

# ASPECTOS DA AVALIAÇÃO NO PROCESSO DE ENSINO E APRENDIZAGEM DE COMPONENTES CURRICULARES DA ÁREA DE ENGENHARIA DE SOFTWARE

**Claudia Pinto Pereira Sena<sup>1</sup>; Hugo Saba<sup>2</sup>; David Moises Barreto dos Santos<sup>3</sup>**

<sup>1</sup> Universidade Estadual de Feira de Santana (UEFS), Departamento de Ciências Exatas  
Km 03, BR-116 Campus Universitário  
CEP: 44.031-460, Feira de Santana, BA  
caupinto.sena@gmail.com

<sup>2</sup> Universidade Estadual de Feira de Santana (UEFS), Departamento de Ciências Exatas  
Km 03, BR-116 Campus Universitário  
CEP: 44.031-460, Feira de Santana, BA  
hugosaba@uefs.br

<sup>3</sup> Universidade Estadual de Feira de Santana (UEFS), Departamento de Ciências Exatas  
Km 03, BR-116 Campus Universitário  
CEP: 44.031-460, Feira de Santana, BA  
davidmbs@uefs.br

**Resumo:** *Avaliar é um processo que deve ser contínuo e possibilitar a correção dos desvios durante o percurso de aprendizagem. Desta forma, este artigo tem por objetivo apresentar aspectos relacionados às avaliações do método de Aprendizagem Baseada em Problemas no processo de ensino e aprendizagem dos estudantes do módulo de Algoritmos e Programação I e do estudo integrado de Engenharia de Software, discutindo as experiências adquiridas e apontando pontos negativos e positivos. Os referidos componentes curriculares fazem parte do currículo do curso de Engenharia de Computação da Universidade Estadual de Feira de Santana, Bahia.*

**Palavras-chave:** *Engenharia de Computação, Avaliação no Processo de Ensino e Aprendizagem, Método de Aprendizagem, Aprendizado Baseado em Problemas.*

## 1. INTRODUÇÃO

A avaliação seja no ensino fundamental, no ensino médio ou no ensino superior, continua sendo uma das questões que mais aflige os docentes, pois, ainda é um “terreno” cheio de dúvidas, incertezas, de constantes acertos e desacertos, e que merece atenção redobrada daqueles que estão envolvidos diretamente com o processo de ensino e aprendizagem. Segundo COSTA (2004), tanto a bibliografia acadêmica quanto a legislação sobre a avaliação confronta-nos com termos como “conhecimentos”, “conteúdos”, “competências”, “capacidades”, “aquisições”, “destrezas”, “habilidades”, “atitudes”, “comportamentos”, etc., sem que sejam imediatamente claras as fronteiras e as correlações existentes entre eles. Segundo a LDB - Lei de Diretrizes e Bases da Educação Nacional (MEC, 1996), a avaliação do desempenho do aluno deve ser contínua e cumulativa, com prevalência dos aspectos qualitativos sobre os quantitativos.

Avaliar, para o dicionário Aurélio, representa a possibilidade de medir, computar, calcular. Entretanto, essa definição acaba sendo um pouco tendenciosa, no sentido de refletir apenas a avaliação quantitativa, desconsiderando, ou deixando em segundo plano, a avaliação qualitativa e processual. Por isso, é importante que os currículos estejam atentos para a avaliação e a qualidade desta, no sentido de garantir o aprendizado do estudante e não apenas mensurar, através de números, a aquisição de conteúdos.

É claro que não é possível desconsiderar toda a estruturação da avaliação no ensino superior, que não é diferente dos outros níveis de escolaridade. Em todos eles, as notas e conceitos são decisivos para a continuidade dos estudos, refletindo o “sucesso” ou o “fracasso” do estudante durante seu processo de aprendizagem. Essa quantificação, muitas vezes, mensura ou, pelo menos, tenta mensurar o que foi aprendido (conteúdos). Entretanto, outras competências, habilidades, atitudes também precisam ser consideradas no processo avaliativo dos estudantes. A avaliação deve ser sistemática, processual, construtiva e formativa, favorecendo a evolução do estudante.

Diante dessa preocupação, aliada ao perfil esperado de um profissional de engenharia (ABENGE, 1991), buscou-se, desde a concepção do Curso de Engenharia de Computação da UEFS, um método que, em sua essência, visasse, dentre outras coisas, permitir o desenvolvimento do raciocínio, da comunicação interpessoal, da iniciativa e da auto-aprendizagem. Essas competências e habilidades são também requisitos colocados na resolução 11/2002 do Conselho Nacional de Educação - CNE, que “Institui Diretrizes Curriculares Nacionais do Curso de Graduação em Engenharia” (CNE, 2002; SANTOS, *et al.*, 2007).

A Aprendizagem Baseada em Problemas (ABP ou PBL – *Problem Based Learning*) vem sendo, portanto, adotado no currículo de Engenharia de Computação, desde o seu início, em 2003, em alguns dos seus componentes curriculares, dentre os quais estão “Algoritmos e Programação I” e “Estudo Integrado de Engenharia de Software”. O detalhamento da aplicação do método ao longo do currículo do Curso de Engenharia de Computação da UEFS e de seus componentes curriculares encontra-se em (SANTOS *et al.*, 2007).

ANASTASIOU e ALVES (2003) expõem várias estratégias com as quais o professor pode levar o ensino, inclusive, a forma de avaliar o aluno em cada uma delas, desde as mais comuns, como a aula expositiva dialogada, até as menos utilizadas. Dentre elas, está também o PBL. Segundo eles, o PBL estimula ou amplia a significação dos elementos apreendidos em relação à realidade ou área profissional.

