

Desenvolvimento de um *plugin* para construção e gestão de objetos de aprendizagem interativos no CMS WordPress

Dionei Junior Marques Mariño, Ricardo Frohlich da Silva

Instituto de Informática – Centro Universitário Franciscano (UNIFRA)
Santa Maria – RS – Brasil

dionei.marino@unifra.edu.br, ricardo.frohlich@unifra.br

Abstract. *This paper exposes the development of a plugin for managing digital and didactic resources, and available questionnaires, to make them reusable in the construction of interactive learning objects. In addition, it proposes a model that allows the interactive use of learning objects generated by the plugin.*

Resumo. *Este trabalho expõe o desenvolvimento de um plugin para o gerenciamento de recursos digitais e didáticos, e de questionários avaliativos, com a finalidade de torná-los reutilizáveis na construção de objetos de aprendizagem interativos. Além disso, propõe um template que possibilite o uso interativo dos objetos de aprendizagem gerados pelo plugin.*

1. Introdução

A Educação a Distância (EaD) vem crescendo rapidamente em todo o mundo. Nota-se que profissionais e instituições - incentivados tanto pelas diversas possibilidades disponibilizadas pelas Tecnologias da Informação e das Comunicações (TICs), como pela diversificação de recursos pedagógicos - veem nesta forma de educação um meio para proporcionar o acesso ao conhecimento, de expandir oportunidades de trabalho e de aprendizagem ao longo da vida, conforme relatado em Sead Ufscar (2015). Schneider *et al* (2011) relata que a evolução da EaD no Brasil, nos últimos anos, tem gerado uma série de modelos educacionais que se utilizam de diferentes tipos de tecnologias, mídias e recursos.

Identifica-se que o desenvolvimento dos computadores pessoais com aumento da qualidade gráfica, os projetos de interfaces focados na usabilidade e acessibilidade, a disponibilidade de *softwares* de desenvolvimento para ambientes virtuais, e a disseminação da Internet propicia que surjam propostas inovadoras para criação de aplicações. Conforme Cunha e Mainente (2011), essas propostas têm auxiliado na definição de um novo paradigma de educação à distância - denominado de aprendizado eletrônico – quem tem o foco no ensino e na aprendizagem em ambientes virtuais, com o uso de recursos digitais diversificados.

A elaboração de objetos de aprendizagem, por parte do professor, pode exigir deste um conhecimento e/ou um gasto considerável de tempo, acarretando na simplificação ou, até mesmo, abandono na elaboração de material mais completo. Dada

esta problemática, pretende-se então propor um *plugin* para facilitar a criação de objetos de aprendizagem interativos em sistemas baseados em WordPress. O *plugin* permite a criação páginas contendo os objetos de aprendizagem (OA) interativos, que tem o foco na gestão de recursos digitais (R) e de questionários avaliativos (Q), agrupando-os em recursos didáticos (RD). Esses objetos de aprendizagem, são compostos por uma sequência de recursos didáticos e questionários avaliativos.

O diferencial deste sistema é o de permitir uma maior autonomia ao professor para criar um objeto de aprendizagem com sequenciamento lógico dos recursos didáticos; além de permitir a inserção de recurso digital complementar quando o questionário de um recurso didático não for corretamente respondido.

1.1. Objetivo geral

O objetivo geral deste trabalho é o desenvolvimento de um *plugin* para que auxilie a criação de objetos de aprendizagem interativos em sistemas baseados em WordPress. Este *plugin* será responsável pelo gerenciamento de recursos digitais e questionários avaliativos através dos *Custom Post Type*; da categorização especializada de recursos didáticos através das Taxonomias, bem como da interação com os objetos de aprendizagem através das *Custom Pages Templates*.

1.2. Objetivos específicos

- Pesquisar e apresentar as tecnologias computacionais para o desenvolvimento do trabalho;
- Identificar e modelar as funções necessárias ao desenvolvimento do *plugin*;
- Desenvolver um *plugin* para criação e gerenciamento dos OAs (*back-end*);
- Desenvolver um *template* para a interação do aluno com os OAs (*front-end*);

2. Referencial teórico

Nesta seção são apresentados conteúdos e conceitos pertinentes ao desenvolvimento deste trabalho, como: educação à distância, objetos de aprendizagem, os recursos tecnológicos para o desenvolvimento do *plugin* e *template*, para o Sistema de gerenciamento de conteúdo (*Content Management System – CMS*) WordPress, e os trabalhos relacionados.

2.1. Conceitos Básicos de EaD

Segundo Moran (1994) *apud* Brauer (2009), Educação à Distância é o processo de ensino-aprendizagem, mediado por tecnologias, no qual professores e alunos estão separados espacial e/ou temporalmente, porém eles podem estar conectados, interligados por tecnologias da informação e da comunicação, como a Internet.

A Figura 1 apresenta a distinção entre os recursos de comunicação e as mídias existentes em 2007.

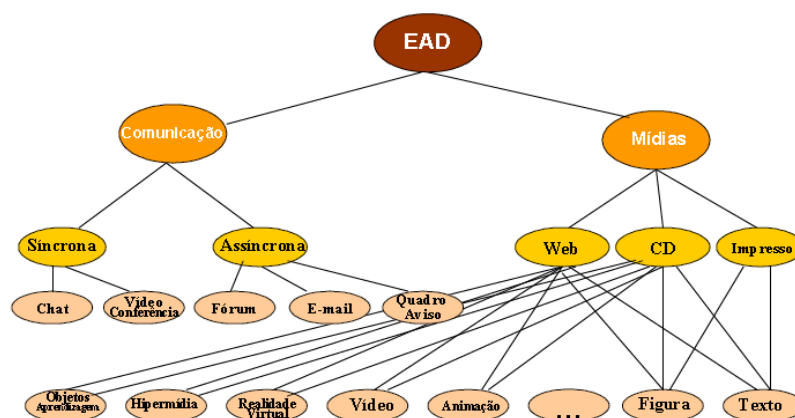


Figura 1. Tecnologias de comunicação e mídias para EaD, Campos *et al* (2007)

Conforme já dito na introdução, este trabalho pretende focar nos objetos de aprendizagem. Segundo Campos *et al* (2007), objetos de aprendizagem são quaisquer recursos digitais que podem ser reutilizados para dar suporte ao ensino. Objetos de aprendizagem podem ser divididos em “pequenos conteúdos” (ou recursos digitais) de forma que estes possam ser reutilizados em ambientes de aprendizagem distintos. Como se pretende que estes recursos sejam reutilizáveis, pode-se, inclusive, personalizá-los e adequá-los às necessidades do público alvo. Note que Silveira (2015) complementa a descrição acima apresentada, indicando que objetos de aprendizagem:

“São quaisquer materiais eletrônicos (como imagens, vídeos, páginas Web, animações ou simulações), desde que tragam informações destinadas à construção do conhecimento (conteúdo autocontido), explicitem seus objetivos pedagógicos e estejam estruturados de tal forma que possam ser reutilizados e recombinaados com outros objetos de aprendizagem (padronização).” [Silveira 2015]

Além disso, Santos *et al* (2007) indica que, para o objeto de aprendizagem ser recuperado e reutilizado, é preciso que esse objeto seja devidamente indexado e armazenado em um repositório.

Clark (1998) *apud* Campos *et al* (2007) salienta que objetos instrucionais possuem objetivos de aprendizagem, contêm uma sequência lógica de utilização, são avaliados por exercícios práticos, e possibilitam receber *feedback*.

