

Israel Henrique Silva de Lima

# **Uma Ferramenta de Mapeamento Conceitual para Moodle integrada ao CMPaaS**

Vitória

2016

Israel Henrique Silva de Lima

# **Uma Ferramenta de Mapeamento Conceitual para Moodle integrada ao CMPaaS**

Monografia apresentada ao Curso de Engenharia de Computação do Departamento de Informática da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do Grau de Bacharel em Engenharia de Computação.

Universidade Federal do Espírito Santo – UFES

Centro Tecnológico

Departamento de Informática

Orientador: Wagner de Andrade Perin

Coorientador: Davidson Cury

Vitória

2016

Israel Henrique Silva de Lima

## **Uma Ferramenta de Mapeamento Conceitual para Moodle integrada ao CMPaaS**

Monografia apresentada ao Curso de Engenharia de Computação do Departamento de Informática da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do Grau de Bacharel em Engenharia de Computação.

Trabalho aprovado. Vitória, 24 de novembro de 2012:

---

**Wagner de Andrade Perin**  
Orientador

---

**Professor**  
Convidado 1

---

**Professor**  
Convidado 2

Vitória  
2016

*Este trabalho é dedicado às crianças adultas que,  
quando pequenas, sonharam em se tornar cientistas.*

*“Não vos amoldeis às estruturas deste mundo,  
mas transformai-vos pela renovação da mente,  
a fim de distinguir qual é a vontade de Deus:  
o que é bom, o que Lhe é agradável, o que é perfeito.  
(Bíblia Sagrada, Romanos 12, 2)*

# Resumo

Segundo a [ABNT \(2003, 3.1-3.2\)](#), o resumo deve ressaltar o objetivo, o método, os resultados e as conclusões do documento. A ordem e a extensão destes itens dependem do tipo de resumo (informativo ou indicativo) e do tratamento que cada item recebe no documento original. O resumo deve ser precedido da referência do documento, com exceção do resumo inserido no próprio documento. (...) As palavras-chave devem figurar logo abaixo do resumo, antecidas da expressão Palavras-chave:, separadas entre si por ponto e finalizadas também por ponto.

**Palavras-chave:** latex. abntex. editoração de texto.

# Abstract

This is the english abstract.

**Keywords:** latex. abntex. text editoration.

# Lista de ilustrações

Figura 1 – Um exemplo de mapa conceitual . . . . .	10
Figura 2 – Interface de um curso no Moodle . . . . .	14
Figura 3 – (a) Integração do Portal do Conhecimento com o CMPaaS. (b) Integração do CMPaaS com serviços externos . . . . .	17
Figura 4 – Arquitetura do CMPaaS . . . . .	18
Figura 5 – Casos de Uso . . . . .	21
Figura 6 – Editor com barra de formatação . . . . .	22
Figura 7 – Componentes do GoJS . . . . .	23
Figura 8 – Um diagrama gerado em GoJS . . . . .	23
Figura 9 – Código dos elementos Shape e TextBlock . . . . .	24
Figura 10 – Rotina que altera a cor de preenchimento de um nó . . . . .	25
Figura 11 – Configurando a tarefa para utilizar o plugin . . . . .	27
Figura 12 – Configurando a tarefa para utilizar o plugin . . . . .	28
Figura 13 – Configurando a tarefa para utilizar o plugin . . . . .	28



# Sumário

<b>1</b>	<b>INTRODUÇÃO</b>	<b>9</b>
<b>2</b>	<b>MAPAS CONCEITUAIS</b>	<b>10</b>
2.1	Origem e definição de mapas conceituais	10
2.2	Mapas conceituais e educação	11
2.3	Mapas conceituais e tecnologia da informação	11
<b>3</b>	<b>EDUCAÇÃO A DISTÂNCIA</b>	<b>13</b>
3.1	Conceitos Básicos de EaD e o Moodle	13
<b>4</b>	<b>CMPAAS</b>	<b>16</b>
4.1	Origem	16
4.2	A plataforma CMaaS	16
4.2.1	Computação em Nuvem	16
4.2.2	Arquitetura	17
<b>5</b>	<b>TECNOLOGIAS UTILIZADAS</b>	<b>19</b>
5.1	JavaScript	19
5.2	Json	19
5.3	PHP	19
<b>6</b>	<b>PLUGIN</b>	<b>20</b>
6.1	O problema	20
6.2	A solução proposta	20
6.3	O projeto	20
6.4	O editor de mapas	22
6.4.1	Implementação das funções de formatação	22
6.5	O plugin para o moodle	25
6.5.1	Implementação	26
6.5.2	Integração com o CMPaaS	27
6.5.3	Instalação e uso	27
<b>7</b>	<b>CONCLUSÃO</b>	<b>29</b>
	<b>REFERÊNCIAS</b>	<b>30</b>

# 1 Introdução

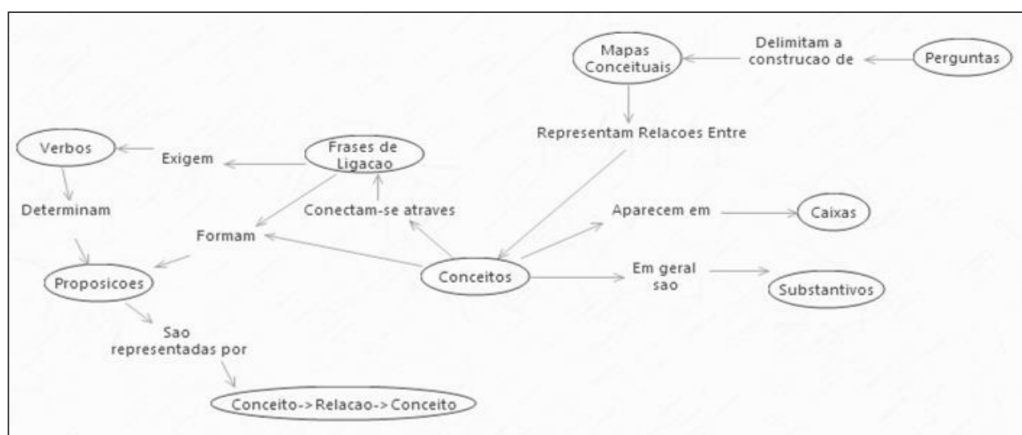
## 2 Mapas conceituais

### 2.1 Origem e definição de mapas conceituais

O mapas conceituais foram criados na década de 70 durante um estudo de Novak que buscava entender como crianças aprendiam conceitos científicos (NOVAK, 2005). Durante seu estudo, Novak observou uma evolução nas proposições utilizadas pelas crianças, porém teve dificuldade de entender como esta mudança cognitiva ocorreu. Para compreender as mudanças na estruturas cognitivas observadas Novak desenvolveu um método baseado nas três ideias de Ausubel sobre a teoria da aprendizagem (AUSUBEL, 1963). A primeira ideia é que novos significados são criados a partir de conceitos e proposições previamente conhecidos pelo aprendiz. A segunda, é que a estrutura cognitiva do indivíduo é organizada hierarquicamente, com conceitos mais gerais ocupando posições superiores, acima dos conceitos mais específicos e complexos. Já a terceira ideia é que durante o processo de aprendizagem os conceitos vão se tornando mais específicos e precisos.