Esse método (PBL) busca tornar o estudante o agente construtor de seu próprio conhecimento, orientado pela figura do docente, ora chamado de tutor. Portanto, o aluno é, constantemente, estimulado a aprender e a fazer parte do processo de construção do aprendizado e constantemente avaliado em relação ao seu conhecimento cognitivo e ao desenvolvimento de habilidades e atitudes necessárias à profissão (DELISLE, 1997; BOUD & FELETTI, 1998; DUCH *et al.*, 2001).

Um dos elementos principais do método é o problema, através do qual o estudante se defronta com situações que simulam problemas reais, e que precisam do esforço coletivo (grupo tutorial) e de reuniões periódicas (sessões tutoriais) para que sejam solucionados. Todo esse processo contínuo (sessões tutoriais, discussões com o grupo tutorial, produtos resultantes da solução encontrada, etc.) e a dinâmica proposta pelo método são avaliados pelo tutor, e também pelo estudante, que tem a possibilidade de se auto-avaliar.

Neste contexto, a principal contribuição deste trabalho consiste em relatar como acontecem as avaliações no processo de ensino e aprendizagem dos estudantes do curso de Engenharia de Computação na Universidade Estadual de Feira de Santana (UEFS), em Feira de Santana – Bahia, identificando pontos positivos, discutindo dificuldades visualizadas pelos atores envolvidos: tutores (docentes) e estudantes, e propondo ações que podem, de alguma

maneira, simplificar o processo avaliativo e evitar situações conflitantes. Para tanto, o artigo foi dividido em mais cinco sessões, além da introdução. A sessão 2 traz uma breve explicação sobre os parâmetros avaliativos considerados pelo método PBL e que servem como base para os trabalhos nos componentes curriculares; as sessões 3 e 4 falam de como são organizados os componentes curriculares “Algoritmos e Programação I” e “Estudo Integrado de Engenharia de Software”; a sessão 5 traz a discussão do Processo de Avaliação nos Componentes Curriculares em questão e a última delas traz as considerações finais.

## **2. O PROCESSO DE AVALIAÇÃO PROPOSTO PELO MÉTODO PBL**

A existência de dificuldades no alinhamento das atividades educacionais baseados em problemas e a avaliação dos alunos [Mamed & Penaforte, 2001], conduzem o método PBL a uma dinâmica analisada sobre dois aspectos: no aspecto global se observa as competências gerais e específicas adquiridas pelo estudante, e no aspecto modular é observada a proficiência de cada aluno na formação do conhecimento em assuntos específicos inerentes a cada módulo do estudo integrado.

No aspecto global, o método PBL adotado no curso buscar avaliar o aluno de várias formas no desenrolar do componente curricular. Neste sentido, basicamente são avaliados: o ciclo PBL, a discussão que gira em torno do problema e a solução final do problema dado.

O ciclo PBL é o processo de discussão entre os alunos durante as sessões tutoriais objetivando atingir a solução do problema. Para tanto, são seguidos passos pré-definidos (SANTOS *et al.*, 2007). Desta forma, é essencial que uma auto-avaliação deste ciclo seja periodicamente realizada a fim de absorver críticas para rever e aperfeiçoar o processo de ensino-aprendizagem. Vale ressaltar que nesta avaliação é tratada a postura do aluno, do grupo e do tutor. Esta avaliação é apenas qualitativa, não sendo quantitativa.

Quanto a avaliação das discussões, o tutor busca, durante a sessão tutorial, avaliar a participação de cada aluno. Não existe uma padronização de critérios usados entre os tutores, porém muitos tentam fazê-la. Entretanto, mesmo usando critérios iguais, a avaliação acaba não sendo uniforme, uma vez que é subjetivo definir, por exemplo, quão boa foi a participação de um aluno.

Para tentar amenizar este aspecto, algumas características e indícios podem ajudar no momento da avaliação:

- processo de raciocínio coerente: nota o desenvolvimento cognitivo do aluno, verificando de onde se originou a idéia até a sua conclusão;
- revisão bibliografia prévia: uma das formas de criar novos conhecimentos é recorrer a conhecimentos já disponíveis na literatura. Portanto, esta atividade é essencial para a geração de uma boa discussão;
- assiduidade: é importante a frequência do aluno e sua pontualidade em cada sessão tutorial sob pena de prejudicar a discussão já iniciada pelo outros alunos;
- habilidade de comunicação oral: o desenvolvimento de uma expressão oral de forma lógica, concisa e ordenada é fundamental para um bom profissional. Então, estas características são estimuladas e observadas no intuito de que o aluno melhore continuamente a referida habilidade. Aqui também está inclusa a habilidade de ouvir criticamente os outros colegas;
- levantamento de questões: a formulação de questões centrais, pertinentes ao problemas, ajuda na reflexão a cerca do problema e no direcionamento de decisões a serem tomadas. Assim, é importante levar esse item em consideração na avaliação;
- criatividade: observar a criatividade do aluno na resolução do problema.

A solução do problema, denominado produto, pode ser apresentada pelos alunos de diversas formas, por exemplo, relatório, código-fonte do programa exigido, diários reflexivos, etc. O tutor (elaborador do problema) é quem é o responsável por definir qual a forma mais adequada. Normalmente, nos componentes curriculares da área de Engenharia de Software pede-se relatório e código-fonte. Para cada problema, um produto deve ser entregue por apenas um aluno ou por uma equipe, dependendo de sua natureza e da escolha do tutor.

Por fim, ainda considerando a avaliação do produto, há o “bate-bola”, que é um momento no qual o aluno discute com o tutor a respeito do mesmo, explicando a solução proposta, principalmente, os detalhes requeridos. O “bate-bola” sempre é feito individualmente com cada aluno do grupo tutorial, independentemente de o produto ter sido elaborado por apenas um aluno ou por uma equipe. Durante este tempo, o tutor ainda pode apontar possíveis pontos para correção, para que, dado um prazo, o aluno re-entregue o produto novamente. Desta forma, acontece um processo de avaliação contínuo.

