



Universidade de Brasília - UnB  
Faculdade UnB Gama - FGA  
Engenharia de Software

# **Evoluindo uma plataforma de rede de colaboração social para um ambiente de aprendizagem virtual**

Autor: Hebert Douglas de Almeida Santos  
Orientador: Prof. Dr. Paulo Roberto Miranda Meirelles

Brasília, DF  
2015





Hebert Douglas de Almeida Santos

**Evoluindo uma plataforma de rede de colaboração social  
para um ambiente de aprendizagem virtual**

Monografia submetida ao curso de graduação em (Engenharia de Software) da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em (Engenharia de Software).

Universidade de Brasília - UnB

Faculdade UnB Gama - FGA

Orientador: Prof. Dr. Paulo Roberto Miranda Meirelles

Brasília, DF

2015

---

Hebert Douglas de Almeida Santos

Evoluindo uma plataforma de rede de colaboração social para um ambiente de aprendizagem virtual/ Hebert Douglas de Almeida Santos. – Brasília, DF, 2015-  
73 p. : il. (algumas color.) ; 30 cm.

Orientador: Prof. Dr. Paulo Roberto Miranda Meirelles

Trabalho de Conclusão de Curso – Universidade de Brasília - UnB  
Faculdade UnB Gama - FGA , 2015.

1. Redes sociais. 2. Software livre. I. Prof. Dr. Paulo Roberto Miranda Meirelles. II. Universidade de Brasília. III. Faculdade UnB Gama. IV. Evoluindo uma plataforma de rede de colaboração social para um ambiente de aprendizagem virtual

CDU 02:141:005.6

---

Hebert Douglas de Almeida Santos

## **Evoluindo uma plataforma de rede de colaboração social para um ambiente de aprendizagem virtual**

Monografia submetida ao curso de graduação em (Engenharia de Software) da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em (Engenharia de Software).

Trabalho aprovado. Brasília, DF, 17 de Dezembro de 2013:

---

**Prof. Dr. Paulo Roberto Miranda  
Meirelles**  
Orientador

---

**Prof. Dr. Edson Alves da Costa Júnior**  
Convidado 1

---

**Prof. Dr. Leonardo Lazarte**  
Convidado 2

Brasília, DF  
2015



# Resumo

Este trabalho de conclusão de curso de engenharia de software tem como objetivo evoluir uma plataforma de redes sociais, para permitir que ele contenha funcionalidades de um Ambiente Virtual de Aprendizagem (AVA). Neste contexto, escolheu-se utilizar a rede colaboração livre Comunidade.UnB, um ambiente para alunos e professores trabalharem de maneira horizontal e colaborativa, que faz uso da plataforma brasileira para redes sociais livre Noosfero. Dessa maneira, realizou-se um levantamento sobre os principais recursos desses ambientes comparando-os com o Noosfero, além de compreender como realizar a evolução da plataforma no desenvolvimento empírico de software. Assim sendo, planeja-se contribuir com a implementação de funcionalidades, que evolua o Comunidade.Unb, para conter recursos de AVA, propondo um hibridismo entre esses ambientes.

**Palavras-chaves:** redes sociais. ambiente virtual de aprendizagem. noosfero. métodos ágeis. software livre.



# Abstract

This Course Conclusion Paper of Software Engineering aims to evolve a social network platform to allow functions of a Virtual Learning Environment (VLE). For this, was chosen a open collaborative network Comunidade.Unb, a environment for students and teachers to work collaboratively, making use of a Brazilian platform for open social networks, Noosfero. This way, was held a survey about the main features of these environments comparing them with Noosfero, and understand how to perform the evolution of the platform in the empirical software development. Therefore , plans to contribute to the implementation of features that evolve the Comunidade.Unb to contain VLE resources, proposing a hybridism between these environments.

**Key-words:** social networking. virtual learning environment. noosfero. agile methods. open-source software.



# Lista de ilustrações

Figura 1 – Issue Tracker no GitLab . . . . .	45
Figura 2 – Arquitetura do Noosfero . . . . .	46
Figura 3 – Relações entre entidades de domínio ambiente, domínio e perfis. Extraído de: (BUCHER, 2013) . . . . .	47
Figura 4 – Entidades de domínio: tipos de perfis. Extraído de: (BUCHER, 2013) . . . . .	48
Figura 5 – Entidades de domínio: tipos de artigos. Extraído de: (BUCHER, 2013) . . . . .	49
Figura 6 – Estrutura de Plugins . . . . .	49
Figura 7 – Exemplo do uso do Comunidade.UnB na disciplina de MES. . . . .	51
Figura 8 – Resultados do questionário para a pergunta 8 . . . . .	73
Figura 9 – Resultados do questionário para a pergunta 9 . . . . .	73
Figura 10 – Resultados do questionário para a pergunta 10 . . . . .	73
Figura 11 – Resultados do questionário para a pergunta 11 . . . . .	73



# Lista de tabelas

Tabela 1 – Tabela de comparação categorias: Conteúdo e Atribuição . . . . .	35
Tabela 2 – Tabela de comparação categoria Comunicação . . . . .	36
Tabela 3 – Tabela de comparação categoria Curso . . . . .	37
Tabela 4 – Tabela de comparação categoria Ferramentas . . . . .	38
Tabela 5 – Tabela de comparação categoria Teste/Quiz . . . . .	39
Tabela 6 – Tabela de comparação categorias: Permissões e papéis, página principal	40
Tabela 7 – Cronograma de atividades para TCC2 . . . . .	61
Tabela 8 – Opções para preenchimento do questionário. . . . .	69



# Lista de abreviaturas e siglas

AGPL	Affero General Public License
AVA	Ambiente Virtual de Aprendizagem
BDD	Behavior Driven Development
CPD	Centro de Informática
DRY	Don't Repeat Yourself
FGA	Faculdade UnB Gama
VLE	Virtual Learning Environment
HTML	HyperText Markup Language
HTTP	HyperText Transfer Protocol
HTTPS	HyperText Transfer Protocol Secure
IRC	Internet Relay Chat
Moodle	Modular Object-Oriented Dynamic Learning Environment
LAPPIS	Laboratório de Produção, Pesquisa e Inovação em Software
LDAP	Lightweight Directory Access Protocol
MVC	Model-View-Controller
NIED	Núcleo de Informática Aplicada à Educação
ORM	Object-relational mapping
Rails	Ruby on Rails
RAM	Random Access Memory
SGBD	Sistema Gerenciador de Banco de Dados
SMT	Tecnologias de Mídia Social
TCC	Trabalho de Conclusão de Curso
TDD	Test Driven Development

UnB	Universidade de Brasília
UNICAMP	Universidade Estadual de Campinas
US	User Story
USP	Universidade de São Paulo
VLE	Virtual Learning Environment

# Sumário

<b>1</b>	<b>INTRODUÇÃO</b>	<b>17</b>
<b>1.1</b>	<b>Justificativa</b>	<b>17</b>
<b>1.2</b>	<b>Objetivos</b>	<b>18</b>
1.2.1	Objetivo Geral	18
1.2.2	Objetivos Específicos	18
<b>1.3</b>	<b>Organização do Trabalho</b>	<b>19</b>
<b>2</b>	<b>MANUTENÇÃO E EVOLUÇÃO DE SOFTWARE</b>	<b>21</b>
<b>2.1</b>	<b>Manutenção e evolução de software nos métodos tradicionais</b>	<b>21</b>
<b>2.2</b>	<b>Manutenção e evolução de software em métodos empíricos de desenvolvimento</b>	<b>22</b>
2.2.1	Software Livre	23
2.2.2	Relações entre métodos ágeis e processo de desenvolvimento de Software Livre	24
2.2.2.1	Práticas baseadas em disciplinas da engenharia de software	26
<b>3</b>	<b>AVA E PLATAFORMAS DE REDES SOCIAIS</b>	<b>29</b>
<b>3.1</b>	<b>Ambiente virtual de aprendizagem</b>	<b>29</b>
<b>3.2</b>	<b>Redes sociais</b>	<b>30</b>
<b>3.3</b>	<b>Comparação entre Noosfero e AVA</b>	<b>32</b>
<b>4</b>	<b>EVOLUINDO UMA PLATAFORMA DE REDE SOCIAL</b>	<b>43</b>
<b>4.1</b>	<b>Noosfero</b>	<b>43</b>
4.1.1	Práticas de desenvolvimento da comunidade Noosfero	44
4.1.2	Arquitetura	45
4.1.3	Modelo de domínio	47
4.1.4	Plugins	49
<b>4.2</b>	<b>Comunidade UnB</b>	<b>50</b>
<b>4.3</b>	<b>Desenvolvimento do NoosferAVA</b>	<b>52</b>
4.3.1	Evolução do <i>plugin Comunidade.UnB</i>	52
4.3.2	Evolução do <i>plugin Work Assignment</i>	53
<b>5</b>	<b>CONSIDERAÇÕES PRELIMINARES</b>	<b>59</b>
<b>5.1</b>	<b>Pesquisa com alunos</b>	<b>59</b>
<b>5.2</b>	<b>Cronograma</b>	<b>61</b>

Referências . . . . .	63
<b>APÊNDICES</b>	<b>67</b>
APÊNDICE A – INSTRUÇÕES PARA PREENCHIMENTO DO QUESTIONÁRIO . . . . .	69
APÊNDICE B – QUESTIONÁRIO PARA PROBLEMATIZAÇÃO .	71
APÊNDICE C – AVALIAÇÃO DOS RESULTADOS . . . . .	73

# 1 Introdução

Nas últimas décadas percebeu-se a grande difusão dos recursos computacionais e do acesso à *Internet*, por conseguinte houve um aumento significativo na quantidade e diversidade de conteúdos e serviços à disposição dos usuários. Araújo (2011) menciona que um dos fatores para esse aumento é que indivíduos que anteriormente limitavam-se ao papel de consumidores de conteúdo, hoje colocam-se também na posição de produtores. Neste contexto, encontramos na Internet inúmeros exemplos de sucesso como *blogs*, enciclopédias colaborativas como a Wikipedia, repositórios para compartilhamento de fotografia e vídeo, como Flickr e Youtube, redes sociais colaborativas, entre outros.

Castells (2007) afirma que tal fenômeno, comumente referenciado como *Web 2.0*, ocorre porque a população acredita que pode influenciar outras pessoas atuando no mundo por meio da sua força de vontade e utilizando seus próprios meios. Isso explica o fato do crescimento no uso de redes sociais nos últimos anos, onde as pessoas querem produzir seus próprios conteúdos gerando discussões construtivas sobre determinado assunto.

O ambiente acadêmico atual faz uso de ferramentas virtuais de aprendizagem, soluções colaborativas e redes sociais para interação. Essas ferramentas auxiliam professores e alunos em suas atividades diárias, facilitam a rápida atualização de conteúdos, facilitam o acesso a públicos geograficamente dispersos, reduz custos logísticos e administrativos e desenvolvem capacidades de auto estudo e autoaprendizagem. Nesse contexto, deve-se buscar solução integrada, de fácil acesso, que agregue as diversas funcionalidades necessárias a um ambiente virtual de aprendizagem.

## 1.1 Justificativa

A Universidade de Brasília (UnB) faz uso de ambiente virtual de aprendizagem, suportado pela ferramenta moodle, que permite a criação de cursos “on-line”, páginas de disciplinas, grupos de trabalho e aprendizagem virtual. Esse é um suporte tecnológico até então utilizado pelos professores como suporte a condução de suas disciplinas. Baseado em métodos empíricos, verifica-se que nesse contexto os alunos o utiliza, apenas para consumirem conteúdos produzidos ou responderem a estímulos vindo do professor.

Neste trabalho realizou-se um questionário afim de investigar se os alunos da Universidade de Brasília utilizam as redes sociais, como o *Facebook*, para o compartilhamento de recursos e informações relacionados as disciplinas cursadas na Universidade. Verificou-se que em sua maioria os alunos utilizam as redes sociais para tal fim, mesmo que os professores não recomendem o uso em suas disciplinas (Resultados no apêndice C). Evi-

denciando a afirmação de [Davis et al. \(2012\)](#) que os alunos de graduação querem estar nas redes sociais e que aqueles que participam de forma mais ativa delas tendem a obter maior riqueza em suas relações sociais.

Para atender este novo contexto onde os usuários também querem produzir seus conteúdos e trabalharem de maneira horizontal e colaborativa, a UnB também possui uma rede colaboração livre. Esta é denominada Comunidade.UnB , desenvolvida para que alunos, professores e servidores técnico-administrativos tenham um ambiente virtual de criação e compartilhamento de conhecimento colaborativo. Essa comunidade foi implementada com base na plataforma Noosfero, que também permite que dentro de cada rede, o usuário tenha o seu espaço, com total flexibilidade de personalização visual e gerenciamento de conteúdo.

Neste contexto, este projeto visa criar instrumentos para que os alunos permaneçam nas redes sociais, mas no âmbito de assuntos acadêmicos, por meio da adição de funcionalidades no Comunidade.Unb, relativas a ambientes virtuais de aprendizagem. Para propor um hibridismo entre uma rede social colaborativa da UnB e um ambiente virtual de aprendizagem, dessa maneira há necessidade de evolução da plataforma Noosfero, que não dispõe de alguns recursos necessários para esses ambientes.

## 1.2 Objetivos

### 1.2.1 Objetivo Geral

Evoluir o Noosfero para permitir que ele contenha funcionalidades de um Ambiente Virtual de Aprendizagem.

### 1.2.2 Objetivos Específicos

- Identificar as funcionalidades presentes em Ambientes Virtuais de Aprendizagem (AVA).
- Comparar as funcionalidades dos AVA com as disponíveis no Noosfero.
- Selecionar as funcionalidades dos AVA a serem implementadas no Noosfero.
- Implementar algumas das funcionalidades selecionadas para os usuários da Comunidade.UnB.

