

# Recursos avançados de demonstração de resultados de um ambiente de avaliação de habilidades metacognitivas em lógica de programação

## IV Workshop de Tecnología Informática Aplicada en Educación (WTIAE)

Clemilson de Oliveira Abreu, Edson Pinheiro Pimentel

IMES – Universidade Municipal de São Caetano do Sul  
Av. Goiás, 3400, 09550-051, São Caetano do Sul, SP, Brasil,

[coabreu@terra.com.br](mailto:coabreu@terra.com.br), [edson.pimentel@imes.edu.br](mailto:edson.pimentel@imes.edu.br)

***Resumo.** Pesquisas atuais apontam que o monitoramento dos conhecimentos prévios é um processo metacognitivo fundamental. Se os estudantes não conseguem diferenciar com precisão o que eles sabem, do que eles não sabem, dificilmente conseguirão estabelecer qualquer espécie de comprometimento em atividades metacognitivas avançadas, como avaliar, planejar ou controlar a própria aprendizagem. O baixo índice de aprendizagem dos estudantes nas disciplinas cujos requisitos exigem o conhecimento de programação de computadores tem sido um grande problema enfrentado em muitas instituições. Como o aprendizado de programação ocorre durante todo o curso, o aluno leva adiante as suas dificuldades, uma vez que geralmente estes não estão capacitados a utilizar as suas habilidades metacognitivas para identificar as próprias lacunas de aprendizagem. Pesquisas têm demonstrado que atividades metacognitivas produzem melhores resultados no que se refere à realização escolar. Este artigo apresenta os recursos avançados de exibição de resultados gerados pela ferramenta AMLP (Assistente Metacognitivo para Lógica de Programação) que visam a promoção e o aprimoramento das habilidades metacognitivas contribuindo positivamente para a melhoria da aprendizagem.*

**Palavras chave:** Atividades Cognitivas, Metacognição, Ambientes Virtuais de Aprendizagem

### 1.Introdução

A aprendizagem de linguagens de programação é essencial para todas as carreiras ligadas à Informática, principalmente para a formação daqueles que terão no desenvolvimento de softwares o produto final do seu trabalho. Dentre os principais problemas apresentados pelos aprendizes de programação destacam-se as dificuldades na compreensão dos enunciados dos problemas e na concepção dos algoritmos, incapacidade de detectar erros de lógica de programação, erros de sintaxe e semântica, dentre outros (SHRAW et al., 1994). Todos estes fatores levam o aluno de Programação, principalmente os de disciplinas introdutórias, a sentir a necessidade de um acompanhamento personalizado que o professor nem sempre disponibiliza e que os métodos de

ensino tradicionais nem sempre dão resposta, pois não se consegue identificar o problema localizado de cada aprendiz (PIMENTEL et al., 2004).

Até meados da década de 70, as investigações no âmbito da aprendizagem centravam-se nas capacidades cognitivas e nos fatores motivacionais como os principais determinantes da realização escolar. Desde então, uma terceira categoria de variáveis tem sido extensivamente estudada: a dos processos metacognitivos (RIBEIRO, 2003). Flavell (1976) define metacognição como a faculdade de planificar, de dirigir a compreensão e de avaliar o que foi aprendido. Estas características também estão presentes na atividade de programação de computadores e sugere que o desenvolvimento e aprimoramento dos processos metacognitivos podem contribuir para a melhoria do aprendizado neste contexto.

Este trabalho apresenta uma ferramenta computacional denominada de Assistente Metacognitivo para Lógica de Programação – AMLP, e os seus recursos de visualização de resultados que visam possibilitar ao aluno e professor uma melhor compreensão dos dados gerados pelo software, como o objetivo de promover as habilidades metacognitivas em aprendizes de lógica de programação, fornecendo dados para o automonitoramento do aluno em relação ao seu desempenho e contribuindo positivamente para o desenvolvimento dos processos metacognitivos.

O artigo está organizado como segue. A seção 2 apresenta os princípios de metacognição e automonitoramento da aprendizagem além de descrever alguns modelos metacognitivos. Na seção 3 discute-se sobre as medidas metacognitivas destacando-se o KMA e o KMB. A seção 4 apresenta a ferramenta AMLP destacando os recursos de visualização de resultados par aprimoramentos dos processos metacognitivos. A seção 5 conclui o trabalho com algumas considerações acerca do trabalho e aprofundamentos necessários.

## 2. Metacognição e o auto-monitoramento da aprendizagem

Conhecer o que se sabe e o que não se sabe é o primeiro passo para melhorar a aprendizagem. Este conhecimento sobre o próprio Nível de Aquisição de Conhecimento (NAC), num determinado domínio é possível através do monitoramento do conhecimento, que é um dos aspectos da Metacognição (PIMENTEL et al., 2004).