### **3. COMPONENTES CURRICULARES DO CURSO DE ENGENHARIA DE COMPUTAÇÃO DA UEFS**

O currículo do Curso de Engenharia de Computação possui componentes curriculares ditos isolados e outros ditos estudos integrados. Todas as disciplinas são chamadas de módulos. Portanto, os módulos isolados são disciplinas que podem trabalhar ou não com o método PBL. Essa escolha depende do professor e das possibilidades da disciplina (módulo), pois algumas favorecem mais o uso do PBL e outras não.

O módulo “Algoritmos e Programação I”, por exemplo, é um desses módulos isolados, trabalhado com o método PBL. Dessa maneira, intercala, em suas 60 horas de carga horária, encontros com os grupos tutoriais (as chamadas sessões tutoriais), e as aulas de caráter expositivo, com professores que assumem papéis de consultores e conferencistas (SANTOS *et al.*, 2007). O consultor assume o papel de esclarecer dúvidas em relação a assuntos tratados nos problemas ou mesmo nas aulas teóricas, enquanto o conferencista assume as aulas teóricas, com todas as variações que elas possam ter: debates, exposição, palestras, etc. Embora possam ser pessoas diferentes, no geral, nesse módulo o conferencista e o consultor é o mesmo professor da disciplina.

O professor assume também o papel de tutor durante as sessões tutoriais, orientando os estudantes durante as discussões sobre o problema proposto. É preciso que o tutor permita autonomia dos estudantes, para que possam construir suas próprias hipóteses, idéias e soluções. Isto, é claro, não significa que o tutor não possa interferir, em momentos oportunos, para garantir a continuidade dos trabalhos, nas sessões tutoriais (SANTOS *et al.*, 2007).

#### **3.1 Componente Curricular “Algoritmos e Programação I”**

Esse módulo tem algumas particularidades importantes, que devem ser levadas em consideração. É um módulo oferecido no primeiro semestre do curso, para calouros recém egressos do ensino médio. Dessa forma, existe também todo o processo de adaptação ao novo ritmo do ensino superior. Além disso, eles se deparam com um novo método que exigem mais autonomia, iniciativa, maior controle e organização do seu tempo de estudo, e habilidades que talvez não possuam, ou não tenham tido oportunidades de desenvolver, como a habilidade de comunicação oral e escrita, de trabalhar em grupos, dentre outras. Por fim, é também o primeiro encontro desses alunos com o desenvolvimento do raciocínio lógico e com uma linguagem de programação, nesse caso a linguagem C.

Todos esses fatores agregados podem trazer ao aluno maior insegurança e desconforto. Por conta disso, é preciso uma preocupação especial em como tratar os conteúdos, em como

apresentar e preparar os estudantes para o PBL, e em como avalia-los adequadamente, dentro da proposta processual e continuada da avaliação. Em relação à preparação do estudante para o PBL, há dois semestres, é construída, conjuntamente com professores e alunos, a “1 Semana de Integração”, encontro que objetiva receber-los, ambientá-los, proporcionar ouvir depoimentos de estudantes que já vivenciam a academia, e participar de oficinas que trazem a prática do PBL. Nessas oficinas, eles se comportam como coordenadores de sessões, secretários de quadro e mesa, tutores, podendo experimentar o que viverão dali pra frente.

Como é um módulo isolado, os problemas propostos durante as sessões tutoriais envolvem apenas conteúdos de desenvolvimento do raciocínio lógico e de programação em C: elaboração de algoritmos em pseudocódigo, fluxograma, estruturas de controle (seqüenciais, condicionais, repetição), tipos de dados simples e compostos, estruturas de dados, ponteiros, modularização, passagem de parâmetros, etc. Entretanto, todo problema elaborado busca tratar esses conteúdos, significando-os, contextualizando-os, trazendo uma situação que envolva aspectos da vida real, principalmente em se tratando de problemas a serem solucionados por futuros profissionais de tecnologia. Essa contextualização tenta aproximar os estudantes de situações com as quais provavelmente lidarão em sua vida profissional, como por exemplo, clientes dizendo o que desejam em seus sistemas. Isso implica que é preciso também que eles possuam discernimento, compreensão, no sentido de “filtrar” no texto o que é central para a solução do problema. É preciso também poder de síntese para, além de codificar a solução, explicá-la com clareza.

Abaixo, é trazido um problema que mostra detalhes importantes no que se refere à avaliação – “Figura 1”. Esses detalhes e aspectos serão tratados na seção 5.

### **Problema 1: Sistema de Alocação em Teatros**

**Tema:** Algoritmos: conceitos, representação e aplicações.

#### **Objetivos**

- Compreender os fundamentos de algoritmos, desde a sua conceituação até suas formas de representação (e.g.: fluxogramas e pseudo-código);
- Aplicar os conceitos de algoritmos, modelando problemas estruturados logicamente e representando suas soluções em linguagem adequada;
- Compreender as etapas de entrada, processamento e saída envolvidas na solução de um problema através do uso do computador;
- Compreender as principais estruturas de seleção e de repetição utilizadas em algoritmos e aplicá-las para resolver problemas simples;
- Interpretar uma especificação de um problema complexo através de abstrações e uso do raciocínio lógico;
- Projetar possíveis soluções para um problema complexo na forma de algoritmos expressos através de fluxogramas e pseudo-código;
- Pesquisar sobre noções básicas de algoritmos em diversas fontes tais como bibliografia básica e sites na Internet;
- Apresentar a solução de um problema complexo através de expressão oral.

#### **Cronograma**

(OBS: No Cronograma, são colocados todos os encontros (sessões tutoriais) pertinentes ao problema em questão, assim com as aulas teóricas, conferências ou consultorias).