## 1.3 Organização do Trabalho

Nesta seção apresenta-se como este trabalho está organizado e o que será abordado em cada um dos capítulos. O trabalho se inicia com o capítulo 2, no qual é abordado conceitos relacionados a manutenção e evolução de software nos métodos tradicionais e empíricos de *software* além de descrições de práticas que serão utilizadas no desenvolvimento deste projeto. No capítulo 3 é discutido os conceitos de ambientes virtuais de aprendizagem, bem como o levantamento de suas principais funcionalidades. E por fim uma comparação de funcionalidades entre esses ambientes e o Noosfero, afim de verificar quais delas a plataforma carece. No capítulo 4 será apresentado a arquitetura e funcionamento da plataforma Noosfero, utilizado pela Comunidade.Unb. Além das propostas de melhoria e evolução deste trabalho para com a plataforma. E por fim no capítulo 5 serão apresentadas uma discussão sobre as funcionalidades até então implementadas, e o cronograma utilizado para a execução do projeto.



## 2 Manutenção e evolução de software

Neste capítulo será abordado a manutenção e evolução de software provendo uma comparação entre os métodos tradicionais e empíricos empregados na engenharia de software, evidenciando algumas práticas adotadas nas comunidades de *software* livre, que serão utilizadas neste trabalho.

### 2.1 Manutenção e evolução de software nos métodos tradicionais

A [ISO/IEC \(1998\)](#) 12207 define o ciclo de vida de um software por meio de agrupamento de processos que são divididos em quatro classes: fundamentais, apoio, organizacionais e adaptação. O processo de manutenção de software está classificado nessa ISO como um dos processos fundamentais, ou seja necessários para que um o software seja implementado.

O Instituto de Engenheiros Eletricistas e Eletrônicos [IEEE 1219 \(1998\)](#), define manutenção de software como a modificação de um produto de software após sua entrega com vistas a corrigir falhas, melhorar o desempenho ou outros atributos, e adaptar o produto a um ambiente modificado de acordo com as necessidades do cliente.

As normas citadas tratam a manutenção de software como se o mesmo fosse um produto genérico. Para [Pfleeger \(2004, p. 380\)](#) a manutenção varia de acordo com o propósito de criação do software. Ainda, segundo a Autora, a probabilidade de o sistema ser modificado varia de acordo com o vínculo de dependência dos requisitos com o sistema do mundo real, pois, quanto mais dependentes forem os requisitos do mundo real maior será a probabilidade do sistema ser modificado.

Para determinar o tipo de manutenção a ser realizado, [Pfleeger \(2004\)](#) separa sistemas em três grupos. O primeiro grupo de sistemas é o que possui os requisitos especificados e formalmente definidos, ou seja, os resultados esperados são bem conhecidos e a solução é estática e não se adapta facilmente a uma mudança no problema que o gerou. O segundo grupo consiste dos sistemas, cujas soluções são aproximadas do mundo real, tem-se a solução completa apenas na teoria pois solucionar o problema em sua totalidade é impraticável ou impossível. O jogo de xadrez é um exemplo desse grupo, uma vez que é impossível calcular todas as movimentações possíveis e suas consequências em tempo hábil para a próxima jogada. O terceiro grupo incorpora a mudança na natureza do mundo real em si, modificando-se a medida em que o mundo muda. Tem como base um modelo do processo abstrato envolvido. Exemplo: sistema que prevê a estabilidade econômica de um país e utiliza como base o modelo econômico em uso.

Analisando a literatura percebe-se que a manutenção de software está dividida em três grandes categorias, listadas a seguir, que podem ser visualizados tanto na norma da [IEEE 1219 \(1998\)](#) e em livros, desde [Lientz e Swanson \(1980\)](#) à [Pfleeger \(2004\)](#), as categorias são as seguintes:

- **Manutenção corretiva:** realizada para controlar funções cotidianos do sistema, corrigindo falhas e defeitos de funcionalidades do programa.
- **Manutenção adaptativa:** visa a implementação de modificações secundárias que têm como origem uma modificação em outra parte do sistema ou do meio externo.
- **Manutenção perfectiva ou evolutiva:** objetivam realizar acréscimo de novas funcionalidades ou mudanças no *software* independente da existência de defeitos.

[Pressman \(2011\)](#) e [Criscuolo \(2008\)](#) acrescentam a manutenção preventiva. Essa está ligada a uma reengenharia, aplicando conceitos de engenharia de software para tornar os softwares mais fáceis de serem corrigidos, adaptados e melhorados ao longo dos anos. Para [Criscuolo \(2008\)](#) a manutenção preventiva permite que o software receba alterações com objetivo de minimizar os riscos de defeitos e reduzir a complexidade de sua estrutura, garantindo um certa segurança à manutenção.

Para o [IEEE 1219 \(1998\)](#) além das categorias citadas, há a manutenção “emergencial”, definida pelo IEEE como uma manutenção corretiva que não foi programada ou planejada, mas que deve ser executada para manter o sistema em funcionamento.

As atividades do processo de desenvolvimento de software ainda são dependentes da participação e criatividade humana, e o crescimento de métodos ágeis e software livre mostram que o código é um dos principais artefatos. Diante disso torna-se necessário na Engenharia de *Software* utilizar métodos empíricos que apoiam o desenvolvimento de *software* de acordo com sua natureza abstrata.

## 2.2 Manutenção e evolução de software em métodos empíricos de desenvolvimento

A ciência empírica diz respeito à aquisição de conhecimentos através de métodos empíricos. No entanto, o que constitui o conhecimento, assim sendo, os métodos para adquiri-lo baseia-se em pressupostos básicos sobre ontologia (ou seja, o que acreditamos existir) e epistemologia (ou seja, como as crenças são adquiridas e que justifica-los). Dessa maneira a pesquisa empírica procura explorar, descrever, prever e explicar fenômenos naturais, sociais, cognitivas ou usando evidências baseadas em observação ou experiência. Trata-se de obtenção e interpretação de evidências, por exemplo: experimentação; obser-

vação sistemática; entrevistas ou inquéritos; ou pelo análise minuciosa de documentos ou artefatos (SJOBORG; DYBA; JORGENSEN, 2007).

Neste capítulo serão abordadas algumas metodologias e práticas no processo de desenvolvimento de software que advém de métodos empíricos, dentre elas destaca-se os métodos ágeis e o processo de desenvolvimento de software livre.

### 2.2.1 Software Livre

De acordo com Meirelles (2013) software expressa uma solução abstrata dos problemas computacionais, neste contexto é o componente que contém o conhecimento relacionado aos problemas a que a computação se aplica, contendo diversos aspectos que ultrapassam questões técnicas, tais como:

- o processo de desenvolvimento do software;
- os mecanismos econômicos que regem o desenvolvimento e o uso do software, a exemplo de gerenciais, competitivos, sociais, cognitivos etc.;
- o relacionamento entre desenvolvedores, fornecedores e usuários do software;
- os aspectos éticos e legais relacionados ao software.

O que define e diferencia o software livre de software proprietário vai do entendimento desses quatro pontos dentro do que é conhecido como *ecossistema do software livre* (MEIRELLES, 2013).

Software livre é definido pela Free Software Foundation como “uma questão de liberdade dos usuários para executar, copiar, distribuir, estudar, mudar e melhorar o software. Levando isto em consideração significa que os usuários do software possuem quatro liberdades essenciais (STALLMAN, 2002):

- executar o programa, para qualquer propósito (liberdade no. 0);
- estudar como o programa funciona, e adaptá-lo para as suas necessidades (liberdade no. 1). Acesso ao código-fonte é um pré-requisito para esta liberdade;
- redistribuir cópias de para ajudar o próximo (liberdade no. 2);
- aperfeiçoar o programa, e liberar os seus aperfeiçoamentos, de modo que toda a comunidade se beneficie (liberdade no. 3). Acesso ao código-fonte é um pré-requisito para esta liberdade.

Para que as liberdades um e três façam sentido, é necessário que o desenvolvedor ou pesquisador tenha acesso ao código-fonte do programa. Portanto, acesso ao código-fonte é uma condição necessária ao software livre (GNU, 2013).

Um programa é considerado um software livre se os usuários possuem todas as liberdades citadas. Dessa forma, você deve ser livre para redistribuir cópias, com ou sem modificações, de graça ou cobrando uma taxa pela distribuição, para qualquer lugar. Ser livre para fazer essas coisas significa, entre outras coisas, que não há necessidade de pedir ou pagar por permissão para fazê-las (SANTOS, 2012).

O fato de que o código-fonte pode ser livremente compartilhado oferece vantagens ao software livre em comparação ao software proprietário. Uma delas é simplificação do desenvolvimento de aplicações personalizadas já que não precisam ser programadas a partir do zero, ou seja, podem se basear em soluções existentes.

A outra vantagem resultante do compartilhamento do código refere-se à possibilidade de melhoria na qualidade, em particular frente aos problemas inerentes à sua complexidade (RAYMOND, 1999).

Isso se deve ao maior número de desenvolvedores e usuários envolvidos com o software, permitindo maiores situações de uso em necessidades variadas, o que propicia a identificação de um número maior de *bugs* e mais sugestões de melhoria, promovendo refatorações que geralmente levam a melhoria do código.

O GNU teve a necessidade de utilizar termos de distribuição que impedissem a transformação de software livre em proprietário. O método usado foi chamado de *Copyleft*. *Copyleft* é um método geral para desenvolver um software livre e exige que todas as versões modificadas e estendidas do programa também sejam livres. Ele utiliza lei de direitos autorais, com finalidade diferente da usual: ao invés de um meio de privatizar o software, torna-se um meio de manter o software livre (STALLMAN; GAY, 2009).

### 2.2.2 Relações entre métodos ágeis e processo de desenvolvimento de Software Livre

Os métodos ágeis assim como processo de desenvolvimento de software livre, são métodos empíricos de software que provêm de um conjunto de metodologias baseada na prática de desenvolvimento de software. Em 2001 a aliança ágil definiu o Manifesto ágil, que definem preferências e princípios que se espera de qualquer método de desenvolvimento dessa categoria. O manifesto é baseado em quatro valores (BECK et al., 2001):

- Indivíduos e interações são mais importantes que processos e ferramentas.
- Software em funcionamento é mais importante que documentação abrangente.

- Colaboração com o cliente (usuários) é mais importante que negociação de contratos.
- Responder às mudanças é mais importante que seguir um plano.

Na dissertação de [Corbucci \(2011\)](#) há evidências de que esses quatro valores demonstram a semelhança entre as práticas comuns utilizadas pelas comunidades de software livre e equipes ágeis. Além disso, o Autor, destaca que muitas práticas disseminadas pelas metodologias ágeis são usadas no dia a dia dos desenvolvedores e equipes das comunidades de software livre:

- código compartilhado (coletivo);
- projeto simples;
- repositório único de código;
- integração contínua;
- código e teste;
- desenvolvimento dirigido por testes;
- refatoração.

Entender esses aspectos no contexto de software livre torna-se relevante porque, ao contrário dos métodos tradicionais de desenvolvimento, o ágeis são adaptativos ao invés de prescritivos e são orientados às pessoas ao invés dos processos. Analisar o processo de desenvolvimento de software livre do ponto de vista da Engenharia de Software e as possíveis sinergias com os métodos ágeis podem contribuir para um melhor entendimento dessa disposição na criação e colaboração em torno de projetos de software livre ([MEIRELLES, 2013](#)).

No processo de desenvolvimento de software livre, após a divulgação e lançamento da primeira versão do projeto, a solução está pronta para uso. Os usuários são desenvolvedores podem colaborar com o projeto afim de evoluí-lo para suprir necessidades pessoais ou de interesse da comunidade, são enviadas aos mantenedores do projeto como *patches*<sup>1</sup>. Os mantenedores analisarão as propostas, e caso concordem com a mudança e implementação, irão aplicá-las ao repositório oficial do projeto. Portanto, mesmo que em projetos maiores outros aspectos sejam levados em consideração ou sigam processos mais burocráticos de colaboração, a essência da colaboração técnica está no envio e análise de trechos de código-fonte ([MEIRELLES, 2013](#)).

As comunidades de software livre são “livres” para escolher e adotar as práticas que mais se adaptam ao seu modelo, dessa maneira é relatado o processo de desenvolvimento

---

<sup>1</sup> arquivos que contém as modificações no código

do Noosfero, um exemplo de desenvolvimento de software livre evidenciado na seção 4.1.1 do capítulo 4.

### 2.2.2.1 Práticas baseadas em disciplinas da engenharia de software

No contexto de gerência e configuração de software grande parte dos projetos de software livre atualmente utilizam algum sistema de controle de versão, visto como uma extensão natural do processo de desenvolvimento. Esse sistema, normalmente faz-se uso de um repositório central, onde são armazenadas todas as versões dos arquivos, e permite paralelizar o desenvolvimento de forma organizada, principalmente quando há um grande número de desenvolvedores (REIS; FORTES, 2001). Permite ainda o *commiter* oficial realizar inspeção e revisão de todo código que é integrado ao repositório principal do projeto de forma não intrusiva no desenvolvimento realizado pelos colaboradores. Reis e Fortes (2001) atestam que a revisão acaba acontecendo quase automaticamente, já que as pessoas são forçadas pela distância a enviar modificações a outras para a integração.

O teste de software é um processo destinado a certificar-se de que o código desenvolvido é previsível e consistente, e que realiza apenas as ações para o qual foi projetado (MYERS; SANDLER; BADGETT, 2011). Teste automatizado é uma prática de criar *scripts* ou programas de computador que exercitam o sistema em teste, capturam os efeitos colaterais e fazem verificações, de forma automática e dinâmica (MESZAROS, 2007).

Segundo Bernardo (2011) os testes automatizados agregam valor ao produto final porque afetam diretamente a qualidade dos sistemas de software, mesmo que os artefatos produzidos não sejam visíveis aos usuários.

