

Um modelo de design instrucional para auxiliar a aprendizagem de conceitos computacionais no curso de Licenciatura em Ciência da Computação

Rio Tinto – PB – Maio 2012

Adelito Borba Farias – Universidade Federal da Paraíba (UFPB) –
adelito.farias@dce.ufpb.br

Categoria: C - Métodos e Tecnologias

Setor Educacional: 3 - Educação Universitária

Classificação das Áreas de Pesquisa em EaD:

Macro: E- Métodos de Pesquisa em EAD e Transferência de Conhecimento / Meso: K- Serviços de Apoio ao Estudante / Micro: M- Design Instrucional

Natureza: B- Relatório de Pesquisa

Classe: 1- Investigação Científica

Resumo

O processo de aprendizagem não pode ficar limitado a práticas pedagógicas tradicionais, tendo em vista a contínua e crescente influência das tecnologias digitais no comportamento das gerações presentes. O uso de softwares educacionais passa a ser um importante fator no aprendizado, ao permitir abordagens colaborativas na construção do conhecimento. Em frente a vários relatos sobre a dificuldade de compreensão do conteúdo de programação - POO em particular, surgiu a ideia de construir um material instrucional. Através de uma pesquisa com alunos do curso de graduação em Licenciatura em Ciência da Computação, esse trabalho busca refletir acerca da utilização de ferramentas, priorizando a modalidade EAD, no auxílio do ensino de programação, no qual o conceito escolhido para estudo foi o de polimorfismo. Para tanto, apresenta uma proposta para construir um ambiente onde os alunos possam aprender e aplicar o conceito de em um ambiente colaborativo.

Palavras Chave: Ambiente de Aprendizagem EAD, Ensino de Programação, Tecnologias na Educação.

1- Introdução

Os cursos de Computação são constituídos de conhecimentos diversos interligados que buscam desenvolver diferentes competências e habilidades necessárias para exercício da profissão [10]. Dentre as várias áreas, os graduandos em Licenciatura em Ciência da Computação, vivenciam duas linhas fundamentais para a formação profissional: ciência da computação e educação. O curso de Licenciatura em Ciência da Computação tem como propósito formar educadores para o ensino de computação em escolas de ensino fundamental, médio e técnico.

Dentro do fluxograma dos cursos de Computação, a Programação Orientada a Objeto, POO, é uma área que desenvolve nos estudantes habilidades específicas para resolver situações ou problemas de natureza complexa encontradas no mundo real. A abordagem orientada a objetos tem se mostrado mais adequada, comparativamente às demais, para ser empregada no desenvolvimento de sistemas de softwares complexos e de grande porte. Neste sentido, POO possui fundamental papel na preparação dos estudantes da área de Computação [5].

Ocorre que há uma enorme dificuldade de absorção dos conteúdos, o provoca desmotivação nos estudantes, fazendo-os abandonar e/ou migrar de curso [9]. Esta dificuldade é comprovada pelo grande índice de reprovação da disciplina de programação.

Visando mensurar as principais dificuldades dos alunos no aprendizado de POO dos cursos de Licenciatura em Ciência da Computação da UFPB, foi desenvolvida uma pesquisa com os estudantes do referido curso. Um dos motivos apontados como complicadores, foram as técnicas algorítmicas que, exploram o raciocínio lógico na construção de sistemas computacionais. Analisando os dados, uma das questões que apontou índice elevado na compreensão do conteúdo, foi relativo ao uso do conceito de Polimorfismo.

Segundo Ferreira [3], polimorfismo pode ser definida como a característica que autoriza diferentes significados serem associados a algo em diferentes contexto. Pondo na perspectiva OO, é um conceito que trata da possibilidade de uma função ser habilitada para operar em uma variedade de tipos de dados, no qual, o efeito de polimorfismo está ligado aos conceitos de tipo abstrato de dados e de herança [14].

Cardelli e Wegner [2] classificam o polimorfismo em linguagens orientadas a objetos de acordo com a Figura 1 a seguir:

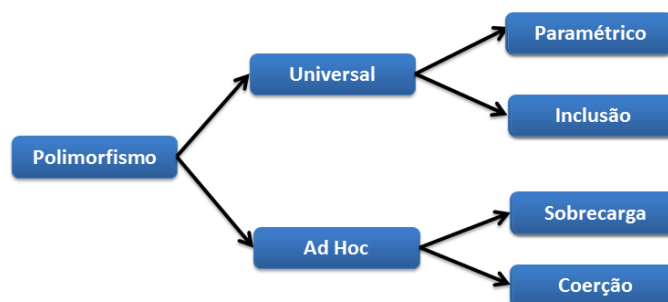


Figura 1. Classificação de Cardelli e Wegner para polimorfismo.