O termo Metacognição foi utilizado formalmente pela primeira vez por John Flavell (1976), da Universidade de Stanford que influenciado pelo trabalho de Jean Piaget realizou pesquisas sobre habilidade do indivíduo em administrar e monitorar a entrada, armazenagem, busca e recuperação dos conteúdos de sua memória (metamemória) e concluiu que :

*“(...) em qualquer tipo de processo cognitivo em ambiente humano ou não humano, uma variedade de atividades de processamento de informações podem acontecer (...). Metacognição se refere, entre outras coisas, ao monitoramento ativo e conseqüente regulação e orquestração destes processos, normalmente no serviço de alguma meta concreta ou objetiva”. (FLAVELL, 1976, p. 232 apud GAMA, 2004).*

Apesar do termo ter sido introduzido por Flavell (1976), desde o início do século, pedagogos e psicólogos demonstraram estar conscientes de que o estudo e a leitura envolvem um tipo de atividade agora denominada metacognitiva. Por exemplo, nas idéias de Thorndike encontram-se aspectos relacionados com o que é atualmente considerado metacognição. Segundo o autor, ler um parágrafo é resolver um problema, pois consiste na seleção dos elementos certos da situação e a sua colocação nas relações certas (RIBEIRO, 2003). O sujeito deve levar a cabo determinadas atividades, como por exemplo: selecionar, dominar, enfatizar, correlacionar e organizar, sob a orientação de um objetivo ou exigência/ requisito. Cavanaugh e Perlmutter (1982 *apud* RIBEIRO,

2003) referem-se ainda ao trabalho de Baldwin (1909), autor que utilizava os questionários introspectivos para examinar as estratégias de estudo, como o precursor do auto-conhecimento dos processos de pensamento e resultados das realizações.

Contudo, apesar da contribuição de muitos teóricos e investigadores para a compreensão da metacognição e da sua relação com a aprendizagem (LEFEBVRE-PINARD & PINARD, 1985 *apud* RIBEIRO, 2003; PRESSLEY, 1986), ocupam lugar de destaque, nesta área, os trabalhos desenvolvidos por John Flavell, Ann Brown e colaboradores.

Uma definição apresentada por Ann Brown, pesquisadora que tem realizado muitos trabalhos sobre os processos cognitivos de compreensão de leitura, exprime:

*“Metacognição refere-se ao entendimento do conhecimento, um entendimento que pode ser ou em um uso efetivo ou em uma descrição explícita do conhecimento em questão” (BROWN, 1987, p. 65 apud GAMA, 2004).*

A definição de Brown chama atenção para um aspecto importante da metacognição: a consciência que cada um possui do seu próprio conhecimento ou entendimento do conhecimento. Pode ser dito que um estudante entende uma atividade cognitiva particular se lhe for permitido usá-la apropriadamente e se for possível discutir sobre o seu uso. Mas há diferentes realidades nos passos para o entendimento, como não é incomum para estudantes usarem o conhecimento efetivamente sem estarem aptos para explicar como eles o fizeram. Para responder por esta variabilidade, o termo “pessoa metacognitiva” é usado freqüentemente para representar uma pessoa que está atenta ao seu próprio conhecimento (bem como às suas limitações).

## 2.1 Modelos Metacognitivos e os Ambientes de Aprendizagem

Existem vários modelos propostos derivados da conceituação da metacognição. Alguns são mais gerais e provêm uma estrutura teórica para metacognição, como os modelos de Flavell e Brown; outros se concentram em especificar os aspectos da metacognição como o conhecimento da estrutura de linguagens por Tunmer e Bowey (1984), ou os processos de memória e metamemória de SHNEIDER (*apud* GAMA, 2004). Tobias e Everson (2002) propõem um modelo modular em que há uma interdependência e uma hierarquia de relacionamento entre as habilidades metacognitivas e Gama (2004) formulou um módulo denominado RA (*Reflection Assistant*) cujo propósito é incrementar o foco nas habilidades metacognitivas utilizando o modelo de Tobias e Everson.

A incorporação nos ambientes interativos de aprendizagem de mecanismos que promovam o desenvolvimento de habilidades metacognitivas é fundamental. Isto permitirá ao aprendiz, definir seus objetivos e estruturar melhor a sua forma de aprender (DERRY, 1992). Estes ambientes devem propiciar também o monitoramento dos aspectos metacognitivos, ou seja, se estão sendo usados ou não, e se estão ou não sendo aprimorados.

Um modelo para o uso de Metacognição e Reflexão em Sistemas Tutores Inteligentes com o objetivo de aumentar a consciência e melhorar a aprendizagem é apresentado num estudo de Gama (GAMA, 2000). Para testar este modelo, Gama desenvolveu o MIRA (Metacognitive Instruction with a Reflective Approach), um software educacional que apresenta problemas de álgebra e oferece recursos para elaboração de equações lineares. As ações dos estudantes são monitoradas ajudando-os a refletir sobre seu próprio aprendizado e características. Isto incentiva o desenvolvimento metacognitivo do aluno (GAMA, 2004).

Assim como o MIRA, o protótipo CoNoteS2, desenvolvido por Hadwin e Winne (2001), visa estimular a auto-regulação do aprendiz através de uma ferramenta computacional, com a

diferença de não se centrar em uma única disciplina ou material didático. A interface pode ser adaptada para qualquer curso, pois podem ser incorporados textos eletrônicos e guias de estudante na área de trabalho do software juntamente com as atividades de estudo.