A integração contínua tem por objetivo integrar o sistema e todas suas dependências para assegurar que nenhuma modificação tenha danificado o sistema, sejam elas alterações no código-fonte, em configurações ou mesmo em dependências e outros fatores externos (DUVALL; MATYAS; GLOVER, 2007). Desse modo, para certificar a integridade do código a integração contínua garante que todos os testes são executados a cada modificação da base de código fornecendo *Feedback* imediato, assegurando que a integração tenha ocorrido de forma satisfatória.

Corbucci (2011) afirma que a equipe utiliza os testes para descobrir eventuais defeitos o mais rapidamente possível, o que facilita e acelera a correção, além de reduzir a probabilidade de que pequenos problemas se transformem em *bugs* ou comportamentos inesperados no futuro.

Baseado nas disciplinas de gerência e configuração e teste, fazendo uso de controle de versão, testes automatizados e integração contínua, é que evoluiremos a plataforma Noosfero com as funcionalidades selecionadas para melhor adaptar a plataforma com

recursos de AVA. Para isso será abordado no próximo capítulo (3) os recursos utilizados nos AVA comparando-os com o Noosfero.



## 3 AVA e Plataformas de redes sociais

Neste capítulo serão apresentadas os conceitos relacionados aos ambientes virtuais de aprendizagem e redes sociais. Além disso há uma comparação dos principais AVA levantados nessa pesquisa com a plataforma Noosfero.

### 3.1 Ambiente virtual de aprendizagem

No atual cenário, em que vivemos a tecnologia da informação está cada vez mais presente em nossa rotina, seja com o uso de celulares, tablets, televisores entre outros e utilizá-la no ambiente educacional pode ser uma grande vantagem de integração à sociedade do conhecimento. Nesse cenário ambientes virtuais de aprendizagem tem sido cada vez mais usados nas mais diversas instituições (acadêmicas, empresariais e tecnológicas) com o objetivo de apoiar o ensino à distância.

Os ambientes virtuais de aprendizagem <sup>1</sup> são sistemas que tem a funcionalidade de *software* para a comunicação entre computador e métodos de entrega de material de cursos online. Muitos desses ambientes permitem simular a sala de aula física em um meio virtual contudo existem ambientes que além de realizar este papel fazem uso da tecnologia para propiciar aos envolvidos novas ferramentas que potencializem a comunicação e facilitem a aprendizagem (SCHLEMMER, 2005).

Os AVA são utilizadas por várias instituições de ensino do mundo. O mercado apresenta uma grande variedade de *softwares* proprietários e livres. Entretanto, apesar dessa diversidade, Behar e BASSANI (2009) destaca que muitas características e ferramentas são comuns entre eles:

- permitem acesso restrito a usuários previamente cadastrados;
- disponibilizam espaço para a publicação de material do professor;
- disponibilizam espaço destinado ao envio/armazenamento de tarefas realizadas pelos alunos;
- disponibilizam um conjunto de ferramentas de comunicação síncrona e assíncrona<sup>2</sup>;
- disponibilizam correio eletrônico (*e-mail*) e mural de recados.

Além dessas características, Aguado (2013) ressalta que os principais componentes encontrados consistem de:

---

<sup>1</sup> Em inglês *Virtual Learning Environment* (VLE)

<sup>2</sup> Como *chat* e fórum de discussões

- disponibilização de conteúdo de ensino;
- mapeamento do conteúdo do curso (quebra do conteúdo em sessões que podem ser acessadas e cumpridas);
- disponibilização de atividades de interação e avaliação;
- acompanhamento do desempenho do aluno;
- suporte online;
- links de internet para conteúdos externos;
- identificações e permissões de acesso.

Apesar dos autores supracitados relatarem as principais características e componentes dos AVA, constata-se que nem todos oferecem as mesmas funcionalidades como pode ser verificado na seção 3.3 desse trabalho. O escopo das opções disponíveis nos sistemas nem sempre é o mesmo, evidenciando a razão do alto número de sistemas no mercado, pois os produtos possuem pontos fortes e fracos, e por isso, nem sempre as funcionalidades atendem aos diversos tipos de instituições e seus respectivos objetivos (AGUADO, 2013).

Segundo Chin (2003), as funcionalidades oferecidas pelo AVA facilitam e fortalecem o aprendizado de conceitos dados presencialmente. Mesmo que os AVA apresentem características principais do ensino tradicional em sala de aula, Chin (2003) propõe que os AVA não devem ser vistos como substitutos à aula presencial, uma vez que sempre deverá existir a interação humana física no processo educacional.

Dentre os vários sistemas disponíveis no mercado alguns necessitam de instalação no computador cliente, porém, a maioria é acessível pela internet, onde a instalação é realizada apenas no servidor. Das ferramentas encontradas atualmente destacam-se as tradicionalmente utilizadas como o Moodle <sup>3</sup> e o BlackBoard <sup>4</sup> detalhadas na seção 3.3.

## 3.2 Redes sociais

Para entender o conceito de redes sociais é importante destacar que são apenas uma camada das mídias sociais. É comum vermos a utilização deste termo para todos os tipos de mídias sociais mediadas por um computador. As tecnologias de mídias sociais <sup>5</sup> podem ser definidas de forma abrangente como a totalidade de produtos e serviços digitais disponibilizados online, como aplicações *web* e *mobile* que permitem aos indivíduos criar e

<sup>3</sup> *Modular Object-Oriented Dynamic Learning Environment*. Disponível em: <<https://moodle.org/>>

<sup>4</sup> Disponível em: <<http://www.blackboard.com/>>

<sup>5</sup> Termo em inglês, *Social Media Technology* (SMT)

compartilhar conteúdos onde sua principal fonte são os usuários, através da comunicação em várias vias (DAVIS et al., 2012).

As definições de rede utilizadas neste trabalho estão relacionadas com as propostas de:

- Emirbayer e Goodwin (1994): rede é um conjunto de relações ou ligações entre um conjunto de atores, no caso atores são os elos às pessoas que se comunicam em uma dada rede.
- Costa (2004, p. 73) forma de organização caracterizada fundamentalmente pela sua horizontalidade, isto é, pelo modo de inter-relacionar os elementos sem hierarquia.
- Tomaél, Alcará e Chiara (2005) estrutura não-linear, descentralizada, flexível, dinâmica, sem limites definidos e autoorganizáveis. Estabelece-se por relações horizontais de cooperação .

Segundo (MARTELETO, 2001), as redes sociais derivam dos conceitos de rede e representam um conjunto de participantes autônomos, que buscam unir suas idéias e recursos em torno de valores e interesses compartilhados. A autora ressalta a ideia de compartilhamento de valores e interesses que, para promover o fortalecimento da rede, dependem do compartilhamento da informação e do conhecimento.

Fundamentado nesses principais conceitos o foco deste trabalho é nos sites de redes sociais, os autores Ellison et al. (2007) definem sites de redes sociais como serviços *web* que permitem a seus usuários criarem perfis e, por meio deles, conectarem com outros usuários, propiciando a busca e o cruzamento de informações dentro da sua lista de conexões.

Fundamentado nestes principais conceitos, este trabalho está focado em sites de redes sociais, que possuem o objetivo de promover a interação em torno das colaborações, isto é, apoiada no entendimento de redes de colaboração. Isto se aplica ao contexto de uma rede social de nicho de uma universidade, como explicitado na seção 4.2 onde é descrita a proposta de disponibilização de uma rede social para a Universidade de Brasília (UnB). A rede da UnB permite que as pessoas entrem na rede para acompanharem uma disciplina, um projeto ou um determinado grupo de trabalho. Os alunos também podem explorar a rede para encontrar comunidades e conteúdos de seus interesses.

Bucher (2013) menciona que esta prática adotada por redes de colaboração não é possível em redes monopolistas e centralizadoras, porque em geral seus conteúdos estão dipostos de forma fragmentada ou são controlados. Isto demonstra que o uso de uma rede de colaboração dentro de universidades é uma escolha mais plausível para a difusão do conhecimento.

Apoiado nas vantagens proporcionadas pelas redes sociais de colaboração e sua implantação na UnB, busca-se neste trabalho associá-las aos conceitos dos ambientes virtuais de aprendizagem tratados na seção 3.1. A proposta visa a criação de funcionalidades que viabilizem o uso de uma plataforma de redes sociais como um ambiente de apoio aos ambientes virtuais de aprendizagem.

### 3.3 Comparação entre Noosfero e AVA

Como o objetivo do trabalho é a evolução de plataforma de redes sociais para um ambiente virtual de aprendizagem, a pesquisa realizada fundamenta-se no levantamento das principais funcionalidades que os AVA atuais utilizam, para verificar quais a plataforma o Noosfero ainda não possui. Desse modo surgiu a necessidade do levantamento das principais ferramentas AVA utilizadas atualmente pelas universidades e instituições de ensino, para proporcionar o mapeamento destas funcionalidades. Os AVA selecionados para comparação são **BlackBoard**, **Sakai**, *Modular Object-Oriented Dynamic Learning Environment* (**Moodle** <sup>6</sup>) e *TelEduc*.

O BlackBoard <sup>7</sup>, é um dos maiores ambientes virtuais de aprendizagem utilizados atualmente por um, total de 72% das TOP 200 melhores universidades do mundo (**BLACKBOARD**, 2015). Para utilização desta plataforma é necessário a aquisição de licenças por parte da instituição que pretende implementá-la. A plataforma tem total suporte da empresa que a mantém e não permite a modificação de sua estrutura interna pelos seus utilizadores. Os ajustes e melhorias de funcionalidades são realizados através do mapeamento dos comentários e sugestões dos usuários.

O Moodle, é um software livre criado em 2001 pelo educador cientista computacional Martin Dougiamas. É desenvolvido na linguagem de programação PHP e desta maneira pode ser instalado em qualquer sistema operacional que tenha suporte à linguagem. Para o seu uso podem ser utilizadas várias bases de dados com suporte a ODBC<sup>8</sup>, tais como MySQL, PostgreSQL e Oracle.

O projeto Sakai<sup>9</sup> ou a comunidade Sakai desenvolve e distribui o software livre Sakai como ambiente virtual de aprendizagem para colaboração e ensino para educadores, por educadores. A plataforma foi desenvolvida em linguagem Java e pode ser executado em várias plataformas diferentes como Linux, Unix, Windows e MAC. Suporta banco de dados MySQL e Oracle. As instituições que utilizam esta plataforma são abrangentes e estão listadas no site da comunidade <sup>10</sup>.

<sup>6</sup> Disponível em: <<https://moodle.org/>>

<sup>7</sup> Disponível em: <<http://www.blackboard.com/>>

<sup>8</sup> Padrão para acesso a sistemas gerenciadores de bancos de dados

<sup>9</sup> Disponível em: <https://sakaiproject.org/>

<sup>10</sup> Disponível em: <https://www.sakaiproject.org/community>

O **TelEduc**<sup>11</sup> é um ambiente virtual de aprendizagem, cujo desenvolvimento iniciou em 1997, a partir de uma dissertação de mestrado de Alessandra de Dutra e Cerceau do Instituto de Computação da Universidade Estadual de Campinas<sup>12</sup>. Foi objeto de pesquisa e desenvolvimento de um projeto coordenado pela Profa Dra Heloísa Vieira da Rocha<sup>13</sup> até 2012. Desde então, o projeto vem sendo evoluído para implementar ajustes e novas funcionalidades segundo (ROCHA; MORAES, 2002), visando implementar necessidades relatadas por seus Usuários. O projeto é mantido pelo Núcleo de Informática Aplicada à Educação<sup>14</sup> da UNICAMP.

O TelEduc é um software livre desenvolvido na linguagem de programação PHP<sup>15</sup> e JavaScript com banco de dados o MySQL<sup>16</sup> para ambientes Linux. Em seu lançamento tornou-se um dos softwares mais utilizados para apoiar a educação à distância nas mais diversas áreas. Seus principais usuários são as universidades públicas e privadas, que utilizam a ferramenta para atividades educacionais, disponibilizando materiais, dando suporte a comunicação e interação entre os participantes.

Foram criadas tabelas comparativas (Tabelas 1, 2, 3, 4, 5, 6) entre as ferramentas selecionadas e a plataforma Noosfero, desse modo fica mais evidente identificar quais funcionalidades a plataforma Noosfero carece. E vale ressaltar que não é objetivo deste trabalho esclarecer ou afirmar qual é a melhor ferramenta, mas verificar as principais funcionalidades dos AVA que são utilizados e aceitos pelas universidades.

As funcionalidades incluídas nas tabelas foram obtidas da literatura apresentada na seção 3.1, e em uso aos ambientes onlines de demonstração, contando ainda, com o auxílio de mecanismos de ajuda fornecidos pelas ferramentas. Assim sendo, as funcionalidades selecionadas foram divididas em oito categorias:

- **Conteúdo:** relacionadas à criação e manutenção de conteúdos publicados pelos usuários.
- **Atribuição:** permitem que os usuários carreguem arquivos do seu computador local para a plataforma.
- **Ferramentas:** mecanismos de apoio ao gerenciamento do conteúdo e ambiente;
- **Teste/Quiz:** permitem a realização de avaliações e acompanhamento de resultados dos alunos;

---

<sup>11</sup> Disponível em: <<http://www.teleduc.org.br/>>

<sup>12</sup> IC/UNICAMP. Disponível em: <<http://www.ic.unicamp.br/>>

<sup>13</sup> <<http://lattes.cnpq.br/6985892121344767>>

<sup>14</sup> NIED/UNICAMP. Disponível em: <<http://www.nied.unicamp.br/>>

<sup>15</sup> Disponível em: <<http://php.net/>>

<sup>16</sup> Disponível em: <<https://www.mysql.com/>>

- **Comunicação:** possibilitam a comunicação síncrona e assíncrona entre os participantes de um curso;
- **Curso:** fundamentais para organização dos cursos;
- **Permissão e papéis:** utilizadas para gerenciar o ambiente, definindo papéis e autorizações de acesso ao ambiente;
- **Página principal:** são as principais funcionalidades evidenciadas na página inicial de cada usuário.

