

Reengenharia da Ferramenta de Criação do Modelo de Domínio para Cursos Adaptativos do Ambiente GRAPPLE

Gustavo C. Publio¹, Edson P. Pimentel¹, Maria G. B. Marietto¹, Ismar F. Silveira²

¹Centro de Matemática, Cognição e Computação - Universidade Federal do ABC (UFABC)

Rua Santa Adélia, 166, Bairro Bangu, Santo André - SP - Brasil. CEP 09210-170.

²Faculdade de Computação e Informática - Universidade Presbiteriana Mackenzie
Rua da Consolação, 930, Bairro Consolação, São Paulo - SP - Brasil. CEP 01302-907.

{gustavo.publio,edson.pimentel,maria.marietto}@ufabc.edu.br,
ismar.silveira@mackenzie.br

Abstract. *The creation of courses that adapts themselves to learners' profiles requires the use of tools for modelling contents to be adapted, i.e., the domain model. In the context of this work it has been analyzed the GAT (GRAPPLE Authoring Tool) from the GRAPPLE framework and has been identified interface problems that makes the use of the tool harder by common educators and also performance problems to model many-concept domains. This article aims to introduce the NGAT prototype developed in the process of Reengineer of the Tool of domain model's creation for adaptive courses of GRAPPLE environment, considering performance and compatibility requirements.*

Resumo. *A criação de cursos que se adaptem ao perfil dos aprendizes requer o uso de ferramentas para a modelagem dos conteúdos a serem adaptados, ou seja, o modelo do domínio. No contexto desse trabalho analisou-se a ferramenta GAT (GRAPPLE Authoring Tool) do framework GRAPPLE e identificou-se problemas de interface que dificultam o uso da ferramenta por educadores comuns e também problemas de desempenho para modelagem de domínios com muitos conceitos. Este artigo tem por objetivo apresentar o protótipo NGAT desenvolvido no processo de Reengenharia da Ferramenta de criação do modelo de domínio para cursos adaptativos do ambiente GRAPPLE, considerando requisitos de desempenho e compatibilidade.*

1. Introdução

A personalização da educação tem sido pensada como uma forma de enfrentamento de alguns desafios presentes no aprendizado online, como os descritos por Huddleston (2008): taxas de evasão superiores àquelas apresentadas em cursos tradicionais, gerenciamento das características de aprendizado dos alunos, autoria de conteúdos e reutilização destes conteúdos, entre outros.

Visando superar tais desafios, o uso de técnicas de Hipermídia Adaptativa combinadas com Sistemas de Gerenciamento da Aprendizagem (SGA) tem fomentado a criação de ambientes de aprendizado adaptativos, enriquecendo a experiência da relação ensino-aprendizado, tanto pelos aprendizes quanto por seus professores/tutores.

Diversos Sistemas de Gerenciamento da Aprendizagem Adaptativos (SGAA) podem ser citados, como o Interbook [Brusilovsky 1998], o AHA! [De Bra 1998], o MOT - My Online Teacher [Cristea 2003] e mais recentemente o projeto GRAPPLE (*Generic Responsive Adaptive Personalised Learning Environment*) [Grapple 2010], desenvolvido por De Bra (2008).

O projeto GRAPPLE é uma abreviação para “Ambiente de Aprendizagem Genérico, Personalizado e Responsivo” (*Generic Responsive Adaptive Personalised Learning Environment*) e consiste numa plataforma Web de hipermídia adaptativa que se integra com os principais SGAs disponíveis no mercado, como Moodle (2002) e Sakai (2005), utilizando uma abordagem orientada a serviços. Dentre suas principais ferramentas, destaca-se aquela responsável pela criação de conteúdos de aprendizado adaptativos, denominada GAT (*Grapple Authoring Tool*). Uma das principais vantagens desta ferramenta GAT em específico é que a mesma consiste em um ambiente visual, voltado para facilitar a criação de conteúdo por parte dos autores. Entretanto, conforme evidenciado no trabalho de Ramos (2013) e também por Gomes (2013), a experiência de profissionais da educação torna-se limitada, devido à necessidade de conhecimento prévio de linguagens de programação, principalmente em termos de desempenho e baixa usabilidade. Esta experiência limitada ocorre principalmente quando há a tentativa de desenvolvimento visual de conteúdos de hipermídia adaptativos para cursos que envolvam dezenas de conceitos de aprendizagem ou mais, ocasionando problemas de desempenho (lentidão) da ferramenta.

