory Access Protocol</i>
API	<i>Application Programming Interface</i> - Interface de Programação de Aplicativos
SSH	<i>Secure Shell</i>

## SUMÁRIO

<b>1 INTRODUÇÃO</b> .....	13
<b>2 OBJETIVOS</b> .....	15
2.1 OBJETIVO GERAL .....	15
2.2 OBJETIVO ESPECIFICO.....	15
2.3 JUSTIFICATIVA .....	15
2.4 ESTRUTURA DO TRABALHO.....	16
<b>3 REFERENCIAL TEÓRICO</b> .....	17
3.1 REDES DE COMPUTADORES .....	17
3.2 TOPOLOGIA DE REDE .....	17
3.3 REDES CLIENTE SERVIDOR .....	18
3.4 SEGURANÇA EM REDES DE COMPUTADORES .....	18
3.5 REDES PONTO A PONTO .....	18
3.6 DISTRIBUIÇÃO LINUX .....	19
3.7 DISTRIBUIÇÃO DEBIAN 7.0 WHEZZY.....	20
3.8 DISTRIBUIÇÃO LINUX UBUNTU 14.0.....	20
3.9 <i>DANSGUARDIAN</i> .....	21
3.10 LIGHT WEIGHT DIRECTORY ACCESS PROTOCOL - LDAP .....	23
3.11 HYPERTEXT PREPROCESSOR- PHP .....	24
3.12 BOOTSTRAP .....	25
3.13 CASCADING STYLE SHEET - CSS.....	26
3.14 HTML.....	26
3.15 CACOO .....	27
3.16 UNIFIED MODELING LANGUAGE - UML .....	27
<b>4 METODOLOGIA</b> .....	<b>29</b>
<b>5 ANÁLISE DOS RESULTADOS</b> .....	<b>31</b>
5.1 FUNCIONAMENTO.....	31
5.1.1 DIAGRAMA DE CASOS DE USO .....	38
5.1.2 Visao Geral .....	38
5.1.3 Ações do Administrador .....	40
5.1.4 Ações do Professor .....	41
5.1.5 LOGIN DO ALUNO .....	42
5.1.6 FLUXOGRAMA .....	42
<b>6 CONCLUSAO</b> .....	<b>44</b>
6.1 TRABALHOS FUTUROS .....	45
<b>REFERÊNCIAS</b> .....	<b>46</b>
<b>APÊNDICE A - Fragmento do arquivo do <i>Dansguardian</i>, contendo a lista de conteúdos banidos de forma padrao</b> .....	<b>49</b>

<b>APÊNDICE B - Fragmento do arquivo do <i>Dansguardian</i>, contendo a lista de conteúdos banidos .....</b>	<b>50</b>
<b>APÊNDICE C - Fragmento do arquivo do <i>Dansguardian</i>, contendo a lista de conteúdos liberados de forma padrao.....</b>	<b>51</b>
<b>APÊNDICE D - Fragmento do arquivo do <i>Dansguardian</i>, contendo a lista de conteúdos liberados.....</b>	<b>52</b>
<b>APÊNDICE E - Script criação da base de dados .....</b>	<b>53</b>

## 1 INTRODUÇÃO

O advento da Internet facilitou o acesso à informações consideradas fundamentais para o gerenciamento e gestão de empresas, bem como o acesso a informações nos ambientes educacionais, (SANGARA, 2011).

“A interação em tempo real pode de fato acrescentar em muito ao desenvolvimento do ambiente no qual se está inserido. A busca por informações imediatas em diferentes partes do mundo, o acesso aos avanços em termo de pesquisas desenvolvidas nos melhores centros educacionais, torna-se muito mais acessível com a utilização de ferramentas interativas” (NEGROPONTE, 2005).

Entretanto, o uso indevido de recursos disponíveis na rede pode gerar desperdício de banda, baixa da produtividade, descumprimento de normas internas e legais, pode acarretar inclusive consequências judiciais.

Neste sentido faz-se necessária a aplicação de regras de uso e implantação de ferramentas que impeçam o desvio de finalidade na utilização da rede institucional. Esse controle deve ser eficiente e ao mesmo tempo flexível para impedir os excessos sem limitar o livre acesso a informações úteis e permitidas.

Na UTFPR-Campus Ponta Grossa, esse controle é feito utilizando conjuntamente o *Dansguardian* e o servidor *Proxy Squid*.

Segundo Tanenbaum (2003), o *Proxy* pode ser definido como um tipo de servidor que redireciona as requisições de um usuário de um serviço fornecido por outro servidor, em uma outra rede.

“*Dansguardian* é um software livre de filtro de conteúdo Web que atualmente é compatível com inúmeros sistemas operacionais e multiplataforma. Esta ferramenta filtra o conteúdo real de páginas com base em vários métodos, incluindo correspondência de frase e filtragem de URL. *Dansguardian* é projetado para ser completamente flexível e permite adequar a filtragem às suas exatas necessidades. As configurações padrões são voltadas para uma escolha primária, mas também coloca o usuário no controle do que deseja bloquear em caso de necessidades mais restritivas” (NAKAMURA E GEUS, 2007).

Sendo assim, o *Dansguardian* é um filtro de conteúdo Web seguindo uma configuração pré-definida.

Porém, esse controle, em determinadas situações, pode ser falho ao ponto de bloquear material, que em tese, não deveria ser bloqueado, impossibilitando assim o

acesso a conteúdo legítimo. Dessa forma, poderá acarretar prejuízos de várias maneiras, seja no desenvolvimento de pesquisas, ou ao sobrecarregar o administrador de redes na demanda de tempo e esforço para a liberação desse material.

Visando aperfeiçoar esse controle, pretende-se auxiliar essa liberação de falsos positivos (conteúdo bloqueados indevidamente) através do desenvolvimento de um sistema que permitirá, por meio de concessão a usuários específicos, a liberação desse material.

## 2 OBJETIVOS

### 2.1 Objetivo Geral

Este projeto tem como objetivo desenvolver uma ferramenta Web que facilite o gerenciamento e a liberação de conteúdos bloqueados na Universidade Tecnológica Federal do Paraná - Câmpus Ponta Grossa (UTFPR-PG).

### 2.2 Objetivos Específicos

- Simular um ambiente de testes semelhante ao utilizado atualmente na UTFPR-PG com a instalação do servidor Ubuntu Server 12.04, ativando o serviço de gerenciamento de conteúdo *Dansguardian* em uma máquina virtual (VM) operando com o servidor Debian.
- Desenvolver um sistema computacional em linguagem de código aberto, que permita melhorar a comunicação entre o usuário final e o administrador de rede.
- Criar um módulo para gerência de usuários e grupos com diferentes níveis de acesso, a fim de tornar o controle de conteúdo mais participativo e eficiente.
- Desenvolver um módulo de armazenamento de solicitações e liberações com a finalidade de permitir um controle mais detalhado através de auditoria de rede.

### 2.3 Justificativa

Atualmente a Universidade Tecnológica Federal do Paraná – Campus Ponta Grossa, UTFPR-PG, possui como mecanismo de gestão de conteúdos acessados via internet o software *Dansguardian*.

Esta gestão é feita de forma não automatizada, ou seja, as ferramentas utilizadas bloqueiam a partir de configurações pré-determinadas gerando, algumas vezes bloqueio indevido de conteúdo legítimo pertinentes a pesquisa.

Quando existe a necessidade de liberação de algum conteúdo bloqueado, existe a necessidade de uma requisição, feita através de correspondência eletrônica, contato telefônico ou pessoalmente ao administrador de rede da UTFPR-PG. A análise do conteúdo solicitado para a liberação é feita de forma individualizada e de acordo com a demanda solicitada.

Partindo desse princípio, pretende-se auxiliar a gestão do mecanismo de desbloqueio, através de um sistema que permita a liberação imediata de conteúdos, considerados, por usuários cadastrados pelo protocolo *Lightweight Directory Access Protocol* (LDAP), como válidos ao ambiente acadêmico, e ao mesmo tempo permitir aos demais usuários da rede da UTFPR-PG solicitar a liberação mediante uma requisição a ser analisada pelo administrador de redes.

## 2.4 Estrutura do Trabalho

Neste trabalho apresentou-se no capítulo I uma visão geral do problema a ser tratado, a proposição de uma resolução para este problema, bem como uma noção preliminar das tecnologias envolvidas nessa solução.

O capítulo II trata sobre a revisão de literatura, onde abordou-se de forma mais detalhada as tecnologias envolvidas na construção da solução do problema proposto com os principais estudos já elaborados sobre o tema e como contribuirão para o êxito de nossa empreitada.

O capítulo III apresentou uma análise da solução apresentada, com o detalhamento do sistema desenvolvido.

Já o capítulo IV apresenta os resultados obtidos através dos trabalhos realizados no sentido de aplicar a solução ao problema proposto.

Finalmente, o capítulo V apresenta as considerações finais, mostrando a aplicabilidade da solução desenvolvida.

## 3 REFERENCIAL TEÓRICO

### 3.1 Redes de Computadores

Uma rede é um conjunto de dispositivos (normalmente conhecidos como nós) conectados por *links* de comunicação. Um nó pode ser um computador, uma impressora ou outro dispositivo de envio/recepção de dados, que estejam conectados a outros nós de rede (FOROUZAN, 2006).

Todo ambiente de rede precisa armazenar informações para possibilitar o seu gerenciamento (autenticação, grupos de usuários, permissões, cotas de armazenamento e impressão, compartilhamentos e etc.). Hoje em dia, a maioria das grandes organizações possui ambientes de rede heterogêneos, com várias plataformas presentes (Linux, Windows, Solaris...) e com redes virtuais fisicamente conectadas, muitas vezes distribuídas geograficamente.

### 3.2 Topologia de Rede

A topologia da rede pode ser pensada como um mapa. Trata-se do “arranjo físico e lógico dos elementos de uma rede. Duas redes têm a mesma topologia se a sua configuração de conexão, de ligação entre seus pontos, for a mesma, embora possam diferir em suas interligações físicas, distâncias entre nós, taxas de transmissão ou tipos de sinal” (TELECOMMUNICATIONS, 2008).

“A topologia física é a configuração física ou diz respeito aos caminhos da interligação dos cabos, roteadores, switches, concentradores, enfim, componentes físicos de uma rede. A topologia lógica de uma rede é a configuração esquemática que reflete o funcionamento da comunicação” (TANEMBAUM, 2003).