Das categorias listadas na Tabela 1, podemos destacar que o Noosfero carece do SCORM <sup>17</sup> e IMS-Content-Package <sup>18</sup> que são funcionalidades importantes para os AVA, pois permitem a importação de conteúdos de outras ferramentas proporcionando flexibilidade para o usuário.

Na análise da Tabela 2, verifica-se que o Noosfero possui funcionalidades básicas de comunicação, mas não possibilita a comunicação síncrona (Bate-papo). Possui ainda, fóruns que normalmente são utilizados para a divulgação de notícias e dúvidas, entretanto, não permite a assinatura de tópicos para o recebimento de notificações.

Como o Noosfero não é um ambiente virtual de aprendizagem, é possível verificar na Tabela 5 que ele não possibilita a criação de questões para obter a resposta de usuários. Também não conta com a funcionalidade que permita ao professor atribuir as notas e acompanhar a situação de cada aluno. O Noosfero possui um *plugin* denominado *Work Assignment* que permite aos usuários enviarem arquivos para o servidor mas não possibilita que o moderador estabeleça uma pontuação para os mesmos.

---

<sup>17</sup> Um modelo de referência seja, conjunto unificado de especificações para a disponibilização de conteúdos e serviços de e-learning (DUTRA; TAROUCO, 2006).

<sup>18</sup> Padrão que permite exportar o conteúdo de um sistema de gerenciamento de conteúdo de aprendizagem ou repositório digital.

Funcionalidades\Sistema	BlackBoard 9.1	Moodle	TelEduc	Sakai	Noosfero
<b>Conteúdo</b>					
Diretórios/Pastas	Sim	Sim	Sim	Sim	Sim
Criar/Extrair arquivos de pastas	Sim	Sim	Sim	Sim	Sim
Editor HTML	Sim	Sim	Sim	Sim	Sim
Filtro de vários idiomas	Não	Sim	<b>Não</b>	<b>Não</b>	Sim
Audio	Sim (incorporado no plugin do QuickTime)	Sim	<b>Não</b>	Sim	
Vídeo	Sim	Sim	<b>Não</b>	Sim	Sim
SCORM	Sim	Sim	<b>Não</b>	Sim	<b>Não</b>
IMS-Content-Package	Sim	Sim			<b>Não</b>
<b>Atribuição</b>					
Upload de um arquivo	Sim	Sim	Sim	Sim	Sim
Upload de vários arquivos	Sim	Sim	<b>Não</b>	Sim	Sim
Texto online	Sim(quiz)	Sim	Sim	Sim	Sim

Tabela 1 – Tabela de comparação categorias: Conteúdo e Atribuição

Funcionalidades\Sistema	BlackBoard 9.1	Moodle	TelEduc	Sakai	Noosfero
<b>Comunicação</b>					
Enviar e-mail	Sim	Sim	Sim	Sim	Sim
Enviar mensagens	Sim	Sim	Sim	Sim	Sim (mural)
Bate papo	Sim	Sim	Sim	Sim	<b>Não</b>
Fórum de discussão	Sim	Sim	Sim	Sim	Sim
Alterar tipo de fórum	Sim	Sim	Sim	Sim	Sim
Fórum:	Sim	Sim	Sim	Sim	Sim
- Postagem anônima	Sim	<b>Não</b>	<b>Não</b>	<b>Não</b>	<b>Não</b>
- anexar um arquivo	Sim	Sim	<b>Não</b>	Sim	<b>Não</b>
- os participantes podem criar tópicos	Sim	Sim	<b>Não</b>	Sim	Sim
- inscrever-se em um fórum	Sim	Sim	<b>Não</b>	Sim	<b>Não</b>
- assinar tópicos	Sim	<b>Não</b>	<b>Não</b>	Sim	<b>Não</b>
- moderar fórum	Sim	<b>Não</b>	Sim	Sim	<b>Não</b>
Pesquisar fóruns	Sim	Sim	Sim	Sim	Sim (busca geral)

Tabela 2 – Tabela de comparação categoria Comunicação

Funcionalidades\Sistema	BlackBoard 9.1	Moodle	TelEduc	Sakai	Noosfero
<b>Curso</b>					
Criar cursos	Sim	Sim	Sim	Sim	Sim
Criar novos papéis	Sim	Sim	<b>Não</b>	<b>Não</b>	Sim
Ferramentas de grupos	Sim	Sim	Sim	Sim	Sim
Relatórios do curso	Sim	Sim	Sim	Sim	<b>Não</b>
Sistema de alerta	Sim	<b>Não</b>	<b>Não</b>	<b>Não</b>	<b>Não</b>
Personalização ou configuração (nome, duração, inscrição, idioma)	Sim	Sim	Sim	Sim	Sim
Formato de curso(fórum, formato dos tópicos, semanal)	<b>Não</b>	Sim	<b>Não</b>		<b>Não</b>
Gerenciar campo menu	Sim	Sim O menu é personalizado de acordo com conteúdo o curso	<b>Não</b>	Sim	Sim
Customizar estilo de curso	Sim (menu, tema)	Sim (tema)	<b>Não</b>		Sim
Estrutura de pastas	Sim	Sim	<b>Não</b>	Sim	Sim
Gerenciar ferramentas	Sim	Sim	Sim	Sim	Sim
Backup do curso / exportação	Sim	Sim	<b>Não</b>	Sim	Sim
Importar curso	Sim	Sim	<b>Não</b>	<b>Não</b>	<b>Não</b>
Restaurar curso	Sim	Sim	<b>Não</b>	<b>Não</b>	<b>Não</b>
Importar conteúdo/características de outros cursos	Sim	Sim	Sim	Sim	Sim
Gerenciador de arquivos	Sim	Sim	<b>Não</b>	Sim	Sim

Tabela 3 – Tabela de comparação categoria Curso

Funcionalidades\Sistema	BlackBoard 9.1	Moodle	TelEduc	Sakai	Noosfero
<b>Ferramentas</b>					
Glossário	Sim (Editável pelo instrutor)	Sim (Os participantes também podem adicionar entradas) + bloco para glossário	<b>Não</b>	Sim	<b>Não</b>
RSS Feeds	<b>Não</b>	Sim (Banco de dados, Fórum, Glossário)	<b>Não</b>	Sim	<b>Não</b>
Blogs	Sim	Sim	Sim	Sim	Sim
Lista de contatos	Sim	Sim	Sim	<b>Não</b>	Sim (lista de participantes)
Calendário (Curso)	Sim	Sim	Sim	Sim	Apenas de eventos
Permissões de acesso	Sim (programada, dependendo de grau, dependendo do nível total do curso, depende do acesso a conteúdo definido, definir suas próprias regras)	Sim (programada, dependendo de grau, dependendo do nível total do curso, depende do acesso a conteúdo definidos / atividades)	Sim	Sim	Sim
Verificação de plágio	Sim (integrado)	<b>Não</b> (Não integrado, mas existem APIs)	<b>Não</b>	<b>Não</b>	<b>Não</b>
Busca de arquivos (curso)	Sim (adicionando um bloco correspondente)	<b>Não</b>	Sim	Sim	Sim
Jornal	Sim	<b>Não</b>	<b>Não</b>		<b>Não</b>
Atividade concluída	Sim	Sim	Sim	Sim	Sim ( <b>Não</b> totalmente implementado)
Ferramentas de voz	<b>Não</b> (é necessário a ferramenta Wimba instalada)	<b>Não</b>	<b>Não</b>	Sim	<b>Não</b>
Permite o uso de módulos e plugins	Sim	Sim	<b>Não</b>	Sim (permite a seleção de ferramentas)	Sim

Tabela 4 – Tabela de comparação categoria Ferramentas

Funcionalidades\Sistema	BlackBoard 9.1	Moodle	TelEduc	Sakai	Noosfero
<b>Teste/Quiz</b>					
Banco de questões	Sim	Sim	Sim	Sim	<b>Não</b>
Exportar resultados	Sim	Sim	Sim	<b>Não</b>	<b>Não</b>
Exportar respostas	Sim	<b>Não</b>	<b>Não</b>	Sim	<b>Não</b>
<b>Tipo de questões</b>					
resposta com arquivo	Sim	Sim	<b>Não</b>	Sim	Sim
resposta por escala	Sim	Sim (múltipla escolha)	<b>Não</b>	<b>Não</b>	<b>Não</b>
preencher espaço vazio	Sim	Sim (difícil de lidar)	<b>Não</b>	Sim	<b>Não</b>
múltipla escolha	Sim	Sim	Sim	Sim	<b>Não</b>
resposta múltipla	Sim	Sim	Sim	Sim	Sim (permite o envio de várias versões de um arquivo)
Verdadeiro/Falso	Sim	Sim	Sim	Sim	<b>Não</b>
Calculos	Sim	Sim	<b>Não</b>	Sim	<b>Não</b>
Resposta dissertativa	Sim	Sim	Sim	Sim	Sim (com a criação de um artigo)
Calculo numérico	Sim	Sim	<b>Não</b>	Sim	<b>Não</b>

Tabela 5 – Tabela de comparação categoria Teste/Quiz

Funcionalidades\Sistema	BlackBoard 9.1	Moodle	TelEduc	Sakai	Noosfero
<b>Permissões e papéis</b>					
Papéis pré-definidos	Sim	Sim	Sim	Sim	Sim
Editar papéis existentes	Sim	Sim	<b>Não</b>	<b>Não</b>	Sim (apenas com permissões)
Criar novos papéis	Sim	Sim	<b>Não</b>	<b>Não</b>	Sim
<b>Página principal</b>					
RSS Feeds	<b>Não</b>	Sim	<b>Não</b>	Sim	<b>Não</b>
Avisos/ Novos fóruns	Sim	Sim	Sim	Sim	<b>Não</b>
Lista de afazeres/ Próximos eventos	Sim	Sim	Sim (Agenda)	Sim	Sim (Calendário)
Mensagens	<b>Não</b>	Sim	Sim	Sim	Sim
Boletim/ Resultados de questionários	Sim	Sim	Sim	Sim	<b>Não</b>
Usuários online	Sim	Sim	Sim	Sim (chat)	<b>Não</b>
Meu calendário	Sim	Sim	Sim	Sim	Sim
Baixar o conteúdo completo do curso	<b>Não</b>	<b>Não</b>	Sim (todos os arquivos)	Sim	<b>Não</b>
Informação pessoal/ Perfil de usuários	Sim	Sim	Sim	Sim	Sim
Cancelar a inscrição de um curso	Sim	Sim	<b>Não</b>	Sim	Sim
Software livre - GPL license	<b>Não</b>	Sim	Sim	Sim	Sim
Extensão a plataformas móveis	Sim	<b>Não</b>	<b>Não</b>	<b>Não</b>	<b>Não</b>

Tabela 6 – Tabela de comparação categorias: Permissões e papéis, página principal

Os AVA mencionados possuem papéis de usuários pré-estabelecidos (Tabela 6): BlackBoard: instrutor, assistente de ensino, construtor curso, nivelador, estudante, convidado; Moodle: Administrador do site, gerente, criador do curso, professor editor, professor, estudante, convidado, usuário; TelEduc: Coordenadores, formadores, alunos, visitantes, colaboradores; Sakai: instrutores, assistente de professor, estudante; Noosfero: Administrador do sistema, moderador, membro, administrador de perfil. Apesar do Noosfero possuir estes papéis preestabelecidos possui a vantagem de criação de papéis customizados, onde o administrador define quais as permissões para cada um deles. Esta funcionalidade foi desenvolvida recentemente e se encontra em aprovação do *merge request*.

O Noosfero carece de uma melhor estruturação dos conteúdos para o gerenciamento de um curso. Conforme evidenciado na Tabela 3, a plataforma não permite tratar os conteúdos na forma de cursos e tampouco a personalização de seus formatos (formato dos tópicos, duração). Além disso a ferramenta de calendário disponível mostra apenas as datas dos eventos criados, não disponibilizando aos alunos todas as atividades e duração do curso de forma limpa e organizada.

Quando analisados os itens da categoria de página principal (Tabela 6) é verificado que o Noosfero necessita de um template próprio para cursos que reúne as informações principais dos alunos. Algumas dessas informações necessitam apenas de uma melhor disposição pela interface, como calendário e mensagens. Já outras exigem implementação como um sistema de avisos, usuários online e boletim.

Neste capítulo foi apresentado o conceito de ambientes virtuais de aprendizagem, bem como as suas funcionalidades e uma comparação com o Noosfero. Este estudo é importante, uma vez que considerando esta análise, algumas funcionalidades serão implementadas no desenvolvimento prático deste trabalho, como proposto na seção 4.3. No entanto, é necessário explorar o Noosfero (Capítulo 4) assim como os processos e desenvolvimento da comunidade. Essa compreensão é importante, uma vez que o Comunidade.UnB utiliza essa plataforma que será o foco principal desse trabalho.



## 4 Evoluindo uma plataforma de rede social

Neste capítulo é discutida a plataforma Noosfero, desde sua arquitetura até os processos de desenvolvimento da comunidade. Além disso, é apresentado o Comunidade.UnB, rede de colaboração livre, no qual são propostas evoluções para que se torne um ambiente híbrido utilizado por professores e alunos.

### 4.1 Noosfero

O Noosfero <sup>1</sup> é uma plataforma livre para criação de redes sociais, desenvolvida em 2007, pela Cooperativa de Tecnologias Livres - Colivre <sup>2</sup>, sob a licença AGPL<sup>3</sup> V3, com a proposta de facilitar a criação de redes sociais personalizadas, livres e autônomas e a geração de conteúdo colaborativo.