Para o polimorfismo universal a mesma função pode ser executado para um conjunto possivelmente ilimitado de tipos de dados diferentes. Já no polimorfismo "ad hoc" uma mesma função poderá apenas ser executada para um conjunto restrito de tipos de dados. O polimorfismo universal está subdividido em: Paramétrico - no qual, a partir de uma única definição de uma função ela pode trabalhar uniformemente. Também conhecido como genericity; e Inclusão - consiste em permitir que um objeto possa "ser visto" como um objeto de diversas classes, fornecendo comportamentos diferentes de acordo com o objeto real em tempo de execução. E o polimorfismo ad hoc é composto por: Coerção – é um cast, no qual a operação converte um argumento de um determinado tipo para outro que está sendo esperado; e Sobrecarga - consiste em prover diferentes implementações para funções com o mesmo nome, no qual os tipos dos argumentos na chamada são utilizados para escolha do método a ser executado.

Este artigo descreve a proposta de um ambiente para a aprendizagem e prática do conceito de Polimorfismo. A proposta parte da perspectiva de proporcionar ao usuário a prática dos conceitos através da resolução de exercícios mostrando a aplicabilidade do conceito. O material também buscará mostrar aos alunos o quão útil é o conceito para desenvolvimento de códigos flexíveis. Para isso, pretende-se fazer uso de roteiros onde o usuário deve suscitar resoluções em que exibam a utilidade do conceito no desenvolvimento de algoritmos. Essas soluções serão postadas em um fórum, para que seja compartilhada a construção do conhecimento entre os envolvidos. A fim de indicar o nível de assimilação do conteúdo, haverá exercício, em que ao fim da resolução, o usuário receberá uma porcentagem de acerto do questionário.

Esse artigo está estruturado em quatro seções: a primeira trata das iniciativas de melhoria do ensino de programação; a segunda seção apresenta os problemas do ensino de programação; a terceira mostra a metodologia usada para criação da proposta; e por último, considerações finais serão apresentadas.

2. O Ensino de Programação

Na conjuntura da pesquisa realizada, foi encontrado trabalhos que tratam de materiais instrucionais para auxiliar no aprendizado do paradigma OO. Porém, os trabalhos focam outros conceitos que não é o proposto neste artigo, o Polimorfismo. Dentre os variados tipos de ferramentas e ambientes propostos na literatura com a finalidade de facilitar o aprendizado de lógica e linguagens de programação, pode-se destacar que os ambientes buscam minimizar a abstração do conteúdo. A seguir serão apresentados alguns desses trabalhos.

O HabiPro (Habits of Programming) [18] é ambiente colaborativo que visa desenvolver nos estudantes "bons hábitos" em programação, ao estimular os estudantes a adquirirem habilidades como a observação e reflexão sobre a estrutura dos algoritmos necessários para se tornarem bons programadores.

Um protótipo de sistema de ensino e aprendizagem de programação proposto por Kemp [11], apresenta programas de computadores previamente prontos e o estudante deve criar um diagrama correspondente ao programa já

implementado. Neste sistema, os estudantes devem associar as estruturas de programação com a representação correspondente aos diagramas.

O projeto do ambiente ASTRAL [15] visou atender as necessidades de preparação de diversos exercícios de implementação de estruturas de dados utilizando animações gráficas, de modo homogêneo sob uma interface consistente.

A ferramenta JavaTool [13] foi pensada para auxiliar aulas de prática de programação, por meio da animação da execução do código, sejam presenciais e/ou na modalidade a distância. O discente poderá animar os exemplos expostos pelo professor bem como os exemplos existentes em um conteúdo digital de uma disciplina de programação.

Outro trabalho relacionado foi proposto por Iepsen [8], em que é apresentado um sistema que recomenda uma série de exercícios de acordo com o nível de conhecimento do estudante. O Persona-Algo, realiza uma personalização dos conteúdos para ficar proporcional ao estágio no qual está o estudante, desenvolvendo habilidades específicas para a elaboração de algoritmos.