2.2. Objetos de Aprendizagem, foco na construção com qualidade

Carneiro e Silveira (2012) permitem uma reflexão, por parte do professor, sobre a construção de um objeto de aprendizagem com qualidade, procurando questionar se o objeto de aprendizagem: apresenta, de forma clara, o objetivo que aluno deve alcançar no que diz respeito ao conhecimento adquirido; pretende apresentar conteúdo novo, ou apenas reforçar o conhecimento adquirido; motiva o aluno a querer compreender melhor o conteúdo apresentado, e que todo objeto de aprendizagem deve conter referências deste conteúdo apresentado.

Silveira e Carneiro (2012) definem 7 (sete) diretrizes que devem ser satisfeitas para que um recurso se torne um objeto de aprendizagem com qualidade. Dessas diretrizes, as que este trabalho pretende satisfazer são:

- priorizar o digital - é necessário priorizar o desenvolvimento de objetos de aprendizagem que não necessitam, para sua utilização, aplicativo ou programa que não esteja disponível gratuitamente na *Web*;
- proporcionar interação - é necessário permitir ações entre os usuários (alunos, professores, tutores, etc.) a partir do objeto e/ou nele.
- proporcionar interatividade - é necessário que o usuário tenha a possibilidade de interagir, executando ações com o objeto.

Para garantir a qualidade na construção de objetos de aprendizagem, Tarouco (2010) indica que é desejável que haja o uso integrado de sistemas de gestão de conteúdo e ambientes virtuais de aprendizagem. A autora relata ainda, que é necessária a compreensão dos seguintes conceitos:

- Sistema de gerenciamento de conteúdo (*Content Management System - CMS*): plataforma de publicação e gerenciamento de conteúdo, que possui diversos recursos para tornar a experiência de publicação mais simples, agradável e atrativa.
- Ambientes Virtuais de Aprendizagem (AVA) ou *Learning Management Systems (LMS)*: permite o controle de acesso, a criação e a gestão de grupos e turmas, um calendário e a integração com os sistemas de informação da organização;
- Sistema de Gerenciamento de Conteúdo de Aprendizagem ou *Learning Content Management Systems (LCMS)*: gerencia o conteúdo de aprendizagem personalizado. Permite a criação e armazenamento de objetos de aprendizagem. Combina os aspectos administrativos e os de gerenciamento de um AVA com as funcionalidades para criação de conteúdo personalizado de um CMS.

Dado o contexto acima apresentado, pode-se concluir que um LCMS deve possuir as seguintes características: ser um repositório de conteúdo, ou seja, de objetos de aprendizagem, permitir a aplicação de autoria, ser uma interface de distribuição dinâmica, e disponibilizar aplicações de controle administrativo.

2.3. Recursos Tecnológicos

Tarouco (2010) e Silveira e Carneiro (2012) defendem que é desejável a adoção de padrões abertos para guiar tanto o projeto quanto o desenvolvimento de objetos de aprendizagem com interatividade. Portanto, esta seção apresenta as tecnologias necessárias para a o desenvolvimento deste trabalho.

2.3.1. HTML5 e CSS3

O *HiperText Markup Language* (HTML ou Linguagem de Marcação de Hypertexto) é uma linguagem para publicação de conteúdo (texto, imagem, vídeo, áudio

e etc) na *Web*. O *Cascade Style Sheet* (CSS ou Folhas de Estilo em Cascata) é um complemento ao HTML, que permite definir o estilo das páginas, através da formatação da informação entregue pelo HTML, de forma que esta informação possa ser consumida da melhor maneira pelo usuário final, conforme W3C Brasil (2015).

2.3.2. Biblioteca JavaScript jQuery

A biblioteca jQuery é um *software* livre e aberto. Segundo Silva (2014), é uma linguagem destinada a adicionar interatividade e dinamismo às páginas *Web*. Proporciona ao desenvolvedor funcionalidades necessárias à criação de scripts que visem incrementar a usabilidade, a acessibilidade e o design para o usuário.

2.3.3. Linguagem PHP

Permite o processamento das informações e a geração rápida de páginas de conteúdo de forma dinâmica pelo servidor. Possui ampla utilização no mercado, é uma linguagem interpretada, e, geralmente, é mesclada junto ao código HTML. Sua sintaxe acessível remete às linguagens “C”, *Java* e *Perl*, conforme PHP.net (2015).

2.3.4. WordPress

O WordPress é um sistema de gerenciamento de conteúdo (CMS ou *Content Manager System*). Baseado em Messenlehner e Coleman (2014), esta plataforma possui recursos e configurações para tornar experiência de publicação, mais fácil, agradável e atrativa, indo ao encontro da proposta deste trabalho, que possui foco no criador, o professor conteudista. Esses criadores são possíveis em função dos seus tipos de usuários, que possuem papéis e capacidades, além de diferentes níveis de permissões e acesso a determinados recursos do painel administrativo. Conforme Codex (2015), esse CMS fornece um diretório de *plugins* e *templates* gratuitos para adicionar recursos ao seu site, ou sistema, além de prover a possibilidade de desenvolver *templates* e *plugins* próprios e disponibilizá-los no diretório mundial, juntamente com a documentação oficial chamada Codex.

As publicações são utilizadas para exibir conteúdo, seus tipos são *posts* e páginas. Uma página geralmente mantém um conteúdo de forma fixa, sendo essa uma página estática. Já a página que contém os *posts*, é chamada de dinâmica ou página de blog, onde o seu conteúdo será organizado conforme os termos escolhidos na criação, conforme Pereira (2015).

Os termos são usados para organizar os conteúdos, sendo possível a utilização de categorias, taxonomias - que são categorias personalizadas, e *tags*. As categorias são a primeira divisão do conteúdo, e as *tags* são utilizadas para subdividir essas categorias. As taxonomias são categorias personalizadas, geralmente utilizadas quando há a criação de tipos de *posts* personalizados (*Custom Post Types*), segundo Mazetto (2013).

Além disso, possui uma hierarquia de arquivos e trata de questões como banco de dados, *templates* e *plugins* através de funções nativas. Essa estrutura completa, a possibilidade de desenvolver *plugins* próprios e personalizados, e as funções nativas são

os principais recursos que permitem ao WordPress ser usado para o desenvolvimento das funcionalidades descritas neste trabalho.

2.4. Trabalhos Relacionados

Nesta seção apresenta-se os trabalhos relacionados com este trabalho. São sistemas que propõem a disseminação de conteúdos *on-line* de forma diferenciada.

O **TIMTec** é uma iniciativa do Instituto TIM, baseada em três pilares: *software* livre, produção de cursos e parcerias com instituições de ensino, segundo Instituto TIM (2015). TIM Tec é então uma plataforma no estilo MOOC (*massive open on-line courses* ou cursos *on-line* abertos e massivos, em tradução livre), desenvolvida em código aberto para que instituições interessadas também possam se apropriar da plataforma e adaptá-la aos seus interesses e objetivos. O *software* pode ser requisitado, instalado e modificado. O **CodeSchool** é um ambiente de aprendizagem *on-line* para desenvolvedores que pretende ensinar por meio de conteúdo com entretenimento (Pluralsight 2015). Cada curso possui um tema criativo e enredo. Os cursos iniciam com conteúdo introdutório e avançam em níveis de aprendizagem. São combinados numa mecânica de jogo com instruções em vídeo, e no navegador de codificação realizam desafios que tornam a aprendizagem tanto educacional quanto memorizável.