Este artigo tem por objetivo apresentar o protótipo NGAT (*New Grapple Authoring Tool*) desenvolvido no processo de Reengenharia da Ferramenta de criação do modelo de domínio para cursos adaptativos do ambiente GRAPPLE. Dois requisitos principais foram definidos para a reengenharia: melhoria do desempenho para grafos com muitos conceitos e compatibilidade com dispositivos móveis e diversos navegadores.

O artigo está organizado como segue: o capítulo 2 introduz o processo de autoria de cursos adaptativos no contexto do ambiente GRAPPLE; o capítulo 3 apresenta o protótipo da ferramenta NGAT, desenvolvido baseado na GAT do GRAPPLE para dar suporte à criação de modelos de domínio. Por fim, o capítulo 4 apresenta as considerações finais.

2. Autoria de Cursos Adaptativos no ambiente GRAPPLE

Escolheu-se o ambiente GRAPPLE para os estudos sobre autoria de cursos adaptativos por este ser compatível com o núcleo do Sakai que também é o núcleo do ambiente Tidia, em uso pela instituição Universidade Federal do ABC (UFABC) em que este projeto está sendo conduzido.

Um dos componentes do projeto GRAPPLE é a GAT, ferramenta de autoria desenvolvida a fim de fornecer interfaces simples e úteis para a criação ou importação de conteúdos, de maneira a garantir uma ampla adoção desse ambiente. A ferramenta (GAT) gera arquivos de saída que contém códigos CAM ou GAM (*Conceptual Adaptation Model* ou *Generic Apadtation Model*), pois a ferramenta se utiliza de uma versão parametrizada dessa linguagem. Trata-se de um formato puramente textual que

também pode ser usado como uma base para o desenvolvimento de ferramentas externas de autoria ou conversores.

A ferramenta GAT possui um mecanismo para a criação de um modelo de domínio, como pode ser visualizado na Figura 1. Tal mecanismo contém conceitos, relações entre conceitos e expressam a estrutura do domínio da aplicação. Apesar de ser uma ferramenta de autoria de conteúdo adaptativo multiuso bastante completa, a GAT apresenta limitações, encontradas a partir do contato direto e uso da ferramenta para a criação de um curso adaptativo completo. Apesar de não existirem referências diretas a estas limitações na literatura, existem evidências que corroboram esta afirmação.

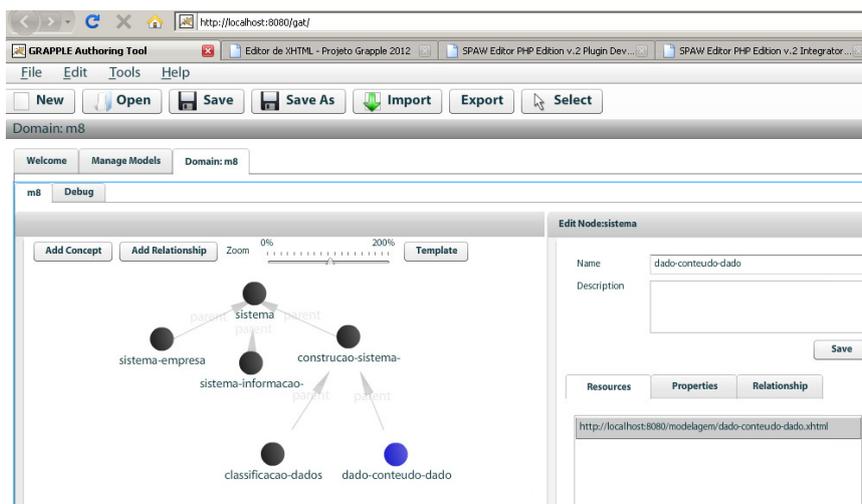


Figura 1. Interface da ferramenta GAT para a criação de um modelo de domínio.