Assim foi criada essa ferramenta gráfica, que tem como objetivo fazer uma representação do conhecimento. Um mapa conceitual é constituído por interligações entre dois ou mais conceitos através de uma palavra de forma a gerar um significado. Ele é construído ligando-se um conceito a outro, através de um arco que contém uma palavra descritiva, que é usada para criar uma relação significativa entre os conceitos. Além disso, os conceitos mais gerais estão dispostos no topo do mapa e os mais específicos na parte inferior dele. A Figura 1 mostra um exemplo de mapa conceitual.

Figura 1 – Um exemplo de mapa conceitual



Fonte: Perin (2014, p. 28)

## 2.2 Mapas conceituais e educação

Os mapas conceituais podem ter diversas aplicações na educação, podem ser utilizados para representar o conhecimento adquirido em uma aula, ou então o conteúdo de um capítulo algum livro lido. Por se tratar de uma representação que é alimentada com novos conceitos conforme eles são adquiridos, pode ser utilizado pelo professor para acompanhar a aprendizagem de um estudante durante todo o período de um curso. Além disso, como os mapas conseguem representar a estrutura cognitiva de um indivíduo, eles podem ser utilizados também como método de avaliação de aprendizagem([PERIN, 2014](#)).

A confiabilidade da avaliação da aprendizagem tradicional, como textos dissertativos e questões de múltipla escolha, tem sido questionada por pesquisadores e educadores. Nestes tipos de teste apenas os acertos são contabilizados e os erros são descartados, isto pode fazer com que informações importantes para a avaliação sejam desprezadas. As provas tradicionais não proporcionam a possibilidade do estudante apresentar como construiu o seu aprendizado. Elas conseguem avaliar somente a aprendizagem mecânica, não mostrando como que o aprendiz alterou suas estruturas cognitivas. Os mapas conceituais tem se destacado como alternativa a avaliação tradicional, pois conseguem demonstrar com facilidade as modificações cognitivas que ocorrem durante o processo de aprendizagem do estudante([DUTRA; FAGUNDES; CAÑAS, 2002](#)).

## 2.3 Mapas conceituais e tecnologia da informação

A construção de mapas conceituais é muito simples, uma caneta e um pedaço de papel são ferramentas suficientes para a confecção desse grafo. Porém a tarefa de revisá-lo, armazená-lo, e editá-lo a longo prazo pode ser muito cansativa e complexa. A introdução do uso de computadores na confecção de mapas pode facilitar a tarefa ([NOVAK; CAÑAS, 2006](#)).

Os primeiros programas de computadores criados com este objetivo se limitavam a mostrar apenas os mapas na tela, sem oferecer nenhum recurso adicional a ferramenta. Com a popularização do uso de computadores e da Internet houve um aumento na oferta de aplicações que tem como objetivo a construção de mapas gráficos. Porém há uma grande confusão entre usuários e desenvolvedores sobre a diferença entre mapas conceituais, organogramas e mapas mentais, devido a semelhança entre os diagramas([PERIN, 2014](#)). Um mapa mental, por exemplo, é muito semelhante a um mapa conceitual pois apresenta ligações entre ideias, mas ao contrario de um mapa mental não possui uma palavra descritiva ligando essas ideias e criando organicidade entre as ideias. Portanto uma aplicação de criação de mapas mentais não permite a construção de um mapa conceitual.

Na seção 2.3 de sua dissertação de mestrado, [Perin \(2014\)](#) realizou uma pesquisa

sobre o estado da prática em mapas conceituais. Nela foi feito um levantamento sobre softwares que permitem a criação de mapas conceituais e quais são as funcionalidades oferecidas por eles. Nesta pesquisa ele observou uma carência de ferramentas específicas para a criação de mapas conceituais, sendo o CMapTools a única ferramenta que oferece suporte específico para a manipulação dos mapas.

## 3 Educação a distância

### 3.1 Conceitos Básicos de EaD e o Moodle

Segundo [CHAVES \(1999\)](#) EAD é um ensino que ocorre quando há uma separação física entre mestre e aprendiz, sendo utilizadas tecnologias de transmissão de voz, dados e imagens para a comunicação entre ambos. A EAD também é definida no Decreto nº 5.622 de 19 de dezembro de 2005([BRASIL, 2005](#)).

Art. 1o Para os fins deste Decreto, caracteriza-se a educação a distância como modalidade educacional na qual a mediação didático-pedagógica nos processos de ensino e aprendizagem ocorre com a utilização de meios e tecnologias de informação e comunicação, com estudantes e professores desenvolvendo atividades educativas em lugares ou tempos diversos([BRASIL, 2005](#)).

Em síntese é possível definir a educação a distância como um método de ensino no qual estudante e professor ficam separados fisicamente e a interação é feita através de tecnologias de comunicação (que podem ser dos mais diversos) de maneira a contornar esta separação.

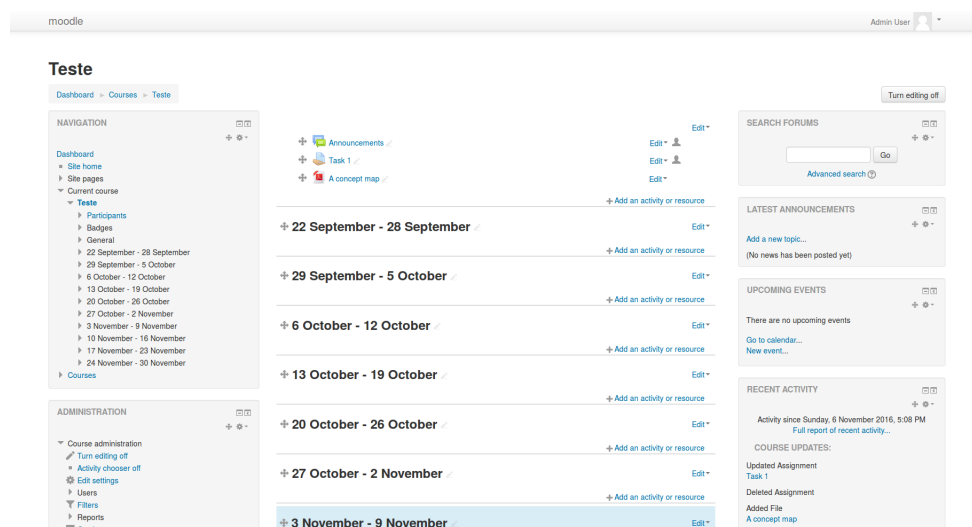
Atualmente a educação a distância está relacionada ao uso de um AVA, que nada mais é que um sistema computacional cuja finalidade é prover suporte a atividades mediadas pelas tecnologias de informação e comunicação([ALMEIDA, 2010](#)). É um ambiente que tem como objetivo integrar mídias e recursos, além de apresentar informações de forma organizada e permitir interação entre estudantes, tutores e professores([FRANCISCATO et al., 2008](#)). Um exemplo de um AVA são os cursos que podem ser criados e gerenciados no Moodle.