O **MAIS Unifra** é um portal, do Setor de EaD do Centro Universitário Franciscano, que disponibiliza conteúdos digitais e objetos de aprendizagem, todos de autoria própria, e produzidos para que a comunidade acadêmica e local possam utilizá-los em diferentes contextos, conforme MAIS Unifra (2015).

2.5. Considerações sobre os trabalhos relacionados

O TIMTec e o CodeSchool são ambientes de aprendizagem com conteúdos pré-existentes e sem possibilidades de criação por um professor que não componha a equipe de produção dentro dessas instituições. Já o MAIS Unifra atende apenas aos pedidos dos membros da instituição. Este servirá para fornecer conteúdos que serão utilizados na construção de objetos de aprendizagem na fase de implantação deste trabalho.

Em virtude do exposto acima, este trabalho pretende atender as demandas não atendidas por essas iniciativas, que são: a criação de objetos de aprendizagem por professores sem equipes de apoio e possibilitar ao aluno, acesso direto aos objetos de aprendizagem, independente de ambientes virtuais de aprendizagem.

A comparação entre as plataformas acima expostas e o trabalho desenvolvido, é o foco no professor. As iniciativas acima dependem de equipes para desenvolver os objetos de aprendizagem, já neste trabalho o professor tem autonomia para criação. Outra comparação com este trabalho é a mudança do conteúdo exposto ao aluno nas situações de erro, ou não aprendido, onde naqueles o conteúdo é revisto da mesma forma, já neste trabalho um novo conteúdo é exposto ao aluno.

3. Metodologia

Para Sommerville (2007), metodologia é a maneira ou forma de utilizar um conjunto coerente e coordenado de métodos para alcançar um objetivo, primando pela qualidade e produtividade nos projetos, é fornecido um roteiro dinâmico e interativo para o desenvolvimento de forma estruturada dos projetos.

A Engenharia de Software define que as fases do modelo de ciclo de vida clássico (ou modelo cascata) são:

- Levantamento e análise de requisitos - é a investigação para tentar descobrir o que o cliente quer. Na **Seção 3.1** é apresentado o resultado da aplicação destas fases, isto é, a identificação e descrição do problema de criação de objetos de aprendizagem interativos;
- Projeto - é a proposta de solução com base no conhecimento adquirido na fase anterior, que utiliza a modelagem UML para ilustrar de forma técnica, através de diagramas. Na **Seção 3.2** é apresentada a solução para o problema descrito na Seção 3.1. Salienta-se que o sistema é implementado com o uso do CMS WordPress;
- Implementação e testes - fase onde o sistema é codificado e convertido em um sistema instalável ou executável. Também são realizados testes de construção do sistema, concentrando-se na especificação documentada na fase de projeto. As telas do *plugin* e *template* desenvolvidos serão apresentadas na **Seção 3.3**, juntamente com um estudo de caso desenvolvido para testes do mesmo; e
- Implantação - realizar a criação de um objeto de aprendizagem junto aos professores que participaram do levantamento de requisitos, tendo como base os conteúdos do portal MAIS Unifra, e construir um manual de uso do *plugin*.

O modelo em cascata passa para a próxima fase apenas quando a fase anterior está completa e finalizada. Os desenvolvimentos no modelo cascata são rígidos, não havendo adiantamentos ou retrocessos. Possuem uma ordenação previamente estabelecida, pois cada fase depende do produto final da anterior para iniciar. Nesse sentido, a adoção do modelo em cascata para o desenvolvimento deste sistema é o mais indicado pois não há equipe de desenvolvimento nem clientes envolvidos.

3.1 Levantamento e análise de requisitos

No Centro Universitário Franciscano, tem-se um setor de produção de objetos de aprendizagem. O levantamento de requisitos foi feito junto a este grupo, mais especificamente junto ao coordenador desse setor. Deste levantamento de requisitos, pode-se concluir a existência dos seguintes problemas: a complexidade no processo de produção de objetos de aprendizagem, grande período utilizado para a criação, e necessidade constante de capacitação de professores conteudistas na compreensão da concepção e construção dos conteúdos dos objetos de aprendizagem.

Pode-se concluir que a elaboração de objetos de aprendizagem - por parte dos professores - exige deste um vasto conhecimento e/ou um longo tempo de elaboração. Acarretando na simplificação ou, até mesmo, abandono na elaboração de material mais completo. Logo, a construção de objetos de aprendizagem interativos e com qualidade demanda um grande esforço por parte do professor e do setor responsável.

3.2 Projeto

O projeto foi modelado através do Diagrama de Casos de Uso, exposto na Figura 2, onde são apresentadas as funcionalidades necessárias. O ator ProfessorConteudista, após efetuar o *login* no sistema, pode efetuar a gestão de recursos didáticos, que são questionários avaliativos e recursos digitais. Essas gestões estão indicadas nos casos de uso denominados por “Gerenciar” que são basicamente um **CRUD** (isto é, "*Create Read Update Delete*" ou "Criar Ler Atualizar Deletar"). O ator ProfessorConteudista pode também gerenciar e publicar páginas de objetos de aprendizagem. O ator Aluno pode interagir com objetos de aprendizagem, após realizar o *login* no sistema.

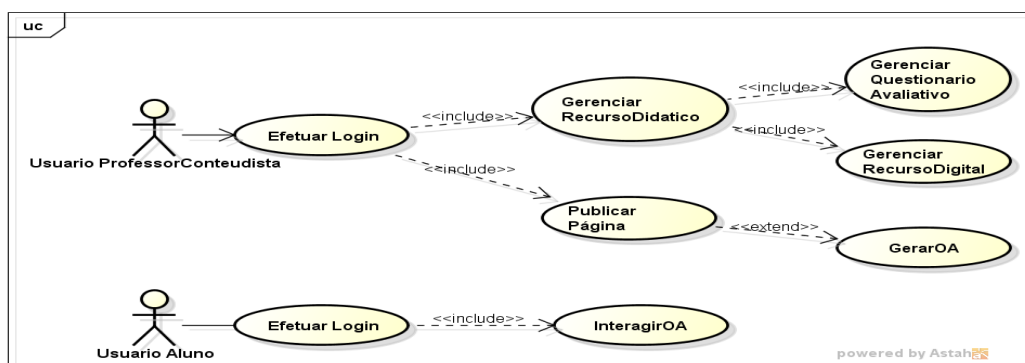


Figura 2. Diagrama de Casos de Uso

No Diagrama de Classes da Figura 3, as classes representam que: o **professor conteudista**, será o proprietário do objeto de aprendizagem; um **objeto de aprendizagem** é composto por um conjunto de recursos didáticos; um **recurso didático** é composto por recursos digitais e um questionário avaliativo; **questionário avaliativo** é composto por perguntas; e **aluno** é um usuário que utiliza os objetos de aprendizagem.

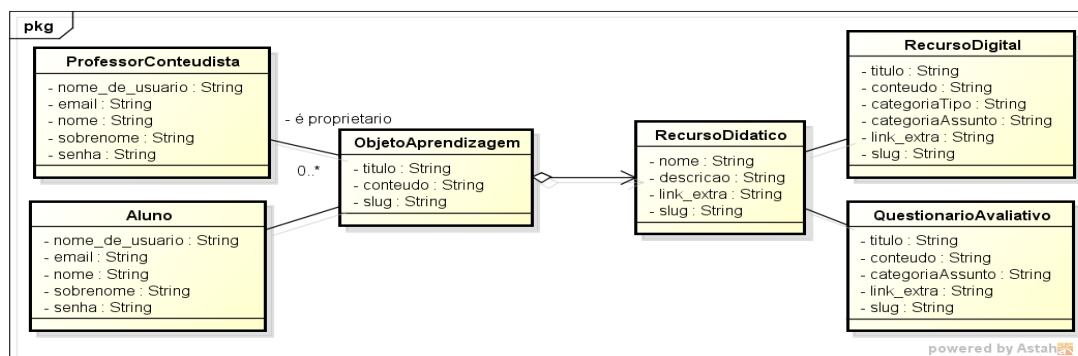


Figura 3. Diagrama de Classes

3.3 Implementação e testes

O objetivo do trabalho é o desenvolvimento de um *plugin* para a criação de objetos de aprendizagem e de um *template* para interação com os objetos de aprendizagem.