No trabalho de Ramos (2013), foram realizados testes de autoria de aplicações adaptativas utilizando a ferramenta GAT, em que foi solicitado que um grupo de alunos desenvolvesse aplicações utilizando as ferramentas do GRAPPLE, incluindo a GAT. A seguir, trechos que evidenciam estas limitações, traduzidos livremente de Ramos (2013), p. 94.

“(...) Em 2011 a maioria dos grupos usou a ferramenta gráfica de autoria do GRAPPLE para definição da adaptação. Em 2012 a maioria dos grupos usou a linguagem GAM diretamente, depois de receberem *feedback* dos estudantes de 2011 sobre as diferenças nas abordagens de autoria. (...) Contrariamente do que se pudesse esperar, estudantes preferiram utilizar a linguagem GAM em detrimento à ferramenta de autoria. Isto pode ser explicado pelo tipo de estudantes (ciências da computação), os quais valorizaram o total controle ao invés da interface de autoria”.

Da mesma forma que os alunos que foram avaliados no trabalho de Ramos (2013), nos experimentos realizados por Gomes (2013) também ficaram evidenciadas algumas limitações da GAT, pois o autor decidiu editar diretamente o código CAM, sem utilizar a interface gráfica da GAT. No âmbito do processo de produção de conteúdos adaptativos, Gomes relata que a ferramenta GAT, apesar de poder ser considerada de fácil utilização por autores sem muito conhecimento técnico, possui limitações operacionais de interface e desempenho para a criação de cursos completos e funcionais,

mesmo em cursos não muito complexos, mas que exijam um número maior do que 20 nós de conceitos.

Para endereçar tais evidentes limitações encontradas na ferramenta GAT, uma nova ferramenta foi desenvolvida através do processo de Reengenharia do software da GAT. O protótipo desta nova ferramenta será apresentado no capítulo a seguir.

3. NGAT – Suporte a Criação de Modelos de Domínio

A NGAT foi concebida com as mesmas funcionalidades da GAT do Projeto GRAPPLE, em termos de construção do Modelo de Domínio, descrita nos trabalhos de Ploum (2009) e Hendrix (2010). No entanto, os seguintes requisitos principais foram estabelecidos na reengenharia da ferramenta GAT:

- Desempenho adequado e interface amigável para grafos com muitos nós (R1);
- Compatibilidade com diversos dispositivos, incluindo os dispositivos móveis *touch screens* (R2);

A seguir, detalha-se o protótipo da ferramenta NGAT, destacando-se a tecnologia utilizada, sua interface, o escopo deste protótipo, suas saídas e testes realizados.

3.1. Sobre Tecnologia Utilizada

A ferramenta GAT original foi desenvolvida utilizando a tecnologia *Java Flex*, que baseia-se numa mistura da linguagem *Java* com a linguagem *Flash/ActionScript*, proprietária e não compatível, por exemplo, com dispositivos móveis. A base utilizada para o desenvolvimento da NGAT é o padrão HTML (*HyperText Markup Language*) em sua quinta versão (HTML5) [W3C 2014]. Além deste padrão, utilizou-se também a tecnologia CSS (*Cascading Style Sheets*) na versão três (CSS3) [Lie 2005] e a linguagem *JavaScript* [Mozilla 2014], que por sua vez foi desenvolvida baseada no padrão ECMA-262 da linguagem *ECMAScript* [ECMAScript 2011].

A compatibilidade entre dispositivos é um grande avanço proporcionado pelo uso de novos padrões (HTML5, CSS3) que oferecem suporte a dispositivos de interface tradicionais (computadores com mouse/teclado) e interfaces móveis digitalizáveis (*touch screens*).