Redes cibernéticas e tecnologias do anonimato rede e como será a ligação entre os usuários dessa rede. A topologia física pode ter um desenho diferente da topologia lógica.

### 3.3 Redes Cliente-Servidor

“Também são chamadas de redes baseadas em servidor, consiste de um servidor centralizado onde os usuários compartilham e acessam recursos da rede, este servidor dedicado controle níveis de acesso de usuário aos recursos compartilhados, os dados compartilhados ficam armazenados em local único facilitando tarefas administrativas como backups.” (TANENBAUM, 2002).

Cada computador conectado à rede é chamado de computador cliente. Em uma rede baseada em servidor, os usuários têm uma conta de usuário e senha para efetuar login no servidor e acessar os recursos compartilhados. Os sistemas operacionais de servidor são desenvolvidos para suportar a carga quando vários computadores clientes acessarem os recursos baseados em servidor, na figura abaixo está demonstrado um exemplo de rede baseadas em servidor.

### 3.4 Segurança em Redes de Computadores

A segurança de qualquer sistema computacional é um aspecto fundamental de sua arquitetura, principalmente quando estes, são acessíveis de redes públicas, como a Internet. A segurança de sistemas distribuídos envolve os seguintes aspectos:

Autenticação: identificação de usuários e serviços

Autorização: acesso seletivo aos recursos

Privacidade: confidencialidade

Disponibilidade

“A autenticação é o processo de identificar o usuário (ou serviço) e validar se essa identificação é autêntica. Sem essa funcionalidade corretamente garantida, um usuário malicioso pode se passar por um usuário legítimo. Da mesma forma para os servidores, sem essa garantia, um serviço malicioso pode se passar pelo serviço autêntico e interceptar dados indevidamente.” (WINCKLER, 2014)

### 3.5 Redes Ponto-a-Ponto

Em uma rede ponto a ponto também chamado de modelo de Grupo de Trabalho ou Workgroup em ambientes Windows, computadores são conectados em grupo para que outros usuários possam compartilhar recursos e informações. Não há um local central para autenticação de usuários, armazenamento de arquivos ou acesso a recursos. Isso significa que os usuários devem lembrar em qual computador do grupo de trabalho está o recurso ou a informação compartilhada que desejam acessar e também lembrar-se de qual o login utilizado neste computador para conseguir acesso ao recurso desejado, neste modelo de redes é dificultado as ações administrativas como alteração de dados de usuários ou backup de dados pois cada recurso pode estar em um computador diferente ou em mais de um computador. (TANENBAUM, 2002).

### 3.6 Distribuição linux

“O sistema operacional procura tornar a utilização do computador mais eficiente e mais conveniente. A utilização mais eficiente busca um maior retorno no investimento feito no hardware, significa mais trabalho obtido do mesmo hardware. Uma utilização mais conveniente vai diminuir o tempo necessário para a construção e utilização dos programas. Um enorme número de conceitos, abstrações, mecanismos e algoritmos foram criados e aprimorados ao longo dos últimos anos.” (SIMÃO SIRINEO, 2001).

Os programas solicitam serviços ao sistema operacional através das chamadas de sistema, entretanto, enquanto as chamadas transferem a execução para o sistema transferem a execução para o sistema operacional. Através de parâmetros, o programa informa exatamente o que necessita.

A parte do sistema operacional responsável por implementar as chamadas de sistema é normalmente chamada de núcleo ou Kernel.

Para o desenvolvimento do sistema WEA, focou-se no sistema operacional Linux, em suas distribuições mais apropriadas, explicadas mais adiante.

“Uma distribuição de Linux é um sistema operacional Unix-like incluindo o kernel Linux e outros softwares de aplicação, formando um

conjunto. Distribuições mantidas por organizações comerciais, como a Red Hat, Ubuntu, SUSE e Mandriva, bem como projetos comunitários como Debian e Gentoo montam e testam seus conjuntos de software antes de disponibilizá-los ao público. Como o Linux e a maior parte dos softwares incluídos em distribuições são livres, qualquer organização ou indivíduo suficientemente motivado podem criar e disponibilizar (comercialmente ou não) a sua própria distribuição. Isso faz com que hoje haja registro de mais de 300 distribuições ativamente mantidas, embora menos de 20 delas sejam largamente conhecidas.” (Campos, Augusto.2006).

### 3.7 Distribuição Linux Debian 7.0 Whezzy

O Debian é um sistema operacional livre, desenvolvido por milhares de voluntários ao redor do mundo que colaboram através da Internet. O pontos chave do projeto Debian são a sua base de voluntários, a sua dedicação ao Contrato Social do Debian e ao Software Livre, e o seu compromisso de fornecer o melhor sistema operacional possível. O Debian 7.0 é mais um passo importante nessa direção. (Debian.org, 2013)

Esta versão da distribuição Debian inclui vários recursos interessantes tais como suporte à multiarquitetura várias ferramentas específicas para implantar nuvens privadas um instalador melhorado, e um conjunto completo de *codecs* de multimídia e *front-ends* que eliminam a necessidade de repositórios de terceiros. (Debian.org)

### 3.8 Distribuição Linux Ubuntu 14.04

Ubuntu é uma antiga palavra africana que significa algo como "Humanidade para os outros" ou ainda "Sou o que sou pelo que nós somos". A distribuição Ubuntu traz o espírito desta palavra para o mundo do software livre.

Ubuntu é um sistema operacional baseado em Linux desenvolvido pela comunidade e é perfeito para notebooks, desktops e servidores. Ele contém todos os aplicativos que você precisa - um navegador Web, programas de apresentação, edição de texto, planilha eletrônica, comunicador instantâneo e muito mais. Uma nova versão do Ubuntu é lançada periodicamente a cada seis meses. Cada nova versão possui

suporte completo, incluindo atualizações de segurança pela Canonical por pelo menos 18 meses (UBUNTU.ORG 2014).

A versão 14.04 o suporte estendido (LTS - Long Term Support) garante atualizações de segurança por 5 anos a partir da data de lançamento.

### 3.9 *Dansguardian*

*Dansguardian* é uma ferramenta baseada em código livre idealizada para funcionar como filtro de conteúdo Web, podendo trabalhar de forma integrada com o *Proxy*. Permite grande flexibilidade em suas configurações proporcionando recursos simples e eficientes no que tange ao objetivo de controle de um ambiente de rede, (RICCI; MENDONÇA, 2006).

O servidor *Proxy* como um servidor intermediário que aceita pedidos de clientes e os encaminha para outros servidores *Proxys*, para o servidor de origem ou atende o pedido a partir do seu próprio cache. Um *Proxy* atua tanto como servidor e como cliente: o *Proxy* é um servidor para o cliente conectado a ele e também é um cliente quando se conecta a servidores Web. Existem servidores proxies disponíveis no mercado que permitem customizações para a realização de filtragem de conteúdo Web, (Luotenen, 1997).

O software livre *Squid* é um servidor *Proxy*, bastante popular, para conexões à Internet usando *HyperText Transfer Protocol* (HTTP), entre outras, desenvolvido para o sistema operacional do tipo Unix e Windows. Seu propósito principal é a realização de cache para redução da utilização da conexão e melhora no tempo de resposta, mas também conta com funcionalidades para controle de acesso e registro de operações.

O bloqueio de páginas Web não é a característica mais forte do *Squid*, mas pode ser feito através de um módulo chamando *SquidGuard*, também disponível como software livre. Sua configuração se dá por meio de arquivos, que são do tipo script, onde é possível definir listas com URL de sites para bloquear ou mesmo desbloquear, ou então definir palavras que serão identificadas na URL para fazer o bloqueio.

O *Squid* também oferece uma série de *Application Programming Interfaces* (APIs) para a criação de módulos. Todavia, o desenvolvimento de módulos para este servidor *Proxy* não é muito comum, possivelmente devido a escassez de informação sobre as APIs e exemplos. A instalação e configuração do *Squid* exigem conhecimentos técnicos, (Luotenen, 1997).

O filtro de conteúdo é uma maneira importante de se trabalhar com bloqueio de conteúdo não só utilizando palavras, mas também bloqueio de perfis de acesso.

Foi definido um perfil de acesso como algo próximo a um grupo de elementos que podem agrupar características comuns de conteúdos existentes em sites que se repetem em um número extremamente grande de páginas, o que inviabiliza o bloqueio de forma individualizada apenas pelo nome. Um bom exemplo são os sites pornográficos, existe uma infinidade de sites com conteúdo pornográficos e outros tantos são criados permanentemente, (Luotenen, 1997).

Inicialmente é criado um perfil do acesso proibido com base em palavras-chave comuns a este perfil. O segundo passo é bloquear este perfil. Quando o usuário faz uma requisição a uma página Web, o servidor primeiramente faz uma análise do conteúdo do site. Com base em palavras-chave e frases pontuadas contidas em seu banco de dados, o servidor determina se o site acessado possui ou não palavras proibidas. Se a quantidade de palavras proibidas ultrapassarem o limite permitido, o site é automaticamente bloqueado e uma mensagem de erro informará ao usuário do ocorrido, (Luotenen, 1997).

Originalmente o *Dansguardian* foi desenvolvido com um filtro de conteúdo para ser utilizado em escolas e para pessoas que tem crianças pequenas em casa. Porém nota-se que o filtro efetuado pelo *Dansguardian* é algo que pode ser usado em várias outras situações, (Barron, D., 2009)

O filtro inclui um conjunto de regras prontas, que contêm palavras, frases e tipos de arquivo frequentemente usados em páginas impróprias, além de uma lista de páginas conhecidas. A filtragem é feita cruzando-se todas essas informações (Barron, D., 2009).

Atualmente o *Dansguardian* é um produto comercial, que é gratuito para uso pessoal e também gratuito para uso sem fins comerciais. Pode ser utilizado em escolas, bem como em escritório, desde que utilizado em seu ambiente interno, (Barron, D., 2009).

O *Dansguardian* poderá ser utilizado em várias distribuições linux como citado acima. No caso em questão será utilizado o ambiente linux com a distribuição Debian 7.0. com arquitetura i386.

A filtragem de páginas funciona em dois níveis. Ao receber a requisição do cliente, o *Dansguardian* verifica se o endereço a ser acessado está em uma das listas de domínios ou IPs proibidos. Caso esteja, o cliente recebe a mensagem de erro e o acesso sequer é feito, economizando banda.

Se não existir nenhum bloqueio relacionado ao domínio, a requisição é liberada. Ao receber os arquivos da página, o *Dansguardian* verifica o conteúdo em busca de expressões e palavras "ruins", freqüentemente encontradas em páginas indesejadas, e também palavras "boas", normalmente encontradas em páginas de bom conteúdo.