Baseado na problemática apresentada e seguindo as propostas de Silveira e Carneiro (2012) - para a construção de objetos de aprendizagem interativos, e de Tarouco (2010) - no que diz respeito as funcionalidades desejáveis do sistema, foi utilizado o CMS WordPress como base para desenvolver o *plugin*, o mesmo fora escolhido em função de compilar todas as tecnologias *Web* necessárias de uma maneira versátil e adaptativa.

Para tal, foi desenvolvido o *plugin* CriaOA para o gerenciamento dos recursos digitais e questionários avaliativos, que são recursos didáticos integrantes dos objetos de aprendizagem.

Sobre o *plugin*, foram utilizadas publicações nativas, como *posts*, páginas e categorias, que se tornaram, respectivamente: o **Custom Post Type** - “**ctp_recurso**” - parte integrante do *plugin* e usada para definir um novo tipo de *post*, chamado de recurso didático, que foram categorizados em recursos digitais e questionários avaliativos; o **Custom Page Template** - “**page.php**”, que define o layout e a forma de exibição os objetos de aprendizagem criados; as **Taxonomias** “**tipo_rd**” e “**assunto_rd**” - parte integrante do *plugin* e categorizam os *posts* em tipos de recursos didáticos (Recursos digitais e Questionários avaliativos) e em assuntos; bem como a **Taxonomia** “**assunto_oa**” - parte integrante do *plugin* e categoriza as páginas dos objetos de aprendizagem.

Todo *plugin* necessita de uma pasta principal (a pasta “*criaoa_p*”), que é colocada na pasta “*plugins*” dentro da pasta “*wp-content*”, e de um arquivo “*nome_plugin.php*” (nesse caso, “*criaoa_p.php*”), bem como, de outras pastas para a organização do seu conteúdo. O arquivo “*criaoa_p.php*” possui um código comentado utilizado para descrever as informações do *plugin*. Essas informações aparecem no item “Descrição”, da aba “*Plugins*”, do painel de controle do WordPress.

Além disso, estão contidos no arquivo comentado acima, os comandos PHP *require_once()*, que garantem a inclusão única dos arquivos passados como parâmetro, esses arquivos contém os *scripts* PHP de criação do *plugin*. O comando também atua de forma a não permitir que a aplicação seja executada em caso de ocorrer algum erro nos códigos fonte.

O arquivo “*cpt_recurso.php*”, está dentro da pasta “*custom-posts*”, e serve para criar o tipo de *post* personalizado, que foi utilizado para criação dos recursos digitais e questionários avaliativos. Neste arquivo existem as seguintes funções:

- **cpt_criar_recurso()** - função criada para receber os parâmetros de criação de um tipo de *post* personalizado;
- **register_post_type()** - função nativa do WordPress que registra o tipo de *post* personalizado com os argumentos passados na função anterior. Sua chamada é realizada dentro da função anterior;

- **add_action()** - função nativa utilizada para fazer a inicialização do tipo de *post* personalizado no painel de controle do WordPress, conforme destaque na Figura 4.

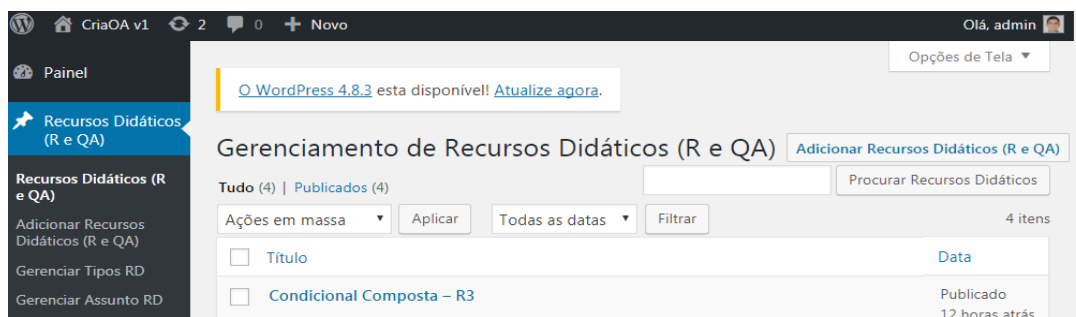


Figura 4. Custom Post Type - “cpt_recurso” adicionado no Painel de Controle (back-end)

Os arquivos “*tipo_rd.php*”, “*assunto_rd.php*” e “*assunto_oa.php*”, que estão dentro da pasta “*taxonomias*”, servem para criar as taxonomias (categorias personalizadas), que são utilizadas para categorização dos tipos de *post* personalizados criados anteriormente. Esses arquivos possuem as seguintes funções:

- **cria_tipo_rd()**, **cria_assunto_rd()** e **cria_assunto_oa()** - funções criadas para receber os parâmetros de criação das taxonomias;
- **register_taxonomy()** - função nativa do WordPress que registra a taxonomia utilizando os argumentos passados na função anterior. Suas chamadas são realizadas dentro das funções anteriores;
- **add_action()** - função nativa utilizada para fazer a inicialização das taxonomias dentro das abas no painel de controle do WordPress. No caso dos “Recursos Didáticos (R e QA)” são criadas as taxonomias “Gerenciar Tipos RD” e “Gerenciar Assunto RD”. No caso das “Páginas” é criada a taxonomia “Gerenciar Assunto OA”, conforme destaque na Figura 5.



Figura 5. Taxonomias - “Gerenciar Tipos RD, Gerenciar Assunto RD e Gerenciar Assunto OA” adicionadas nas abas do Painel de Controle (back-end)

Conforme Pereira (2015), um *template* é composto de forma ideal, pelos seguintes arquivos: *functions*, *header*, *footer*, *sidebar*, *frontpage*, *index*, *home*, *archive*, *page*, *single*, *search*, *author*, *404*, *category* e *taxonomy*, todos com extensão “.php”. Para efeitos deste trabalho, apenas não foram identificadas a necessidade de realizar alterações nos arquivos *sidebar*, *archive*, *author*, *404* e *category*.

O desenvolvimento do *template* InterageOA focou nos arquivos “*single.php*” e “*page.php*”, pois são os arquivos que fazem o resgate dos dados incluídos nos recursos didáticos e páginas de objetos de aprendizagem, respectivamente.

Para resgatar esses dados e formatar essas informações, são necessárias o uso de algumas funções nativas do CMS WordPress, conforme a seguir:

- **get_the_title()** e **get_the_content()**, **get_the_permalink()** - funções de resgate de conteúdo, e a última de criação de link de acesso ao conteúdo;
- **WP_Query**, **have_posts()** e **the_post()**, **next_post_link()** - funções relativas a busca em banco de dados, resgate de conteúdo, e criação de links de acesso ao próximo *post*, ou recurso didático ;
- **wp_head()** e **get_header()** - funções de registro e resgate do cabeçalho;
- **wp_footer()** e **get_footer()** - funções de registro e resgate do rodapé;
- **register_nav_menu()** e **wp_nav_menu()** - funções de registro e posicionamento de menu;

O *layout* do tipo de *post* personalizado segue o mostrado no lado esquerdo da Figura 6, e o da página de objetos de aprendizagem no lado direito da mesma Figura.