A maioria destas aplicações estão baseadas na exposição e animação de algoritmos, podendo ser utilizadas para tarefas tão diversas, a exemplo: reter a atenção dos alunos durante as apresentações, explicar conceitos de uma forma visual e encorajar um processo de aprendizagem firmado na prática. Este último aspecto parece extremamente relevante, uma vez que os ambientes de programação (IDEs como Eclipse¹, Netbeans²) são direcionados aos profissionais, e a informação que fornecem é normalmente bastante complexa para ser compreendida por estudantes recém introduzidos na esfera do mundo computacional.

O trabalho de Vahldick [16] exhibe que diante da necessidade de um ambiente mais próximo da realidade dos estudantes, a utilização de ambientes mais simples, com interface acessível e interação pode contribuir para estimular os alunos a experimentar, avaliar e corrigir os seus próprios algoritmos, o que os auxilia no processo de aprendizagem de programação.

Uma das colunadas deste artigo, é a pesquisa realizada por um grupo de estudantes do curso de Licenciatura em Ciência da Computação da UFPB – Campus IV que, ao perceberem quão grande era o número de estudantes reprovados e/ou repetentes das disciplinas de POO, decidiram por fazer tal pesquisa. Os alunos que participaram da entrevista, pertenciam aos quarto e quinto período do mesmo curso. A pesquisa aponta que um dos conteúdos considerados como grande obstáculo de aprendizagem e prática foi o Polimorfismo.

Amparados por essa pesquisa e pelo fato dos materiais encontrados não tratam diretamente do polimorfismo, decidiu-se trabalhar na construção de um material instrucional focado no referido conceito, buscando expor quão valioso é para compreensão de reuso de código, também como a sua aplicação em sistemas computacionais.

¹ <http://www.eclipse.org/downloads>

² <http://netbeans.org/downloads>

2.1 Problemas no Ensino de Programação

Para a construção deste projeto, iniciou o processo de compreensão de como ocorre a aprendizagem de algoritmos e, elaboração de material instrucional voltados para este tema. É pertinente para uma nação que pleiteia acender do posto de consumidor tecnológico, e assim, ocupar o cargo de produtor de tecnologia, que seja melhorada a qualidade do processo de ensino/aprendizagem.

Para os cursos da área de Computação, também é válida esta observação, considerando que um dos maiores entraves é o elevado número de reprovação e/ou evasão das disciplinas de algoritmo e programação. Este é um dentre as várias causas que fazem com que a quantidade de alunos iniciantes seja inversamente proporcional à quantidade de estudantes concluintes [10].

A linguagem de programação é o conteúdo predominante nos cursos das áreas correlatas de Computação. O aprendizado de programação é um processo difícil e exigente, e vai além da aquisição de conhecer a sintaxe e a semântica de uma linguagem de programação [6].

Linguagens de programação possuem níveis de complexidade elevado. A abstração também é uma das principais dificuldades dos alunos. Segundo Vahldick [16], a Programação Orientada a Objetos é um tópico essencial na Ciência da Computação, pois é o paradigma mais presente nas linguagens e ambientes de programação na atualidade (C++, Java, C#, Python, Ruby, PHP, Perl).

Esses fatos nos levaram a propor um material instrucional que viesse a auxiliar no ensino de POO. O fato de existirem poucos materiais voltados para o ensino do Polimorfismo e por esse conceito ser importante, é apresentada uma proposta de um ambiente para aprendizagem e prática do mesmo.

3. Metodologia Proposta

Esse trabalho fez uso de técnicas qualitativas de pesquisa e adotou um processo de design instrucional para elaboração da proposta do material aqui tratado.

No processo de design instrucional foi de acordo com o proposto por Filatro [4]. Nosso procedimento foi composto das seguintes etapas: levantamento de requisitos, elaboração do planejamento do material e elaboração de StoryBoard.

O ponto inicial dos trabalhos foi a etapa de levantamento de requisitos. O produto final dessa etapa foi a elaboração de um relatório de análise de necessidades que apresentava a demanda e a caracterização do público-alvo. Baseados nesse relatório, o planejamento do material didático foi tomando corpo, culminando na elaboração de uma matriz instrucional com os seguintes elementos: objetivos de aprendizagem de cada unidade, conteúdo a ser abordado, descrição das atividades, ferramentas utilizadas, duração de cada

unidade e forma de avaliação. Por fim, um Storyboard foi construído com objetivo de obter uma sequência completa de ações, a fim de compreender e testar o material proposto antes da etapa de implementação final.