Cabe destacar também o uso de algumas bibliotecas baseadas na linguagem JavaScript: (a) jQuery e sua variação jQueryUI (*Query User Interface*) [jQuery 2014] que foram utilizadas para facilitar a criação da interface gráfica da aplicação; (b) cytoscape.js [Cytoscape 2014] criada para fornecer subsídios para a criação e análise de grafos. Por ser muito utilizada para representação de dados biológicos, com grafos na ordem de milhares de nós e relacionamentos, esta biblioteca possui um ótimo desempenho, sobretudo em grafos densos, isto é, com maior quantidade de objetos.

3.2. Sobre o Escopo do Protótipo NGAT

A GAT do projeto GRAPPLE abrange três componentes, cada qual responsável por uma etapa do processo de autoria de conteúdo: (i) Modelo de Domínio – DM (*Domain Model*); (ii) Modelo de Tipos de Relacionamentos de Conceitos - CRT (*Concept Relationship Types*) e (iii) Modelo do Usuário – UM (*User Model*).

O protótipo da ferramenta NGAT, na versão descrita nesse trabalho, possui apenas o editor de Modelo de Domínio. Dessa forma, a saída do código CAM da ferramenta será parcial, contendo apenas os dados referentes ao Modelo de Domínio, sem os dados dos CRTs e do Modelo de Usuário. Essas funcionalidades estão previstas para a continuidade do trabalho.

3.3. Interface da Ferramenta NGAT

A primeira caixa de diálogo a ser preenchida pelo usuário é de criação do modelo do domínio conforme mostra a Figura 2. Nela o usuário irá informar o título e a descrição do modelo a ser criado.

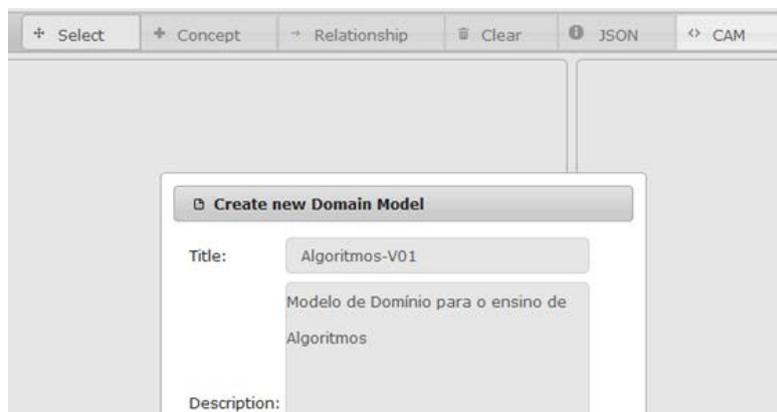


Figura 2. Caixa de Diálogo Inicial: Criação do Modelo de Domínio.

Após a inserção das informações iniciais, a interface principal da NGAT é exibida conforme mostra a Figura 3 enumerando-se suas principais áreas: a barra de ferramentas (1); a área de trabalho (2) e o painel de edição (3). Essas áreas são detalhadas a seguir.

Barra de Ferramentas (1)

A NGAT dispõe de uma barra de ferramentas, conforme o destaque “1” da Figura 3 na qual o usuário pode selecionar, através de botões, as seguintes funcionalidades:

- **SELECT:** selecionar conceitos ou relacionamentos no mapa, o que possibilitará a edição de seus respectivos conteúdos.
- **CONCEPT:** monitorar a área de trabalho para adicionar um novo nó no centro do exato local de cada *click* do usuário
- **RELATIONSHIP:** adicionar novos relacionamentos ligando um conceito a outro.
- **CLEAR:** apagar todo o conteúdo da área de trabalho após confirmação da exclusão dos dados.
- **JSON:** visualizar e exportar o código do Mapa de Domínio criado no formato JSON (*JavaScript Object Notation*).
- **CAM:** visualizar e exportar o código do Mapa de Domínio criado no formato CAM (*Conceptual Adaptive Model*), que consiste num arquivo XML (*eXtensible Markup Language*) estruturado no padrão IMS-VDEX [IMS 2004] - estrutura de dados aceita pelo compilador do GALE.