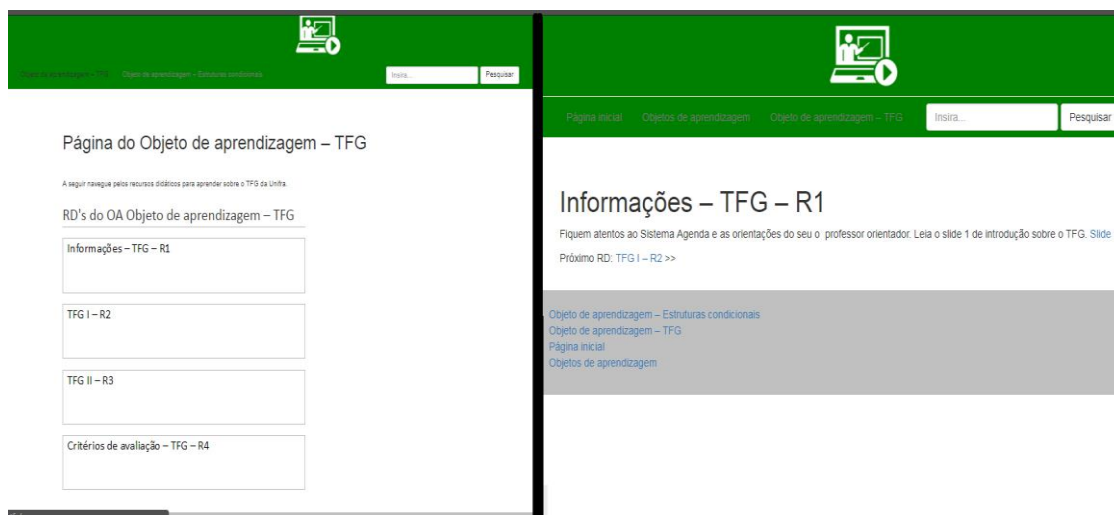


Figura 6. *Layout* do *template* InterageOA, referente aos arquivos “*page.php*” (LE) e “*single.php*” (LD), visão do aluno (*front-end*)

Para que a utilização do *plugin* CriaOA e do *template* InterageOA seja possível, é necessário adaptar os usuários nativos do sistema. Assim o Administrador, Autor e Assinante, tornam-se, respectivamente: O **Administrador do Sistema** - responsável pelo desenvolvimento e criação de novas funcionalidades, sob demanda do professor conteudista; O **Professor Conteudista** - responsável pela criação de recursos didáticos, e concepção dos objetos de aprendizagem. E o **Aluno** - responsável pela leitura e interação com objetos de aprendizagem.

Estudo de caso

O *plugin* e *template* para construção de objetos de aprendizagem interativos, atuam conforme as diretrizes apresentadas em Silveira e Carneiro (2012), de forma a auxiliar o professor na criação e gestão dos objetos de aprendizagem, bem como o aluno na compreensão do problema, através da divisão do problema em subproblemas originando recursos didáticos e, posteriormente objetos de aprendizagem para interação.

Este trabalho apresenta o *plugin* “CriaOA” para a criação e gerenciamento de objetos de aprendizagem interativos, com foco na gestão de recursos digitais, de recursos didáticos (composto por recursos digitais e questionários avaliativos, organizados em sequência), e de páginas de objetos de aprendizagem (composto então por uma sequência de recursos didáticos com questionário avaliativo).

Para acessos ao sistema, o professor conteudista deverá realizar o *login* (lado esquerdo), e será redirecionado para o painel de controle, devendo clicar no botão “Recursos Didáticos (R e QA)” (seta do lado direito), para que comece a realizar a criação dos recursos, conforme Figura 7.

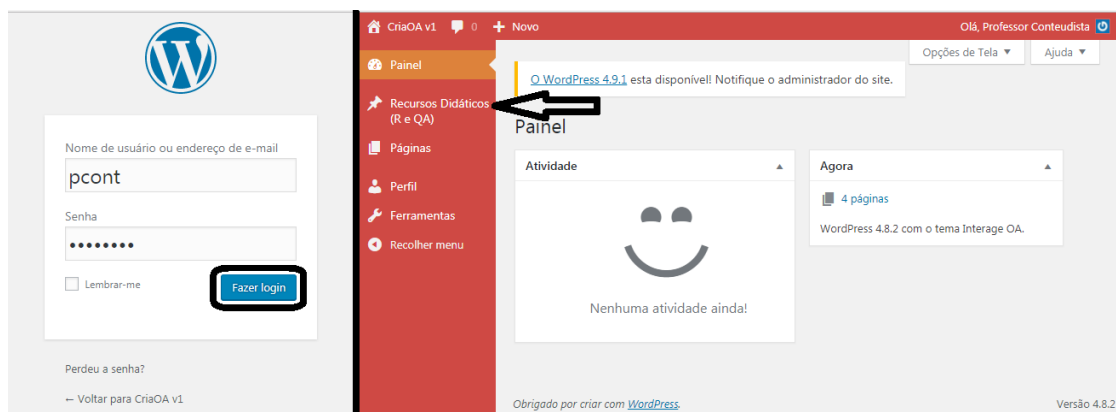


Figura 7. Tela de login e acesso ao painel de controle, visão do professor (back-end)

A criação dos recursos e a organização dos objetos de aprendizagem ocorrem após o Professor Conteudista clicar no botão “Adicionar Recursos Didáticos”, conforme ilustrado no lado esquerdo na Figura 8. Após o preenchimento dos campos de texto com o título e conteúdo, deverá ser selecionado “Tipo de RD” (primeira seta), e depois o “Assunto de RD” (segunda seta), deve-se clicar em “Publicar” (terceira seta), conforme ilustrado no lado esquerdo da mesma Figura.

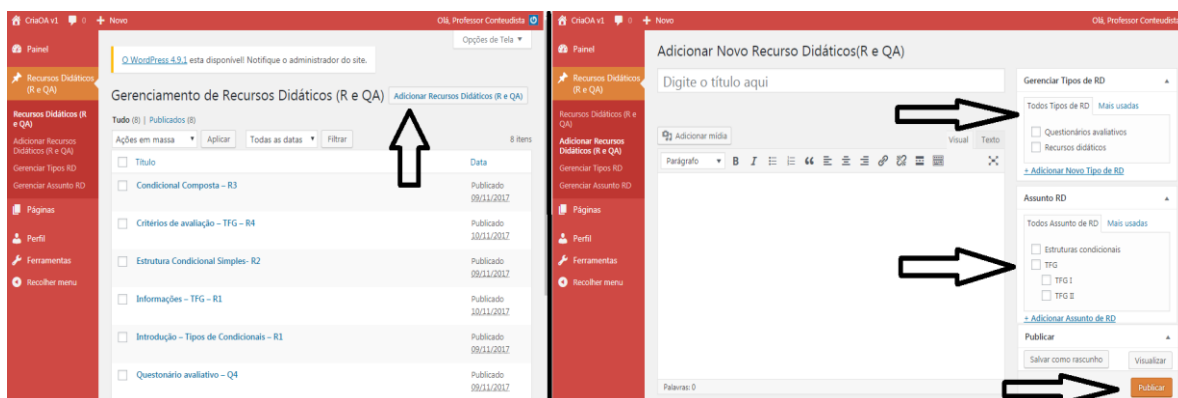


Figura 8. Tela de criação dos recursos, visão professor (*back-end*)

O após a criação dos conteúdos, o professor deverá realizar o mesmo procedimento para a criação das páginas dos objetos de aprendizagem, no botão “Páginas”. Após isso, o sistema está pronto para ser acessado pelo aluno.