Um modelo de referência para melhorar a aprendizagem por exemplos, com o suporte de auto-explicação é apresentado por Conati e Vanhlen (2000). Auto-explicação é o processo de gerar explicações e justificativas para si mesmo, quando se estuda um exemplo. Muitos estudos têm mostrado que estudantes que se “auto-explicam” aprendem mais.

Já o uso de metacognição no domínio de programação é ainda uma área pouco explorada. O trabalho de Shaft (1995), investiga o uso de metacognição por programadores na escolha de uma estratégia para a compreensão dos programas. Os resultados deste estudo mostram que os programadores usam metacognição quando estão estudando programas de computador. Os resultados sugerem também que o uso de metacognição influencia o quanto bem eles compreendem os programas.

### 3. Avaliação e Medidas Metacognitivas

O desenvolvimento de métodos eficientes para avaliar metacognição tem sido uma tarefa difícil. Processos metacognitivos são geralmente avaliados fazendo inferências das observações dos estudantes, através de entrevistas ou até mesmo de auto-avaliações.

Tobias e Everson (2002) criaram um índice denominado KMA (Knowledge Monitoring Accuracy) para medir e avaliar a diferença entre as estimativas do aprendiz sobre seu conhecimento num domínio particular e seu conhecimento atual determinado por desempenho em avaliações. Gama (2004) incrementou a maneira de calcular o KMA adicionando a possibilidade de prever e desempenhar também a resolução parcial de problemas. Além disso, Gama criou o índice KMB (Knowledge Monitoring Bias), para medir e identificar o tipo de desvio do aprendiz no monitoramento do seu conhecimento. Isto permite identificar, por exemplo, que um aprendiz com KMA baixo é pessimista – prevê que vai errar e acerta, ou otimista – prevê que vai acertar e erra. Este tipo de informação permite um feedback mais adequado para cada aprendiz.

. Osborne (*apud* GAMA, 2004) conclui que o KMA parece ser uma medida naturalista e robusta de monitoramento do conhecimento que possui uma boa confiabilidade e excelente validade interna . Comparando as medidas gerais de metacognição, metamemória e metacompreensão destinadas às propriedades psicométricas e a suas conveniências para uso pedagógico o KMA vem no topo da lista.

Da mesma forma, Tobias e Everson (2002) efetuaram estudos que investigavam a relação entre o KMA e outras medidas metacognitivas de auto-informação. Eles o compararam às medidas “*Learning And Study Strategies Inventory – LASSI*” (Inventário de Estratégias de Estudo e Aprendizagem) (WEINSTEIN et al., 1987 *apud* TOBIAS e EVERSON, 2002), ao “*Motivated Strategies for Learning Questionnaire – MSLQ*” (Estratégias Motivadas ao Questionário de Aprendizagem) (PINTRICH et al., 1991 *apud* TOBIAS e EVERSON, 2002), e ao “*Metacognitive Awareness Inventory – MAI*” (Inventário de Consciência Metacognitiva) (SHRAW e DENISON, 1994). Uma vantagem que se pode observar no KMA sobre esses instrumentos de auto-informação é que ele possui uma probabilidade menor de ser afetado pelas dificuldades de assunto de recordação envolvidas nas respostas dos questionários metacognitivos. Isto acontece porque não é requisitado aos estudantes que informem os processos metacognitivos usados enquanto os trabalhos são efetuados.

Segundo Tobias e Everson (2002), a pesquisa sugere que o KMA, comparado com escalas para auto-informação de testes de ansiedade, é menos sensível às tendências dos estudantes em aspectos favoráveis. Além disso, baixos relacionamentos foram encontrados entre esses instrumentos e o KMA, mostrando que esses auto-informativos tendem a se correlacionar um com o outro, indicando que as características das medidas usadas nos métodos são similares. Tobias e Everson (2002) notaram que a maioria dos estudos se correlacionavam entre o KMA e as medidas de aptidão escolar, e a tendência da aprendizagem seria maior que aqueles instrumentos de auto-avaliação. A conclusão deles é que o KMA avalia os componentes metacognitivos que estão mais relacionados à aprendizagem em sala de aula, do que a medida de auto-informação. Em outra recente pesquisa de revisão de diferentes instrumentos de avaliação metacognitivos existentes PINTRICH et al.(2000) confirma que as contagens efetuadas com este instrumento têm relação global mais alta com os resultados de aprendizagem.

Com base nos estudos de Tobias e Everson, Gama (2004) adaptou o instrumento KMA reinterpretando o significado da fórmula em pontuações individuais que evoluem a cada ciclo de predição/desempenho. Cada pontuação, que ela denomina de *KMA*, corresponde à precisão da predição naquele ciclo particular. Como há dois possíveis valores tanto para a avaliação quanto ao desempenho o KMA tem que ser definido para um total de quatro casos. A fórmula de Tobias & Everson é obtida dando ao *KMA* os valores expostos na Tabela 3.1.