Foi possível compreender o nível de dificuldade da aprendizagem, após a análise dos dados coletados e da investigação realizada com estudantes que já cursaram a disciplina de POO. Aproximadamente 67% dos entrevistados possuem dificuldade sobre o conceito foco deste trabalho, Polimorfismo, e cerca de 48% prefere estudar por meio de software que auxiliem no processo de aprendizado. Estes dados serviram de estímulo para elaboração deste material instrucional.

Após o estudo dos dados, foi organizado uma matriz instrucional que apresenta os conteúdos de Polimorfismo, todas as atividades que serão realizadas, a forma de avaliação dos usuários para cada tópico, e os instrumentos necessários para realizar tais atividades, bem como a definição das atribuições dos envolvidos no processo: professor e aluno.

Visando compreender como seria o progresso das atividades, como também identificar erros referentes à usabilidade, foi construído um Storyboard com telas do aplicativo, onde é mostrado detalhadamente um tópico relacionado ao Polimorfismo.

4. Resultados

Na intenção de verificar a coerência desse trabalho, os dados obtidos da pesquisa, através do questionário, permitiu conhecer o nível dos estudantes, bem como encontrar estratégias para implementar atividades que melhorem a compreensão da abstração algorítmica, por meio de ambientes que auxiliem os estudantes neste processo. Desta forma, foi realizado um design instrucional que culminou nas fases a seguir.

4.1 Relatório de Análise de Necessidades

As linguagens de programação de computadores são temas recorrentes na formação dos licenciados em computação, segundo Valadares [17], as técnicas de construção de algoritmos e programação constituem o principal núcleo de conhecimento no contexto das disciplinas que compõem os cursos de informática de nível superior. Portanto, aprender a programar é um processo difícil e exigente para a maioria dos alunos. Mesmo a aprendizagem de conceitos básicos e a sua aplicação na resolução de problemas concretos coloca problemas difíceis a muitos estudantes [12].

Partindo desta perspectiva, os requisitos necessários para a criação e desenvolvimento de um material instrucional, busca solucionar os problemas no entendimento envolvendo a disciplina de POO, tendo como temática principal as dificuldades encontradas na compreensão dos conceitos relacionando o uso do Polimorfismo.

Desta forma, após a constatação da dificuldade dos discentes em suas atividades, referentes à aplicação deste conceito em suas soluções

algorítmicas, foi proposto desenvolver um tutorial, visto que grande parcela dos alunos aprovam o uso de livros e apostilas, de acordo com o resultado da pesquisa, para auxiliar no processo de aprendizagem do conceito de Polimorfismo.

4.2 Matriz instrucional

Elaborada para ter uma visão ampla de cada tópico de aprendizagem, para estabelecer as atividades essenciais para atingir as metas, foi obtido 4 unidades. Um que merece destaque, é a unidade que faz aplicação do conceito, intitulada: Aplicação do polimorfismo. A identificação dos casos onde se aplica o conceito é o principal objetivo desta unidade, bem como, resolução de roteiros que mostrem a utilidade no desenvolvimento de softwares.

4.3 Storyboard

A construção do Storyboard serviu para a compreensão de como seria realizada a inserção dos conteúdos montados na matriz instrucional, descrevendo os por menores da estrutura e do fluxo da informação, dos conteúdos e da interface [4]. Assim, neste intuito, é possível identificar possíveis erros de interação e usabilidade, antes da implementação do software.

Utilizou protótipos de média fidelidade, que são de rápida construção, porém, utilizam ferramentas computacionais e possibilitam simular o comportamento de interação da interface. Não requerem dos projetistas conhecimento técnico [1].

Abaixo é exibida duas telas, que, respectivamente, são a Figura 2: tela inicial, onde o usuário escolherá o tópico para estudo; e a Figura 3: tela de um tópico escolhido, no qual o usuário terá acesso aos materiais do tópico.