O *Modular Object Oriented Distance Learning (Moodle)* é sistema gerenciador de cursos que foi criado por Martin Dougiamas em 1999. Ele é uma aplicação *open-source*, o que significa que ele é livre para ser instalado, utilizado, modificado e distribuído ([DOUGIAMAS; TAYLOR, 2003](#)).

O *Moodle* trabalha por padrão com cinco perfis de usuário: administrador, criador de cursos, professor, aluno e visitante. O administrador é o responsável técnico, é ele quem realiza a instalação e configuração do ambiente, além de manter ele funcionando corretamente. Já o criador de cursos tem como responsabilidade a criação e configuração dos cursos disponíveis na plataforma. Por sua vez, o professor tem como função o acompanhamento dos estudantes e a inserção de recursos e tarefas nos cursos. O aluno é quem realiza o curso, é ele quem vai utilizar os recursos e tarefas disponíveis no AVA. Por fim, o visitante é um usuário que só tem acesso as informações disponíveis na tela inicial do

sistema. A Figura 2 mostra a interface de um curso no Moodle.

Figura 2 – Interface de um curso no Moodle



Fonte: Elaborada pelo autor

Ele foi escolhido para ser utilizado neste trabalho pois é amplamente usado por instituições em todo o mundo, possuindo mais de 73 mil sites registrados em 232 países (MOODLE, 2016), e possui uma grande comunidade que contribui para correção de erros e criação de novas ferramentas. Além disso ele é utilizado para gerenciar os AVAs da Universidade Federal do Espírito Santo, e é muito utilizado também em diversas instituições no país, sendo o Brasil o terceiro maior utilizador da plataforma, conforme pode ser visto na Tabela 1.

O fato do Moodle ser modular, ou seja, ser composto por módulos instaláveis, configuráveis e estendíveis, permite que sejam desenvolvidos plugins e componentes que adicionam novas funcionalidades a plataforma, como é o caso do que está sendo desenvolvido neste trabalho.

Tabela 1 – Os 10 países que mais utilizam o Moodle.

País	Sites Registrados
Estados Unidos	10.131
Espanha	7.067
Brasil	4.401
Reino Unido	3.486
México	3.464
Alemanha	2.444
Itália	2.414
Austrália	2.324
Colômbia	2.264
Rússia	1.993

Fonte: [Moodle \(2016\)](#)



## 4 CMPAAS

### 4.1 Origem

Na [seção 2.2](#) vimos como os mapas conceituais são importantes para educação, seja para auxiliar o estudante a aprender novos conceitos, ou como método de avaliação, além de algumas outras aplicações educacionais.

Apesar de haver este grande interesse acadêmico em ferramentas computacionais para criação e manipulação de mapas, a fragmentação das pesquisas nesta área impede que o desenvolvimento destas aplicações ocorra de forma acelerada. Além disso, a integração de ferramentas já existentes facilitaria o progresso de novas pesquisas.

Se houvesse uma infraestrutura de apoio a criação e manipulação de mapas conceituais o desenvolvimento de novas ferramentas seria facilitado. Um pesquisador poderia criar uma nova solução mais facilmente se não houvesse a necessidade dele se preocupar com toda uma estrutura de gestão de mapas. Por exemplo, alguém que queira desenvolver uma aplicação que avalie mapas conceituais poderia utilizar o conteúdo existente previamente em uma solução computacional de criação e armazenamento de mapas.

Um outro problema apontado por [Perin \(2014\)](#) é a dificuldade que a sociedade tem para acessar os resultados das pesquisas.

Consideramos importante a criação de um mecanismo de acesso eficiente aos resultados das pesquisas científicas, para que a comunidade possa contribuir para a evolução delas. O que propomos, portanto, é o lançamento de bases para uma convivência mais estreita entre o mundo acadêmico e a sociedade em geral. ([PERIN, 2014](#)).

Assim foi proposto por [Perin \(2014\)](#) a criação de uma plataforma denominada CMPaaS, cujo objetivo é possibilitar que a comunidade em geral acesse o resultado de pesquisas acadêmicas e prover uma infraestrutura que permita que esta comunidade crie e estenda suas funcionalidades.

### 4.2 A plataforma CMaaS

#### 4.2.1 Computação em Nuvem

O termo computação em nuvem teve sua origem em 2006 durante uma palestra Eric Schmidt sobre como o Google gerenciava seus servidores ([TAURION, 2009](#)). A palavra nuvem é uma abstração para a Internet e toda a sua complexidade de infraestrutura, arquiteturas e componentes. A computação em nuvem é um paradigma no qual o processamento,

armazenamento e ferramentas computacionais são oferecidos como um serviço através da Internet. Aplicações baseadas nesta tecnologia possuem a característica de serem extensíveis e facilmente incorporadas a outras que precisem consumir os seus serviços (PERIN, 2014).

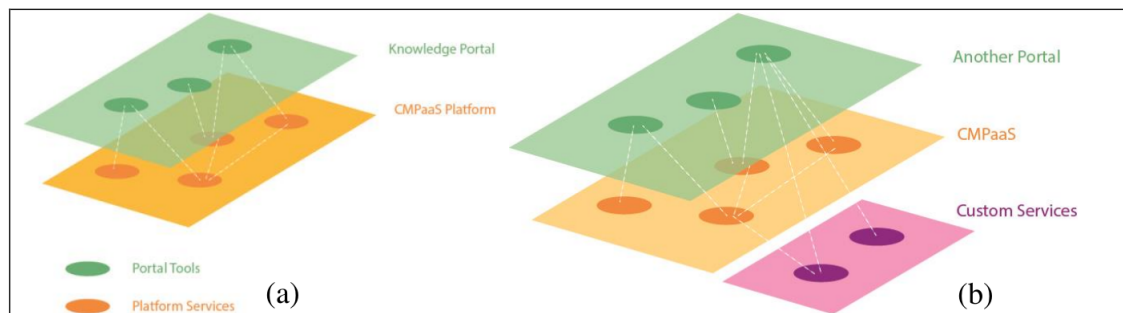
A plataforma CMPaaS usa esta capacidade de expansão e integração da computação em nuvem para oferecer serviços de criação, edição e gestão de mapas conceituais que poderão ser acessados por qualquer usuário no mundo.