#### **Problema:**

Uma empresa que gerencia alocação de espaços em teatros, embora já trabalhe com isso há algum tempo, possui seu controle ainda manual. Dessa forma, seu atendimento torna-se lento, por conta da busca manual que é feita por espaços desocupados.

Como a empresa está aumentando o seu número de clientes e, com isso, também a exigência por serviço mais rápido e eficiente, pensou em informatizar o serviço de alocação de espaços em teatros. Conversando com um amigo, o dono dessa empresa foi orientado a procurar os alunos da Disciplina Algoritmos e Programação I, do Curso de Engenharia de Computação da UEFS, para que estes desenvolvessem a solução para esse problema.

A recomendação da empresa de gestão de teatros é que a solução seja entregue dentro do prazo previsto e conforme as suas especificações (especificações do cliente), descritas abaixo:

- O programa deve ser capaz de registrar reservas de lugares em um teatro;
- Considere que cada teatro pode ter, no máximo, 24 fileiras de cadeiras (A, B, C, D...Z);
- Considere que cada fileira tem, no máximo, 100 cadeiras, onde cada um dos lugares é identificado com uma letra (coluna) e um número (Ex: A04, B23, etc.);
- O programa deverá solicitar ao usuário qual lugar deseja ocupar;
- Caso o lugar indicado não esteja vago, o programa deverá avisar ao usuário para que escolha um novo lugar;
- Caso o lugar indicado esteja vago, este deverá ser reservado ao usuário;
- Sempre que uma dada fileira (primeira, segunda, etc.) estiver totalmente ocupada, o programa deverá informar ao usuário antes que ele efetue a escolha;
- Ao final de cada reserva, o programa deverá indicar:
  - O total de lugares ocupados por fileira
  - O total de lugares vagos por fileira
  - O total de lugares ocupados no teatro
  - O total de lugares vagos no teatro

#### **Produto**

Você foi devidamente contatado pela empresa e prontamente aceitou o desafio de desenvolver este novo sistema, apesar de ainda está iniciando os seus estudos nas mais avançadas técnicas de programação e de desenvolvimento de softwares da atualidade.

Você deve fazer um relatório, contendo duas formas lógicas de representação para o problema em questão (fluxograma e pseudocódigo). Seu relatório deve ser entregue **impresso**, impreterivelmente, no dia 10/06, além de postado no site do módulo de Algoritmos e Programação I, no Moodle, em 07/06. A apresentação oral do trabalho será realizada na sala tutorial, também no dia 10/06.

#### **Recursos para Aprendizagem**

- MANZANO, J. A. N. G., OLIVEIRA, J. F. **Algoritmos** : Lógica para Desenvolvimento de Programação. São Paulo: Érica, 1996.
- MANZANO, J. A. N. G. **Estudo Dirigido de Linguagem C**. São Paulo: Érica: 1997.
- FORBELLONE, A. V. L., EBERSPACHER, H. F. **Lógica de Programação** : A Construção de Algoritmos e Estrutura de Dados. 2. ed. São Paulo: Makron Books, 2000.
- FARRER, H., et al. **Algoritmos Estruturados**. 3. ed. Rio de Janeiro: LTC, 1999.

Figura 1 – Exemplo de um Problema no Modulo Isolado de Algoritmos e Programação I.

### **3.2. Componente Curricular “Estudo Integrado de Engenharia de Software”**

O Estudo Integrado (EI) de Engenharia de Software já foi aplicado em quatro semestres distintos e sua carga horária foi planejada para abranger um total de 180 horas sendo dividido em 3 módulos, contendo 60 horas de duração cada um: Engenharia de Software, Análise e Projeto de Sistemas e Bancos de Dados.

A abordagem de ensino deste Estudo Integrado tem possibilitado uma experiência profissional e ao mesmo tempo em que permite ao estudante vivenciar situações reais de engenharia de software. Um importante benefício desta abordagem proposta de ensino da Engenharia de Software é a vivência profissional participativa, obtida através do desenvolvimento de um sistema real.

O processo de desenvolvimento de software do EI teve sua origem na sua primeira edição, quando foi adotada uma adaptação do Processo Unificado, considerando o ciclo de vida espiral. Entretanto, ainda é um desafio alcançar a finalização do produto, passando por todo o ciclo de desenvolvimento de software. Neste sentido, na última edição foi adaptado um processo baseado no processo ágil *eXtremming Programming* (XP), no intuito de que os alunos conseguissem finalizar o software e ter uma maior experiência na etapa de manutenção, que foi pouco explorada em edições anteriores.

A Figura 2 apresenta a primeira etapa do projeto, representando detalhes importantes no que se refere à avaliação (Seção 5).

## **. Problema 1: Acervo Digital**

**Tema:** Análise e definição de requisitos

### **Objetivos**

- Conhecer os fundamentos da Engenharia de Software;
- Compreender como se dá a comunicação entre os membros da equipe;
- Realizar tarefas em conjunto de forma mais eficiente;
- Compreender os mecanismos necessários para identificar o tempo para desenvolvimento e o custo de um artefato de software;
- Realizar levantamento de custo de software com vistas a definir o preço do software;
- Dominar técnicas para elicitar requisitos do usuário;
- Levantar requisitos usando técnicas e metodologias apropriadas a situação;
- Definir, especificar e monitorar metas de qualidade;
- Identificar riscos e criar planos de contingência para administrar ou minimizar o seu impacto;
- Usar testes de aceitação como uma das formas de atender os requisitos do usuário.

### **Problema**

Você faz parte de uma equipe de desenvolvimento que foi contratada para confeccionar um sistema de acervo dinâmico para ser usado pela UEFS. Este sistema deverá ser capaz de gerar diferentes tipos de acervos específicos, por exemplo, processos jurídicos, insetos, recortes de jornais, etc.