Além das funcionalidades de rede social, com foco na produção e compartilhamento de conteúdo, o Noosfero permite que dentro da rede cada usuário e comunidade tenha o seu espaço com total flexibilidade de personalização visual e gerenciamento de conteúdo. São exemplos de portais que utilizam o Noosfero: o Participa BR <sup>4</sup>, Stoa<sup>5</sup>, Portal da FGA<sup>6</sup> e o novo Portal do Software público Brasileiro <sup>7</sup>.

O Noosfero foi desenvolvido na linguagem de programação Ruby, atualmente na versão 2.2.0, e utiliza o *framework* aplicações web *Ruby on Rails* <sup>8</sup>, versão 3.2.21. E utiliza também padrões arquiteturais de software Model-View-Controller (MVC), e o padrão de plugins que serão apresentados na seção 4.1.2.

A escolha da linguagem *Ruby* foi decisiva no Noosfero, pois possui uma sintaxe simples, que facilita a manutenibilidade do sistema, característica importante em projetos de software livre que tendem a atrair colaboradores externos (MEIRELLES, 2013). A escolha do *Rails* foi influenciada pelos seus conceitos básicos que auxiliam em sua produtividade: “Não Repita a Si Mesmo” (DRY-*Don't Repeat Yourself*) e “convenção sobre configuração” (*Convention Over Configuration*) (AKITA, 2006). Além disso, por questões de segurança, o Noosfero é homologado para a versão *stable* do *Debian*, acompanhando a versão do *Ruby* e do *Rails* dessa distro.

<sup>1</sup> Disponível em: <<http://www.noosfero.org>>

<sup>2</sup> Disponível em: <<http://www.colivre.coop.br>>

<sup>3</sup> Licença de software GNU

<sup>4</sup> Disponível em: <<https://www.participa.br/>>

<sup>5</sup> Disponível em: <<http://stoa.usp.br/>>

<sup>6</sup> Disponível em: <<http://fga.unb.br/>>

<sup>7</sup> Disponível em: <<https://portal.softwarepublico.gov.br/>>

<sup>8</sup> Disponível em: <<http://rubyonrails.org/>>

### 4.1.1 Práticas de desenvolvimento da comunidade Noosfero

Para que novos desenvolvedores colaborem com o Noosfero, a comunidade utiliza em seu próprio *site* uma seção para o desenvolvimento. Nessa seção, encontram-se os manuais com os passos para instalação do ambiente, descrição dos *plugins* disponíveis, instruções para a personalização da plataforma através de temas além de informações arquiteturais da plataforma. Para realizar o controle de itens a fazer, como a implementação de novas funcionalidades ou correção de bugs é utilizado um *Issue Tracker* do repositório oficial no GitLab <sup>9</sup>.

Uma vez que o desenvolvedor ou usuário tenha registro no GitLab, é possível utilizar o *Issue Tracker* para cadastrar novas funcionalidades ou *bugs* de maneira simples seguindo os seguintes passos:

1. preencher o campo título;
2. definir a descrição do item, se existir, é necessário especificar com qual *plugin* o item está relacionado;
3. associar um desenvolvedor responsável pela sua implementação *issue*;
4. definir as *labels* da funcionalidade, onde é especificado ao que está relacionado a nova *issue*.

Após a criação da *issue*, todos os membros da comunidade podem visualizar os dados e especificações, bem como discutir sobre os propósitos e formas de implementação da funcionalidade.

A Figura 1 evidencia o *Issue Tracker* do Noosfero onde é possível verificar os itens que foram mapeados e seus respectivos *status*, se ainda estão abertas ou fechadas, além de um filtro para os autores e os marcadores envolvidos à cada uma delas. Dessa forma, é possível priorizar os itens identificados como sendo de maior importância para os usuários e desenvolvedores.

Priorizando a comunicação entre os desenvolvedores, a comunidade compartilha canais <sup>10</sup> de comunicação pelo IRC (*Internet Relay Chat*). Nesses os desenvolvedores podem compartilhar conhecimento e realizarem discussões técnicas sobre as implementações.

A implementação é realizada pelo desenvolvedor que tem a responsabilidade de manter a qualidade do código produzido bem como realizar todos os testes relacionados à funcionalidade implementada. Uma vez que este primeiro passo esteja concluído o código é submetido a um *merge-request* onde um dos desenvolvedores do *core* efetua a revisão

<sup>9</sup> Disponível em: <<https://gitlab.com/noosfero/noosfero>>

<sup>10</sup> Canais do Noosfero: #noosfero-br e #noosfero

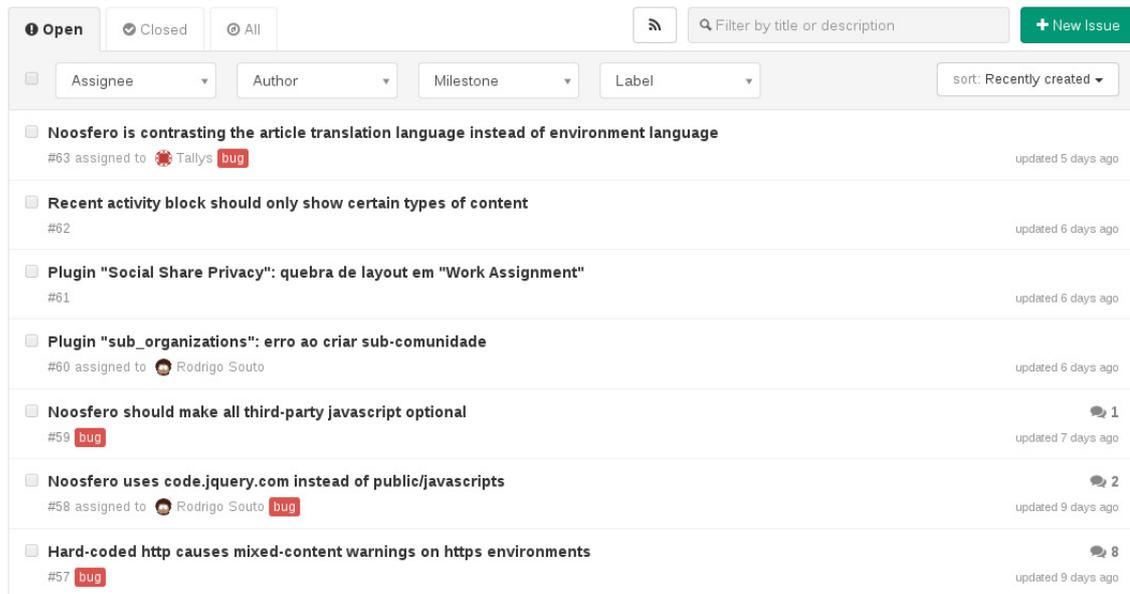


Figura 1 – Issue Tracker no GitLab

para verificar se está de acordo com os padrões esperados, e aprova ou não, a inclusão do código na *branch* principal do Noosfero.

A comunidade Noosfero recomenda práticas de desenvolvimento como o TDD, *Test Driven Development* ou Desenvolvimento orientado a testes), combinado com o BDD <sup>11</sup> (*Behavior Driven Development* ou Desenvolvimento Guiado por Comportamento, que auxiliar o desenvolvedor a criar testes e integrar regras de negócio com a linguagem de programação, mantendo o foco no comportamento do software (NORTH, 2006).

Para realizar o controle de versão e gerenciamento do código fonte é utilizado o *Git*, uma ferramenta livre de versionamento distribuído de código fonte. O repositório oficial do Noosfero encontra-se no software livre Gitlab com um espelho no Github<sup>12</sup>. Na página de desenvolvimento da comunidade existe uma série de recomendações sobre o envio de *patches* para o Noosfero, incluindo como versionamentos e solicitações de inclusão de seu *patch*, ou *merge-request*.

#### 4.1.2 Arquitetura

Para a evolução de um software de forma adequada é importante o conhecimento da arquitetura do sistema, para não comprometer todo o planejamento realizado na concepção do projeto. Desse modo conhecer e entender a arquitetura de funcionamento do Noosfero é uma etapa fundamental para o desenvolvimento de novas funcionalidades para a plataforma.

<sup>11</sup> <<https://cukes.info/>>

<sup>12</sup> <<https://github.com/noosfero/noosfero>>

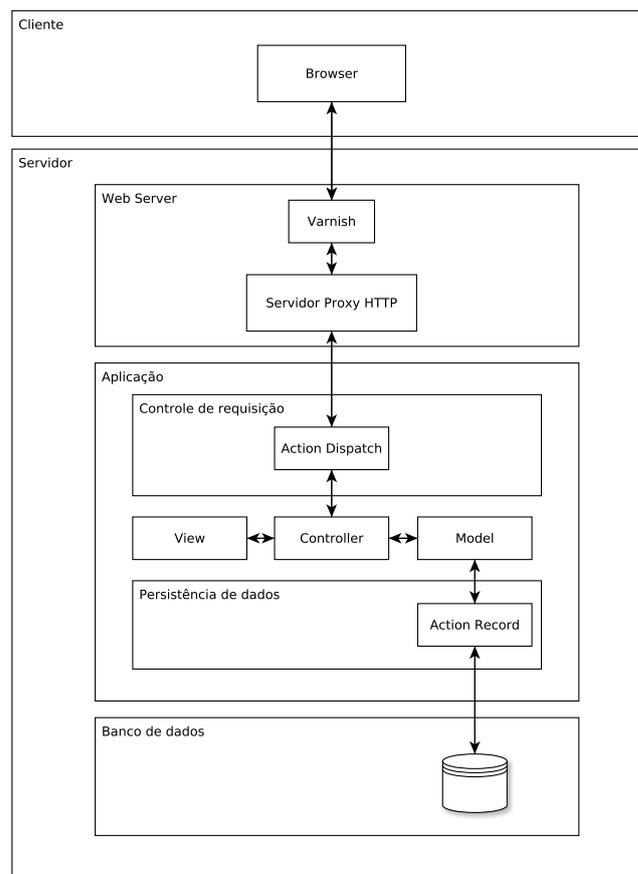


Figura 2 – Arquitetura do Noosfero

A Figura 2 apresenta uma visão de alto nível da arquitetura do Noosfero. Basicamente tem-se uma arquitetura cliente-servidor onde o cliente via **Browser** solicita um conteúdo ou uma função para o servidor Noosfero, que aguarda requisições de entrada para processá-las e compartilhar recursos com o cliente.

Do lado do servidor, temos basicamente três camadas de abstração: *Web-Server*, *aplicação* e *Banco de dados*. Na primeira camada temos dois componentes responsáveis por processar e acelerar todas as requisições de entrada e saída:

- **Varnish:** é um acelerador para sites web dinâmicos com alto volume de conteúdo que utiliza *proxy* HTTP Reverso. Sua eficiência deve-se ao fato dele armazenar o conteúdo HTTP requisitado na memória RAM, fazendo com que o servidor não consulte e processe diversas vezes o mesmo conteúdo solicitado.
- **Servidor Proxy HTTP:** pode ser utilizado o Apache ou Nginx <sup>13</sup> que ajudam a melhorar o desempenho funcionando como um servidor proxy HTTP reverso que processa as requisições de entrada e saída e as encaminha para a aplicação executá-las.

<sup>13</sup> <<http://nginx.org>>

Na camada da aplicação, foi considerada uma camada responsável pelo controle de requisições, que é efetuada pelo componente **Action Dispatch**, que lida com o mapeamento de todas as requisições, *cookies* e sessão para suas respectivas *controllers*.

Na aplicação, utiliza-se o padrão de arquitetura de software MVC<sup>14</sup> onde a **controller** controla o fluxo da aplicação, relacionando as entidades de *model* e de *view* através de chamadas de métodos. A **model** representa as entidades do domínio da aplicação, onde a lógica do sistema são implementadas. A **view** é a interface de comunicação com o usuário, ou seja as páginas HTML apresentadas no navegador.

Ainda na camada da aplicação tem-se o **Active Record** que é um ORM<sup>15</sup>, um mapeador entre objetos e registros de uma tabela, onde cada classe de modelo possui uma tabela correspondente à ela no banco de dados.

Por fim temos a camada de banco de dados que recebe requisições da camada de persistência de dados e por meio de um sistema gerenciador de banco de dados (SGBD) realizam operações na base de dados.

### 4.1.3 Modelo de domínio

Para Larman (2002, p. 160), um modelo de domínio é a representação visual de classes conceituais ou objetos do mundo real em um domínio, que também podem ser chamados de modelos conceituais, modelos de objetos de domínio e modelos de objetos de análise. Dessa forma, é necessário o seu entendimento para realizar a evolução da plataforma.

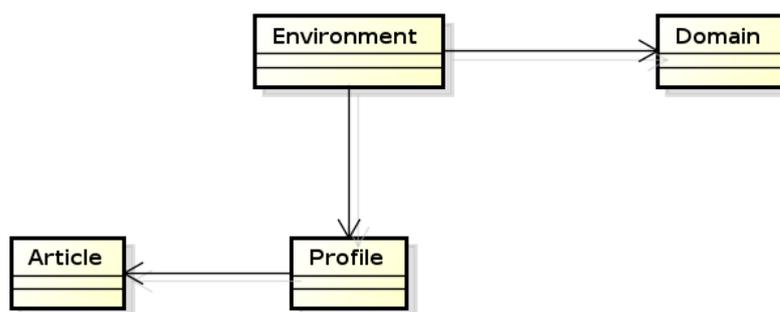


Figura 3 – Relações entre entidades de domínio ambiente, domínio e perfis. Extraído de: (BUCHER, 2013)

O Noosfero é uma plataforma que tem suporte a vários ambientes de rede social dentro de uma mesma instalação. A Figura 3 mostra-se o funcionamento geral do Noosfero com suas quatro principais classes **Domain**, **Environment**, **Profile** e **Article** (Em

<sup>14</sup> Model-view-controller

<sup>15</sup> object-relational mapping

português: Domínio, Ambiente, Perfil e Artigo respectivamente). Analisando o modelo verifica-se que na implementação é necessário que exista pelo um domínio e partir disso é possível criar várias instâncias de Ambiente na aplicação.