O aluno deve realizar o *login* no sistema (circulado vermelho no lado esquerdo), e clicar no botão “CriaOA v1” (circulado vermelho e superior) para acessar o sistema, conforme Figura 9.

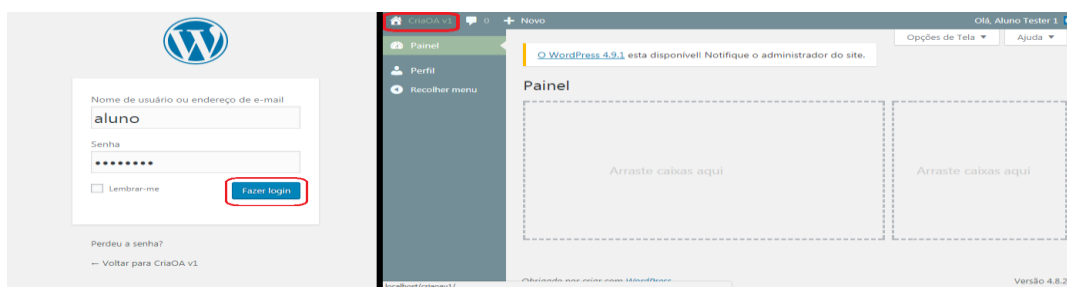


Figura 9. Tela de *login* e acesso ao painel de controle, visão do aluno (*front-end*)

A partir deste momento, ele clicará no link da página “Objetos de aprendizagem” (seta do lado esquerdo) e será direcionado para as páginas dos objetos de aprendizagem “Páginas de OA”, devendo clicar em “Objeto de aprendizagem - TFG”, conforme exibido no lado direito da Figura 10.

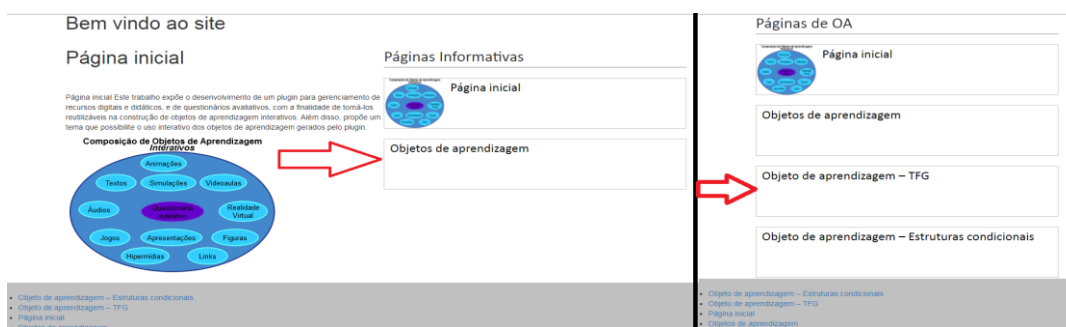


Figura 10. Layout do *template* InterageOA para interação com o objeto de aprendizagem, visão do aluno (*front-end*)

A partir deste momento, ele estará acessando a página onde serão mostrados os recursos didáticos, devendo clicar no primeiro recurso, chamado “Informações – TFG – R1” (seta do lado esquerdo), e será direcionado ao recurso didático para sua leitura (título circulado em vermelho no lado direito). Logo abaixo do conteúdo do recurso didático, há um *link* (seta do lado direito) que o levará para os próximos recursos e até chegar ao questionário, conforme exibido na Figura 11.

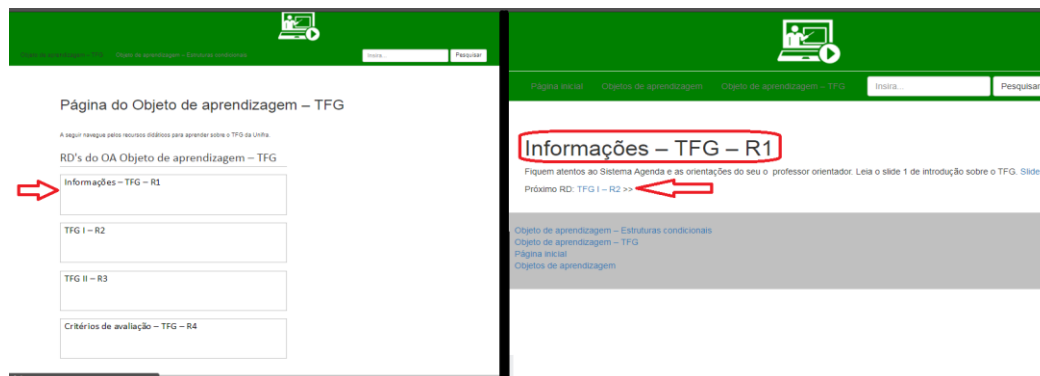


Figura 11. Layout do *template* InterageOA para interação com os recursos didáticos, visão do aluno (*front-end*)

Problemas encontrados

Na criação dos questionários avaliativos, a implementação através de um tipo de *post* personalizado não foi adequada, pois não foi possível suportar dinamicamente o resgate de informações do banco de dados. A tentativa de readequação do *template* “InterageOA” através das *Custom Page Template*, para extrair o conteúdo das perguntas e criar dinamicamente o formulário HTML, para o teste dos conhecimentos do aluno, também não ocorreu da forma esperada e, portanto, não foi concluída em sua totalidade.

Essa criação dinâmica, seria responsável por possibilitar a interação final com os objetos de aprendizagem de forma adequada e realizar o *feedback* do aprendizado ao aluno.

4. Conclusões e Trabalhos Futuros

Conforme a problemática apresentada, sobre a complexidade de criação e dependência de outros setores, por parte do professor conteudista para criar objetos de aprendizagem reutilizáveis e interativos, e conforme o projeto proposto, de criação de um *plugin* de gestão e de um *template* interativo, conclui-se que este trabalho desenvolveu de forma parcial os seus objetivos.

A criação e a categorização dos recursos digitais e didáticos foram implementadas com sucesso. A realização da gestão e categorização dos objetos de aprendizagem também foi implementada de forma satisfatória, ambas implementações foram possíveis através do desenvolvimento do *plugin* “CriaOA”.

Contudo, a utilização de um tipo de *post* personalizado para a criação de questionários avaliativos não foi adequada, tendo em vista a dificuldade de conseguir a adequação do *template* “InterageOA” para extrair o conteúdo das perguntas e criar dinamicamente o formulário HTML para o teste dos conhecimentos do aluno. Essa criação dinâmica, que seria responsável por possibilitar a interação final com os objetos de aprendizagem de forma adequada e dar o *feedback* do aprendizado ao aluno, não foi desenvolvida em sua totalidade.

Como trabalhos futuros, as sugestões são tanto qualitativas quanto quantitativas. No que se refere a estudos qualitativos, existe a avaliação da usabilidade do *template* (*front-end*) junto aos alunos do ensino 100% EaD, ensino híbrido bem como do ensino presencial (com uso do OA guiado pelo professor).

Já no que se refere a estudos quantitativos, existe a avaliação usabilidade do *back-end* (*plugin*), junto aos professores que lecionam no ensino 100% EaD, ensino híbrido e ensino presencial, de forma a verificar se o *plugin* facilita a criação de OAs pelos professores conteudistas, e reduz a dependência da equipe técnica envolvida.