Primeiramente, vocês devem fazer um estudo de viabilidade. Em seguida, caso realmente seja viável, sua equipe deverá iniciar as atividades de análise e definição de requisitos. É importante especificar o tempo gasto para desenvolver cada requisito. Visando a qualidade do software, o processo de desenvolvimento a ser usado é o XP, portanto, você e sua equipe devem definir padrões de desenvolvimento e documentação.

O gerente de desenvolvimento da Assessoria Especial de Informática da UEFS, Abraão Vieira, será o cliente do sistema (ramal 8223 ou telefone 3224-8223).

Vocês sentem que essa é a grande chance que estava faltando e, portanto, precisam se esforçar ao máximo para que o produto desenvolvido atenda as necessidades do cliente. Mas também não querem ter prejuízo com o produto e sabem que o sucesso no seu desenvolvimento será vital para o futuro da empresa.

### **Produto**

Haverá dois produtos a serem entregues. O primeiro é um relatório do estudo de viabilidade de desenvolvimento do software. A data final será dia 13/11.

O segundo produto deverá ser uma proposta escrita para o desenvolvimento do sistema proposto, a qual deve ser elaborada pela equipe do grupo tutorial e entregue até o dia 22/11. Nesta proposta deverá conter, pelo menos, os seguintes artefatos:

- Documento de Visão
- Metas de qualidade
- Análise de Risco
- Estimativa de tempo
- Cronograma de Atividades
- Papéis de cada membro da equipe
- Projeto Arquitetural

Uma apresentação desta proposta deverá ser realizada no dia 23/11, ocorrendo o bate-bola no dia 26/10.

### **Recursos para aprendizagem**

SOMMERVILLE, IAN. "Engenharia de Software", 6ª Edição, Addison Wesley, 2003.

PFLIEGER, SHARI L.. "Engenharia de Software: teoria e prática", Pearson, 2004.

PRESSMAN, ROGER S.. "Engenharia de Software", 5ª Edição, Pearson, 1999.

CARVALHO, Ariadne m. B. Rizzoni; CHIOSSI, Thelma C. dos Santos. "Introdução à engenharia de software", Ed. da UNICAMP, 2001.

Ghezzi, Carlo; Jazayeri, Mehdi; Mandrioli, Dino. "Fundamentals of software engineering". 2a edição. Prentice-Hall, 2003.

Gane, Chris. "Análise estruturada de sistemas". Rio de Janeiro : Livros Tecnicos e Cientificos, 1984.

LARMAN, C. Utilizando UML e padrões: uma introdução a análise e projeto orientados a objetos. Porto Alegre: Bookman, 2004.

Instituto de Engenharia de Software, <http://www.sei.cmu.edu/>  
Ferramentas para auxiliar no acompanhamento e gerência de projetos:  
<http://www.dotproject.net/>  
<http://www.xplanner.org/>  
Ferramenta para auxiliar na edição de documentos:  
<http://www.openoffice.org>.  
Ferramenta para definição de testes de aceitação:  
<http://easyaccept.org/>

•

Figura 2 – Exemplo de um Problema no Estudo Integrado de Engenharia de Software.

Na verdade, o problema do Estudo Integrado de Engenharia de Software tem um caráter de projeto, isto é, é um problema dividido em sub-problemas (etapas). Concluída a primeira etapa, dá-se início às etapas complementares gerando ao final de cada uma delas um *release* do software. Seguindo o processo baseado em XP, para cada *release*, são alocadas duas iterações de duas ou três semanas. Em cada iteração, há atividades de planejamento, acompanhamento, desenvolvimento e testes que são avaliadas continuamente. Um ponto importante deste componente é a presença de um cliente externo que define os requisitos do software e ajuda no processo de avaliação do resultado.

Em cada release o software é incrementado com novas funcionalidades ao mesmo em que novos conceitos são abordados no componente curricular. Desta forma, os artefatos a serem entregues em cada etapa variam, conforme descrito a seguir:

- Etapa 1: Documento de Visão; Metas de qualidade; Análise de Risco; Estimativa de tempo; Cronograma de Atividades; Papéis de cada membro da equipe; Projeto Arquitetural;
- Etapa 2: Atualizações de documentos anteriores, com detalhamento dos requisitos de sistema e do projeto arquitetural; Modelo de Dados; Histórico da Análise de Risco; Histórico das Métricas calculadas; Documento de configuração; Implementação de requisitos;
- Etapa 3: Atualizações de documentos anteriores; Implementação de requisitos;
- Etapa 4: Atualizações de documentos anteriores; Código-fonte com JAVADOC; Arquivo WAR e manual de instalação; Refinamento de Implementação; Testes de Unidade; Testes de aceitação.

Ao final do estudo integrado, um importante benefício observado é a vivência profissional participativa dos alunos, frente a um desenvolvimento de sistema real. O resultado dos trabalhos são bem avaliados, e comparado a realidade exigida pelo mercado de trabalho, apresentam uma grau de funcionalidade que os permitem serem implantados.

#### **4. DISCUSSÃO DO PROCESSO DE AVALIAÇÃO NOS COMPONENTES CURRICULARES EM QUESTÃO**

Antes de iniciar a descrição e discussão do processo avaliativo nos componentes curriculares “Algoritmos e Programação I” e “Estudo integrado de Engenharia de Software”, é pertinente situar esse processo diante de algumas considerações sobre avaliação, PBL, e a relação entre estes, a fim de compreender de que maneira foi pensada a avaliação para estes módulos.

O PBL, enquanto método de aprendizagem, busca desenvolver nos estudantes algumas habilidades, e não somente a apreensão dos conteúdos. Dentre elas, pode-se listar:

- Aprendizado independente;



- Solução de Problemas;
- Desenvolvimento do pensamento crítico;
- Habilidades de comunicação oral;
- Habilidade de comunicação escrita.