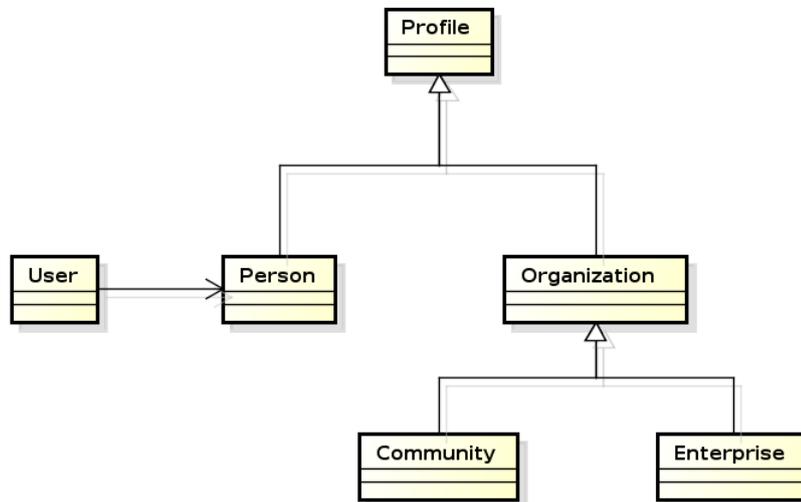


Figura 4 – Entidades de domínio: tipos de perfis. Extraído de: (BUCHER, 2013)

A entidade Profile é uma generalização das entidades *Person* (Pessoa) e *Organization* (Organização), como pode ser visto na Figura 4. Nesse mesmo modelo percebe-se que Organization é especializada nas entidades concretas *Community* (Comunidade e Enterprise (Empreendimento). A herança é um mecanismo pelo qual qual uma classe sub-classe pode estender uma super-classe, onde basicamente isola-se métodos ou atributos em comum dentro de uma classe pai (super-classe), enquanto as especialidades são responsabilidade das classes filhas (sub-classe).

Por questões de design do código da aplicação foi criada uma entidade *User*, ou Usuário, que é mantida separada da entidade Pessoa, que é quem implementa a lógica de autenticação da aplicação. Desta forma a lógica de autenticação fica separada da lógica de visualização e personalização do perfil.

Por fim, as entidades mostradas na Figura 5 representam os principais tipos de conteúdos disponíveis no Noosfero, onde a classe *Article*, ou Artigo, é uma especialização de todos os conteúdos disponíveis como: artigos de texto, pastas, blogs, galerias de imagens, fórum, arquivos e feeds de notícias.

O modelo de domínio aqui apresentado contempla o *core* do noosfero. Para o acréscimo de melhorias e funcionalidades é necessário compreender a visão arquitetural dos *plugins* d a plataforma, que será abordado na próxima seção.

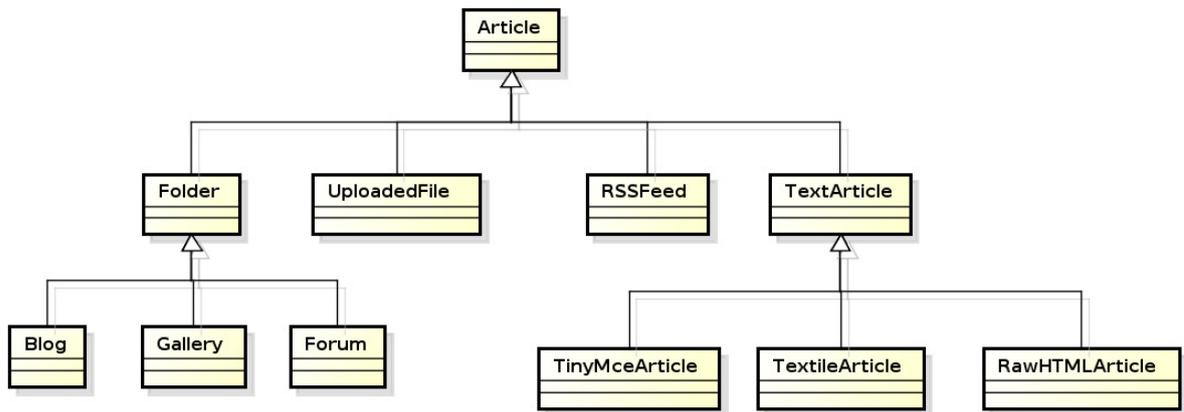


Figura 5 – Entidades de domínio: tipos de artigos. Extraído de: (BUCHER, 2013)

#### 4.1.4 Plugins

Como é software em constante evolução, a arquitetura do Noosfero foi criada para ser altamente expansível, fazendo-se o uso de *plugins*. Essa arquitetura permite que em cada ambiente fique a critério do usuário quais os *plugins* ou novas funcionalidades serão habilitadas, o que torna o sistema flexível e modular.

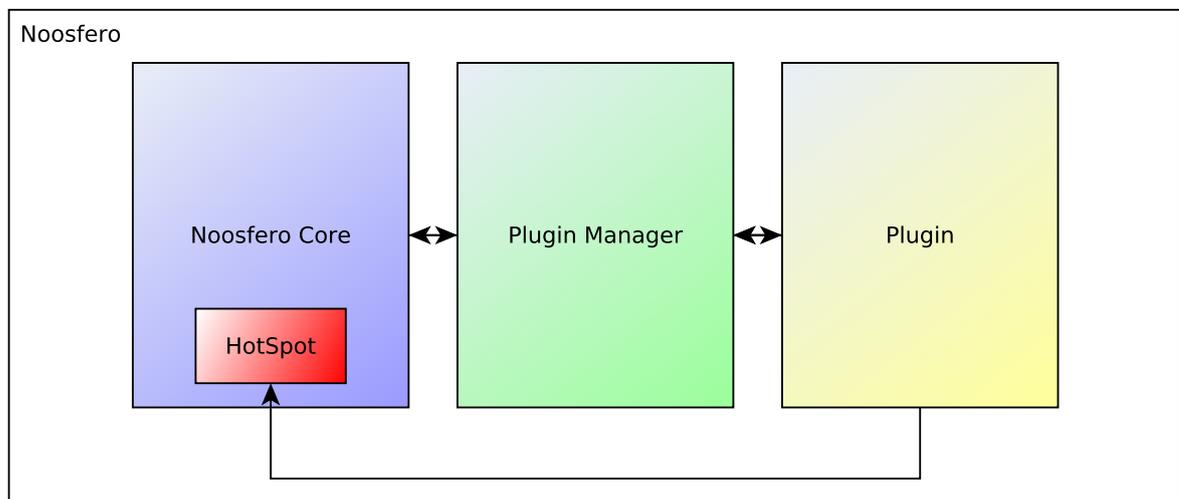


Figura 6 – Estrutura de Plugins

A Figura 6 é uma abstração alto nível do funcionamento interno dos plugins no Noosfero. No *core* do Noosfero temos os *hotspots*, que são pontos de flexibilidade que permitem associar diferentes comportamentos na execução do sistema, permitindo a inserção de trechos de código e ou alteração de um determinado método sem comprometer suas funcionalidades básicas.

Os *hotspots* são gerenciados por uma camada de abstração denominada *Plugin Manager*, ou Gerenciador de *Plugins*, que são chamadas pelo *core* através de um método

principal conhecido como *dispatch*. Basicamente, o ciclo de execução pode ser descrito da seguinte maneira: durante a execução de alguma funcionalidade o método *dispatch* é invocado por alguma funcionalidade do core, deste modo o gerenciador de plugins verifica todos os *Plugins* que fazem uso daquele *hotspot* e encaminha para cada um deles a execução de suas ações de acordo com sua implementação.

Essa arquitetura extensível adotada pelo Noosfero auxilia no controle da qualidade de código das novas funcionalidades. A camada de *Plugins* localiza-se fora do código do seu núcleo, em uma pasta denominada *plugins* em que desenvolvedor cria novas funcionalidades sem modificar o comportamento *core* do Noosfero, fazendo uso dos *dispatch*.

Adicionalmente como mencionado na Seção 4.1.1, o Noosfero faz uso de testes para manter a integridade de seu código, desse modo esta prática é estendida aos *plugins* que devem englobar seus respectivos testes para evitar a inserção de *bugs* e mudanças inesperadas no comportamento do sistema.

Assim sendo a evolução proposta para este trabalho será realizada através de plugins, como proposto na Seção 4.3

## 4.2 Comunidade UnB

A Comunidade.UnB é uma rede colaboração livre desenvolvida para que alunos, professores e servidores técnico-administrativos tenham um ambiente virtual de criação e compartilhamento de conhecimento colaborativo. É um ambiente virtual para o compartilhamento de ideias, produção de conteúdo colaborativo de modo que possam publicá-los para que possa ser de utilidade para outras pessoas ou parcelas da sociedade, uma vez que acredita-se que este é um dos papéis de uma Universidade (BUCHER, 2013).

Inspirada na rede social de colaboração Stoa<sup>16</sup>, da Universidade de São Paulo (USP), a Comunidade.UnB foi criada em 2013 a partir de um trabalho de conclusão de curso de Daniel Costa Bucher e até então está disponibilizada em ambiente de testes. Permite ao usuário a criação de seu espaço pessoal e a liberdade de publicar suas ideias, ou o conteúdo que desejar, por exemplo, na forma de blogs pessoais, blogs de disciplinas, pesquisas em andamento, dentre outras, além de compartilhar esse conteúdo para ser acessível para outros usuários dentro e fora da rede.

O Noosfero, descrito na Seção 4.1, foi a plataforma utilizada para o desenvolvimento da Comunidade.UnB, por dispor de um grande potencial e devido às suas funcionalidades avançadas, que permitem a criação e o compartilhamento de conteúdo de forma satisfatória. Além de dispor de uma comunidade ativa e de posição geográfica favorável em relação ao seu núcleo de desenvolvimento que se encontra no Brasil, facilitando

<sup>16</sup> Disponível em: <<https://social.stoa.usp.br/>>

The screenshot shows the 'Comunidade UnB' website interface. At the top, there is a navigation bar with links for 'Entrar ou Registre-se', 'Pessoas', 'Comunidades', 'Conteúdos', and 'Eventos'. Below this is the 'Comunidade UnB' logo and a search bar. The main content area is titled 'Sobre o Noosfero' and includes a description of the platform as a web-based social and solid economy network. It also mentions the 'Noosfero MES-UnB' project, which is a form of knowledge acquisition for a software maintenance and evolution course. A sidebar on the left lists various community features like 'Community's profile', 'Blog', 'Agenda', and 'Fórum'. A right sidebar contains a 'Sobre o Noosfero' menu with options like 'Site Oficial', 'Desenvolvimento', and 'Lista de E-mail', along with a 'MES 01/2014' section listing 'Dívida Técnica' and 'Work Progress'.

Figura 7 – Exemplo do uso do Comunidade.UnB na disciplina de MES.

a comunicação com os seus principais desenvolvedores.

Apesar da Comunidade.UnB estar disponibilizada como um ambiente de testes e não possuir uma vasta divulgação pela Universidade de Brasília, no seu primeiro ano de criação contava com 153 usuários e 14 comunidades. Ao final de maio de 2015 contabilizava 376 usuários e 30 comunidades demonstrando que houve crescimento.

Alguns professores adotaram a Comunidade.UnB como um ambiente de apoio ao Moodle, AVA oficial adotado pela UnB. Na Figura 7 é apresentado um exemplo de seu uso na disciplina de Manutenção e Evolução de Software (MES) ministrada pelo professor Paulo Roberto Miranda Meirelles.

Esse é um exemplo de comunidade criada dentro do Comunidade.UnB, que possui características que se assemelham ao ambiente Moodle, como evidenciado na Seção 3.3, carece de alguns funcionalidades importantes mas com a vantagem da possibilidade de acesso do público ao conteúdo, e a continuidade do conteúdo desenvolvido por outras pessoas que eventualmente se juntem ao longo do tempo. Vale lembrar que em uma publicação de conteúdo os níveis de privacidade podem ser alterados basicamente entre públicos e privados.

As limitações da plataforma e a proposta de uso da Comunidade.UnB por professores para a criação de disciplinas, estimulam o desenvolvimento de funcionalidades que levem à plataforma Noosfero a assemelhar-se a um ambiente virtual de aprendizagem. Dessa maneira na Seção 4.3 é apresentada a proposta de estudo desse trabalho para desenvolvimento de funcionalidades e minimização das diferenças entre o Noosfero e os

ambientes virtuais de aprendizagem.

## 4.3 Desenvolvimento do NoosferAVA

Nesta seção apresentaremos as funcionalidades que serão desenvolvidas para contribuir para a adequação do Noosfero a um ambiente virtual de aprendizagem, para exercer a função de apoio aos AVA adotados pela universidade. Serão apresentados os requisitos funcionais levantados para que esta possa suprir as necessidades levantadas.

Além das funcionalidades relatadas, julgou-se necessária a evolução do *plugin Comunidade.UnB*, iniciado por Bucher (2013), com objetivo de integrar o Noosfero com o serviço de Lightweight Directory Access Protocol (LDAP). Uma base LDAP é utilizada pela UnB para manter os dados de todos os seus alunos e funcionários.

As funcionalidades serão apresentadas utilizando o formato de histórias de usuários<sup>17</sup>, de acordo com as práticas ágeis. Além disso, os critérios de aceitação serão expostos no formato de cenários de uso, utilizando BDD.

### 4.3.1 Evolução do *plugin Comunidade.UnB*

O objetivo do *plugin* do Comunidade.Unb é que o usuário tenha acesso através dos mesmos dados utilizados para acessar outros serviços, como, por exemplo, o serviço de matrícula<sup>18</sup> para alunos ou o serviço de lançamento de notas para professores.

A versão anterior do *plugin* permitia que apenas os novos usuários realizassem o cadastro de suas matrículas, dessa maneira aqueles que já estavam cadastrados não tinham acesso ao portal. Devido às limitações de acesso definidas pelo *plugin*, onde apenas usuários com matrícula cadastrada tem acesso ao sistema.