No que tange novas funcionalidades à ser inseridas no WordPress, pode-se: incluir o Google Analytics, gerando análises e métricas relevantes para manutenção ou exclusão de objetos de aprendizagem, baseado no número de visitas, tempo de permanência médio em cada recurso didático ou objeto de aprendizagem; incluir a figura do tutor, como tipo de usuário colaborador, de forma a auxiliar na construção, condução ou interação do aluno com o objeto de aprendizagem; e incluir um *plugin* gerador de código para redirecionamento dos ambiente virtuais de aprendizagem Canvas e Moodle aos objetos de aprendizagem.

5. Referências

- Brauer, S. (2009) Fundamento de Educação a Distância. São Luis. UNICEUMA, 2009. Disponível em: <https://pt.scribd.com/doc/16337198/Fundamentosde-Educacao-a-Distancia>. Acessado em novembro de 2017.
- Carneiro, M.L.F. e Silveira, M.S. (2012) Objetos de Aprendizagem Sob o Ponto de Vista dos Alunos: Um Estudo de Caso. Revista Novas Tecnologias em Educação (RENOTE), Porto Alegre, v. 10, n. 3.
- Codex Brasil. “Documentação do WordPress”. Disponível em: https://codex.wordpress.org/pt-br:P%C3%A1gina_Inicial. Acessado em novembro 2015.
- Cunha, T. M. F. e Mainente, C. A. (2011) Utilização de ambientes virtuais 3D no ensino de ciência da computação: estado da arte. Universidade Municipal de São Caetano do Sul (USCS), São Caetano do Sul. Maio. Disponível em: http://www.uscs.edu.br/pesquisasacademicas/images/pesquisas/thiago_cilene.pdf.
- Instituto TIM. “O que são MOOCs”. Disponível em: <http://timtec.com.br/pt/o-que-sao-moocs/>. Acessado em outubro 2017.

- MAIS Unifra. “Sobre o MAIS Unifra”. Disponível em: <http://maisunifra.com.br/sobre-o-mais/>. Acessado em novembro 2017.
- Mazetto. (2013) “Tableless - WordPress Modus Operandi”. Disponível em: <http://tableless.com.br/wordpress-modus-operandi/>. Acessado em novembro 2017;
- Messenlehner, B; Coleman, J. Criando Aplicações Web com WordPress - WordPress como um Framework de Aplicações. 1ª Edição. Editora Novatec. 2014;
- Moran, J.M. (1994). Interferências dos meios de comunicação no nosso conhecimento. Revista Brasileira de Comunicação, São Paulo, v. 17, n. 2, jul./dez. Disponível em: <http://www.eca.usp.br/eca.prof/moran>. Acessado em novembro 2017.
- PHP.net. “Manual do PHP”. Disponível em: https://secure.php.net/manual/pt_BR/preface.php. Acessado em novembro 2017.
- Pluralsight. “Code School is now a Pluralsight company”. Disponível em: <https://www.codeschool.com>. Acessado em novembro 2017.
- Pereira, D. M. C. Programando em WordPress. 1ª Edição. Editora Novatec. 2015.
- Santos, L. M. A., Flores, M. L. P., Tarouco, L. M. R. (2007) Objetos de aprendizagem: Teoria apoiada por computador. Novas Tecnologias na Educação, CINTED-UFRGS, Porto Alegre, V. 6, Nº 2.
- Schneider, E. I.; Medeiros, L.F.; Urbanetz, S.T. (2011) A Interatividade no Processo Educativo Via Rotas de Aprendizagem em EaD. Intersaberes (Facinter), v. 11.
- Secretaria Geral de Educação a Distância da Universidade Federal de São Carlos (SEAD UFSCAR). “Educação a Distância”. Disponível em <http://www.sead.ufscar.br/ead>. Acessado em novembro 2017.
- Silva, M. S. (2014) JQuery - A Biblioteca do Programador JavaScript. 3ª Edição. Editora Novatec.
- Silveira, M.S. e Carneiro, M.L.F. (2012) Diretrizes para a Avaliação da Usabilidade de Objetos de Aprendizagem. Anais do Simpósio Brasileiro de Informática na Educação, Rio de Janeiro, Brasil. Disponível em: <http://www.br-ie.org/pub/index.php/sbie/article/view/1713/1474>. Acessado em novembro 2017.
- Silveira, M. S. (2015) Qualidade de Uso de Objetos de Aprendizagem. (Apresentação de Trabalho/Conferência ou palestra).
- Sommerville, I. (2007), Engenharia de Software. 8ª edição. Editora Prentice-Hall.
- Tarouco, L.M.R. (2010) Sistemas de gestão de conteúdo para objetos de aprendizagem: características desejáveis e soluções existentes, v. 15, p. PDF-A-69.
- W3C Brasil. CSS - Curso W3C Escritório Brasil. Disponível em: <http://www.w3c.br/pub/Cursos/CursoCSS3/css-web.pdf>. Acessado em novembro 2017.

Apêndice A - Manual de instalação do WordPress, *plugin* e *template*

Instalando o WordPress

- 1) Instalar um servidor *apache* e iniciar É indicado o Xampp;
- 2) Acesse *localhost/phpmyadmin*, e insira o banco de dados “*criaoav1.sql*”;
- 3) Download da versão “*wordpress-4.8.1-pt_BR*”;
 - a) Após o download extraia os arquivos e copie, entre na pasta *htdocs*, crie a pasta “*criaoav1*” e cole os arquivos do WordPress;
 - b) Acesse *localhost/criaoav1*, a instalação do WordPress iniciará, crie seus acessos e utilize o banco já inserido, chamado *criaoav1.sql*;
 - c) Aguarde a instalação para prosseguir.

Caso haja alguma impossibilidade de instalação do WordPress, seguem os dados utilizados para a criação da versão de desenvolvimento:

criaoav1
Endereço: localhost/criaoav1 BD: <i>criaoa_v1</i> Nome de usuário: root Senha: 20criaoa17 ou sem senha Servidor do banco de dados: localhost Prefixo da tabela: wp_
Criando o site
Título: CriaOA v1 Usuário: admin Senha: trocardenivel

Acessando o site

Para visualizar o site: **localhost/criaoav1**

Para administrar: **localhost/criaoav1/wp-admin**

Instalando e ativando o Plugin CriaOA

- 1) Extraia os arquivos, copie e cole eles no endereço *htdocs/criaoav1/wp-content/plugins*;
- 2) Acesse o painel de controle como admin, vá até o menu Plugins e clique no botão ativar, do plugin “CriaOA - Plugin”

Instalando e ativando o Template InterageOA

- 1) Extraia os arquivos e cole a pasta no endereço *htdocs/criaoav1/wp-content/themes*
- 2) Faça o login no sistema como admin, acesse o painel de controle, vá até o menu Aparência > Temas e clique no botão ativar do Tema InterageOA.

Apêndice B – Diagrama de Atividades

Na Figura 12, o Diagrama de Atividades mostra que ambos atores devem realizar o *login* no sistema. Sendo que o ator **Aluno** pode interagir com objetos de aprendizagem, e o **ProfessorConteudista** deve inserir todos os recursos digitais para o recurso didático desejado e o questionário avaliativo ao final. Após ter todos os recursos didáticos inseridos, o ProfessorConteudista pode gerar a página do objeto de aprendizagem e publicá-la.