Desempenho	Estimativa	
	resolverá	não resolverá
Correto	1	- 1
Não Correto	- 1	1

**Tabela 5.1:** Valores dados por Gama na reinterpretação do cálculo do KMA de Tobias e Everson

Quando a avaliação estiver correta, o *KMA* é 1 (um), ou seja, em outras palavras predisse que não poderia resolver o problema e realmente não o resolveu corretamente, ou predisse que poderia resolver o problema e o resolveu corretamente e -1 (menos um) quando a avaliação está incorreta, ou seja, em outras palavras predisse que era capaz de resolver o problema, mas não o resolveu corretamente, ou predisse que não poderia resolvê-lo, mas o resolveu corretamente.

Pode-se notar que na fórmula proposta por Tobias e Everson (2002) para o KMA, a predição e o desempenho são binários e, conseqüentemente, a variável KMA também ficou com somente dois valores. Para aumentar a flexibilidade e poder diferenciar entre os estudantes que não resolveram nada do problema corretamente daqueles que apresentaram um caminho parcial da solução correta, foi efetuada uma mudança para permitir um terceiro valor intermediário entre a predição e o desempenho. Essa alteração é representada na Tabela 3.2.

O Knowledge Monitoring Bias (KMB) é uma medida que foi criada por Gama (2004) para completar a avaliação efetuada pelo KMA, pois o mesmo não provê uma contagem detalhada sobre os tipos de inexatidões que o estudante pode apresentar. Por exemplo, um estudante que recebeu a atribuição de ter um perfil de baixo KMA porque ele constantemente está predizendo que resolve os problemas corretamente, mas as suas soluções estão invariavelmente erradas, e de outro estudante que tende a calcular que ele não resolverá os problemas completamente correto, mas na maioria das vezes, ele alcança uma solução correta.

<b>Predição →</b> ↓ <b>Performance</b>	<b>estimado que não resolveria</b>	<b>estimado que resolveria parcialmente</b>	<b>estimado que resolveria</b>
<b>demonstrado não correto</b>	1	-0,5	- 1
<b>demonstrado parcialmente correto</b>	- 0,5	1	- 0,5
<b>demonstrado correto</b>	- 1	- 0,5	1

**Tabela 3.2:** Matriz de valores para Knowledge Monitoring Accuracy (KMA)

O KMB leva em conta o desvio que pode ter em uma avaliação precisa do monitoramento do seu conhecimento. Se não houver nenhuma divergência, diz-se que o estudante é preciso na avaliação do seu conhecimento ou realístico. Caso contrário, três situações são possíveis: (i) o estudante freqüentemente prediz que resolverá os problemas, mas não tem sucesso, demonstrando que fez uma avaliação otimista do seu conhecimento; (ii) o estudante prediz freqüentemente que ele não resolverá os problemas, entretanto ele tem sucesso em resolvê-los, demonstrando assim que é pessimista ao tentar avaliar seus conhecimentos; e (iii) ele, às vezes, é tanto otimista na avaliação do seu conhecimento quanto pessimista, em um padrão aleatório (GAMA, 2004).

A pontuação do KMB depende do KMA demonstrado. Assim, a possível faixa de valores para o KMB também está entre -1 e 1. As possíveis contagens de KMB têm um significado diferente do KMA. As estimativas precisas recebem a contagem 0, significando uma inclinação nula; 1 é designado à estimativa otimista mais alta; igualmente é designado -1 à estimativa pessimista mais alta; são nomeadas como -0.5 as estimativas pessimistas medianas e +0.5 as estimativas otimistas medianas. A Tabela 3.3 ilustra o descrito.

As medidas KMA e KMB, com seus respectivos significados, podem ser melhor visualizados nas Tabelas 3.4 e 3.5, respectivamente

<b>Predição →</b> ↓ <b>Performance</b>	<b>estimado que não resolveria</b>	<b>estimado que resolveria parcialmente</b>	<b>estimado que resolveria</b>
<b>demonstrado não correto</b>	0	0,5	1
<b>Demonstrado parcialmente correto</b>	- 0,5	0	0,5
<b>demonstrado correto</b>	-1	-0,5	0

**Tabela 3.3:** Matriz de valores para Knowledge Monitoring Bias (KMB) de Gama

<b>Índice KMA</b>	<b>Classificação</b>	<b>Interpretação sobre a estimativa do aprendiz</b>
[-1.00 a -0.25]	KMA baixo	Não estima corretamente seu conhecimento na maioria das situações
[-0.25 a +0.50]	KMA médio	Algumas vezes estima corretamente seu conhecimento mas comete, com freqüência, erros médios ou erros grandes na sua estimativa.
[+0.50 a +1.00]	KMA alto	Na maioria das vezes estima corretamente o seu conhecimento
<b>Fórmula KMA = { [AP * 1.00] + [(EMO + EMP) x - 0.50] + [(EGO + EGP) * -1.00] } / QP</b>		

**Tabela 3.4:** Classificação, interpretação e fórmula do KMA

Índice KMB	Classificação	Interpretação sobre a estimativa do aprendiz
KMA alto	Realista	Estima com precisão o seu conhecimento tendo um alto KMA
] 0.25 a 1.00]	Otimista	Estima que pode resolver o problema mas não o faz em muitas situações
] -1.00 a -0.25]	Pessimista	Estima que <i>não</i> pode resolver o problema mas o faz, em muitas situações
] -0.25 a +0.25]	Aleatório	Estima seu conhecimento ora de modo otimista e outras pessimista
Fórmula KMB = { [EMO * 0.50] + [EMP * -0.50] + [EGO * 1.00] + [EGP * -1.00] } / QP		