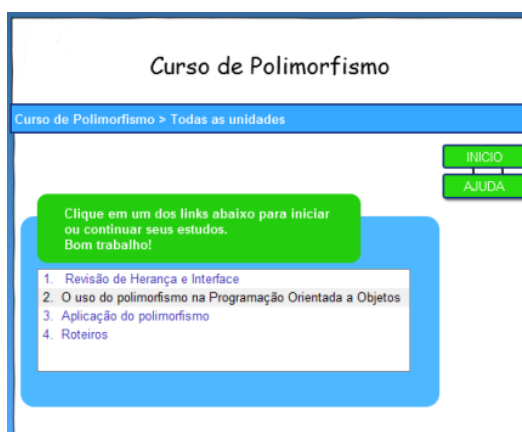


Figura 2

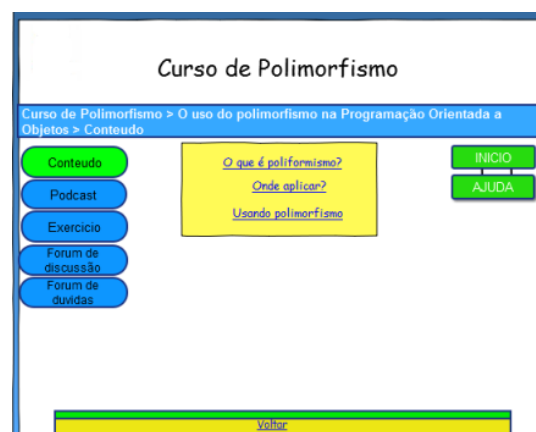


Figura 3

4.4 A Proposta

Diante do cenário apresentando, que está alicerçado nos dados colhidos com os estudantes, surge a oportunidade de construir um material que auxilie tais estudantes do curso de Licenciatura em Ciência da Computação da UFPB – Campus IV, referente ao conteúdo de Polimorfismo, da disciplina de POO.

O material proposto é um tutorial, que aborda o conteúdo de Polimorfismo, e mostre quão útil é o assunto para a vida no ambiente profissional. Nele, constará teorias, exercícios com aplicações reais do conceito, e alguns textos para autoavaliação dos estudantes, no qual, eles poderão verificar o andamento do seu aprendizado.

Ao término, o indivíduo irá criar e aplicar estratégias para a utilização do conteúdo em suas atividades, e assim, usaram o senso crítico para poder identificar a melhor forma de solucionar os problemas com o uso do Polimorfismo.

5.Considerações Finais e Trabalhos Futuros

As reflexões apresentadas neste trabalho expõe a experiência adquirida com um projeto de elaborar um ambiente que busca mostrar o potencial pedagógico das tecnologias digitais em prol de aulas mais motivadora e colaborativas, possibilitando dar um salto de qualidade na educação.

Desta forma, nosso trabalho tem por objetivo aprimorar o aprendizado do conceito de Polimorfismo, dando suporte ao usuário, no caso, estudante, a buscar melhor compreensão do conteúdo por meio de uma metodologia que envolva resolução e compartilhamento de exercícios, focando na aplicabilidade do conceito. Para isto, a fim de ser fixado o conteúdo, o estudante é estimulado a formular resoluções de problemas que farão uso do conceito de Polimorfismo.

Diante da grande quantidade de ambientes educacionais disponíveis e da importância de criteriosas análises técnicas e pedagógicas, Para que o esta proposta cumpra sua missão, pretende-se aprimorar os métodos de avaliação através de equipes multidisciplinares, com participação de profissionais de diferentes áreas do conhecimento [7]. Pretende-se também testar a usabilidade e eficiência do ambiente, a fim de tornar o ambiente acessível para usuários com deficiência visual.

Referencias

[1] Aguiar, Y., Lula, B., Lima, C., Lima, G., Gouveia, R. Uso de Protótipos no Processo de Concepção de Interfaces do Usuário. In II Congresso de Pesquisa e Inovação da Rede Norte Nordeste de Educação Tecnológica, João Pessoa, Brazil. Anderson, R.E. Social impacts of computing: Codes of professional ethics. *Social Science Computing Review* 10, 2 (Winter 1992), 453-469, 2007.

[2] Cardelli, L.; Wegner, P. On understanding types, data abstraction, and polymorphism. 1985. *ACM Comput. Surv.* (ACM) 28: 150. doi:10.1145/242224.242415. ISSN 0360-0300. Disponível em:

<http://lucacardelli.name/Papers/OnUnderstanding.A4.pdf>. Acesso em: Outubro 2011.