## 4.2.2 Arquitetura

O CMPaaS é uma plataforma orientada a serviços, ou seja, ela utiliza a arquitetura SOA. Isto significa que as funcionalidades oferecidas pela plataforma são facilmente extensíveis e aproveitadas por novas aplicações criadas por qualquer pessoa ao redor do mundo.

A plataforma foi implementada com a utilização de Web Services, que precisam de uma aplicação com interface para serem consumidos. Assim, o CMPaaS precisa estar associado a um portal que permitirá que o usuário final utilize as ferramentas oferecidas pela plataforma. Este portal foi nomeado como "Portal do Conhecimento" (PERIN, 2014).

Figura 3 – (a) Integração do Portal do Conhecimento com o CMPaaS. (b) Integração do CMPaaS com serviços externos

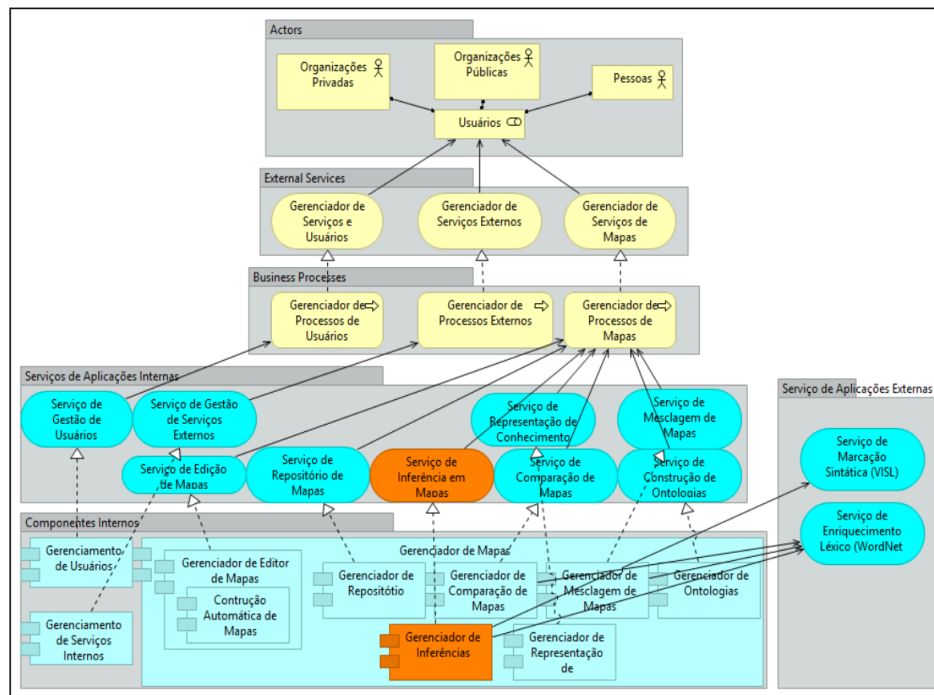


Fonte: Perin (2014, p. 81)

A Figura 3 ilustra a arquitetura do CMPaaS. Podemos observar como as aplicações fornecidas pelo portal utilizam os serviços da plataforma, muitas vezes consumindo mais de um serviço que é oferecido por ela. Além disso, é ilustrado também como um portal externo pode aproveitar os serviços disponíveis no CMPaaS, como é o caso do plugin que será apresentado posteriormente neste trabalho. Ele trata-se de um editor de mapas conceituais que armazena os mapas, criados por ele, na plataforma. Assim ele utiliza dois serviços fornecidos pelo CMPaaS, o de armazenagem de mapas e o de autenticação.

A estrutura interna da plataforma é composta por cinco camadas(PERIN, 2014), conforme mostra a Figura 4.

Figura 4 – Arquitetura do CMPaaS



Fonte: Perin (2014, p. 82)

A camada de serviços externos tem como finalidade prover funcionalidades que serão utilizadas pela comunidade, ou seja, são as aplicações que serão utilizadas por instituições públicas e privadas ou qualquer indivíduo. Os processos de negócio são a camada cuja a responsabilidade é gerenciar todos os processos que ocorrem internamente na plataforma. Já a camada de serviços de aplicações internas gerencia os serviços utilizados pelos processos de negócio, esta camada tem como responsabilidade tornar os componentes internos disponíveis para os processos da plataforma. A camada de componentes internos tem como função gerenciar os componentes que produzem todos os serviços da plataforma, é nesta camada que as aplicações de mapas conceituais funcionam. Por último, a camada de serviços de aplicações externas é composta por serviços externos que são utilizados por componentes internos(PERIN, 2014).

## 5 Tecnologias utilizadas

### 5.1 JavaScript

O JavaScript é uma linguagem orientada a objeto criada em 1995 por Brendan Eich que foi idealizada para permitir que pessoas que não são programadoras pudessem estender as funcionalidades de sites de Internet(RICHARDS et al., 2010). Ela é uma linguagem interpretada, e apesar de ter sido criada para desenvolvimento web, ela é uma linguagem de propósito geral, e pode ser usada para o desenvolvimento de qualquer tipo de aplicação(FLANAGAN, 2006).

Ela foi escolhida para ser utilizada neste projeto pois o editor de mapas disponível no CMPaaS foi desenvolvido com esta tecnologia e, além disso, trata-se de uma linguagem compatível com todos os navegadores modernos.

### 5.2 Json

O JavaScript Object Notation (Json) é uma linguagem em formato de texto cuja função é a serialização de dados estruturados. A serialização é o processo de transformar objetos em um fluxo de bytes para ser armazenado em um banco de dados ou disco. Ele pode ser utilizado para representar tipos primitivos, como cadeia de caracteres e números, ou tipos estruturados, como objetos e vetores(CROCKFORD, 2006). Ele é utilizado para intercambio de dados entre os serviços da plataforma CMPaaS.

### 5.3 PHP

PHP é um acrônimo recursivo para PHP: Hypertext Preprocessor, originalmente ele foi criado como uma linguagem de script estruturada cuja finalidade era o desenvolvimento de aplicações com a funcionalidade de geração de HTML dinâmico. Com o passar do tempo a linguagem evoluiu e passou a oferecer recursos para o desenvolvimento orientado a objeto(MINETTO, 2007).

A plataforma Moodle e seus plugins são desenvolvidos em PHP, por isto esta linguagem foi utilizada neste projeto.

## 6 Plugin

### 6.1 O problema

Vimos no [Capítulo 2](#) que os mapas conceituais são muito importantes para a educação, principalmente como alternativa para os métodos de avaliação tradicionais. Além disso, no [Capítulo 3](#) foi apresentado o conceito de educação a distância e como ela é oferecida através do uso de AVAs. Sendo o Moodle apontado como um gerenciador de cursos amplamente utilizado por instituições brasileiras.