### 3.10 ***Lightweight Directory Access Protocol - LDAP***

Segundo a OpenLDAP Foundation (2003), LDAP é uma implementação de um protocolo leve para acesso a diretórios, LDAP funciona sobre uma conexão TCP/IP ou sobre qualquer outra conexão orientada a serviços (THE OPENLDAP FOUNDATION, 2003).

“LDAP utiliza um modelo de conexão cliente-servidor, um ou mais servidorLDAP contém as informações organizadas em formato de árvore, o cliente se comunica e faz uma pergunta, o servidor responde ou direciona a um outro servidor onde poderá ser adquirido maiores informações, não importa qual servidor LDAP o usuário se comunicará, este sempre enxergará a mesma estrutura de árvore” (THE OPENLDAP FOUNDATION, 2003).

As informações da fonte de dados são armazenadas no banco de dados do LDAP são organizadas de forma similar ao DNS, feita de forma hierárquica partindo da raiz e chegando, por exemplo, até a um dispositivo de rede (ERICH, 2005).

A fundação OpenLDAP, diz que as informações no servidor LDAP são armazenadas de forma hierárquica, seguindo um estrutura parecida com árvores e este geralmente é organizada seguindo limites geográficos ou internos da organização que usa o serviço (The OPENLDAP FOUNDATION, 2003).

Um servidor LDAP é responsável pela autenticação do usuário na rede e as informações deste usuário ficam armazenadas na sua base de dados do servidor e o

mesmo permite ou não que o cliente realize consultas e modificações. Podendo este também ser utilizado como agenda de contatos, no qual foi utilizado inicialmente (ERICH, 2005).

A informação presente no servidor LDAP é protegida de acesso desautorizado usando um mecanismo de autenticação para provar a identidade do solicitante ao servidor, o LDAP também provém integridade e confiabilidade das informações (THE OPENLDAP FOUNDATION, 2003).

Devido a essa estrutura de armazenamento, o LDAP é ideal para armazenar informações que se comportam hierarquicamente como, por exemplo, informações de uma estrutura organizacional (presidência, diretorias, gerências, departamentos, supervisões...), informações sobre níveis de acesso de uma determinada aplicação (aplicação, tipo de acesso, grupo de usuários) ou até mesmo uma estrutura geográfica (países, estados, cidades, municípios...), (ALMEIDA, M. DE ABADE, 2004)

No estudo em questão utilizaremos o LDAP para autenticação de usuário, visto que, nem todos os usuários poderiam efetuar a liberação de conteúdo através do sistema de liberação automático.

### 3.11 *Hypertext Preprocessor – PHP*

A linguagem Personal Home Page (PHP), foi concebida nos Estados Unidos em 1994 por Rasmus Lerdorf, e consiste em um motor de interpretação bem simples, de código aberto, flexível e de fácil utilização, (Lerdorf, Rasmus, 2000).

Durante sua evolução a linguagem PHP apresentou duas versões PHP 3 e PHP 4 que conseguiram uma plataforma potente e estável para a programação de páginas do lado do servidor.

Estas versões serviram muito de ajuda para a comunidade de desenvolvedores, tornando possível que PHP se tornasse a linguagem mais utilizada na Web para o desenvolvimento de páginas avançadas. (Lerdorf, Rasmus, 2000).

Entretanto, mesmo com toda a estabilidade há pontos restritivos no desenvolvimento de PHP que foram tratadas com a versão 5.

Estes pontos foram solucionados através do conceito de Orientação ao objeto introduzido ao contexto do desenvolvimento Web com a versão 5.

A orientação a objetos é uma maneira de programar que trata de modelar os processos de programação de uma maneira próxima à realidade: tratando a cada componente de um programa como um objeto com suas características e funcionalidades. (Kay Alan, 1943)

Referimo- nos principalmente à programação orientada a objetos (POO) que, apesar de estar suportada a partir de PHP3, só implementava uma parte muito pequena das características deste tipo de programação

O PHP pode ser utilizado na maioria dos sistemas operacionais, também é suportado pela maioria dos servidores Web. Para o desenvolvimento e operacionalização de uma aplicação Web, basta apenas um interpretador do PHP, um servidor Web e um navegador Web.

O sistema será desenvolvido em PHP pelo fato de ser uma aplicação Web, no ambiente de rede da UTFPR-PG.

### 3.12 ***Bootstrap***

*Bootstrap* é um framework 'front-end' open-source baseado em HTML, CSS e Javascript. Foi desenvolvido pela equipe do Twitter e atualmente utilizado em diversos sites (inclusive no próprio Twitter). São muitas as facilidades que oferece, desde recursos prontos (como botões, barras de navegação, menus, painéis, etc) até temas customizáveis. A partir da versão 2 oferece nativamente a função de *design* responsivo, facilitando o uso da mesma aplicação em diferentes tipos de dispositivos e navegadores.

### 3.13 ***Cascading Style Sheet – CSS***

CSS é uma linguagem de estilo que descreve a apresentação de um documento HTML (ou XML).

CSS descreve como elementos devem ser renderizados na tela, no papel, ou em outras mídias.

Quase todos os navegadores suportam CSS hoje em dia e muitas outras aplicações de fazer, também. Para escrever CSS, você não precisa de mais do que

um editor de texto, mas existem muitas ferramentas disponíveis, que tornam ainda mais fácil.

A linguagem CSS funciona através de regras com o principal objetivo de estilizar visualmente o conteúdo HTML da página. Com as regras, podemos selecionar um ou mais elementos em uma página e aplicar estilos de acordo.

A letra "C" em CSS significa "Cascata". Isto significa que algumas das propriedades aplicadas a elementos HTML são "herdadas" por seus elementos internos (também chamados de "filhos").

A versão do CSS atual no desenvolvimento desse trabalho é a CSS3. O CSS3 difere do CSS 2.1 onde o CSS3 foi dividido em "módulos". Ele contém a "especificação CSS 2.1" (que foi dividido em pedaços menores) e também recebeu novos módulos foram adicionados.

### 3.14 HTML

*HyperText Markup Language* (HTML) é a linguagem principal de quase todo o conteúdo da Web. A maioria do que você vê na tela de seu navegador está descrita, fundamentalmente, usando HTML. Mais precisamente, o HTML é a linguagem que descreve a estrutura e o conteúdo semântico de um documento da Web. O conteúdo em uma página da Web é marcado com elementos HTML, como <img>, <title>, <p>, <div>, e assim por diante.

HTML é um padrão internacional cujas especificações são mantidas pelo World Wide Web Consortium e o WHATWG. Ele considerou "uma qualidade de vida" e é tecnicamente sempre embaixo de construções. A versão sempre-atual da especificação de HTML menciona-se como HTML5.

Neste trabalho foi utilizado a última versão do HTML no caso o HTML5. É a última versão do HTML e XHTML. O Padrão HTML define uma linguagem simples que pode ser escrita usando a sintaxe descontraída de HTML, mas também o mais rigoroso XML, e também endereços as necessidades de aplicações Web.

O HTML5 não descreve o estilo e a formatação do conteúdo, mas só o próprio conteúdo e a sua significação. O estilo e a formatação são definidos e controlaram Folhas de Estilo de Cascadeamento de utilização (CSS).

### 3.15 **Cacoo**

*Cacoo* é uma ferramenta de desenho online fácil de usar que permite criar uma variedade de diagramas como mapas de site, wireframes, UML e gráficos de rede. *Cacoo* pode ser usado de forma gratuita. *Cacoo* pode ser inserido em uma variedade de aplicações Web, como *Wikis* e *Blogs*.

Ele também permite e criar e inserir diagramas facilmente em documentos on-line. Quaisquer modificações aparecem instantaneamente, assim sempre a versão mais recente do diagrama é sempre apresentado. O utilitário online promove a colaboração e compartilhamento através de diagramas com flexibilidade, agilidade, e beleza. Os diagramas desenvolvidos podem ser exportado em formato PNG PDF EPS entre outros.

### 3.16 **Unified Modeling Language – UML**

É um acrônimo para a expressão *Unified Modeling Language*. Pela definição de seu nome, vemos que UML é uma linguagem que define uma série de artefatos que nos ajuda na tarefa de modelar e documentar os sistemas orientados a objetos que desenvolvemos (FORLAN, 1998).

A *Unified Modeling Language* (UML) é uma linguagem de modelagem não proprietária de terceira geração. A UML não é uma metodologia de desenvolvimento, o que significa que ela não diz para você o que fazer primeiro e em seguida ou como projetar seu sistema, mas ela lhe auxilia a visualizar seu desenho e a comunicação entre objetos.

Basicamente, a UML permite que desenvolvedores visualizem os produtos de seus trabalhos em diagramas padronizados. Junto com uma notação gráfica, a UML também especifica significados, isto é, semântica

É importante distinguir entre um modelo UML e um diagrama (ou conjunto de diagramas) de UML. O último é uma representação gráfica da informação do primeiro, mas o primeiro pode existir independentemente. O XMI (XML *Metadata Interchange*) na sua versão corrente disponibiliza troca de modelos, mas não de diagramas.

Os objetivos da UML são: especificação, documentação, estruturação para sub-visualização e maior visualização lógica do desenvolvimento completo de um sistema de informação. A UML é um modo de padronizar as formas de modelagem.

A UML tem origem na compilação das “melhores práticas de engenharia” que provaram ter sucesso na modelagem de sistemas grandes e complexos. Sucedeu aos conceitos de Booch, OMT (Rumbaugh) e OOSE (Jacobson) fundindo-os numa única linguagem de modelagem comum e largamente utilizada. A UML pretende ser a linguagem de modelagem padrão para modelar sistemas concorrentes e distribuídos.

A UML ainda não é um padrão da indústria, mas esse objetivo está a tomar forma sob os auspícios do Object Management Group (OMG). O OMG pediu informação acerca de metodologias orientadas a objetos que pudessem criar uma linguagem rigorosa de modelagem de software. Muitos líderes da indústria responderam na esperança de ajudar a criar o padrão.

Os esforços para a criação da UML tiveram início em outubro de 1994, quando Rumbaugh se juntou a Booch na Rational. Com o objetivo de unificar os métodos Booch e OMT, decorrido um ano de trabalho, foi lançado, em outubro de 1995, o esboço da versão 0.8 do Unified Process – Processo Unificado (como era conhecido). Nesta mesma época, Jacobson se associou à Rational e o escopo do projeto da UML foi expandido para incorporar o método OOSE. Nasceu então, em junho de 1996, a versão 0.9 da UML.