**Tabela 3.5:** Classificação, interpretação e fórmula do KMB

#### 4. Um Assistente Metacognitivo para Lógica de Programação

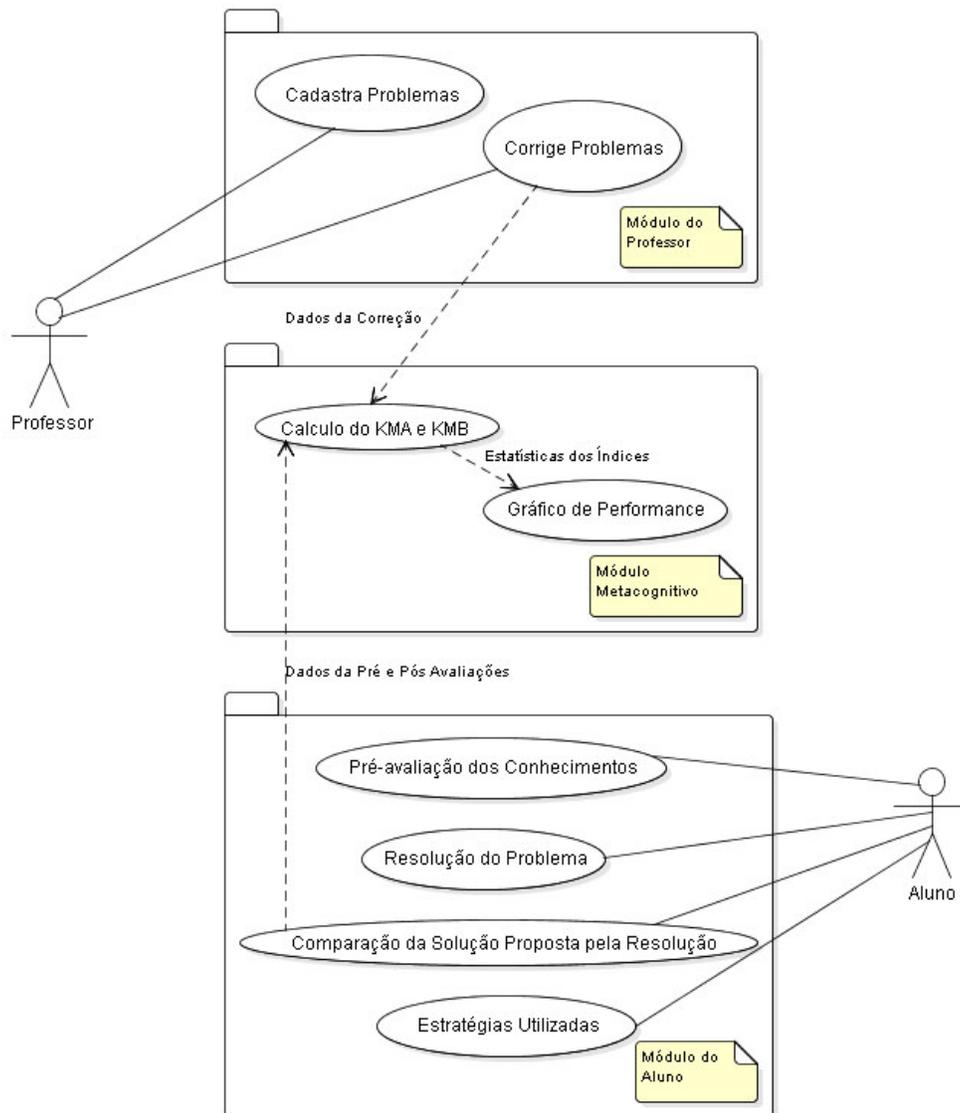
O Assistente Metacognitivo para Lógica de Programação (AMLPL) é um ambiente de avaliação avaliação interativo, que fornece dados para o automonitoramento do aluno em relação ao seu desempenho e contribui positivamente para o desenvolvimento dos processos metacognitivos.

A arquitetura lógica, que dá suporte ao auto-monitoramento e acompanhamento contínuo da aprendizagem, é composta por três módulos distintos denominados Aluno, Professor e Metacognitivo (ABREU et al., 2005), conforme o diagrama de casos de uso da figura 4.1. Estes módulos são compostos de várias etapas que visam coletar dados suficientes para que possa ser efetuado o cálculo dos índices KMA e KMB e exibido o resultado em forma de gráfico. Entre as etapas, destacam-se as que seguintes:

1. criação do problema pelo professor a ser resolvido pelo aluno – nesta etapa são incluídos os problemas que deseja-se que os aprendizes resolvam. Além de ser incluído o enunciado do problema e uma possível resolução para o mesmo, o problema é classificado por tipo de complexidade (fácil, médio e difícil);
2. pré-avaliação do aluno sobre a solução do problema – após efetuar o log-in no software, o aluno é direcionado automaticamente para um problema que o mesmo ainda não resolveu. Junto com o enunciado deste problema, são apresentadas três opções, mutuamente exclusivas, onde a pessoa indica se é capaz de resolver o problema parcialmente, completamente ou senão é capaz de resolvê-lo;
3. verificação da resolução do aluno pelo professor – o responsável em aplicar o teste acessa os problemas que já foram pré-avaliados e efetuados pelo aluno, através da ferramenta que recupera automaticamente os exercícios já encerrados, e verifica se a solução proposta pelo aprendiz está correta, errada ou parcialmente correta, optando pela a opção que é apresentada na tela.
4. geração dos gráficos com os resultados da análise metacognitiva – com base nos dados informados na pré-avaliação do aluno e com os dados passados pelo professor no momento da verificação da solução proposta, a ferramenta gera dois gráficos de barras tridimensionais, onde um indica a análise metacognitiva por problemas e o outro apresenta o cálculo do KMA e KMB por tipo de problema resolvido (fácil, médio e difícil).

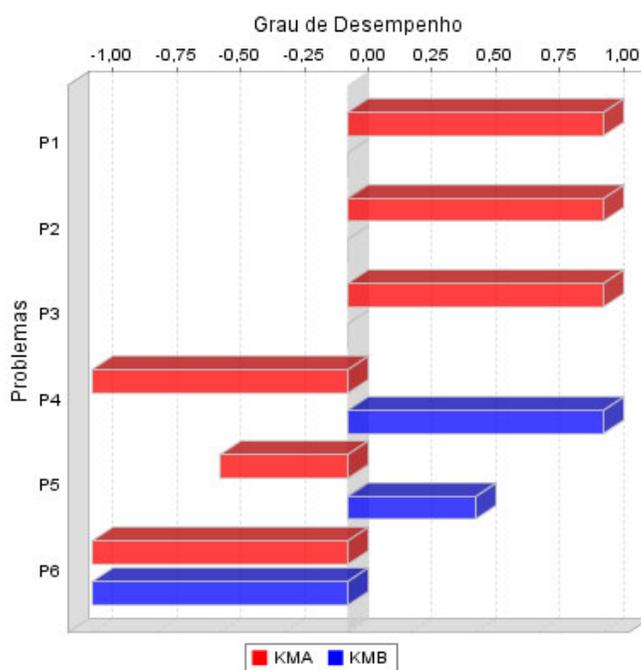
##### 4.1 Recursos para Visualização dos Resultados

A base para o aperfeiçoamento das habilidades metacognitivas é a visualização e a reflexão sobre o perfil metacognitivo atual e o monitoramento da evolução. Neste sentido, os recursos gráficos de visualização do AMLPL desempenham papel fundamental.



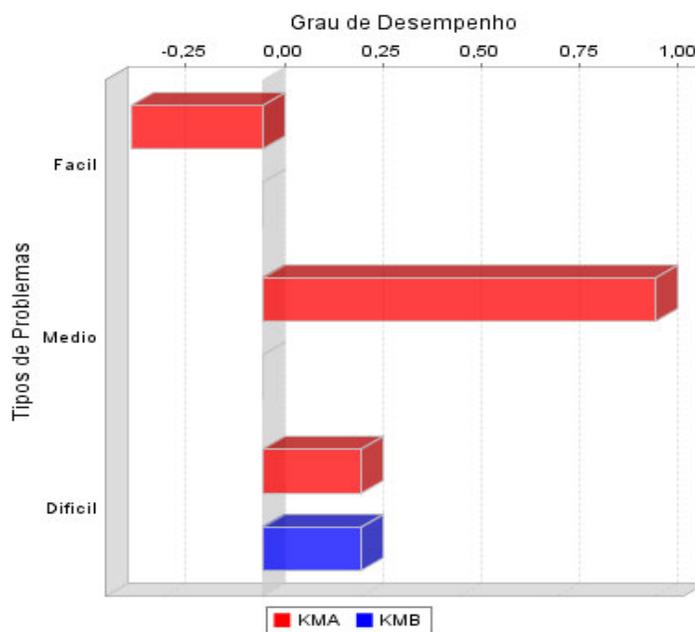
**Figura 4.1:** Diagrama de caso de uso do relacionamento entre os módulos

Os gráficos são montados no momento em que o aluno efetua log-in na ferramenta de acordo com a quantidade de problemas que o mesmo efetuou até o momento. O gráfico apresentado na figura 4.2 demonstra um exemplo em que o aluno efetuou seis problemas, de complexidades diversas, e em cada um deles obteve um índice distinto de KMA e KMB. No eixo das abscissas tem-se o grau de desempenho obtido enquanto no eixo das ordenadas tem-se a identificação dos problemas resolvidos pelo aluno. Pode-se notar que nos três primeiros problemas o aluno atingiu o nível máximo dos índices, obtendo KMA alto - que indica que ele diz que pode resolver o problema e resolve - e sem nenhuma variação, ou seja, sem KMB, que é utilizado para indicar se a pessoa é otimista ou pessimista em suas auto-avaliações. Observando os demais problemas apresentados no gráfico verifica-se que existe a presença de variação nos acertos. No problema P4 o aluno, em sua pré-avaliação, disse que erraria o problema, mas após a correção pelo professor, ficou constatado que ele acertou, obtendo assim um baixo valor para o índice KMA. Conseqüentemente, pontuou o valor máximo para o índice KMB, pois ao invés de errar o exercício, como predisse, o acertou, causando uma variação extrema entre a auto-avaliação e o constatado.