Dessa maneira, possui todo um ciclo sistemático (SANTOS *et al.*, 2007) que busca atingir esses objetivos. De maneira semelhante, BEAUMONT *et al.* (2004) traz o modelo 3P (*Presage, Process, Product*) adaptado ao PBL, que ilustra de maneira geral como se organiza o método, desde o ponto anterior à introdução do problema (*presage*), o processo de aprendizagem através de problemas (*process*), e os resultados obtidos após a solução do problema (*product*). O esquema desse modelo pode ser visualizado na “Figura 3”.

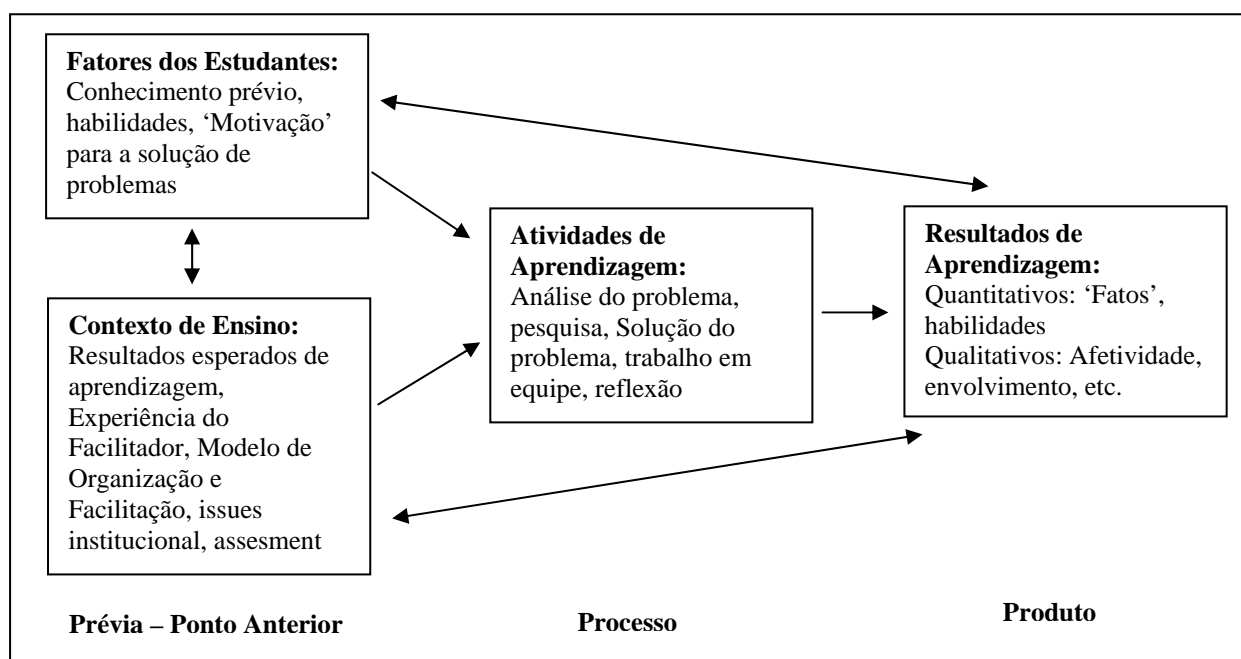


Figura 3 – Modelo 3P, adaptado ao PBL (redesenhado de BEAUMONT *et al.*, 2004)

Esse modelo retrata bem a relação do processo de ensino e aprendizagem com o estudante. O centro desse processo deixa de ser o professor e passa a ser o próprio aluno. É claro que o professor, ora chamado de tutor, consultor ou conferencista, é responsável por facilitar este caminhar, conduzir a apreensão e aquisição de conhecimentos. A predisposição do aluno e seus conhecimentos prévios colaboram muito para o desenvolvimento desse processo.

Nessa fase inicial, aqui chamada de Prévia – Ponto Anterior, nos primeiros contatos entre os atores (estudante e tutor), é de extrema importância que sejam esclarecidas todas as regras de convivência, todos os critérios de avaliação, toda e qualquer dúvida em relação ao método e sua sistemática. Segundo MAMEDE e PENAFORT (2001), o processo de comunicação sobre a avaliação deve ser regular e claro, o que implica que todos os envolvidos precisam ter clareza sobre as características e métodos dos sistemas de avaliação. De maneira semelhante, o processo de divulgação dos resultados (*feedback*) aos estudantes deve fornecer informações precisas e confiáveis sobre seu desempenho, promovendo seu crescimento.

Imbuídos dessa preocupação inicial, é que são realizadas Oficinas PBL com os alunos que ingressam no Curso de Computação, para que eles se inteirem do método utilizado. Os critérios de avaliação também são divulgados juntamente com o plano da disciplina, na primeira semana de aulas. Nesse caso, o aluno fica ciente de que serão avaliados

processualmente, através das sessões tutoriais, com critérios estabelecidos e o quantitativo associado. No caso do módulo “Algoritmos e Programação I”, são estabelecidos pesos para as avaliações de conteúdo (40%), para os produtos resultantes dos problemas (40%) e para o desempenho dos mesmos durante as sessões (20%), perfazendo um total de 10 pontos, para cada uma das unidades.

Em relação ao “Estudo Integrado de Engenharia de Software”, os pesos estipulados são outros: 70% para os produtos resultantes dos problemas e 30% para o desempenho.

Iniciados os encontros dos grupos tutoriais (processo), o estudante tem seu primeiro contato com o problema. Em cada uma das sessões, ele é avaliado pelo seu envolvimento no trabalho em grupo, pela sua participação, pela sua colaboração, e também avaliados em relação ao papel assumido por cada um deles. O coordenador, por exemplo, deve conduzir a sessão, incentivar outros colegas a participarem, permitir que todos tenham a oportunidade de se expressar e não deixar que outros diálogos não pertinentes ao problema tomem espaço na sessão. Todo esse acompanhamento é feito pelo tutor durante os encontros dos grupos. Como o grupo tutorial é o mesmo durante todo o semestre, o tutor acaba conhecendo de perto cada um dos seus alunos, suas capacidades, suas habilidades e também suas fragilidades. De qualquer maneira, esse acompanhamento processual precisa ser quantificado, até mesmo por conta de como se organiza toda a educação em nosso país. Para tanto, alguns formulários avaliativos foram sugeridos pelos professores, tanto para a avaliação de desempenho (processual), quanto para a avaliação do produto.