Ao buscar na lista de plugins no Moodle alguma ferramenta que permite o uso de mapas conceituais como resposta de atividades avaliativas, só foi encontrado um único plugin. Este plugin permite que questionários possam ser respondidos com uma mapa. Além disso, a última atualização foi no início de 2015 e ele não é compatível com as novas versões da plataforma. Portanto, não há nenhuma ferramenta que permita o uso de mapas conceituais em tarefas de cursos gerenciados pela plataforma em suas versões mais recentes.

### 6.2 A solução proposta

Este trabalho propõe a criação de um plugin que adicione ao Moodle de resposta de uma tarefa com um mapa conceitual. Além disso, ele será integrado a plataforma CMPaaS, consumindo serviços oferecidos por ela.

### 6.3 O projeto

O Moodle é composto por vários módulos independentes, sendo cada um deles responsável por uma funcionalidade que a plataforma oferece. O escopo deste trabalho é o desenvolvimento de um novo módulo. O plugin que foi criado neste trabalho é do tipo assignment submission plugin, ou seja, é um módulo de submissão de tarefa. O módulo de tarefa permite que um professor crie uma atividade, avaliativa ou não, para os estudantes realizarem.

Este tipo de plugin é dividido em três partes:

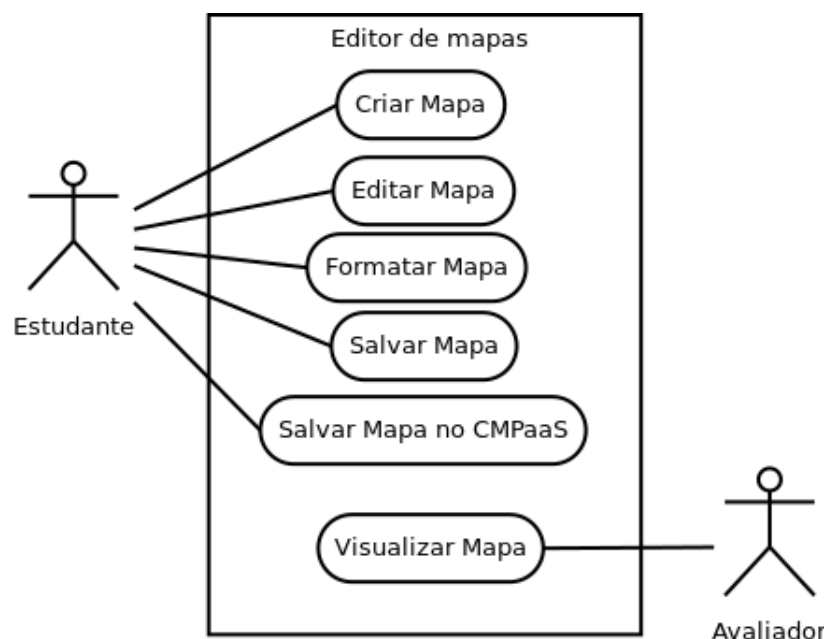
- configurações, que são opções de comportamento do módulo.
- sumário de envio, que será visto pelo estudante e avaliadores na tela inicial da tarefa.
- formulário de envio, que é onde o estudante responde a tarefa.

Este plugin oferece as seguintes funcionalidades:

- criação de um novo mapa conceitual utilizando um editor de mapas. Esta funcionalidade é representada pelo caso de uso Criar Mapa.
- armazenagem do mapa criado no banco de dados do Moodle. Esta funcionalidade é representada pelo caso de uso Salvar Mapa.
- edição de um mapa previamente salvo no banco de dados do Moodle. Esta funcionalidade é representada pelo caso de uso Salvar Mapa.
- alteração da aparência de uma mapa que está sendo criado ou modificado no editor. Esta funcionalidade é representada pelo caso de uso Formatar Mapa.
- possibilidade de salvar o mapa que está sendo criado na plataforma CMPaaS. Esta funcionalidade é representada pelo caso de uso Salvar Mapa no CMPaaS.
- visualizar um mapa salvo para que possa ser avaliado. Esta funcionalidade é representada pelo caso de uso Visualizar Mapa.

Os atores que irão utilizar o plugin são o Estudante, que irá criar e editar mapas conceituais que serão submetidos a avaliação, e um Avaliador, que irá visualizar os mapas criados pelos estudantes e avalia-los. Os atores e casos de uso são representados na [Figura 5](#).

Figura 5 – Casos de Uso



Fonte: Elaborado pelo autor

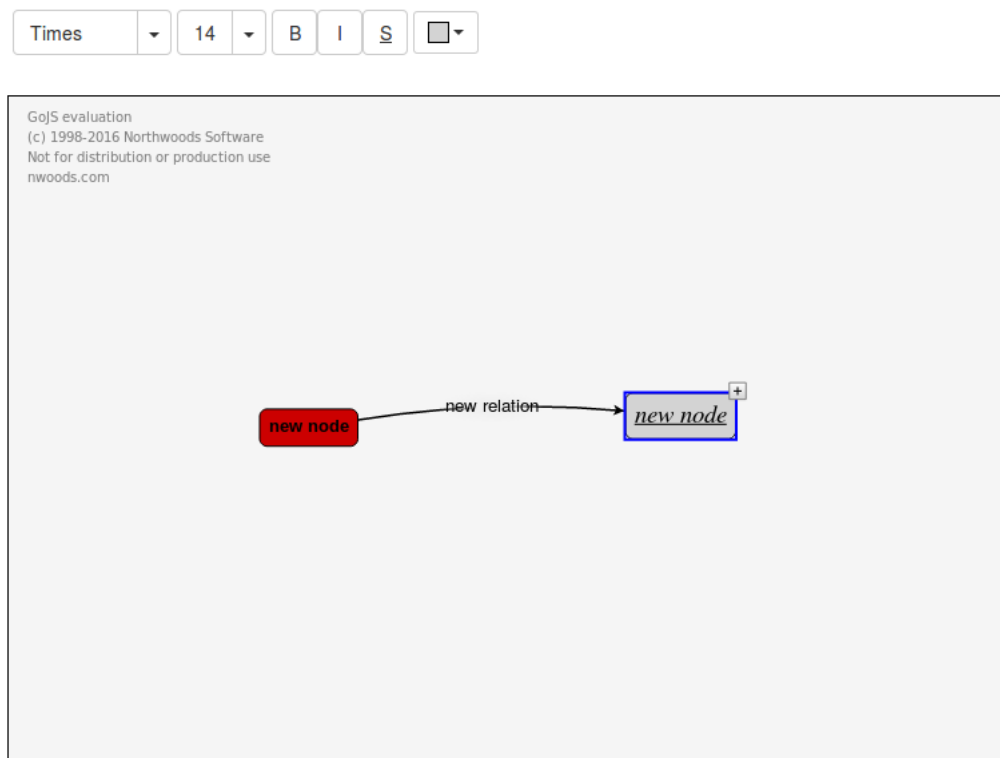
## 6.4 O editor de mapas

Na criação do formulário de envio, foi utilizado o editor de mapas conceituais disponível no CMPaaS. Ele foi desenvolvido utilizando GoJS, que é uma biblioteca em JavaScript para criação de diagramas para navegadores de Internet.