**Figura 4.2:** Gráfico do desempenho obtido em cada problema

A possibilidade de visualizar a evolução das medidas metacognitivas permite ao aprendiz verificar se estas estão ou não sendo aperfeiçoadas e tomar medidas no sentido de, por exemplo, refletir melhor sobre os seus conhecimentos antes de registrar a sua previsão de desempenho. A figura 4.2 exibe um resultado mais sintetizado para que o aluno possa ter uma visão mais global do seu desempenho, por nível de dificuldade dos problemas.

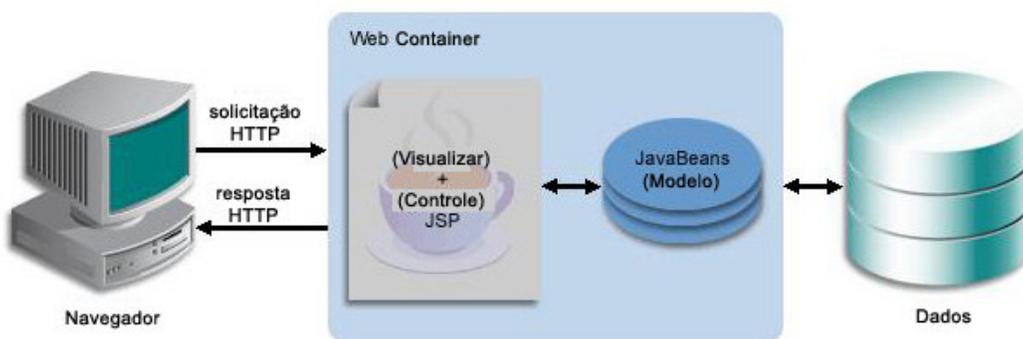


**Figura 4.3:** Gráfico do desempenho obtido por tipo de problema

## 4.2 Arquitetura dos Recursos Tecnológicos

A ferramenta foi desenvolvida com base na plataforma *Java 2 Enterprise Edition* (J2EE), utilizando o modelo de três camadas *Model View Controller* (MVC). A Plataforma J2EE é um conjunto de especificações coordenadas e um guia de práticas que juntos permitem o desenvolvimento, instalação, execução e gerenciamento de aplicações n-camadas no servidor. A plataforma J2EE, que expande a *Plataforma Java 2 Standard Edition* (J2SE), é uma plataforma completa, robusta, estável, segura e de alta performance, voltada para o desenvolvimento de soluções corporativas. O principal objetivo da Plataforma J2EE é reduzir a complexidade e o tempo do desenvolvimento de aplicações corporativas.

O modelo de três camadas físicas divide um aplicativo de modo que a lógica de negócio resida no meio das três camadas físicas. Isto é chamado de camada física intermediária ou camada física de negócios. A maior parte do código escrito reside na camada de apresentação e de negócio. A arquitetura MVC - (Modelo Visualização Controle) fornece uma maneira de dividir a funcionalidade envolvida na manutenção e apresentação dos dados de uma aplicação, conforme figura 4.4.



**Figura 4.4:** Modelo I do padrão MVC

O componente Modelo mantém o estado persistente do negócio e fornece ao controlador a capacidade de acessar as funcionalidades da aplicação encapsuladas pelo próprio modelo. Um componente de Visualização renderiza o conteúdo de uma parte particular do modelo e encaminha para o controlador as ações do usuário; acessa também os dados do modelo via controlador e define como esses dados devem ser apresentados.

Um controlador define o comportamento da aplicação e é ele que interpreta as ações do usuário e as mapeia para chamadas do modelo. Em um cliente de aplicações Web essas ações do usuário poderiam ser cliques de botões ou seleções de menus. As ações realizadas pelo modelo incluem ativar processos de negócio ou alterar o estado do modelo. Com base na ação do usuário e no resultado do processamento do modelo, o controlador seleciona uma visualização a ser exibida como parte da resposta a solicitação do usuário. Há normalmente um controlador para cada conjunto de funcionalidades relacionadas.

Utilizando os conceitos descritos optou-se, como padrão para a camada de visualização e controle, a tecnologia *Java Server Pages* (JSP), que é orientada à criação de páginas Web com programação em Java. Com JSP podemos criar aplicações que podem ser executadas em vários servidores Web, de múltiplas plataformas, já que Java é em essência uma linguagem multiplataforma. As páginas JSP são compostas de código HTML/XML misturado com tags especiais para programar scripts de servidor em sintaxe Java.

Para a camada Modelo foram utilizados os componentes de software JavaBeans que são projetados para serem unidades reutilizáveis, que uma vez criados podem ser reusados sem modificação de código, e em qualquer propósito de aplicação, seja um applet, um servlet ou qualquer outra.