Finalmente em 1997, a UML foi aprovada como padrão pelo OMG (Object Management Group), um consórcio internacional de empresas que define e ratifica padrões na área de Orientação a Objetos.

Os diagramas previstos pela metodologia UML são definidos como de estrutura: Classe, objetos, interface, componentes, colaboração, nó. De comportamento caso de uso, iteração e de estados. De agrupamento como diagramas de pacote, modelo, subsistema, sequência.

## 4 METODOLOGIA

Podemos definir método como um apanhado de processos pelos quais é possível conhecer uma realidade, produzir determinado objeto ou desenvolver certos procedimentos ou comportamentos, (OLIVEIRA,1999).

Para abordar um assunto seguindo o método científico, devemos escolher entre procedimentos sistematizados para descrever e explicar o objeto de estudo seguindo critérios básicos, (FACHINI, 2001).

Ao se realizar uma pesquisa seguindo o método científico, pode-se optar por abordagens quantitativas ou qualitativas, embora haja autores que discordem desta divisão tão engessada, (GOODE E HATT, 1999).

A maior parte dos estudos aponta para a impossibilidade de se impor alguma supremacia entre aos métodos. Para o presente trabalho optou-se por realizar um Estudo de Caso, considerado um tipo de análise qualitativa.

Estudo de caso é uma inquirição empírica que investiga um fenômeno contemporâneo dentro de um contexto da vida real, quando a fronteira entre o fenômeno e o contexto não é claramente evidente e onde múltiplas fontes de evidência são utilizadas, (Yin, 2001).

O método do Estudo de Caso enquadra-se como uma abordagem qualitativa e é frequentemente utilizado para coleta de dados na área de estudos organizacionais, considerando que caracteriza-se pela objetividade e rigor, (Goode,1969).

No cenário que pretende-se elaborar o estudo proposto, o ambiente acadêmico representado pela UTFPR-PG, caracteriza-se por, não só permitir o uso do método estudo de caso, como também se mostrar extremamente pertinente o uso de tal método, uma vez que, segundo Yin (2001) o método do Estudo de Caso é adequado quando são propostas questões de pesquisa do tipo “por que” e “como”, e exatamente essas respostas que pretendemos oferecer ao final do trabalho.

Primeiramente “por que” é importante para a instituição o desenvolvimento de um sistema que automatize o controle de conteúdos acessados no ambiente de rede da UTFPR-PG, e “como” será desenvolvida a solução para a situação atual de gerenciamento de sites.

O desenvolvimento de aplicações é um tipo de pesquisa bastante comum que apresenta soluções para determinado problema organizacional já diagnosticado. (UCHÔA & UCHÔA, 2007).

Na UTFPR-PG o problema de bloqueio indevido de conteúdo foi diagnosticado e o desenvolvimento de uma ferramenta para automatizar sua liberação é extremamente relevante ao ambiente educacional.

## 5 ANÁLISE DOS RESULTADOS

### 5.1 Funcionamento

Ao realizar uma tentativa de acesso a um conteúdo Web, a requisição é direcionada ao *Dansguardian*, que permite a saída do pedido até o destino. Quando essa requisição retorna, de acordo com as regras estabelecidas o conteúdo é liberado ou bloqueado pelo *Dansguardian*.

Ao bloquear o conteúdo o *Dansguardian* exibe uma tela Figura 1, mostrando ao usuário quais motivos do bloqueio.



Figura 1: Tela de Bloqueio

Fonte: D BARON (2009)

O sistema proposto neste estudo permitirá ao usuário da rede da UTFPR-PG, caso considere o conteúdo que tentou ter acesso ser relevante e necessário ao

desenvolvimento de suas atividades ou estudos, clicar no campo “Solicitar Liberação”.

O sistema o direcionará ao módulo de autenticação Figura 2, este módulo faz uma verificação dos usuários cadastrados através da base de dados do serviço LDAP, e o direciona de acordo com os níveis de privilégios estabelecidos pelo sistema aos módulos seguintes. Os níveis de privilégios e seus respectivos acessos são:

Aluno: Acesso a solicitação de site e ao histórico de solicitações por usuário

Professor: Acesso a liberação de site e aos mesmos acessos do grupo Aluno

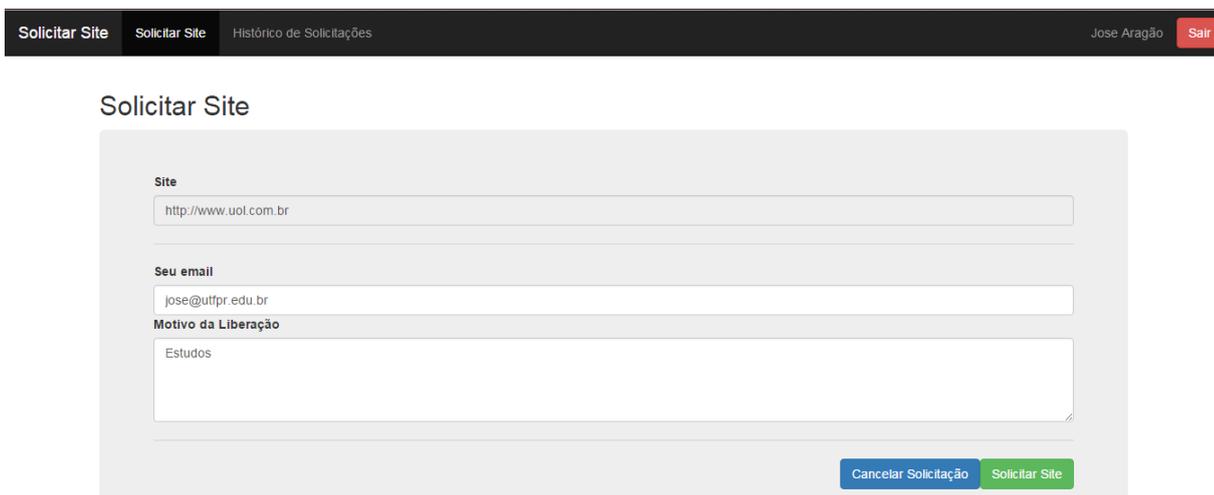
Adm: Acesso aos Logs e a Configuração do Sistema além dos acessos do grupo Aluno e Professor



A imagem mostra uma interface de usuário para login. No topo, o texto "Faça seu login" está centralizado. Abaixo dele, há dois campos de entrada de texto empilhados: o primeiro é rotulado "Login" e o segundo "Senha". Abaixo dos campos, há um botão azul com o texto "Entrar" em branco.

**Figura 2: Tela de login**  
**Fonte: Autoria própria**

Ao ser realizada a autenticação, independente do nível de acesso estabelecido ao usuário, o sistema direcionará até a tela de solicitação de site, Figura 3. Nesse processo ocorre a inclusão dessa solicitação no banco de dados e a requisição fica marcada como “pendente”.



The screenshot shows a web application interface with a dark header. The header contains navigation links: 'Solicitar Site', 'Solicitar Site', and 'Histórico de Solicitações'. On the right side of the header, the user's name 'Jose Aragão' and a 'Sair' button are visible. The main content area is titled 'Solicitar Site' and contains a form with three input fields: 'Site' (containing 'http://www.uol.com.br'), 'Seu email' (containing 'jose@utfpr.edu.br'), and 'Motivo da Liberação' (containing 'Estudos'). At the bottom right of the form, there are two buttons: 'Cancelar Solicitação' (blue) and 'Solicitar Site' (green).

**Figura 3: Tela de Solicitação**  
Fonte: Autoria própria

Feito a solicitação do site, o usuário será redirecionado para a tela de confirmação de solicitação, Figura 4. O usuário receberá um e-mail contendo a confirmação dessa solicitação.



The screenshot shows a confirmation screen with a dark header. The header contains navigation links: 'Confirmação de Solicitação' and 'Histórico de Solicitações'. On the right side of the header, the user's name 'Jose Aragão' and a 'Sair' button are visible. The main content area is titled 'Confirmação de Solicitação' and contains a message: 'O site http://www.uol.com.br foi solicitado com sucesso. Foi enviado para o e-mail jose@utfpr.edu.br o número de acompanhamento nº: 7. Aguarde a resposta diretamente em seu e-mail, ou acompanhe o andamento no menu 'Histórico de Solicitações'. Obrigado!'. At the bottom right of the message, there is a 'Sair' button.

**Figura 4: Tela de Confirmação**  
Fonte: Autoria própria

O usuário ainda poderá entrar no sistema e verificar o seu histórico através da tela de Histórico de Solicitações, Figura 5, onde aparecem todos os sites que foram solicitados pelo mesmo, separados em três diferentes painéis, Sites Pendentes, Sites Negados e Sites Liberados. Essa tela é chamada automaticamente quando qualquer usuário (independentemente do grupo) acessar o sistema diretamente (sem clicar no link Solicitar Liberação na tela de bloqueio).

Histórico de Solicitações de Jose Aragão

Sites Pendentes			
Nº Acompanhamento	Site	Solicitado por	Solicitado em
1	www.globo.com	Jose Aragão	24-10-2015 21:39:32
4	globo.com	Jose Aragão	25-10-2015 15:02:54
7	http://www.uol.com.br	Jose Aragão	26-10-2015 19:16:43

Sites Negados					
Nº Acompanhamento	Site	Solicitado por	Solicitado em	Negado por	Negado em
3	www.globo.com	Jose Aragão	24-10-2015 21:39:32	Wellington Wilczak Pereira	24-10-2015 21:39:32

Sites Liberados						
Nº Acompanhamento	Site	Solicitado por	Solicitado em	Liberado por	Liberado em	Liberado até
2	www.globo.com	Jose Aragão	24-10-2015 21:39:32	Wellington Wilczak Pereira	24-10-2015 21:39:32	Tese

**Figura 5: Tela de Histórico do Usuário**  
**Fonte: Autoria própria**

O Professor e o Administrador terão o acesso a Liberação de Site, para tal operação deve-se clicar no site pendente de autorização na tela de lista de sites, Figura 6.