[3] Ferreira, A.P.L.; Ribeir,L. Programação Orientada a Objeto com Grafos. 2006. Disponível em: <http://www.lbd.dcc.ufmg.br/colecoes/jai/2006/009.pdf>. Acesso em: Outubro 2011.

[4] Filatro, Andrea. Design Instrucional na prática. Pearson: São Paulo, 2008.

[5] Filho, A.M.S. Introdução à Programação Orientada a Objetos. Revista Espaço Acadêmico, Nº 35, abril/2004. Mensal. ISSN 1519.6186.

[6] França, E.L.; Felix, Z.C.; Souza, M.S.; Carneiro, T.B.; Sousa, P.R.C; Filho, C.A.P.D. Utilização de Objetos de Aprendizagem em Sistemas Tutores Inteligentes para o ensino da Programação. VII SEGeT - Simpósio de Excelência em Gesntão e Tecnologia. 2010.

[7] Gladchef, A. P.; Sanches R.; Silva D. M. Um instrumento de avaliação de qualidade de software educacional: como elaborá-lo. Rio de Janeiro, 2001. Disponível em: <http://revistas.pucsp.br/index.php/pensamentorealidade/article/view/8484/6296>. Acesso em: Novembro 2011.

[8] Iepsen, E. F. ; Bercht, M. ; Reategui, E. B. . Persona-Algo: Personalização dos Exercícios de Algoritmos auxiliados por um Agente Afetivo.. In: Anais do Simpósio Brasileiro de Informática na Educação - SBIE 2010, 2010, João Pessoa.

[9] Junior, D.P., Freitas, R.L. Estratégias para melhorar os processos de abstração na disciplina de Algoritmos. XXI Simpósio Brasileiro de Informática na Educação. João Pessoa, PB, Brasil, 2010.

[10] Júnior, J.C.R.P.; Rapkiewicz, C.E.; Delgado, C.; Xexeo, J.A.M. Ensino de Algoritmos e Programação: Uma Experiência no Nível Médio. XXV Congresso da Sociedade Brasileira da Computação, 2005. Disponível em: <http://200.17.137.110:8080/licomp/Members/jeanemelo/plonelocalfolderng.2006-04-10.7475913377/PEP/Aula5/arq0033-2005-medio.pdf>. Acesso em: Julho 2011.

[11] Kemp, R., Todd, E. & Lu, J.Y. A novel approach to teaching an understanding of programming. In U. Hoppe, F. Verdejo & J. Kay (eds.), Proc. 11th Int. Conf. on AIEd, pp. 449-451, Amsterdam, IOS Press, 2003.

[12] Mendes, A. J. N. Software educativo para apoio à aprendizagem de programação. Disponível em: http://www.c5.cl/ieinvestiga/actas/tise01/pags/charlas/charla_mendes.htm. Acesso em: Maio 2011.

[13] Mota, M.P., Pereira, L.W.K. e Favero, E.L. JavaTool: Uma Ferramenta para o Ensino de Programação. In: Congresso da Sociedade Brasileira de Computação. Belém – Pará, 2008. XXVIII Congresso da Sociedade Brasileira de Computação. p. 127-136

[14] Neto, R. M .F. Polimorfismo e Acoplamento Dinâmico. 2010. Disponível em: <http://www.catalao.ufg.br/cc/roberto/apostilas/polimorfismo.pdf>. Acesso em: Outubro 2011.

[15] Rezende, P.J. ;Garcia, I.C. Astral: Animação Gráfica de Algoritmos e Estruturas de Dados - Uma Abordagem Construtiva, VIII Simpósio Brasileiro de Computação Gráfica e Processamento de Imagens, 317-318. 1995.

[16] Vahldick, A. Uma Experiência Lúdica no Ensino de Programação Orientada a Objetos. XVIII Simpósio Brasileiro de Informática na Educação. São Paulo, SP, Brasil, 2007.

[17] Valadares, C. Ambiente de Educação a Distância Sobre a Linguagem de Programação Java. Disponível em: <http://www.tise.cl/archivos/tise2006/20.pdf>. Acesso em: Maio 2011.

[18] Vizcaíno, A., Contreras, J., Favela, J. & Prieto, M. An adaptive, collaborative environment to develop good habits in programming. In C. Frasson, G. Gauthier, & K. VanLenh (eds.), Proc. 5th Int. Conf. on ITS, pp .262-271, Berlin, Springer-Verlag, 2000.