Foi criada nova versão do *plugin* que permite cadastrar a matrícula dos usuários que já possuem registro no Comunidade.UnB. Dessa maneira, o *plugin Comunidade.UnB* poderá ser ativado permitindo a autenticação apenas dos usuários cadastrados no LDAP da universidade. Abaixo segue a história de usuário com seus respectivos cenários de uso.

#### Histórias de usuário

##### 1. Autenticação via LDAP

**Como** um aluno da Universidade Brasília

**Gostaria de** me autenticar na rede através da minha matrícula e senha

**Para** utilizar os mesmos dados de cadastro da UnB.

<sup>17</sup> Em inglês *User Stories*(US)

<sup>18</sup> Disponível em: <<https://matriculaweb.unb.br>>

Cenários de uso:

a) Acesso sem matrícula cadastrada

[**Dado**] que sou aluno da UnB

[**E**] possuo cadastro ativo na base de dados da UnB

[**E**] possuo cadastro ativo na base de dados do Comunidade.UnB

[**Quando**] eu acessar o portal

**Como** um aluno da Universidade Brasília

[**Então**] devo ser direcionado para uma página com o título “Cadastrar Matrícula”

[**E**] devo ver os campos “Matrícula”, “Senha” e “confirmação de senha” em branco.

b) Registro de matrícula de usuários existentes

[**Dado**] que sou aluno da UnB

[**E**] me encontro na página de Cadastrar matrícula do Comunidade.UnB

[**Quando**] eu preencher os campos

“matrícula” com “100103979”,

e “senha” com a senha da base dados da UnB,

e “confirmação de senha”

[**E**] clicar no botão “Registrar”

[**Então**] eu devo ser direcionado para meu perfil

### 4.3.2 Evolução do *plugin Work Assignment*

Nesta subseção serão apresentadas um primeiro levantamento das histórias correspondentes a evolução do *plugin Work Assignment*. Até então, o *plugin* possui apenas a funcionalidade de permitir o envio de arquivos para o servidor em um determinado período de tempo.

A proposta de evolução é a criação de um sistema de notas que permita ao professor atribuir notas as atividades enviadas e dessa maneira acompanhar a situação de cada aluno. Do ponto de vista do aluno, o mesmo poderá visualizar todas as notas das atividades em cada disciplina, avaliando se o seu desempenho está satisfatório.

Além disso, é necessário uma evolução da funcionalidade que permite a definição do tempo restante para o envio, que atualmente é realizado de maneira manual, não possibilitando o estabelecimento de intervalos de tempo.

Histórias de usuário

1. Definir tempo restante

**Como** um professor

**Gostaria de** definir o tempo restante para cada atividade no Work Assignment

**Para** gerenciar o envio de atividades pelos alunos em um determinado período.

Cenários de uso:

a) Definir tempo

[**Dado**] que esteja logado como professor

[**E**] selecione a opção “Gerenciar conteúdo”

[**Quando**] eu clicar em trabalho a ser entregue

[**Então**] devo visualizar a opção “Ativar tempo de entrega”

[**E**] informar a data e hora máxima para envio da atividade.

b) Modificar tempo

[**Dado**] que esteja logado como professor

[**E**] eu tenha atividades a ser entregue em aberto

[**E**] eu esteja visualizando as informações da atividade

[**Quando**] eu selecionar opção “Editar”

[**E**] visualizar a opção “Tempo de entrega”

[**Então**] deve ser permitido que eu altere o tempo definido

[**E**] visualize o resultado da alteração após a confirmação.

c) Permitir entrega de atividades após período

[**Dado**] que esteja logado como professor

[**E**] selecione a opção “Gerenciar conteúdo”

[**E**] vavegar até a página trabalho a ser entregue

[**Quando**] selecionar a opção “Ativar tempo de entrega”

[**Então**] devo visualizar a opção “Permitir entrega após o período”

[**Para**] que eu possa selecioná-la.

2. Tempo restante de atividades

**Como** um aluno

**Gostaria de** visualizar o tempo restante da atividade

**Para** enviar um arquivo na data correta.

Cenários de uso:

a) Visualizar tempo

[**Dado**] que esteja logado como Aluno  
[**E**] eu esteja inscrito em um Curso  
[**E possui atividade em aberto**]  
[**Quando**] eu selecionar a opção de visualizar a atividade  
[**Então**] eu devo visualizar se a atividade está em aberto  
[**E**] qual o tempo restante para o envio da atividade.

### 3. Professor gerencia notas

**Como** um professor

**Gostaria de** gerenciar as notas dos integrantes da comunidade

**Para** manter o controle sobre a pontuação de todos os alunos.

Cenários de uso:

#### a) Definir grupo de atividades

[**Dado**] que esteja logado como professor  
[**E**] que a funcionalidade de notas esteja habilitada no plugin  
[**E**] selecionar a opção “Gerenciar notas”  
[**E**] e eu selecionar o curso desejado  
[**Quando**] eu clicar em “Cadastrar grupo de atividades”  
[**E**] eu preencho os campos  
“Nome do grupo”,  
e “Lista de Atividades”  
[**E**] eu clique em “Salvar”  
[**Então**] recebo uma mensagem de confirmação  
[**E**] visualizo todas os grupos de atividades criados.

#### b) Visualizar notas de todos os alunos de uma determinada atividade

[**Dado**] que esteja logado como professor  
[**E**] que a funcionalidade de notas esteja habilitada no plugin  
[**E**] selecionar a opção “Gerenciar notas”  
[**E**] e eu selecionar o curso desejado  
[**Quando**] eu selecionar atividade  
[**E**] algum aluno tenha enviado a atividade  
[**Então**] devo visualizar todas as atividades enviadas e suas respectivas notas.

#### c) Visualizar notas de todos os alunos de um grupo de atividades

[**Dado**] que esteja logado como professor

[E] que a funcionalidade de notas eteja habilitada no plugin  
[E] selecionar a opção “Gerenciar notas”  
[E] e eu selecionar o curso desejado  
[Quando] eu clicar em “Visualizar notas por grupo de atividades”  
[E] algum aluno tenha enviado a atividade  
[Então] devo visualizar todas as atividades daquele grupo e suas respectivas notas.

#### 4. Atribuir notas aos alunos

**Como** um professor

**Gostaria de** atribuir notas as atividades enviadas pelos alunos

**Para** avaliar o rendimento de cada um deles.

Cenários de uso:

##### a) Atribuir notas

[Dado] que esteja logado como professor  
[E] que a funcionalidade de notas eteja habilitada no plugin  
[Quando] selecionar uma atividade na lista de atividades  
[Então] devo visualizar a opção atribuir nota  
[E] devo visualizar o campo “nota” para preenche-lo.

##### b) Alterar notas

[Dado] que esteja logado como professor  
[E] que a funcionalidade de notas eteja habilitada no plugin  
[Quando] selecionar uma atividade na lista de atividades  
[E] atividade já possua notas  
[Então] devo visualizar a opção alterar nota  
[E] devo visualizar o campo “nota” para preenche-lo.

#### 5. Publicar notas aos alunos

**Como** um professor

**Gostaria de** publicar as notas atribuídas as atividades

**Para** que os alunos possam visualizar suas respectivas avalizações

Cenários de uso:

- a) Disponibilizar notas de uma determinada atividade  
[**Dado**] que esteja logado como professor  
[**E**] que a funcionalidade de notas esteja habilitada no plugin  
[**Quando**] selecionar uma atividade na lista de atividades  
[**Então**] devo visualizar a opção “permitir visualização”  
[**E**] devo ter a opção de ativá-lo.
- b) Omitir notas de uma determinada atividade  
[**Dado**] que esteja logado como professor  
[**E**] que a funcionalidade de notas esteja habilitada no plugin  
[**Quando**] selecionar uma atividade na lista de atividades  
[**Então**] devo visualizar a opção “permitir visualização”  
[**E**] devo ter a opção de desativá-lo.

#### 6. Aluno visualiza notas

**Como** um aluno

**Gostaria de** visualizar minhas notas

**Para** inteira-me sobre minha pontuação nas atividades.

Cenários de uso:

- a) Visualizar cursos com notas disponíveis  
[**Dado**] que estou logado como “Hebert”  
[**E**] o plugin Work Assignment esteja ativo  
[**E**] participo de alguma comunidade que utilize o plugin de notas  
[**E**] existem notas de atividades disponíveis  
[**E**] o professor tenha permitido sua visualização  
[**Quando**] eu navegar até a página “Minhas notas”  
[**Então**] tenho que ver todos as comunidades (cursos) com notas disponíveis.
- b) Detalhar notas de cada curso  
[**Dado**] que estou logado como “Hebert”  
[**E**] o plugin Work Assignment esteja ativo  
[**E**] participo de alguma comunidade que utilize o plugin de notas  
[**E**] existem notas de atividades disponíveis  
[**E**] o professor tenha permitido sua visualização

[E] navegar até a página “Minhas notas”

[E] visualizar todos as comunidades (cursos) com notas disponíveis

[Quando] eu selecionar a “Visualizar detalhes” de alguma comunidade

[Então] tenho que ver todos os grupos de atividades

[E] as notas disponibilizadas pelo professor.

No contexto da UnB, o desenvolvimento dessa funcionalidade é importante, pois segundo o documento da instituição UnB (2009, p. 37), as notas para os alunos auxilia assimilação progressiva de conhecimento, e sabe-se que permite um maior *feedback* para o aluno, pois nem sempre fica transparente o seu desempenho ao longo do semestre.

Em resumo, as funcionalidades apresentadas neste capítulo irão colaborar para uso do Comunidade.UnB, tornando-o uma plataforma híbrida entre um ambiente virtual de aprendizagem e uma rede social, para a troca de conhecimento através das comunidades e perfis de usuários. Assim, permitindo englobar departamentos, organizações e projetos específicos da Universidade de Brasília.

## 5 Considerações preliminares

Nesta primeira etapa do trabalho, foram elaboradas algumas melhorias para a evolução do Comunidade.UnB. Dentre elas, a atualização do Noosfero da Comunidade.UnB<sup>1</sup>, que ainda se encontrava em *Rails* 2 no Debian 6 e não dispunha de suporte pela comunidade. Da mesma forma, o tema teve que ser evoluído para ficar compatível com a nova versão da plataforma.

Como evidenciado na Seção 4.3.1, realizou-se a evolução do *plugin* Comunidade.UnB, permitindo que os usuários já cadastrados acessem desde que tenham cadastro no LDAP da UnB. Essa funcionalidade ainda não está habilitada mas pode ser visualizada no repositório<sup>2</sup> do Noosfero para o Portal UnB Gama.

Como as demais funcionalidades a serem implementadas dizem respeito a evolução do *plugin Work Assignment*, que permitirá aos professores utilizarem o Comunidade.UnB como um ambiente virtual de aprendizagem. Dessa maneira espera-se que o Comunidade.UnB transforme-se em um ambiente híbrido para os alunos e professores compartilharem o conhecimento de maneira construtiva e evolutiva.

### 5.1 Pesquisa com alunos

Para melhor prosseguirmos com este trabalho, realizamos uma pesquisa junto aos alunos da FGA sobre o uso de redes sociais em seu dia a dia e sobre o seu uso para apoio as disciplinas cursadas na universidade. Para isso, foi criado um questionário com quatro questões de múltipla escolha, e para os itens associou-se a uma escala tipo *Likert* de cinco pontos: Nunca; Raramente; Às vezes; Frequentemente; Sempre (Instruções de preenchimento no Apêndice A). Com este tipo de escala, pode-se mapear todas as respostas seguindo a lógica em que há duas alternativas negativas, duas alternativas positivas e uma intermediária.

Nesta seção, apresentamos as questões elaboradas (Apêndice B) e as respostas coletadas através da ferramenta *Google Forms*. O questionário foi aplicado na rede social *Facebook*, em grupos específicos da FGA. Foram coletadas, ao todo noventa e quatro respostas voluntárias ao questionário. Abaixo segue as questões com as respectivas análises de seus resultados, que estão representados graficamente no Apêndice C.

<sup>1</sup> Não possui DNS associado mas pode ser acessado pelo endereço <<http://164.41.9.37/>>

<sup>2</sup> Repositório: <<https://gitlab.com/unb-gama/noosfero>> Branch: ldap\_unb\_plugin\_rails3

## Com que frequência você utiliza redes sociais?

A primeira pergunta tinha como objetivo levantar com que frequência os alunos utilizam as redes sociais. Na Figura 8, nota-se que 98% dos alunos que responderam ao questionário utilizam as redes sociais sempre ou frequentemente, o que evidencia o fato que atualmente os estudantes dessa geração estão inseridos neste contexto.

## Você utiliza alguma rede social como ferramenta de apoio as disciplinas?

O intuito da segunda pergunta foi verificar se os alunos utilizam alguma ferramenta de redes social como ferramenta de apoio as disciplinas. Na Figura 9, verifica-se que apenas 16% dos alunos às vezes utilizam e de outro ponto de vista 83% dos alunos fazem uso da rede para tal. Demonstrando que mesmo a universidade utilizando AVA os alunos se apoiam em redes sociais para discutir e compartilhar conteúdos referentes as disciplinas cursadas.

## Os professores incentivam o uso de redes sociais para suas disciplinas?

O objetivo desta pergunta foi verificar se mesmo que a universidade indique o uso de AVA se os professores incentivam o uso das redes sociais em suas disciplinas. Na Figura 10, percebe-se que pouco mais da metade dos alunos (52 %) tem suas respostas em um ponto intermediário do questionário, mostrando que os professores são bem imparciais quanto a isso. Apesar que do restante, tem-se mais respostas favoráveis do que contra.

## Mesmo que o professor não recomende o uso de redes sociais para discussão de conteúdos de suas disciplinas, você as utiliza?