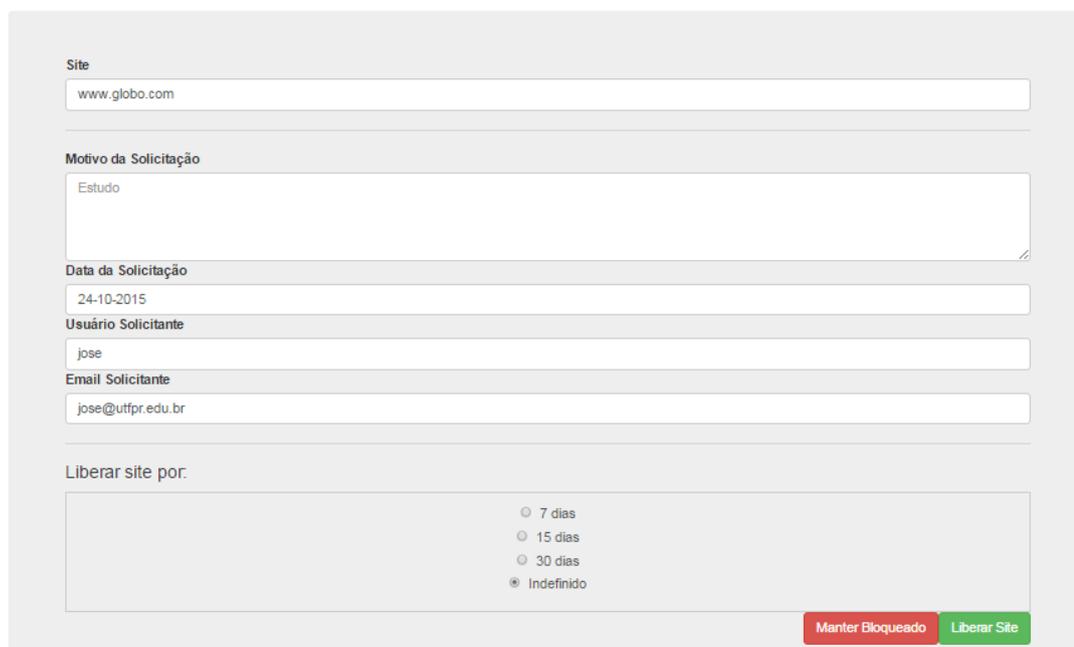
Sites pendentes de autorização

Foi solicitado o site: www.globo.com Motivo: Estudo Solicitado por: jose Solicitado em: 24-10-2015 21:39:32
Foi solicitado o site: www.globo.com Motivo: Estudo Solicitado por: jose Solicitado em: 24-10-2015 21:39:32

**Figura 6: Tela de Lista de Sites Pendentes**  
**Fonte: Autoria própria**

Ao clicar no site disponível para liberação, o usuário será direcionado a página de liberação de site, Figura 7, onde poderá ser mantido o bloqueio do site ou liberado por determinado período (7, 15, 30 dias ou período indefinido). Qualquer das decisões ocasionará em um envio de e-mail para o usuário solicitante e a inclusão da liberação do site diretamente no *Dansguardian*.

#### Liberar Site



O formulário 'Liberar Site' contém os seguintes campos e opções:

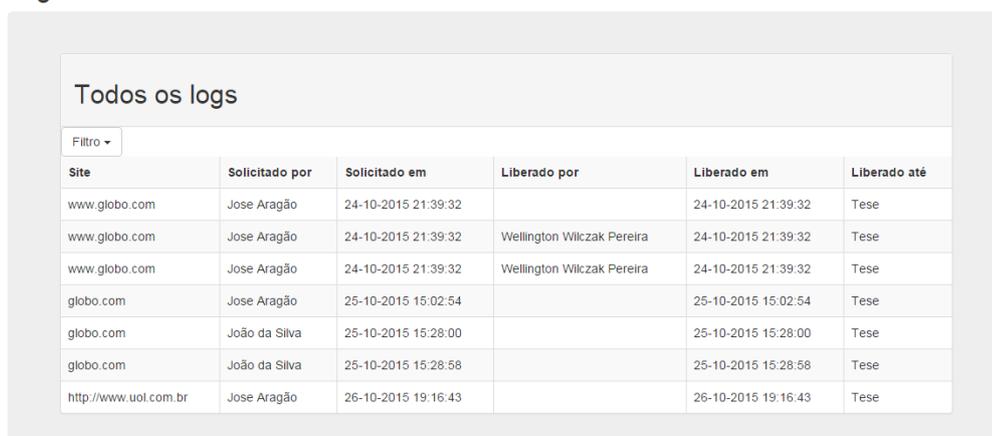
- Site:** Campo de texto com o valor 'www.globo.com'.
- Motivo da Solicitação:** Campo de texto com o valor 'Estudo'.
- Data da Solicitação:** Campo de texto com o valor '24-10-2015'.
- Usuário Solicitante:** Campo de texto com o valor 'jose'.
- Email Solicitante:** Campo de texto com o valor 'jose@utfpr.edu.br'.
- Liberar site por:** Grupo de opções de radio button:
  - 7 dias
  - 15 dias
  - 30 dias
  - Indefinido
- Botões:** 'Manter Bloqueado' (em vermelho) e 'Liberar Site' (em verde).

**Figura 7: Tela de Liberação de Site**  
**Fonte: Autoria própria**

O administrador de rede terá acesso aos Logs de Liberação e a tela de Configuração do Sistema.

Na tela de *Logs*, Figura 8, o mesmo terá a opção de mostrar todos os sites, mostrar somente sites pendentes de liberação, somente sites negados ou somente sites liberados. Também ficará disponível para visualização o IP do *host* de onde foi solicitada a liberação e de onde foi liberado o site.

#### Logs de Liberacao

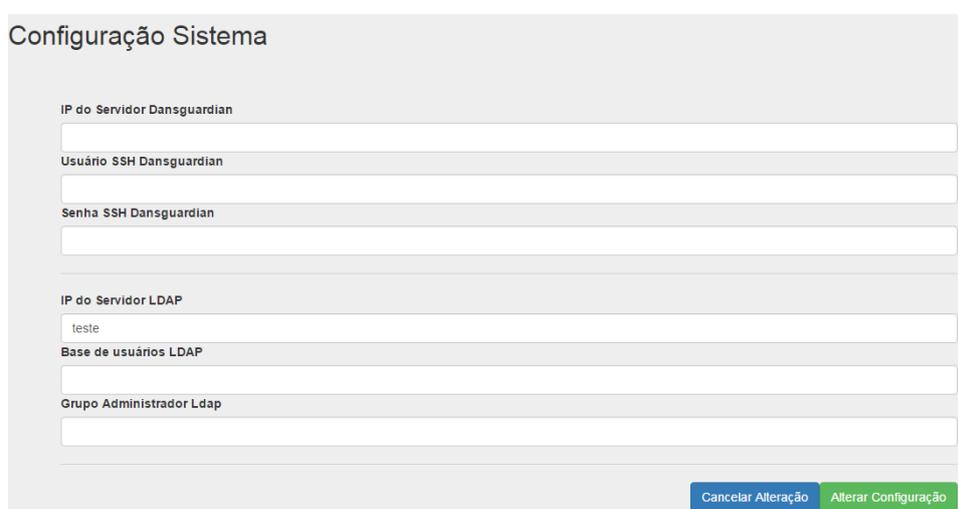


The screenshot shows a web interface titled 'Logs de Liberacao'. At the top, there is a section 'Todos os logs' with a 'Filtro' dropdown menu. Below this is a table with the following columns: Site, Solicitado por, Solicitado em, Liberado por, Liberado em, and Liberado até. The table contains seven rows of log entries.

Site	Solicitado por	Solicitado em	Liberado por	Liberado em	Liberado até
www.globo.com	Jose Aragão	24-10-2015 21:39:32		24-10-2015 21:39:32	Tese
www.globo.com	Jose Aragão	24-10-2015 21:39:32	Wellington Wilczak Pereira	24-10-2015 21:39:32	Tese
www.globo.com	Jose Aragão	24-10-2015 21:39:32	Wellington Wilczak Pereira	24-10-2015 21:39:32	Tese
globo.com	Jose Aragão	25-10-2015 15:02:54		25-10-2015 15:02:54	Tese
globo.com	João da Silva	25-10-2015 15:28:00		25-10-2015 15:28:00	Tese
globo.com	João da Silva	25-10-2015 15:28:58		25-10-2015 15:28:58	Tese
http://www.uol.com.br	Jose Aragão	26-10-2015 19:16:43		26-10-2015 19:16:43	Tese

**Figura 8: Tela de Logs**  
**Fonte: Autoria própria**

O usuário Administrador também possui acesso a tela de configuração do sistema, Figura 9, onde deverão ser inserido os dados para correto funcionamento. Exemplos de configurações inseridas nessa tela são: IPs dos servidores onde ocorrerá os acessos (*Dansguardian* e OpenLDAP), usuário e senha de acesso ao serviço SSH do *Dansguardian*. Base onde estão armazenados os usuários do LDAP, grupo administrativo e grupo de professores do LDAP. Além de configurações inerentes à conta de envio dos e-mails (servidor de saída, porta de saída, usuário, senha).



Configuração Sistema

IP do Servidor Dansguardian

Usuário SSH Dansguardian

Senha SSH Dansguardian

IP do Servidor LDAP

teste

Base de usuários LDAP

Grupo Administrador Ldap

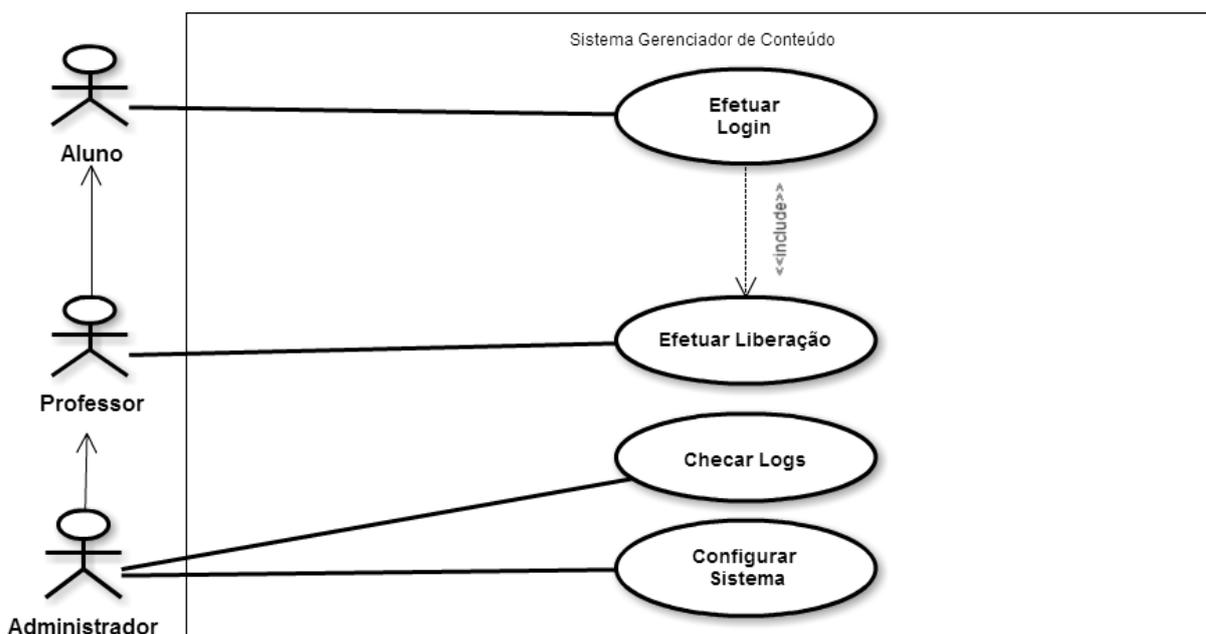
Cancelar Alteração Alterar Configuração

**Figura 9: Tela de Configuração do Sistema**  
Fonte: Autoria própria

### 5.1.1 Diagramas de caso de uso e fluxograma

Descrevendo as principais funcionalidades do sistema e suas interações com os usuários.