Este editor foi desenvolvido de forma a salvar os mapas no formato Json, de forma a facilitar que outras aplicações utilizem os dados que ele produz. Ele foi feito desta forma devido a proposta do CMPaaS de permitir a interoperabilidade dos serviços oferecidos.

A primeira parte do trabalho foi realizar um aprimoramento deste editor, adicionando funções para a formatação dos mapas criados por ele. Inicialmente não existia a possibilidade de alterar a fonte da letra dos conceitos ou a cor de fundo dos nós. Uma barra de ferramenta foi adicionada ao editor original para que funções de formatação estivessem disponíveis.

Figura 6 – Editor com barra de formatação



Fonte: Elaborada pelo autor

### 6.4.1 Implementação das funções de formatação

O editor de mapas é dividido em dois componentes, um Model e um Diagram. O Model é o que contém os dados do mapa que está sendo criado, nele os nós e links são

descritos por vetores de objetos JavaScript. Já o Diagram é usado para visualizar os dados contidos no modelo.

Na [Figura 7](#) temos um trecho de código que gera um diagrama simples em GoJS. Nele é criado um Model que possui um vetor com três objetos JavaScripts, nomeados Alpha, Beta e Gamma. Este vetor de objetos da classe Node representa três nós que serão apresentados no diagrama. O Model nomeado myModel é então adicionado ao Diagram chamado myDiagram. Este trecho de código gera o diagrama ilustrado na [Figura 8](#).

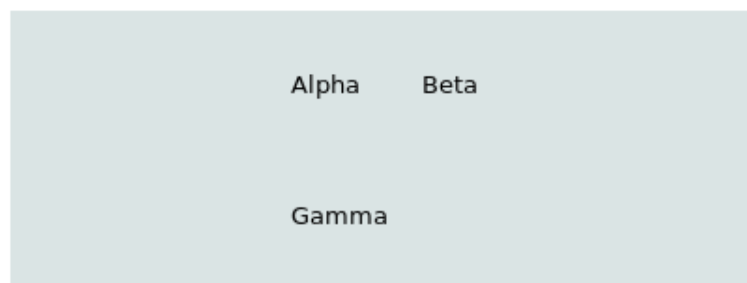
Figura 7 – Componentes do GoJS

```
myDiagram =
  $(go.Diagram, "myDiagram",
    { initialContentAlignment: go.Spot.Center,
      "toolManager.mouseWheelBehavior": go.ToolManager.WheelZoom,
      "undoManager.isEnabled": true,
      "clickCreatingTool.archetypeNodeData": { text: "new node" }
    });

var myModel = $(go.Model);
myModel.nodeDataArray = [
  { key: "Alpha" },
  { key: "Beta" },
  { key: "Gamma" }
];
myDiagram.model = myModel;
```

Fonte: Elaborada pelo autor

Figura 8 – Um diagrama gerado em GoJS



Fonte: Elaborada pelo autor

Cada nó do mapa criado pelo editor é representado por um objeto da classe Node, que por sua vez, é composto por blocos que determinam a sua aparência. Os blocos que o editor utiliza são o Shape e o TextBlock. A classe Shape é utilizada para mostrar uma forma



geométrica colorida. Já a classe `TextBlock` tem como função mostrar um texto. Ambas as classes possuem diversas propriedades que servem para determinar a sua aparência e o seu comportamento no diagrama.

Os trechos de código abaixo são usados para criar um `Shape` e um `TextBlock`. A forma geométrica desenhada é retângulo com largura de 40 pixels, altura de 60 pixels, com margem de 4 pixels e preenchimento na cor vermelha. Já o `TextBlock` criado é um texto “a Text Block” de cor vermelha.

Figura 9 – Código dos elementos `Shape` e `TextBlock`

```
$(go.Shape, "Rectangle", { width: 40, height: 60, margin: 4, fill: "red" })
$(go.TextBlock, { text: "a Text Block", stroke: "red" })
```

Fonte: Elaborada pelo autor

As novas funcionalidades de formatação adicionadas ao editor modificam as propriedades dos objetos das classes `Shape` e `TextBlock`, alterando assim a aparência deles. Foram criados seis botões de formatação com as funções de alterar o tipo de fonte, aumentar o tamanho da fonte, alterar o estilo do texto para negrito, itálico e sublinhado, e para alterar a cor de preenchimento dos nós. Os botões de formatação de fonte alteram a propriedade `font` da `TextBlock`, que deve ser uma string CSS. Já o botão que altera a cor do nó modifica a propriedade `fill` da `Shape`.

As funcionalidades de formatação foram implementadas em JavaScript no arquivo `toolbar.js`. Elas foram desenvolvidas de forma semelhante, utilizando eventos disparados pelos botões da barra de ferramenta.

Para realizar alteração de estilo da fonte e de cor do nó foram implementadas funções que modificam os atributos dos nós. Para cada botão da barra de ferramenta foi criada uma rotina que aguardava um evento disparado por ele. Quando ele ocorre uma função é chamada e realiza a modificação referente ao botão que foi acionado.

A [Figura 10](#) mostra a rotina que altera a cor de preenchimento de um nó. Inicialmente é criada uma espera para o evento que será disparado pelo botão, no caso o evento `input`. Quando ele ocorre é chamada a função que altera a propriedade `fill` do objeto `Shape` de todos os nós selecionados, mudando assim a cor dos mesmos. As funcionalidades de estilo da fonte são implementadas com a mesma rotina, o que muda é a propriedade alterada, que passa a ser a `font` do objeto `TextBlock` de todos os nós selecionados.

Figura 10 – Rotina que altera a cor de preenchimento de um nó

```
colorButton = document.getElementById("color");
colorButton.addEventListener("input", function() {
    myDiagram.startTransaction("change color");
    var it = myDiagram.selection.iterator;
    while (it.next()) {
        var node = it.value;
        var shape = node.findObject("SHAPE");
        if (shape !== null) {
            shape.fill = colorButton.value;
        }
    }
    myDiagram.commitTransaction("change color");
});
```

Fonte: Elaborada pelo autor

## 6.5 O plugin para o moodle

Para a elaboração deste trabalho foi utilizado um plugin já existente no Moodle como modelo. Todos os plugins de envio de tarefa devem ter uma estrutura de arquivo padrão, esta estrutura será detalhada abaixo.