Na quarta e última pergunta buscou-se investigar se os alunos utilizam as redes sociais como uma ferramenta de apoio as disciplinas mesmo que os professores não recomende seu uso para tal. Dos que responderam 75% não levam em consideração tais recomendações e fazem o uso, demonstrando que os alunos preferem estar nas redes sociais onde expõem suas opiniões a fim de promover o compartilhamento de conteúdo (Resultados na Figura 11).

Esta pesquisa exploratória mostra que mesmo com a existência de AVA os alunos a utilizam as redes sociais, que não foram criadas com esse objetivo, para a discussão de assuntos relacionados ao ambiente acadêmico.

## 5.2 Cronograma

O cronograma para o trabalho realizado neste TCC é apresentado na Tabela 7. As atividades planejadas são:

1. Ativar autenticação via LDAP UnB;
2. US - Definir tempo restante;
3. US - Tempo restante de atividades;
4. US - Atribuir notas aos alunos;
5. US - Professor gerencia notas;
6. US - Publicar notas aos alunos;
7. US - Aluno visualiza notas;
8. Disponibilização de novas funcionalidades no Comunidade.UnB;
9. Escrita do trabalho de conclusão de curso 2.

Atividades	Jul 2015	Ago 2015	Set 2015	Out 2015	Nov 2015	Dez 2015
1	•	•				
2	•	•				
3		•				
4			•	•		
5			•	•		
6				•	•	
7					•	
8				•	•	•
9		•	•	•	•	•

Tabela 7 – Cronograma de atividades para TCC2

O Laboratório de Produção, Pesquisa e Inovação em Software (LAPPIS) da FGA é um ambiente para alunos e professores trabalhem de forma colaborativa na produção de software. Esse utiliza práticas ágeis e vários de seus integrantes colaboram com o Noosfero, sendo utilizado para o desenvolvimento do trabalho.

Levando em consideração as práticas ágeis utilizadas no LAPPIS serão realizadas *sprints* com duração de quinze dias. Para cada *sprint* é proposto o desenvolvimento de uma história de usuário. Após a finalização do desenvolvimento e testes, as histórias são submetidas aos integrantes do LAPPIS, que vão realizar uma avaliação do desenvolvimento. Feita a revisão, as melhorias e correções propostas por estes membros serão

trabalhadas em uma *sprint*, onde serão trabalhadas duas histórias simultaneamente, que se relacionam em nível de implementação, a fim de atingir as propostas da revisão. Em outras palavras, após a execução de duas *sprints* a terceira é destinada a correção.

# Referências

- AGUADO, A. G. *Dimensões de qualidade em Ambientes Virtuais de Aprendizagem*. Tese (Doutorado) — Universidade de São Paulo, 2013. Citado 2 vezes nas páginas 29 e 30.
- AKITA, F. *Repensando a web com rails*. [S.l.]: Brasport, 2006. Citado na página 43.
- ARAÚJO, T. C. *AppRecommender: um recomendador de aplicativos GNU/Linux*. Tese (Doutorado) — Universidade de São Paulo, 2011. Citado na página 17.
- BECK, K. et al. *The agile manifesto*. 2001. Citado na página 24.
- BEHAR, P. A.; BASSANI, P. Avaliação da aprendizagem em ambientes virtuais. *Modelos pedagógicos em educação a distância*. Puerto Alegre: Artmed, p. 93–113, 2009. Citado na página 29.
- BERNARDO, P. C. *Padrões de testes automatizados*. Tese (Doutorado) — Universidade de São Paulo, 2011. Citado na página 26.
- BLACKBOARD, G. A. *Sobre a Bb - Construindo a Melhor Experiência Educacional*. 2015. Disponível em: <<http://blackboard.grupoa.com.br/sobre/sobre-a-bb>>. Acesso em: 11 de Junho de 2015. Citado na página 32.
- BUCHER, D. C. Rede de colaboração social para universidades brasileiras: um estudo de caso de implantação e desenvolvimento distribuído de uma plataforma livre na universidade de Brasília. 2013. Citado 7 vezes nas páginas 9, 31, 47, 48, 49, 50 e 52.
- CASTELLS, M. A era da intercomunicação. *Caminhos para uma comunicação democrática*. São Paulo, Instituto Paulo Freire, 2007. Citado na página 17.
- CHIN, P. *Virtual Learning Environments*. [S.l.]: Learning and Teaching Support Network, 2003. Citado na página 30.
- CORBUCCI, H. *Métodos ágeis e software livre: um estudo da relação entre estas duas comunidades*. Tese (Doutorado) — Universidade de São Paulo, 2011. Citado 2 vezes nas páginas 25 e 26.
- COSTA, L. B. da. *Redes: uma introdução às dinâmicas da conectividade e da auto-organização*. [S.l.]: WWF-Brasil, 2004. Citado na página 31.
- CRISCUOLO, M. *Qualidade de Produto de Software: uma abordagem baseada no controle da complexidade*. Tese (Doutorado) — Universidade de São Paulo, 2008. Citado na página 22.
- DAVIS, C. H. et al. Social media in higher education: A literature review and research directions. University of Arizona and Claremont Graduate University, 2012. Citado 2 vezes nas páginas 18 e 31.
- DUTRA, R. L. de S.; TAROUÇO, L. M. R. Objetos de aprendizagem: uma comparação entre scorm e ims learning design. *RENOTE*, v. 4, n. 1, 2006. Citado na página 34.

- DUVALL, P. M.; MATYAS, S.; GLOVER, A. *Continuous integration: improving software quality and reducing risk*. [S.l.]: Pearson Education, 2007. Citado na página 26.
- ELLISON, N. B. et al. Social network sites: Definition, history, and scholarship. *Journal of Computer-Mediated Communication*, Wiley Online Library, v. 13, n. 1, p. 210–230, 2007. Citado na página 31.
- EMIRBAYER, M.; GOODWIN, J. Network analysis, culture, and the problem of agency. *American journal of sociology*, JSTOR, p. 1411–1454, 1994. Citado na página 31.
- GNU, O. S. *O que é software livre? @ONLINE*. 2013. Disponível em: <<http://www.gnu.org/philosophy/free-sw.en.html>>. Citado na página 24.
- IEEE 1219. *Std 1219 – IEEE Standard for Software Maintenance*. New York, NY, USA, 1998. Citado 2 vezes nas páginas 21 e 22.
- ISO/IEC. *Systems and software engineering — Software life cycle processes*. [S.l.], 1998. Citado na página 21.
- LARMAN, C. *Utilizando UML e padrões*. [S.l.]: Bookman, 2002. Citado na página 47.
- LIENTZ, B. P.; SWANSON, E. B. Software maintenance management. *Addison—Wesley*, 1980. Citado na página 22.
- MARTELETO, R. M. Análise de redes sociais: aplicação nos estudos de transferência da informação. *Ciência da informação*, SciELO Brasil, v. 30, n. 1, p. 71–81, 2001. Citado na página 31.
- MEIRELLES, P. R. M. *Monitoramento de métricas de código-fonte em projetos de software livre*. Tese (Doutorado) — Universidade de São Paulo, 2013. Citado 3 vezes nas páginas 23, 25 e 43.
- MESZAROS, G. *xUnit test patterns: Refactoring test code*. [S.l.]: Pearson Education, 2007. Citado na página 26.
- MYERS, G. J.; SANDLER, C.; BADGETT, T. *The art of software testing*. [S.l.]: John Wiley & Sons, 2011. Citado na página 26.
- NORTH, D. Introducing behaviour driven development. *Better Software*, 2006. Citado na página 45.
- PFLEEGER, S. L. Engenharia de software: teoria e prática. 2ª Edição, *Prentice Hall*, 2004. Citado 2 vezes nas páginas 21 e 22.
- PRESSMAN, R. S. *Engenharia de software*. [S.l.]: McGraw Hill Brasil, 2011. Citado na página 22.
- RAYMOND, E. S. *The Cathedral & the Bazaar*. Sebastopol, CA, USA: O’Reilly & Associates, Inc., 1999. Citado na página 24.
- REIS, C.; FORTES, R. P. de M. Caracterização de um modelo de processo para projetos de software livre. *São Carlos, SP*, 2001. Citado na página 26.

- ROCHA, H. V. d.; MORAES, M. O ambiente teleduc para educação à distância baseada na web: Princípios, funcionalidades e perspectivas de desenvolvimento. *Educação a distância: Fundamentos e práticas*. Campinas, SP: UNICAMP/NIED, p. 197–212, 2002. Citado na página 33.
- SANTOS, A. P. O. dos. *Aplicação de práticas de usabilidade ágil em software livre*. Tese (Doutorado) — Universidade de São Paulo, 2012. Citado na página 24.
- SCHLEMMER, E. Ambiente virtual de aprendizagem (ava): uma proposta para a sociedade em rede na cultura da aprendizagem. *Aprendizagem em ambientes virtuais: compartilhando ideias e construindo cenários*. Caxias do Sul, RS: Educus, 2005. Citado na página 29.
- SJOBERG, D. I.; DYBA, T.; JORGENSEN, M. The future of empirical methods in software engineering research. In: IEEE COMPUTER SOCIETY. *2007 Future of Software Engineering*. [S.l.], 2007. p. 358–378. Citado na página 23.
- STALLMAN, R. M. What is free software. *Free Society: Selected Essays of*, 2002. Citado na página 23.
- STALLMAN, R. M.; GAY, J. *Free software, free society: Selected essays of Richard M. Stallman*. [S.l.]: CreateSpace, 2009. Citado na página 24.
- TOMAÉL, M. I.; ALCARÁ, A. R.; CHIARA, I. G. D. Das redes sociais à inovação. *Ciência da informação, Brasília, SciELO Brasil*, v. 34, n. 2, p. 93–104, 2005. Citado na página 31.
- UNB. *Guia do Professor*. 2009. Disponível em: <<http://www.unb.br/noticias/downloads/guiadoprofessorfinal.pdf>>. Acesso em: 18 de Junho de 2015. Citado na página 58.



# Apêndices



# APÊNDICE A – Instruções para preenchimento do questionário

Caros,

O estudo a seguir faz parte de uma pesquisa realizada por um aluno de Engenharia de *software*, da Universidade de Brasília – **UnB**, o qual será utilizado como forma de obtenção de conhecimento sobre o tema.

Esta pesquisa visa investigar se os alunos da Universidade de Brasília utilizam as redes sociais, como o *Facebook*, para o compartilhamento de recursos e informações relacionados as disciplinas cursadas na Universidade. Para tanto, necessito de sua colaboração respondendo ao questionário sobre o tema. É importante ressaltar que os dados aqui levantados serão compilados e utilizados apenas para fins acadêmicos, na problematização da pesquisa.

Instruções para preenchimento do questionário:

1. Você deverá selecionar um item para indicar o grau de concordância com as assertivas apresentadas na tabela 8.

<b>NUNCA</b>	<b>RARAMENTE</b>	<b>ÀS VEZES</b>	<b>FREQUENTEMENTE</b>	<b>SEMPRE</b>
--------------	------------------	-----------------	-----------------------	---------------

Tabela 8 – Opções para preenchimento do questionário.

**Lembrando** que quanto mais próximo de **Nunca**, **MENOR** o grau de concordância. Quanto mais próximo de **Sempre**, **MAIOR** o grau de concordância.

2. O questionário deve ser respondido pelos alunos das Universidade de Brasília;
3. Não existem respostas “certas” ou “erradas”. O importante é respondê-las de forma **sincera**.
4. Procure não deixar nenhuma resposta em branco. Sua participação é muito importante para a finalização deste trabalho.

Desde já agradeço sua atenção e cooperação. Coloco-me à disposição para os esclarecimentos que se fizerem necessários.

Cordialmente,

Hebert Douglas de Almeida Santos <hebertdoug@gmail.com>



## APÊNDICE B – Questionário para problematização

1. Com que frequência você utiliza redes sociais?
2. Você utiliza alguma rede social como ferramenta de apoio as disciplinas?
3. Os professores incentivam o uso de redes sociais para suas disciplinas?
4. Mesmo que o professor não recomende o uso de redes sociais para discussão de conteúdos de suas disciplinas, você as utiliza?



# APÊNDICE C – Avaliação dos resultados

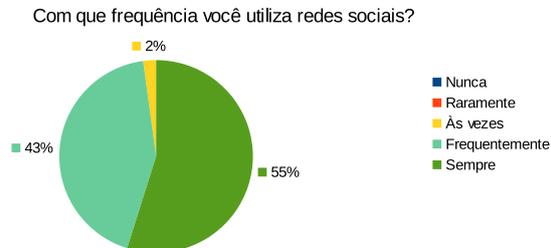


Figura 8 – Resultados do questionário para a pergunta 8

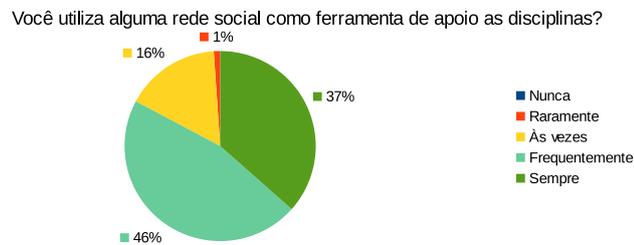


Figura 9 – Resultados do questionário para a pergunta 9

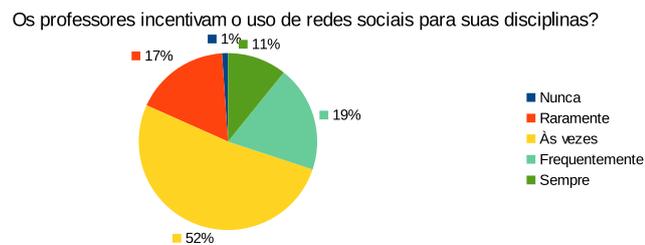


Figura 10 – Resultados do questionário para a pergunta 10



Figura 11 – Resultados do questionário para a pergunta 11