### 5.1.2 Visão Geral



**Figura 10: Visão geral**  
Fonte: Autoria própria

Descrição:

Nome: Visão Geral
Descrição Sucinta: Funcionamento elaborado a partir da ação dos três atores (Aluno, Professor e Administrador) no fluxo do sistema.
Atores: Aluno, Professor e Administrador.
Pré-Condições: Bloqueio de conteúdo pelo <i>Dansguardian</i>
Fluxo Básico: <ol style="list-style-type: none"> <li>1- O usuário tenta acessar um conteúdo Web através de um browser no ambiente de rede da UTFPR-PG</li> <li>2- O <i>Dansguardian</i> bloqueia o conteúdo acessado e exibe uma mensagem ao usuário</li> </ol>

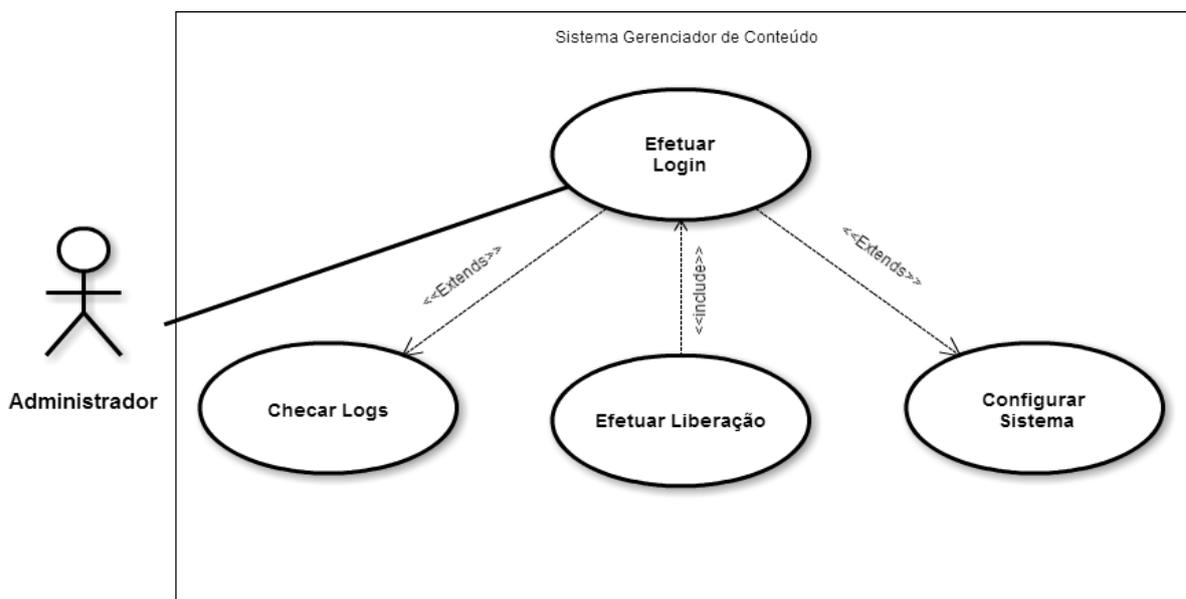
- 3- O usuário clica na opção de liberação exibida na tela de mensagem do *Dansguardian*
- 4- O sistema exibe a tela de login para acesso ao sistema
- 5- O sistema envia uma mensagem eletrônica ao solicitante sobre sua solicitação.
- 6- O sistema envia os dados da solicitação ao módulo de controle e auditoria interna de acesso exclusivo do administrador.
- 7- Caso de uso é encerrado

Fluxo alternativo:

Alternativa ao passo 4

- A- No caso de o ator ser um aluno, o sistema apresenta a tela de solicitação de liberação.
  - A1- O sistema exibe uma mensagem de feedback ao aluno informando status (Sucesso ou Insucesso).
- B- No caso de tratar-se de um professor, o sistema exibe a tela de liberação do conteúdo, acontece à liberação e as informações são enviadas ao módulo de controle e auditoria.

### 5.1.3 Ações do Administrador



**Figura 11: Ações do Administrador**  
 Fonte: Autoria própria

Descrição:

Nome: Ações do Administrador
Descrição Sucinta: Descreve as ações exclusivas do administrador do Sistema
Atores: Administrador
Pré-Condições: Ser o usuário administrador do sistema
Fluxo Principal: <ol style="list-style-type: none"> <li>1- O administrador efetua login no sistema.</li> <li>2- O administrador verifica as informações pertinentes às liberações de conteúdo efetuadas pelos usuários dotados de permissão para executar liberações.</li> <li>3- O administrador do sistema efetua liberações solicitadas pelos usuários sem permissão para executar liberações.</li> <li>4- O administrador configura o sistema seguindo critérios de necessidade e segurança.</li> </ol>
Fluxo Básico: <ol style="list-style-type: none"> <li>1- O usuário administrador do sistema efetua o login.</li> <li>2- O sistema apresenta a tela de configurações.</li> <li>3- O administrador acessa o módulo de controle e auditoria.</li> </ol>

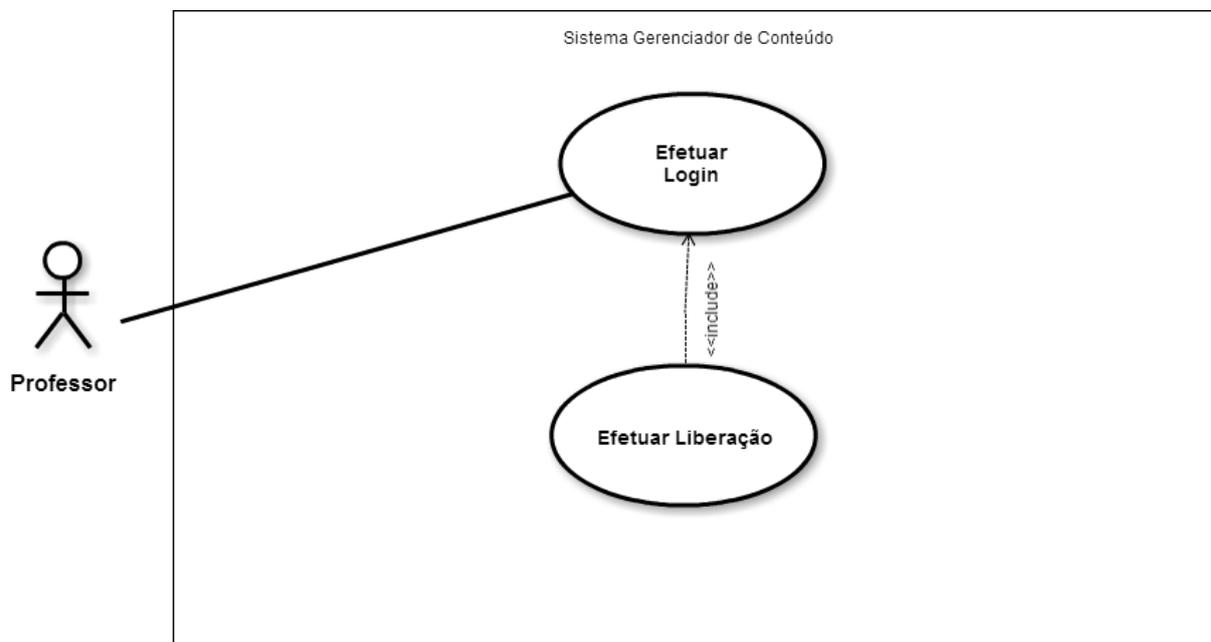
4- O administrador acessa a tela de liberação.

Fluxo alternativo:

Alternativa ao passo 3:

- A- No passo 3 o administrador verifica as informações de log conforme as políticas de liberações estabelecidas.
- B- No passo 3 o administrador faz uma análise dos sites pendentes e efetua ou veta a liberação.