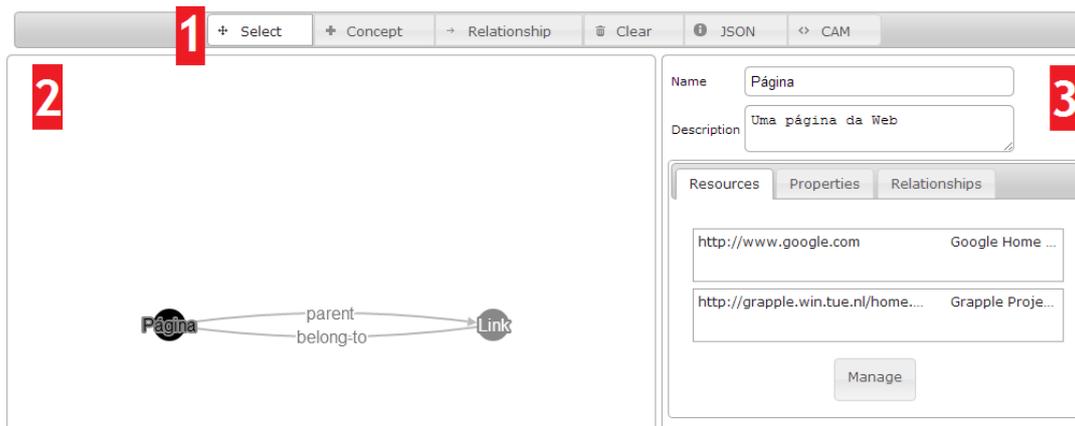


Figura 3. Caixa de Diálogo Inicial da NGAT: Criação do Modelo de Domínio.

Área de Trabalho (2)

A área na qual o usuário da ferramenta desenha seu Modelo de Domínio é a chamada área de trabalho da NGAT, apresentada na Figura 4. Nela o usuário pode adicionar os conceitos e/ou relacionamentos, ou ainda selecioná-los para editar os dados do objeto por meio do painel de edição. Esta área foi criada com o auxílio da biblioteca *cytoscape.js*, que fornece classes para criação e manipulação de grafos, sejam eles direcionados ou não.

Na GAT do Projeto GRAPPLE a criação de relacionamentos entre conceitos ocorre por meio da caixa de diálogo o que torna o processo menos direto, pois aumenta o diálogo com o usuário. A NGAT permite que o usuário desenhe novos relacionamentos apenas selecionando o botão *Relationship* na barra de ferramentas e clicando e arrastando o relacionamento a partir de um ponto específico no nó de origem até o nó de destino. A figura 5 ilustra a criação de relacionamentos na NGAT.

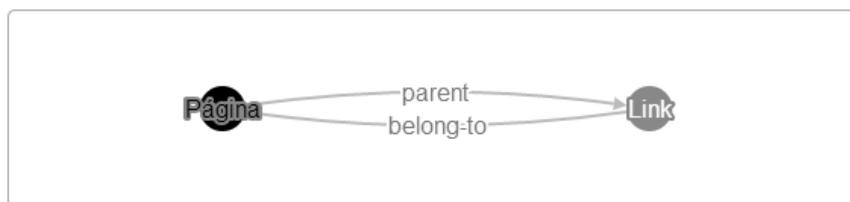


Figura 4. Área de Trabalho da NGAT

Para a visualização do mapa completo ou de uma área específica o usuário pode aproximar (*zoom in*) ou distanciar (*zoom out*) o diagrama criado, através do movimento da roda do *mouse* (no computador) ou abrindo/fechando dois dedos, num movimento de pinça (dispositivos móveis). Na GAT original o controle do nível de *zoom* somente pode ser feito por meio da barra de seleção, tornando a ação menos amigável.

Outro recurso disponível na interface da área de trabalho é o menu de contexto, fornecido pelo *plugin* da classe *cxtMenu*. Ele é acionado clicando com o botão direito no objeto, seja ele um conceito ou um relacionamento, ou tocando com a ponta de dois dedos, no caso de dispositivos com *touchscreen*. O menu de contexto referente a um conceito contém as opções *Connect* e *Delete*, que permitem criar ou apagar um novo relacionamento. Caso o objeto alvo do menu de contexto seja um relacionamento, as

opções *Invert* e *Delete* são mostradas, permitindo ou inverter o sentido de um relacionamento ou apagá-lo. A Figura 6 ilustra o efeito do menu de contexto para os dois tipos de objetos.