- version.php: este arquivo contém informações sobre a versão do plugin, é utilizado para que o Moodle instale e atualize o plugin corretamente.
- settings.php: este arquivo permite que se adicione opções personalizadas para a página configuração do plugin.
- lang/en/submission\_nomedoplugin.php: este é o arquivo de linguagem, ele é usado para internacionalização do plugin.
- db/access.php: e utilizada para adicionar capacidades adicionais ao plugin. Este arquivo é opcional, não sendo necessário se o plugin não tiver capacidades adicionais.
- db/upgrade.php: este arquivo define a rotina de atualização do plugin.
- db/install.xml: este arquivo define as tabelas de banco de dados que o plugin vai utilizar.
- db/install.php: contem o código de instalação do plugin.
- db/locallib.php: eh o arquivo mais importante, é ele que define todas as funcionalidades do plugin.

### 6.5.1 Implementação

O plugin utilizado como base para o desenvolvimento deste trabalho foi o de envio de texto online, nele o estudante tem a possibilidade responder uma tarefa por meio de um texto escrito diretamente na plataforma. Este projeto foi desenvolvido aplicando engenharia reversa a este plugin, buscando entender a sua funcionalidade e assim descobrir como modifica-lo para tranforma-lo em um editor de mapas.

O módulo de envio de texto online pode ser dividido em duas partes, uma é o editor de textos onde o estudante realiza a sua tarefa e a outra e o sumário da atividade onde tanto o estudante quanto o avaliador tem acesso ao conteúdo da tarefa enviada. O módulo desenvolvido neste trabalho mantém esta estrutura, alterando o editor de textos por um de mapas conceituais e alterando o sumário de forma que o conteúdo apresentado nele passa a ser um json do mapa invés de um texto.

As alterações feitas no plugin de envio de texto online se concentraram no arquivo `locallib.php`, que é o que determina o comportamento e funcionalidades do módulo. Dentro deste arquivo, as mudanças ocorreram nas funções `get_form_elements()`, que constrói o formulário de envio de tarefa, e no arquivo `save()`, que realiza a submissão do conteúdo criado pelo estudante.

A primeira parte do trabalho foi substituir o formulário de envio de texto pelo editor de mapas conceituais. Esta modificação foi realizada na função `get_form_elements()`. O código que realizava a inserção do editor de texto foi substituído por um outro, que insere um `iframe` contendo o editor de mapas do CMPaaS.

Feito isto, o plugin já apresentava o editor de mapas, porém ainda era necessário salvar o mapa criado no banco de dados do Moodle. Para realizar isto o json do editor contido no `iframe` precisava ser salvo em algum campo de formulário. Para solucionar isto, foi criado um entrada de dados, oculta na página da tarefa, responsável por receber o json do mapa criado no editor.

Assim, quando o estudante aciona o botão de submissão da tarefa, o json do mapa que ele criou e salvo em um campo de formulário e então armazenado no banco de dados do Moodle.

Para finalizar, foi necessário criar uma forma de o editor de mapas carregar o mapa salvo pelo estudante, para caso ele tivesse interesse em editar um mapa já armazenado. A solução para isto foi realizar o caminho inverso que foi feito ao salvar o mapa criado. Ao carregar o formulário de edição de mapas o json armazenado no banco de dados e carregado em um campo oculto e um código JavaScript se encarrega de obter o conteúdo dele e carregar no mapa.

Então a primeira parte do desenvolvimento do plugin foi concluída, o estudante conseguia criar um mapa no editor, salvar o seu conteúdo na plataforma e editar novamente

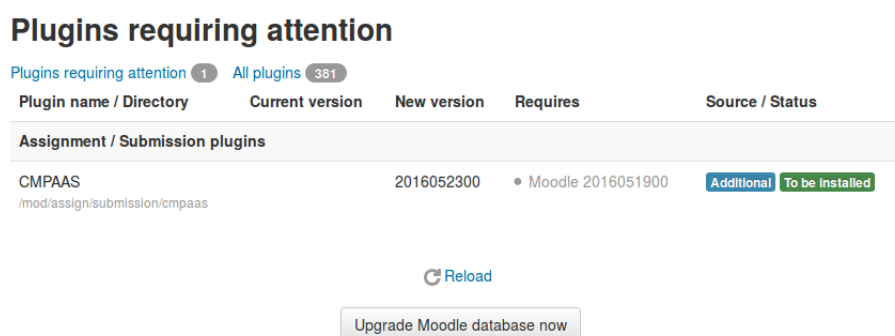
caso fosse necessário. Porém ainda havia a necessidade de permitir que o avaliador tivesse acesso ao conteúdo gerado pelo aluno e, além disso, proposta do trabalho era a criação de um plugin integrado com o CMPaaS, portanto a segunda parte do trabalho e realizar esta integração e criar uma forma de o avaliador visualizar o mapa.

## 6.5.2 Integração com o CMPaaS

### 6.5.3 Instalação e uso

A pasta contendo os arquivos do plugin deve ser copiada para o diretório do moodle mod/assign/submission/. Após copiar os arquivos para este local é necessário acessar o moodle com um usuário com perfil de administrador. Assim que efetuar login irá aparecer uma mensagem de instalação de plugin, conforme a [Figura 11](#).

Figura 11 – Configurando a tarefa para utilizar o plugin



Fonte: Elaborada pelo autor

Depois do plugin ser instalado é necessário realizar a configuração da tarefa para utiliza-lo. Para que a tarefa utilize o plugin para submissão de dados é necessário seleciona-lo na tela de configuração da atividade, conforme ilustrado na [Figura 12](#).

A utilização do plugin é bem simples. Ao acessar uma tarefa o estudante deve clicar no botão de envio de tarefa e ao fazer isto o editor de mapas ira ser mostrado na tela. Após criar ou editar o mapa o aluno deve clicar no botão “Salvar alterações” e o json será salvo no banco de dados da plataforma e aparecerá no sumário, conforme [Figura 13](#).