Tabela 1 – Formulário Avaliativo das Sessões Tutoriais

	Alunos	Aluno 1	Aluno 2	Aluno3
Critérios				
Coordenador				
Sec. Quadro				
Sec. Mesa				
Participação (I, H, Q)				
Participação (seguimento)				
Pontualidade				
Criatividade				
Expressividade				
Oralidade				
Cumprimento das metas				

A “Tabela 1”, mostra um formulário avaliativo para as sessões tutoriais. Nele, são registrados: a participação de cada aluno no que diz respeito às suas contribuições sobre idéias, hipóteses e questões relacionadas ao problema (I, H, Q), contribuições trazidas após estudo bibliográfico – a partir da segunda sessão tutorial (seguimento) - pontualidade, criatividade, expressividade (facilidade de expressar suas idéias), oralidade (capacidade de se comunicar, de falar em público), cumprimento das metas da sessão anterior, sua postura enquanto coordenador, secretário de quadro e secretário de mesa.

É claro que este formulário é apenas uma possibilidade, e é importante que se entenda que todo registro qualitativo envolve subjetividade, que quando expressa em números pode não refletir adequadamente o que foi avaliado, necessitando de ajustes ou re-avaliações. As re-avaliações acontecem continuamente; em cada uma das sessões, o estudante tem a possibilidade de melhorar seu comportamento em relação à sessão anterior. Para tanto, é claro, é preciso um *feedback* do tutor, o que acontece ao final de cada encontro. Nesse momento, o tutor avalia oralmente a sessão, os participantes e pede que eles se auto-avaliem,

no sentido de corrigir o rumo do aprendizado. Essa prática reflete o que é dito por HOFFMAN (2003). Após várias reflexões sobre avaliação classificatória, ela conclui que o aproveitamento máximo do aluno se dará pelas oportunidades que o mesmo obtiver no meio no qual está inserido. Já em relação à avaliação mediadora, acredita que o acompanhamento do aluno em todos os instantes gera uma certa interação, prestigiando todo o contexto do aprendizado.

A avaliação do produto nos módulos e estudos integrados do Curso de Engenharia de Computação da UEFS acontece, basicamente, através de duas maneiras: avaliação do relatório e, também, avaliação de código-fonte, modelo de software ou qualquer outro resultado esperado pela disciplina. É importante que, durante a elaboração do problema, haja a preocupação quanto à clareza dos objetivos a serem atingidos pelos alunos, assim como quais os resultados (produtos) esperados ao final de cada problema. Isto pode ser observado na “Figura 1”, que traz um exemplo de problema elaborado para o módulo “Algoritmos e Programação I”.

Em cada um dos problemas, são definidos os **objetivos** norteadores dos estudos dos alunos e das expectativas dos tutores. O tópico **produto** define claramente quais os artefatos devem ser entregues após todas as discussões das sessões tutoriais e estudos dirigidos. Os **objetivos**, juntamente com o **produto** e os **recursos de aprendizagem** norteiam, conduzem os alunos para a obtenção da solução, juntamente com a orientação do tutor.

Nos relatórios, busca-se desenvolver a habilidade de comunicação escrita, o poder de síntese do referencial teórico estudado, utilizando as normas ABNT como parâmetro. Além disso, toda a solução encontrada para a situação-problema deverá estar descrita no relatório, mostrando, além do domínio de conteúdos, a capacidade de sintetizar e expor textualmente o caminho percorrido no aprendizado. A “Tabela 2” traz o detalhamento dos critérios utilizados para a avaliação dos relatórios que descrevem a solução do problema. Vale lembrar que é dada ao estudante a oportunidade de, após a correção do relatório pelo tutor, rever seus erros, recorrer e entregar nova versão do relatório. Esta ação oportuniza aos estudantes repensar seus erros e melhorar seu desempenho (avaliação de caráter formativo e também somativo).

Tabela 2 – Formulário Avaliativo do Relatório

Item - valor de cada item 1,0	Aluno1	Aluno2	Aluno3
Organização do texto, normas ABNT			
Gramática e ortografia			
Introdução coerente e motivadora			
Base teórica suficiente e coerente com o texto			
Conceitos teóricos corretos e adequados			
Detalhamento e explicação da solução			
Resultados/figuras com simulação do produto			
Discussão dos resultados obtidos			
Conclusões sobre o projeto			
Referências apropriadas e citadas no texto			
Total no Relatório			

Quando um dos produtos é o código-fonte, observa-se primeiramente o atendimento dos requisitos funcionais descritos para o problema, além de critérios como legibilidade, comentários (documentação), e outros. Esses critérios podem ser visto na “Tabela 3”.

Tabela 3 – Formulário Avaliativo de Código-fonte

ITEM	Aluno1	Aluno2	Aluno3
Clareza do Código - 1,5			
Coerência com o problema - 2, 0			
Simplicidade / Legibilidade - 1,5			
Correção - 2,0			
Comentário - 1,5			
Sintaxe / Identificação - 1,5			
Total no Código			

Todas essas avaliações quantitativas (produtos) são depois submetidas a uma média, retratando em valores o desempenho do aluno. Após a correção do relatório, as considerações feitas pelo tutor são discutidas em um encontro previamente agendado, chamado de “bate-bola”. O bate-bola, como o próprio nome sugere, busca estabelecer entre o tutor e o aluno um diálogo sobre os resultados obtidos, momento no qual o aluno esboça o conhecimento adquirido durante a construção do produto, e tem a possibilidade de tirar dúvidas pessoais com o docente. Esse encontro, por se tratar de um encontro individual em diálogo livre, tem por objetivo maior, exaurir questões em aberto, permitindo também um detalhado acompanhamento do aluno.