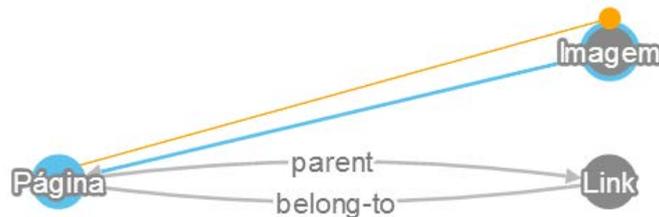


Figura 5. Criação de Relacionamentos na NGAT.

Painel de Edição (3)

O painel de edição é uma barra lateral, paralela à área de trabalho, na qual são exibidas opções de edição dos dados, que variam de acordo com o tipo do objeto selecionado: conceito (nó) ou relacionamento (aresta).

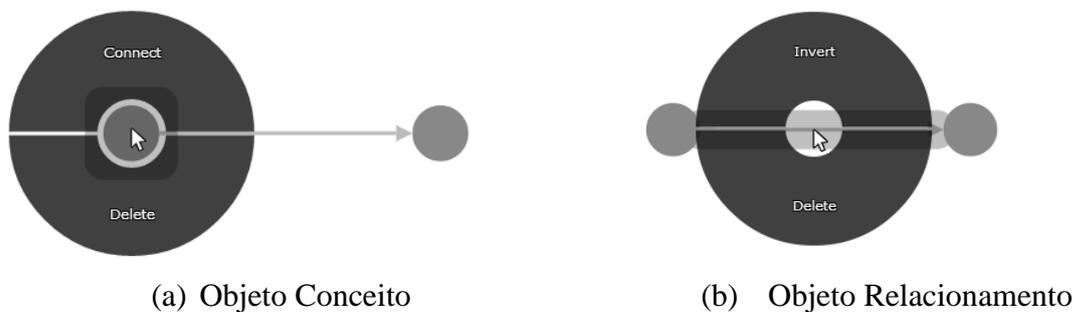


Figura 6. Menu de Contexto.

A Figura 7 ilustra a edição de um conceito ou nó. Há também três abas, através das quais é possível visualizar e gerenciar os Recursos (*Resources*), Propriedades (*Properties*) ou Relacionamentos (*Relationships*) atrelados ao nó selecionado.



Figura 7. Painel de Edição de um Conceito (nó).

Em termos de interface, este painel de edição dos dados de um nó selecionado se assemelha àquele disponível na ferramenta GAT original. Entretanto, caso o objeto selecionado seja um relacionamento (aresta no grafo), a NGAT apresenta neste painel outra interface, para edição dos dados do relacionamento selecionado, conforme ilustra a

Figura 8, adaptando-se, portanto ao objeto específico, tornando-se mais amigável para o usuário.

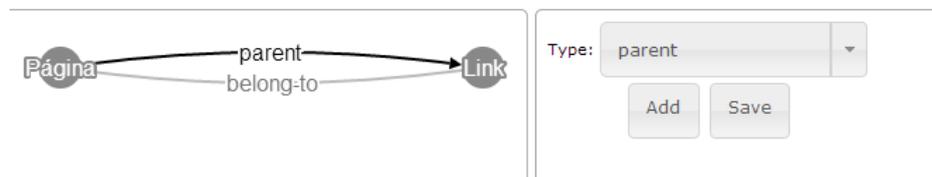


Figura 8. Painel de Edição de um Relacionamento.

3.4. Exportação de Códigos da NGAT

A NGAT permite a exportação do código do Modelo de Domínio em dois formatos: JSON e XML, a partir dos botões presente na barra de ferramentas. O XML é para atender ao formato esperado (código *.cam*) pelo compilador do GALE, baseado no padrão IMS VDEX [IMS 2004].