#### 5.1.4 Ações do Professor



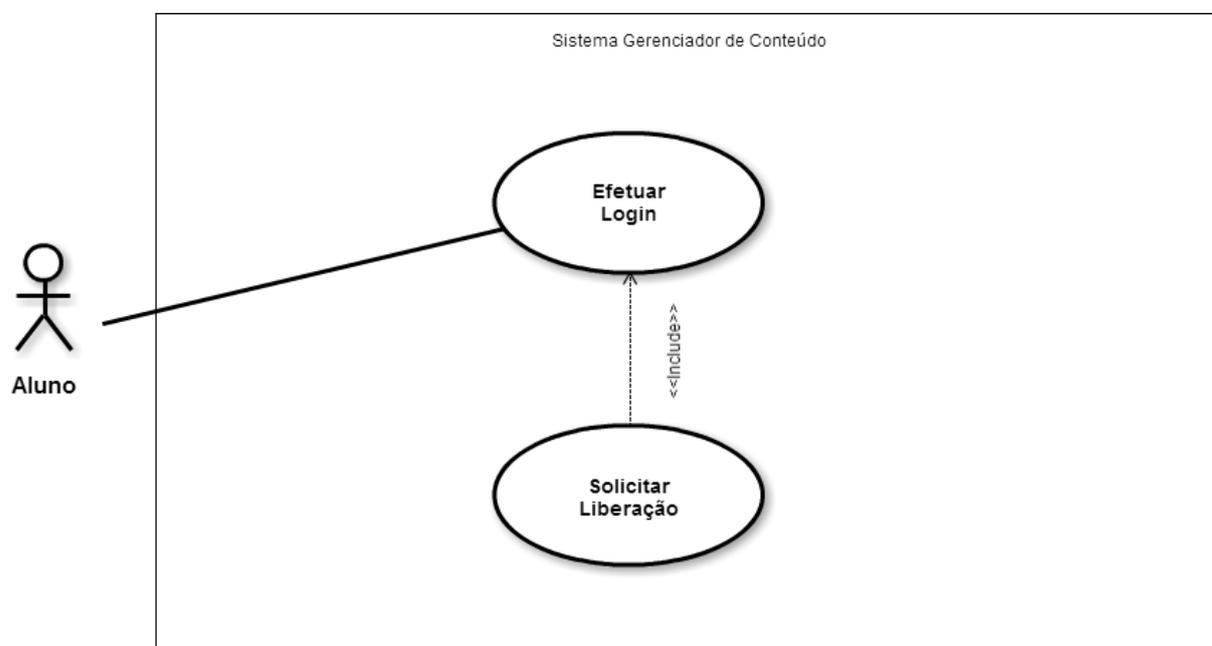
**Figura 12: Ações do Professor**  
Fonte: Autoria própria

Descrição:

Nome: Ações do Professor

Descrição Sucinta: Descrever ações do professor
Atores: Professor
Pré-requisitos: Ser professor da UTFPR-PG
Fluxo Básico: O professor acessa um conteúdo dentro do ambiente de rede da UTFPR-PG O conteúdo é bloqueado pelo Dansguardian A tela de bloqueio é exibida com a opção de clicar na liberação O professor clica na tecla de solicitar liberação A tela de login é exibida A tela de liberação é exibida O ator libera o conteúdo solicitado Uma mensagem de status é exibida (sucesso ou fracasso) O sistema envia uma mensagem eletrônica para o endereço de e-mail do ator Fim do caso de uso.

### 5.1.5 Login do Aluno



**Figura 13: Login do Aluno**  
**Fonte: Autoria própria**

Descrição:

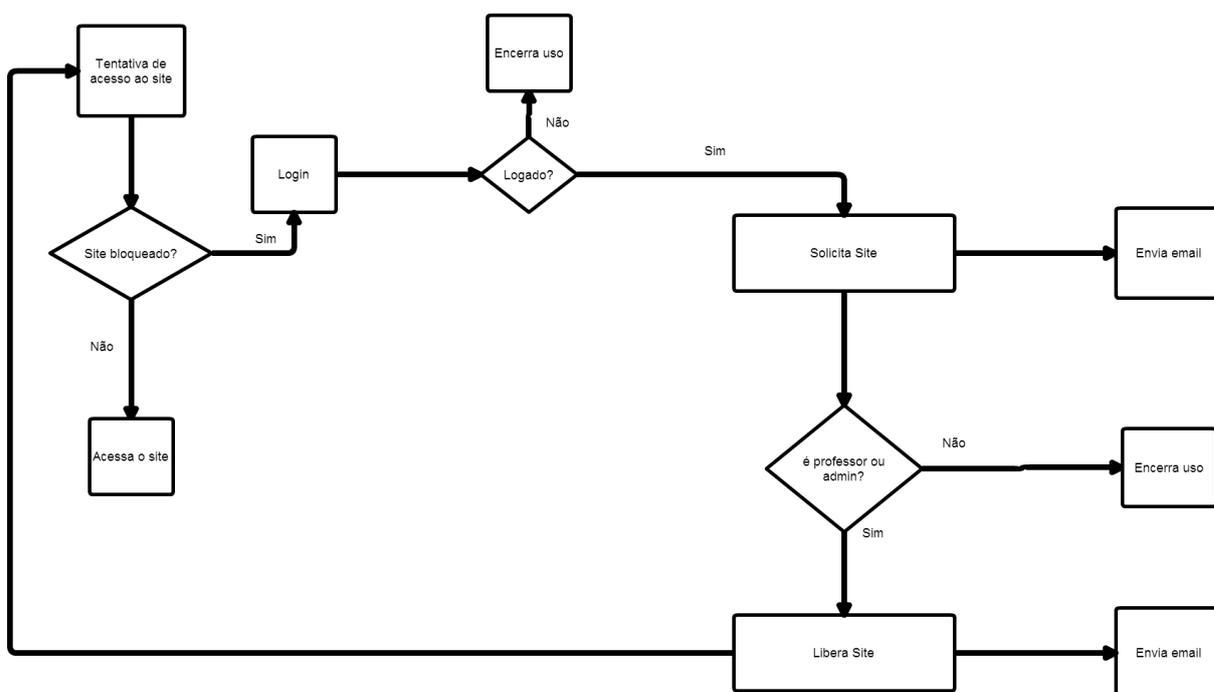
Nome: Login do aluno

Descrição Sucinta: Ação de efetuar o login por parte do ator aluno
Atores: Aluno
Pré-requisitos: Conteúdo bloqueado pelo Dansguardian
Fluxo Básico: O ator tem um conteúdo bloqueado pelo Dansguardian. A tela de bloqueio é apresentada com a opção de liberação de página A tela de login é exibida O sistema exibe a tela de solicitação de conteúdo O sistema exibe a mensagem de feedback.
Fluxo alternativo: Fluxo alternativo ao passo 2 O ator desiste do acesso e fecha a tela de bloqueio do Dansguardian.

### 5.1.6 Fluxograma

Descrevendo o fluxo de sequencia das atividades do sistema

## Fluxograma



## 6 CONCLUSÃO

Quando pretende-se desenvolver uma ferramenta computacional para resolução de algum problema, o objetivo é tornar a administração de um recurso mais rápida e simples, com menores possibilidades de erros e ao mesmo tempo proporcionar maior versatilidade.

Neste sentido o presente trabalho objetivou a elaboração de um sistema prático e automatizado para a liberação de conteúdos bloqueados indevidamente, tornando o acesso à rede da UTFPR-PG mais seguro e eficiente, cumprindo o papel ao qual se destina - permitir o livre acesso a conteúdos inerentes a pesquisas e administração da instituição, e ao mesmo tempo impedir desvios de finalidade na utilização dos ativos de rede.

Para cumprir tal tarefa a rede da UTFPR-PG utiliza-se do *Dansguardian*, ferramenta que filtra o conteúdo proveniente da Internet a partir de regras pré-estabelecidas pelo administrador.

A ferramenta desenvolvida neste trabalho de conclusão de curso produziu uma interface Web, capaz de integrar via LDAP, os usuários com permissão de liberação de conteúdo junto ao *Dansguardian*, de forma automática e segura, além de permitir a usuários não autorizados a liberar imediatamente conteúdo bloqueado a fazer solicitações de liberação que será gravado pelo sistema e disponibilizado ao administrador de redes para análise e possível liberação subsequente.

A ferramenta desenvolvida neste projeto mostrou-se satisfatória para os fins aos quais se destina. Conseguiu integrar junto ao LDAP usuários considerados autorizados a realizar liberação de conteúdo de forma imediata, além de permitir que os demais usuários que fazem parte da mesma base, mas que não possuem autorização imediata de liberação realize solicitações para posterior.

Todas as liberações imediatas e solicitações de liberação ficam gravadas em um arquivo para análise e consulta - log de liberação e solicitação, e neste módulo inserido no sistema que vai proporcionar um controle mais eficiente, no processo de auditoria realizado periodicamente pelo setor de Tecnologia da informação.

Enfim, o sistema mostrou-se capaz de auxiliar na questão de bloqueio indevido de conteúdos (falso positivos), agilizando o processo de desbloqueio, tornando o processo de gerenciamento de conteúdos Web mais participativo e

controlado, consubstanciando-se em um sistema passível de ser utilizado imediatamente dentro do ambiente acadêmico.

## 6.1 Trabalhos Futuros

O que se sugere após o desenvolvimento do sistema de gerenciamento, que ele possa ser utilizado em outros ambientes não limitando-se ao meio institucional da UTFPR Campus Ponta Grossa.

Poderá ser inserido no meio empresarial e efetuado a eficiência do mesmo, se caso possa ou não corresponder a demanda de gerência de segurança nas redes corporativas. Devido a ser um sistema utilizado com ferramentas *open-source* poderá ser aprimorado e inserido novas funcionalidades.

Esse sistema não se finda em si mesmo, ou seja, poderá ser colocado em produção. Porém será necessário efetuar avaliação se o mesmo corresponde ao uso e se necessita de aprimoramento para que tenha a interface mais amigável. Também poderá ser acrescentada configuração para conexão com *Proxy*.

## REFERÊNCIAS

ALMEIDA, M. DE ABADE - **LDAP, Integração de bases de autenticação**. 2004. Tese de Doutorado. Universidade Federal de Lavras.

CACOO. **Easy collaborative diagramming**. Disponível em: <[https://cacoo.com/lang/pt\\_br/?ref=header](https://cacoo.com/lang/pt_br/?ref=header)>. Acesso em: 03 de outubro de 2015.

EMMACK, Donald. "**Add Web porn filtering and other content filtering to Linux desktops**." Linux Journal 2006.151 (2006): 9.

FRACHIN, Odília. **Fundamentos de metodologia**. São Paulo: saraiva. 2001.

FURLAN, José Davi. **Modelagem de objetos através da UML-the unified modeling language**. Makron books, 1998.

GOODE, W. J. & HATT, P. K. - **Métodos de pesquisa**. Cia Editora Nacional, SP, 19969, 3ª ed.

Lerdorf, Rasmus. **PHP Pocket Reference**. O'Reilly & Associates:1<sup>st</sup> edition January 2000. 120 pages.

Lin, Po-Ching, "**An early decision algorithm to accelerate Web content filtering**." Information Networking. **Advances in Data Communications and Wireless Networks**. Springer Berlin Heidelberg, 2006. 833-841.

Lin, Ying-Dar, "**Designing an integrated architecture for network content security gateways**." Computer 11 (2006): 66-72.

OLIVEIRA, Silvio Luiz de. **Tratado de metodologia científica**. São Paulo: Pioneira. 997.

SANTOS, Maximiller dos. **Uso de LDAP implementado em software livre para integrar a autenticação dos controladores de domínio MS-Active Directory e Samba/Linux.** 2014.