## 5. CONSIDERAÇÕES FINAIS

O trabalho apresentado neste artigo relata a estratégia avaliativa no processo de ensino e aprendizagem utilizada nos componentes curriculares “Algoritmos e Programação I” e “Estudo Integrado de Engenharia de Software” no curso de Engenharia de Computação da UEFS. Vale ressaltar que esse processo avaliativo está intimamente relacionado ao método de aprendizagem adotado – PBL.

A adoção do método PBL permitiu um bom desenvolvimento de algumas das competências, citadas anteriormente, do artigo quarto da resolução 11/2002, aprovada pelo CNE (CNE, 2002): “V - identificar, formular e resolver problemas de engenharia”; “VIII - comunicar-se eficientemente nas formas escrita, oral e gráfica”; “IX - atuar em equipes multidisciplinares” e “XIII - assumir a postura de permanente busca de atualização profissional”.

Em relação ao módulo “Algoritmos e Programação I” e ao “Estudo Integrado de Engenharia de Software, Análise e Projeto de Sistemas e Banco de Dados”, o método vem possibilitando ao estudante experiência profissional, e vivência com situações reais, de forma participativa e colaborativa.

Em se tratando do processo avaliativo, todo ele é delineado pelo método PBL, variando percentuais atribuídos a cada um dos itens avaliados (quantitativo). Essa variação de ponderações é característica da natureza de cada módulo – seja ele isolado ou estudo integrado.

Como todo processo de avaliação, enfrenta-se também dificuldades. A primeira delas é a falta de uniformização quanto aos critérios avaliativos. Por isso, esse trabalho sugere alguns formulários avaliativos, principalmente como norteadores para os alunos. A falta de clareza quanto à avaliação, ou a falta de uniformização entre os docentes, é uma das principais queixas dos alunos. Outra dificuldade encontrada diz respeito à postura assumida por alguns dos alunos, que se mostram contrários à utilização do método, ou por conta de uma história escolar que privilegiou “o ensino” e não a “aprendizagem”, ou por conta de características pessoais, como introspecção, timidez ou algo parecido.

O grupo de docentes vem se preocupando com essa situação e tem como objetivo próximo fazer um levantamento, junto a todo corpo discente, sobre a aplicação do método, sobre a avaliação empregada, de maneira que possa existir uma reflexão e ação a respeito.

## REFERÊNCIAS BIBLIOGRÁFICAS

ABENGE - Associação Brasileira de Ensino de Engenharia. **Perfil do Engenheiro do Século XXI**. Brasília, mai. 1991, 19p.

ANASTASIOU, L.G.C. e ALVES, L.P. (Orgs). **Processos de Ensino na Universidade: pressupostos para as estratégias de trabalho em aula**. Joinville,SC: UNIVILLE, 2003.

BEAUMONT, C.; SACKVILLE, A.; CHENG, C. S. Identifying Good Practice in the use of PBL to teach computing. **Italics E-journal**, Ddd, p. 1-19. 10 jan. 2004.

BOUD, D.; FELETTI, G. **The Challenge of Problem-Based Learning**. London: Kongan Page, 1998.

CNE. **Resolução CNE/CES 11/2002**. Diário Oficial da União, Brasília, 9 de abril de 2002. Seção 1, p. 32.

COSTA, A. P. **Como avaliar o aprender a (Competências) e o aprender que (Conteúdos)?**. Porto, 2004. Disponível em: <<http://www.cef-spf.org/>>. Acesso em 27 de junho de 2008.

DELISLE, R. **How to use problem-based learning in the classroom**. Alexandria: Ascd, 1997.

DUCH, B. J.; GROH, S. E.; ALLEN, D. E. **The power of problem-based learning: a practical how to for teaching undergraduate course in any discipline**. Sterling: Stylus Publishing, 2001.

HOFFMAN, J. **Avaliação mediadora: uma prática em construção da pré-escola à universidade**. Porto Alegre: Editora Mediação, 1993. 20ª Edição revista, 2003.

MAMEDE, S.; PENAFORTE, J. **Aprendizagem Baseada em Problemas: anatomia de uma nova abordagem educacional**. Fortaleza: Hucitec, 2001.

MEC - Ministério da Educação. **Lei de Diretrizes e Bases da Educação Nacional**. Disponível em: <<http://portal.mec.gov.br/>>. Acesso em 24 de junho de 2008.

SANTOS, D. M. B, PINTO, G. R. P. R., SENA, C. P. P., BERTONI, F. C., BITTENCOURT, R. A., Aplicação do Método de Aprendizagem Baseada em Problemas no Curso de Engenharia de Computação da Universidade Estadual de Feira de Santana In: XXXV Congresso Brasileiro de Educação em Engenharia, 2007, Curitiba. **COBENGE 2007**. Curitiba - PR: , 2007. p.2A07-1 - 2A07-14.

## **ASSESSMENT ISSUES IN THE TEACHING-LEARNING PROCESS OF THE CURRICULAR COMPONENTS OF THE SOFTWARE ENGINEERING AREA**

**Abstract:** *Assessment is a process that must be continuous and allow correcting abuses during learning journey. Thus, this article presents issues related to assessments of the problem-based learning method in the teaching-learning process of the Algorithms and Programming I module and the Software Engineering Integrated Study, discussing learned lessons and pointing positive and negative points. These curricular components are part of the curriculum of Computing Engineering course of the Feira de Santana University, Bahia.*

**Key-words:** Computing Engineering, Teaching-Learning Assessment, Learning Method, Problem-Based Learning.