O formato JSON é utilizado pela biblioteca *cytoscape.js* para armazenar os dados do grafo construído. Este vem se tornando o principal formato para transmissão de dados utilizado no desenvolvimento de aplicações em detrimento ao XML, principalmente naquelas que envolvem a comunicação entre servidores remotos (como banco de dados) com aplicações em dispositivos móveis. Esta migração de formatos tem ocorrido devido ao melhor desempenho do JSON em comparação ao XML, conforme observado por Nurseitov (2009). A redução passa a ser considerável quando se trata de conexões de dados instáveis e sujeitas a um maior índice de interferências e ruídos, como é o caso da rede celular que atende os dispositivos móveis [Wang 2011][Gao 2011].

3.5. Testes e Resultados

A fim de verificar o atendimento aos dois principais requisitos estabelecidos na reengenharia da ferramenta GAT por meio do desenvolvimento do protótipo NGAT, realizaram-se-se dois teste.

O primeiro teste, do requisito R1, sobre o desempenho adequado e interface amigável para grafos com muitos nós foi realizado repetindo o experimento de Gomes (2013). A NGAT apresentou desempenho satisfatório, isto é, sem lentidão ou travamentos no processo de criação, mesmo quando o modelo (grafo) envolvia diversos conceitos (nós). Além disso, mostrou-se amigável no manuseio e reposicionamento dos conceitos.

O segundo teste, do requisito R2, sobre compatibilidade com diversos dispositivos, incluindo os dispositivos móveis *touch screens* foi realizado, experimentando-se a NGAT em dispositivos móveis (*tablets* e *smartphones*) e também em diversos navegadores e sistemas operacionais. A NGAT comportou-se adequadamente nos sistemas operacionais *Windows* (Versões XP, 7, 8 e 8.1), *Linux* (distribuições Ubuntu e Mint) e *Mac OS* (Versão X), nos navegadores Google Chrome, Mozilla FireFox, Safari da Apple e Microsoft Internet Explorer (versões 8, 9 e 10).

4. Considerações Finais e Trabalhos Futuros

A personalização da educação ao perfil do estudante (ou grupo) tem sido uma das principais vantagens apontadas para a disseminação de recursos de Tecnologia Digital da Informação e Comunicação (TDIC) mediando o processo de ensino-aprendizagem. No entanto, a produção de materiais instrucionais adaptativos tem sido feita em baixa escala, especialmente pela dificuldade de autoria deste tipo de conteúdo adaptativo por parte de professores sem conhecimento em linguagens de programação.

Considerando este contexto, este artigo investigou no âmbito do Projeto GRAPPLE, o processo autoria de conteúdos adaptativos por professores não especialistas em ferramentas computacionais e sua utilização em ambientes populares de E-learning. Foram evidenciadas limitações da ferramenta de autoria do ambiente (GAT), e como forma de apresentar soluções para estas limitações, foi apresentado um protótipo de uma nova ferramenta (NGAT), desenvolvido através da reengenharia da ferramenta GAT, o qual mostrou através de testes desempenho eficiente, mesmo na construção de cursos com dezenas de conceitos, além de uma interface mais amigável, com recursos de auxílio à produção de modelos de domínio indisponíveis na ferramenta GAT.

Como sugestões de trabalhos futuros, a conclusão do protótipo, construindo além do editor de modelo de domínios, também o editor de regras e CAM pode ser considerado o próximo passo desta pesquisa; além disso, o uso da linguagem HTML5 ainda pode proporcionar o uso concorrente da ferramenta por múltiplos usuários, propiciando uma edição colaborativa do curso a ser criado.