SCRIMGER.R.; LASALLE.P.; PARIHAR.M.; GUPTA.M. **TCP/IP - A Bíblia.** Rio de Janeiro: Campus, 2002. p.642.

Sermersheim, Jim. "**Lightweight Directory Access Protocol (LDAP): The protocol.**" (2006).

Silva, Alexsandro Queiroz da. "**Implantação de servidor Proxy utilizando o Squid em modo não autenticado, junto com o Dansguardian para controle de conteúdo na instituição de ensino.**" (2015).

SMITH, Ellen. **Securely Implementing LDAP.** SANS Institute, July, 2001.

TANENBAUM, A. S. – **Redes de Computadores – 4ª Ed.,** Editora Campus (Elsevier), 2003.

LUOTONEN, A. **Web Proxy Servers.** Prentice Hall, 1997.

THE OPENLDAP FOUNDATION. **OpenLDAP 2.4 Administrator's Guide.** Disponível em: <<http://www.openldap.org/doc/admin24/OpenLDAP-Admin-Guide.pdf>>. Acesso em: 03 de outubro de 2015.

TRIGO, H. Clodonil. **OpenLDAP – Uma Abordagem Integrada.** São Paulo: Novatec, 2007. VIACONNECT, Autenticação OpenLDAP vs Microsoft Active Directory. 2012. Disponível em:<[http://www.viaconnect.com.br/catalogo/catalogo\\_active\\_openldap.PDF](http://www.viaconnect.com.br/catalogo/catalogo_active_openldap.PDF)> Acesso em: 15 de outubro de 2015.

YIN, Robert K. **applications of case study reserach**. Thousand Oaks, california: Sage Publications. 2001.

ZEILENGA, Kurt. ***Lightweight Directory Access Protocol (Idap): Technical specification road map***. 2006.

Campos, Augusto. **"O que é uma distribuição Linux."** BR-Linux. Florianópolis, março de (2006).

APÊNDICE A – Fragmento do arquivo de configuração do *Dansguardian*, contendo a lista de conteúdos banidos, de forma padrão.

```
#domains in banned list
#Don't bother with the www. or the http://

#The bannedurllist is for blocking PART of a site
#The bannedsitelist is for blocking ALL of a site

#As of DansGuardian 2.7.3 you can now include
#.tld so for example you can match .gov for example

#The 'grey' lists override the 'banned' lists.
#The 'exception' lists override the 'banned' lists also.
#The difference is that the 'exception' lists completely switch
#off *all* other filtering for the match. 'grey' lists only
#stop the URL filtering and allow the normal filtering to work.

#An example of grey list use is when in Blanket Block (whitelist)
#mode and you want to allow some sites but still filter as normal
#on their content

#Another example of grey list use is when you ban a site but want
#to allow part of it.

#To include additional files in this list use this example:
#.Include</etc/dansguardian/anotherbannedurllist>

#You can have multiple .Includes.

# Time limiting syntax:
# #time: <start hour> <start minute> <end hour> <end minute> <days>
# Example:
##time: 9 0 17 0 01234
# Remove the first # from the line above to enable this list only from
# 9am to 5pm, Monday to Friday.

# List categorisation
#listcategory: "Banned Sites"

#List other sites to block:

badboys.com

#Blanket Block. To block all sites except those in the
#exceptionsitelist and greysitelist files, remove
#the # from the next line to leave only a '***':
#**

#Blanket SSL/CONNECT Block. To block all SSL
#and CONNECT tunnels except to addresses in the
#exceptionsitelist and greysitelist files, remove
#the # from the next line to leave only a '**s':
```

Apêndice B – Fragmento do arquivo de configuração do *Dansguardian* contendo a configuração com os sites banidos

```
#domains in banned list
#Don't bother with the www. or the http://

#The bannedurllist is for blocking PART of a site
#The bannedsitelist is for blocking ALL of a site

#As of DansGuardian 2.7.3 you can now include
#.tld so for example you can match .gov for example

#The 'grey' lists override the 'banned' lists.
#The 'exception' lists override the 'banned' lists also.
#The difference is that the 'exception' lists completely switch
#off *all* other filtering for the match. 'grey' lists only
#stop the URL filtering and allow the normal filtering to work.

#An example of grey list use is when in Blanket Block (whitelist)
#mode and you want to allow some sites but still filter as normal
#on their content

#Another example of grey list use is when you ban a site but want
#to allow part of it.

#To include additional files in this list use this example:
#.Include</etc/dansguardian/anotherbannedurllist>

#You can have multiple .Includes.

# Time limiting syntax:
# #time: <start hour> <start minute> <end hour> <end minute> <days>
# Example:
##time: 9 0 17 0 01234
# Remove the first # from the line above to enable this list only from
# 9am to 5pm, Monday to Friday.

# List categorisation
#listcategory: "Banned sites"

#List other sites to block:

badboys.com
clickjogos.com.br
steam.com
origin.com
xvideos.com
netcartas.com.br

#Blanket Block. To block all sites except those in the
#exceptionsitelist and greysitelist files, remove
#the # from the next line to leave only a '***':
#**
```

APÊNDICE C – Fragmento do arquivo de configuração do *Dansguardian*, contendo a lista de conteúdos liberados, de forma padrão.

```
#Sites in exception list
#Don't bother with the www. or
#the http://
#
#These are specifically domains and are not URLs.
#For example 'foo.bar/porn/' is no good, you need
#to just have 'foo.bar'.
#
#You can also match IPs here too.
#
#As of DansGuardian 2.7.3 you can now include
#.tld so for example you can match .gov for example

# Time limiting syntax:
# #time: <start hour> <start minute> <end hour> <end minute> <days>
# Example:
##time: 9 0 17 0 01234
# Remove the first # from the line above to enable this list only from
# 9am to 5pm, Monday to Friday.

# Blanket exception. To allow all sites except those in the
# exceptionsitelist and greysitelist files, remove
# the # from the next line to leave only a '***':
#**

# Blanket SSL/CONNECT exception. To allow all SSL
# and CONNECT tunnels except to addresses in the
# exceptionsitelist and greysitelist files, remove
# the # from the next line to leave only a '**s':
#**s

# Blanket IP exception. To allow all sites specified only as an IP,
# remove the # from the next line to leave only a '*ip':
#*ip

# Blanket SSL/CONNECT IP exception. To allow all SSL and CONNECT
# tunnels to sites specified only as an IP,
# remove the # from the next line to leave only a '*ips':
#*ips

windowsupdate.microsoft.com|
```

Apêndice D – Fragmento do arquivo de configuração do *Dansguardian* contendo a configuração com os sites liberados

```
#Sites in exception list
#Don't bother with the www. or
#the http://
#
#These are specifically domains and are not URLs.
#For example 'foo.bar/porn/' is no good, you need
#to just have 'foo.bar'.
#
#You can also match IPs here too.
#
#As of Dansguardian 2.7.3 you can now include
#.tld so for example you can match .gov for example

# Time limiting syntax:
# #time: <start hour> <start minute> <end hour> <end minute> <days>
# Example:
##time: 9 0 17 0 01234
# Remove the first # from the line above to enable this list only from
# 9am to 5pm, Monday to Friday.

# Blanket exception. To allow all sites except those in the
# exceptionsitelist and greysitelist files, remove
# the # from the next line to leave only a '***':
#**

# Blanket SSL/CONNECT exception. To allow all SSL
# and CONNECT tunnels except to addresses in the
# exceptionsitelist and greysitelist files, remove
# the # from the next line to leave only a '***s':
#**s

# Blanket IP exception. To allow all sites specified only as an IP,
# remove the # from the next line to leave only a '*ip':
#*ip

# Blanket SSL/CONNECT IP exception. To allow all SSL and CONNECT
# tunnels to sites specified only as an IP,
# remove the # from the next line to leave only a '*ips':
#*ips

windowsupdate.microsoft.com
utfpr.edu.br
globo.com
google.com.br
uol.com.br
facebook.com
```

## Apêndice E – Script criação da base de dados

```
1 SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
2 SET time_zone = "+00:00";
3
4
5 /*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
6 /*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
7 /*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
8 /*!40101 SET NAMES utf8 */;
9
10 --
11 -- Base de Dados: `dansguardian`
12 --
13 CREATE DATABASE IF NOT EXISTS `dansguardian` DEFAULT CHARACTER SET utf8 COLLATE utf8_general_ci;
14 USE `dansguardian`;
15
16 -----
17
18 --
19 -- Estrutura da tabela `config`
20 --
21
22 CREATE TABLE IF NOT EXISTS `config` (
23   `ip_dans` varchar(15) NOT NULL,
24   `user_dans` varchar(100) NOT NULL,
25   `pass_dans` varchar(100) NOT NULL,
26   `ip_ldap` varchar(15) NOT NULL,
27   `base_ldap` varchar(50) NOT NULL,
28   `grupo_admin_ldap` varchar(100) NOT NULL
29 ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
30
31 -----
32
33 --
34 -- Estrutura da tabela `site`
35 --
```

```
24 `user_dans` varchar(100) NOT NULL,  
25 `pass_dans` varchar(100) NOT NULL,  
26 `ip_ldap` varchar(15) NOT NULL,  
27 `base_ldap` varchar(50) NOT NULL,  
28 `grupo_admin_ldap` varchar(100) NOT NULL  
29 ) ENGINE=InnoDB DEFAULT CHARSET=utf8;  
30  
31 -----  
32  
33 --  
34 -- Estrutura da tabela `site`  
35 --  
36  
37 CREATE TABLE IF NOT EXISTS `site` (  
38 `id_solicitacao` int(11) NOT NULL AUTO_INCREMENT,  
39 `url` varchar(100) NOT NULL,  
40 `motivo` text NOT NULL,  
41 `usuario_solicitacao` varchar(100) NOT NULL,  
42 `nome_usuario_solici` varchar(250) NOT NULL,  
43 `email_usuario_solicitaca` varchar(100) NOT NULL,  
44 `dt_solicitacao` datetime NOT NULL,  
45 `ip_solicitacao` varchar(15) NOT NULL,  
46 `usuario_liberacao` varchar(100) DEFAULT NULL,  
47 `nome_usuario_libera` varchar(250) DEFAULT NULL,  
48 `dt_liberacao` datetime DEFAULT NULL,  
49 `dias_liberados` int(11) DEFAULT NULL,  
50 `status` int(11) NOT NULL,  
51 PRIMARY KEY (`id_solicitacao`)  
52 ) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=7 ;  
53  
54 /*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;  
55 /*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;  
56 /*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;  
57
```