## 7. Conclusões e aprofundamentos necessários

A incorporação de ferramentas de auto-monitoramento da aprendizagem em Ambientes Interativos de Aprendizagem e/ou Sistemas Tutores Inteligentes certamente cria melhores condições para que as pessoas tenham o controle da própria aprendizagem, ingrediente essencial tanto para o dia-a-dia da educação formal quanto para a aprendizagem continuada ao longo da vida.

O domínio de conhecimento de “Lógica de Programação”, em cursos de graduação em Informática, pela dificuldade de aprendizagem dos alunos, visível nos altos índices de reprovação, tem demonstrado ser um terreno fértil para pesquisas que busquem o aumento dos padrões de aprendizado. Nesta área, o monitoramento dos conhecimentos prévios, um processo metacognitivo fundamental, parece ser ainda mais indicado.

Os resultados da pesquisa de Gama (2004) demonstram que os índices KMA e KMB podem ser úteis para o auto-monitoramento do aluno em relação ao seu desempenho. A incorporação destes mecanismos em um ambiente interativo como AMLP podem contribuir para a agilização do desenvolvimento de habilidades metacognitivas.

Como aprofundamentos necessários, do ponto de vista de estudo de caso, pretende-se intensificar a aplicação de testes, utilizando diferentes categorias de problemas de programação, que permitam a um grupo de alunos desenvolver habilidades metacognitivas através de retorno contínuo. Do ponto de vista técnico da ferramenta, pretende-se automatizar o processo de seleção de problemas a serem apresentados pela ferramenta utilizando propriedades como o NAC, o grau de dificuldade e os estados metacognitivos atuais do aluno (KMA e KMB) apresentando, por exemplo, problemas mais simples ou mais complexos. Outra melhoria é a integração da ferramenta a um Sistema de Tutoria Inteligente juntamente com a inclusão de um novo módulo, o Domínio de Conhecimento, que seria útil para confrontar com os problemas incluídos e assim deixar claro quais os conceitos do domínio estão sendo avaliados.

## Referências

- Abreu, C. O., Pimentel E. P., Batistela P., Omar N. (2005). Um ambiente de avaliação para o desenvolvimento de habilidades metacognitivas em lógica de programação. GCETE'2005 – Global Congress on Engineering and Technology Education.
- Bransford, John D. et al. (2002). How People Learn – Brain, Mind, Experience, and School. Washington : National Academy Press.
- Conati, C and Vanlehn, K. (2000). Toward computer-based support of metacognitive skills: a computational framework to coach self-explanation. International Journal of Artificial Intelligence in Education.
- Derry, S. J. (1992). Metacognitive models of learning and instructional systems design. In M. Jones and P. H. Winne (eds.), Adaptive Learning Environments - Foundations and Frontiers.

- Flavell, J. H. (1976). Metacognitive Aspects of Problem Solving. In L.B. Resnick (ed.), *The Nature of Intelligence*, Hillsdale, NJ: Lawrence Erlbaum Associates.
- Gama, C.(2004). Towards a model of Metacognition Instruction in Interactive Learning Environments. Tese de Doutorado. University of Sussex, Inglaterra, 2004 (em fase de Conclusão).
- Gama, Claudia. Metacognitive Awareness: a Pilot Study in a Software Design Course; School Of Cognitive and Computing Science, University of Sussex.
- Hadwin, A.F. & Winne, P. H.(2001). CoNoteS2: A software Tool for Promoting Self-Regulation. *Educational Research and Evolution*, 7(2-3), 313-334.
- Pimentel, E. P. & Omar, N & França, V F. (2004). A caminho de um modelo para a incorporação de Automonitoramento da Aprendizagem em STI. In: X WIE – Workshop de Informática e Educação da Sociedade Brasileira de Computação, Salvador, BA.
- Pintrich, P. R, Wolters, C. A., and Baxter, G. (2000). Assessing metacognition and self-regulated learning. In G. Schraw and J. Impara (eds.), *Issues in the measurement of metacognition*, pp. 43–97. The University of Nebraska Press, Lincoln, NE: Buros Institute of Mental Measurements.
- Pressley, M. (1986). The relevance of good strategy user model to the teaching of mathematics. *Educational Psychologist*.
- Ribeiro, C. (2003). Metacognição: Um apoio ao Processo de Aprendizagem, *Psicologia: Reflexão e Crítica*, 16(1), pp.109-116.
- Rocha, Helena. V. (1991). Representações Computacionais Auxiliares ao Entendimento de Conceitos de Programação. Unicamp.
- Schraw, G & Dennison, R. S. (1994). Assessing Metacognitive Awareness. *Contemporary Educational Psychology*.
- Shaft, T. M.(1995). Helping Programmers Understand Computer Programs: The Use of Metacognition. *DATABASE Advances*, 26(4), 25-46.
- Shang, Y., Shi, H., and Chen, S.-S (2001). An Intelligent Distributed Environment for Active Learning,” *ACM Journal of Educational Resources in Computing*, Vol. 1.
- Tobias, S.; Everson, H. T. (2002). Knowing what you know and what you don't: further research on metacognitive knowledge monitoring. College Board Research Report 2002-3, College Entrance Examination Board: New York.