Referências Bibliográficas

- Brusilovsky, P.; Eklund, J.; Schwarz, E. (1998) “Web-based education for all: a tool for development adaptive courseware”. *Computer Networks and ISDN Systems*, Elsevier, v. 30, n. 1, p. 291–300.
- Brusilovsky, P. (2003) “Developing Adaptive Educational Hypermedia Systems: from Design Models to Authoring Tools”. Pittsburgh, EUA: [s.n.]. School of Information of Science, University of Pittsburgh.
- Cristea, A. (2003) “Adaptive patterns in authoring of educational adaptive hypermedia”. *Educational Technology and Society*, v. 6, p. 1–5.
- Cristea, A. (2005) “Authoring of adaptive hypermedia”. *Educational Technology & Society*, v. 8, p. 6–8.
- Cytoscape, C. (2014) “Cytoscape.js”. Acesso em 18 de março de 2014. Disponível em: <<http://cytoscape.github.io/cytoscape.js/>>.
- De Bra, P.; Calvi, L. (1998) “Aha: a generic adaptive hypermedia system.” In: *Proceedings of the 2nd Workshop on Adaptive Hypertext and Hypermedia*. [S.l.: s.n.].
- De Bra, P.; Pechenizkiy, M.; Sluijs, K. V. D.; Smits, D. (2008) “Grapple: Integrating adaptive learning into learning management systems. Proceedings of the World”. *Conference on Educational Multimedia, Hypermedia & Telecommunications*, Viena, Austria, v. 20, p. pp. 5183–5188.

- ECMAScript, E.; A, E. C. M. et al. (2011) “ECMAScript”. Acesso em 18 de março de 2014. Disponível em: <<http://www.ecma-international.org/publications/files/ECMA-ST/Ecma-262.pdf>>.
- Gao, J.; Duan, H. (2011) “Research on data transmission efficiency of json”. CNKI Journal of Electronic Technology & Information Science, Computer Engineering and Design, v. 7, p. 2267–2270.
- Gomes, E. H. (2013) “Personalização do E-Learning Baseado no Nível de Aquisição de Conhecimento do Aprendiz”. Dissertação (Mestrado) — Universidade Federal do ABC.
- GRAPPLE (2010). “Generic Responsive Adaptive Personalized Learning Environment Project”. Acesso em 18 de março de 2014. Disponível em: <<http://grapple.win.tue.nl>>.
- Hendrix, M. (2010) “Supporting authoring of adaptive hypermedia”. Tese (Doutorado) — University of Warwick, Reino Unido.
- Huddleston, J.; Pike, J. (2008) “Seven key decision factors for selecting e-learning”. Cogn Tech Work, p. 237–247. Springer-Verlag London Limited, Reino Unido.
- IMS. (2004) “IMS VDEX”. Acesso em 18 de março de 2014. Disponível em: <<http://www.imsglobal.org/vdex/>>.
- jQuery, F (2014) “jQuery: write less, do more.” Acesso em 18 de março de 2014. Disponível em: <<http://jquery.com/>>.
- Lie, H. W. (2005) “Cascading Style Sheets”. Tese (Doutorado) — University of Oslo.
- Moodle. (2002) “Moodle.org: open-source community-based tools for learning”. Acesso em 18 de março de 2014. Disponível em: <www.moodle.org>.
- Mozilla. (2014) “JavaScript Specification”. Acesso em 18 de março de 2014. Disponível em: <<https://developer.mozilla.org/en-US/docs/Web/JavaScript>>.
- Nurseitov, N.; Paulson, M.; Reynolds, R.; Izurieta, C. (2009) “Comparison of json and xml data interchange formats: A case study”. Caine, v. 9, p. 157–162, 2009.
- Ploum, E. (2009) “Authoring of adaptation in the GRAPPLE project”. Tese (Doutorado) — Technische Universiteit Eindhoven, Eindhoven, Países Baixos.
- Ramos, V. F. C. (2013) “Adaptive Hypermedia Courses: Qualitative and Quantitative Evaluation and Tool Support”. Tese (Doutorado) — Universidade Federal do Rio de Janeiro.
- Sakai. (2005) “Sakai Project | collaboration and learning: for educators by educators”. . Acesso em 18 de março de 2014. Disponível em: <<http://www.sakaiproject.org>>.
- W3C (2014). “A vocabulary and associated APIs for HTML and XHTML”. Acesso em 18 de março de 2014. Disponível em: <<http://www.w3.org/html/wg/drafts/html/CR/>>.
- Wang, G. (2011) “Improving data transmission in web applications via the translation between XML and JSON”. In: Communications and Mobile Computing (CMC), 2011, Third International Conference on. [S.l.: s.n.]. p. 182–185.