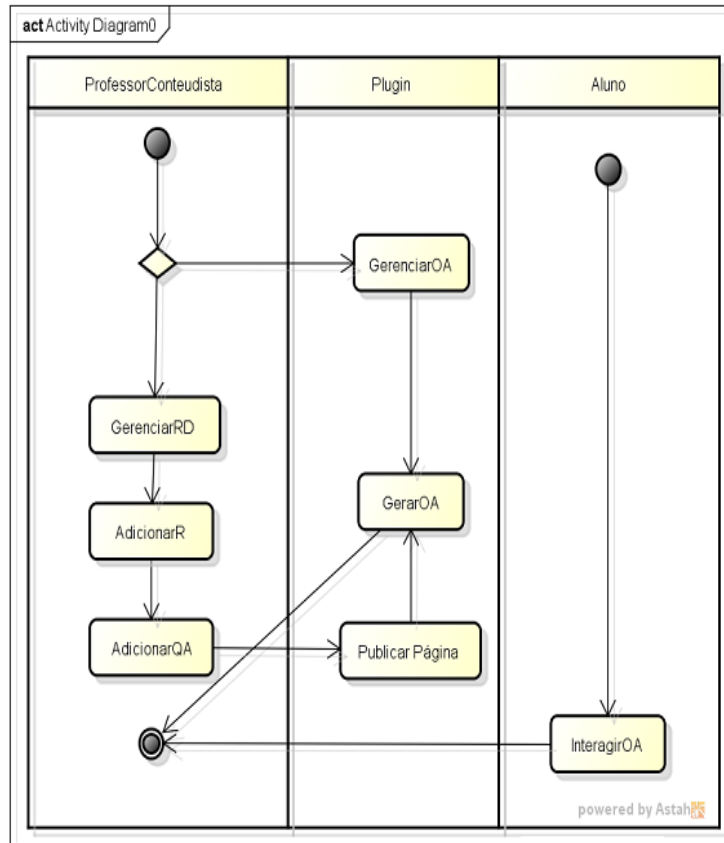


Figura 12. Diagrama de Atividades

Durante o uso dos objetos de aprendizagem (OA), na atividade InteragirOA, cada recurso didático (RD) é seguido pela aplicação de um questionário avaliativo (Q) interativo para que, ao final do recurso didático, seja verificado o conhecimento adquirido pelo aluno. O resultado do questionário determinará qual será o próximo recurso didático a ser disponibilizado pelo sistema: caso o aluno responda corretamente a avaliação será direcionado para o recurso didático que prossegue na apresentação do conteúdo; caso não atenda, será redirecionado para um recurso didático que complementa as informações apresentadas, conforme ilustrado na Figura 13.

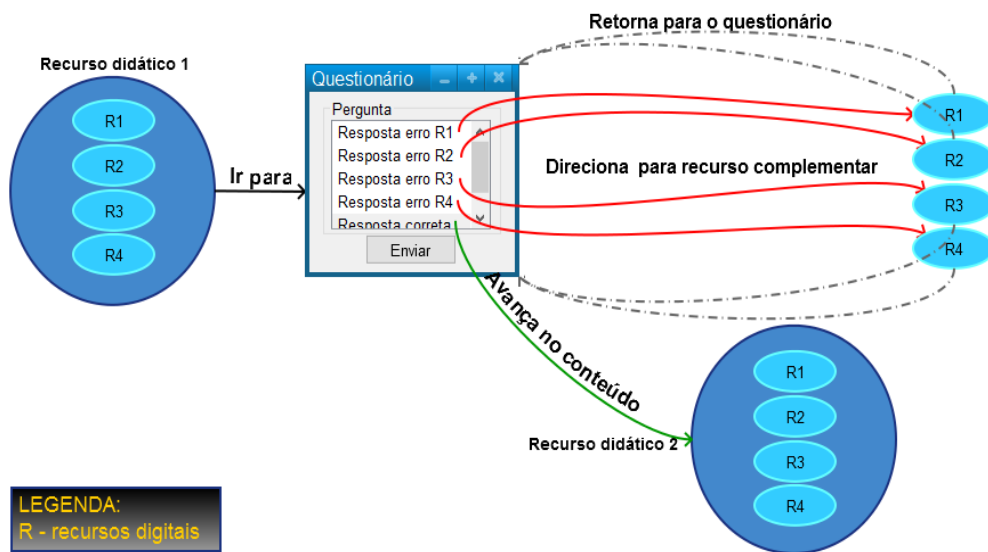


Figura 13. Interatividade do aluno com o objeto de aprendizagem através do *template (front-end)*

7. Anexos

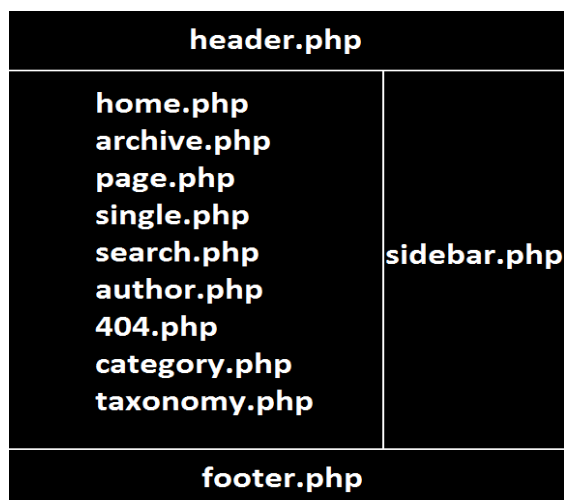


Figura 1. Estrutura de *template* do WordPress separada por arquivos, Pereira (2015)

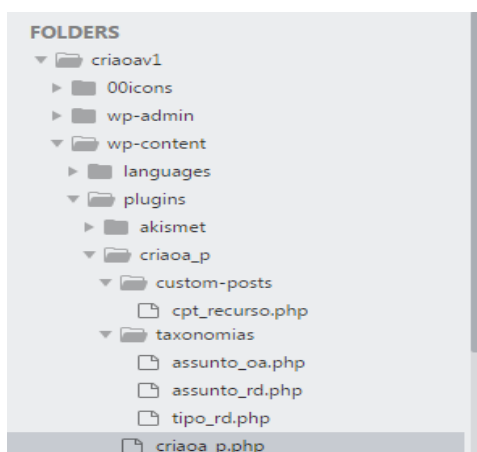


Figura 2. Estrutura de pastas e organização dos arquivos do *plugin* desenvolvido

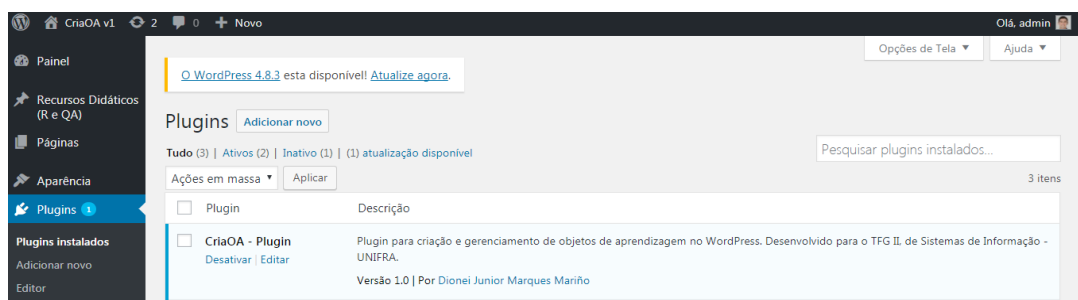


Figura 3. *Plugin* desenvolvido instalado no CMS WordPress (*back-end*)