Figura 12 – Configurando a tarefa para utilizar o plugin

▼ Submission types

Submission types ☒ CMPAAS ☐ File submissions ☐ Online text ☐ OneNote submissions

Word limit  ☐ Enable

Maximum number of uploaded files

Maximum submission size

Word limit  ☐ Enable

Maximum number of uploaded OneNote pages

OneNote page size

Fonte: Elaborada pelo autor

Figura 13 – Configurando a tarefa para utilizar o plugin

Submission status	
Submission status	Submitted for grading
Grading status	Not graded
Due date	Monday, 17 October 2016, 1:00 AM
Time remaining	Assignment was submitted 24 days 11 hours late
Last modified	Thursday, 10 November 2016, 12:10 PM
CMPAAS	<div><div></div><div><pre>{ "class": "go.GraphLinksModel", "nodeDataArray": [ { "text": "Capítulo 1", "key": "-1", "loc": "-210.72041083170865 -56.56608954917805", [ "text": "Mapas Conceituais", "loc": "-239.37692552041673 7.193817273917574", "key": "-2", { "text": "pg", "key": "-3", "loc": "- 105.16707740625003 -178.52745996875012", { "text": "Introdução", "loc": "-252.56325178125002 -118.68975534375005", "key": "-4", [ "text": "Capítulo 2", "loc": "-55.547746890625035 -100.05013790625003", "key": "-5", { "text": "O que são", "key": "-6", "loc": "- 277.00478203125004 182.299521796875", { "text": "Origem", "loc": "-238.72180319229176 244.2439551801676", "key": "-7", [ "text": "Quem criou", "loc": "-165.34591573916674 153.26424210204252", "key": "-8", { "text": "Importância", "loc": "-119.78266395791675 70.17831238329248", "key": "-9", { "text": "EAD", "loc": "34.56441057812481 -19.644399468750038", "key": "-10", { "text": "Moodle", "loc": "4.067960390624819 58.08114768749999", "key": "-11", { "text": "Mapas e EAD", "loc": "75.09302934374983 56.74105204687493", "key": "-12", { "text": "Capítulo 3", "loc": "57.65034029687496 -119.78162589062502", "key": "-13", { "text": "Cmpaas", "loc": "108.89972418749988 -35.35560053124999", "key": "-14", { "text": "new node", "loc": "34.65408426083326 153.26424210204254", "key": "-15", { "text": "new node", "key": "-16", "loc": "-60 224" } ], "linkDataArray": [ { "from": "-3", "to": "-1", "text": "" }, { "from": "-3", "to": "-4", "text": "" }, { "from": "-3", "to": "-5", "text": "" }, { "from": "-2", "to": "-6", { "from": "-2", "to": "-7", { "from": "-1", "to": "-2", { "from": "-2", "to": "-8", { "from": "-2", "to": "-9", { "from": "-5, "to": "-10", { "from": "-10, "to": "-11", { "from": "-10, "to": "-12", { "from": "-3, "to": "-13", { "from": "-13, "to": "-14", { "from": "-8, "to": "-15" } ] }</pre></div></div>
Submission comments	<a href="#">Comments (0)</a>

Fonte: Elaborada pelo autor

## 7 Conclusão

## Referências

ALMEIDA, M. E. B. de. Tecnologia e educação a distância: Abordagens e contribuições dos ambientes digitais e interativos de aprendizagem. *Revista Brasileira de Educação a Distância*, p. 6, 2010.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. *NBR 6028*: Resumo - apresentação. Rio de Janeiro, 2003. 2 p.

AUSUBEL, D. P. The psychology of meaningful verbal learning. Grune and Stratton, New York, 1963.

BRASIL. Decreto 5.622, de 19 de dezembro de 2005. Regulamenta o artigo 80 da Lei no 9.394, de 20 de dezembro de 1996, que estabelece as diretrizes e bases da educação nacional. 2005. Disponível em: <[http://www.planalto.gov.br/ccivil\\_03/\\_ato2004-2006/2005/decreto/d5622.htm](http://www.planalto.gov.br/ccivil_03/_ato2004-2006/2005/decreto/d5622.htm)>. Acesso em: 08/11/2016.

CHAVES, E. Tecnologia na educação, ensino a distância, e aprendizagem mediada pela tecnologia: conceituação básica. *Revista Educação da Faculdade de Educação da Pontifícia Universidade Católica de Campinas*, v. 3, n. 7, 1999.

CROCKFORD, D. The application/json media type for javascript object notation (json). 2006.

DOUGIAMAS, M.; TAYLOR, P. Moodle: Using Learning Communities to Create an Open Source Course Management System. *Research.Moodle.Net*, p. 1–16, 2003. Disponível em: <[http://research.moodle.net/pluginfile.php/15/mod\\_{\\\_}data/content/1121/Moodle-Dougiamas-2003.>](http://research.moodle.net/pluginfile.php/15/mod_{\_}data/content/1121/Moodle-Dougiamas-2003.>)

DUTRA, Í. M.; FAGUNDES, L. d. C.; CAÑAS, A. J. Um ambiente integrado para apoiar a avaliação da aprendizagem baseado em mapas conceituais. *XIII Simpósio Brasileiro de Informática na Educação - SBIE - Unisinos*, p. 49–59, 2002.

FLANAGAN, D. *JavaScript: the definitive guide*. [S.l.]: "O'Reilly Media, Inc.", 2006.

FRANCISCATO, F. T. et al. Avaliação dos Ambientes Virtuais de Aprendizagem Moodle, TelEduc e Tidia - Ae: um estudo comparativo. *Novas Tecnologias na Educação*, v. 6, n. 2, p. 5–10, 2008.

MINETTO, E. L. Frameworks para desenvolvimento em php. *São Paulo: Novatec*, 2007.

MOODLE. *Moodle Statistics*. 2016. Disponível em: <<https://moodle.net/stats/>>. Acesso em: 16/11/2016.

NOVAK, J. D. Results and Implications of a 12-Year Longitudinal Study of Science Concept Learning. *Research in Science Education*, v. 35, n. 1, p. 23–40, mar 2005. ISSN 0157-244X. Disponível em: <<http://link.springer.com/10.1007/s11165-004-3431-4>>.

NOVAK, J. D.; CAÑAS, A. J. The origins of the concept mapping tool and the continuing evolution of the tool. *Information Visualization*, v. 5, n. 3, p. 175–184, 2006. ISSN 1473-8716. Disponível em: <<http://ivi.sagepub.com/lookup/doi/10.1057/palgrave.ivs.9500126>>.

PERIN, W. d. A. *iMap – UM MECANISMO DE INFERÊNCIA PARA MAPAS CONCEITUAIS*. 120 p. Dissertação (Mestrado) — Universidade Federal do Espírito Santo, 2014.

RICHARDS, G. et al. An analysis of the dynamic behavior of JavaScript programs. *Proceedings of the 2010 ACM SIGPLAN conference on Programming language design and implementation*, p. 1, 2010. ISSN 03621340. Disponível em: <<http://doi.acm.org/10.1145/1806596.1806598>{\%}5Cn[http://dl.acm.org/ft{\\\_}gateway.cfm?id=1806598{\&}type=pdf{\%}5Cnhttp://portal.acm.org/citation.cfm?doid=180](http://dl.acm.org/ft{\_}gateway.cfm?id=1806598{\&}type=pdf{\%}5Cnhttp://portal.acm.org/citation.cfm?doid=180)>.

TAURION, C. Cloud computing: computação em nuvem: transformando o mundo da tecnologia da informação. *Rio de Janeiro: Brasport*, v. 2, n. 2, p. 2–2, 2009.