

Uso de tecnologia adaptativa para implementação de sistema de aprendizagem de algoritmos baseado na plataforma Google Android

G. C. Farto

Abstract — The purpose of this article is to present the development of an algorithms learning system based on Google Android platform, in order to provide a new resource of teaching and learning for students of IT courses. Concepts of adaptive technology were used to make the application more dynamic.

Keywords – sistema de aprendizagem; Mobile Learning; algoritmos; tecnologia adaptativa; Java; Google Android

I. INTRODUÇÃO

¹As disciplinas de Algoritmos e Lógica de Programação, geralmente lecionadas nas primeiras etapas dos cursos de Tecnologia da Informação (TI), são consideradas desafiadoras por grande parte dos alunos, pois exige a formulação e desenvolvimento de soluções de problemas utilizando-se os conceitos base de matemática e lógica.

Por meio de um levantamento realizado em parceria com a Fundação Educacional do Município de Assis (FEMA) e Instituto Municipal de Ensino Superior de Assis (IMESA), obteve-se indicadores sobre a quantidade de alunos aprovados e reprovados de cursos de tecnologia de 2009, 2010 e 2011.

O gráfico ilustrado na Figura 1 apresenta as estatísticas de alunos aprovados, reprovados com exame, reprovados por frequência e reprovados sem exame que frequentavam a disciplina de Algoritmos e Estruturas de Dados I dos cursos de Análise e Desenvolvimento de Sistemas (ADS) e Tecnologia em Processamento de Dados (TPD) no período analisado.

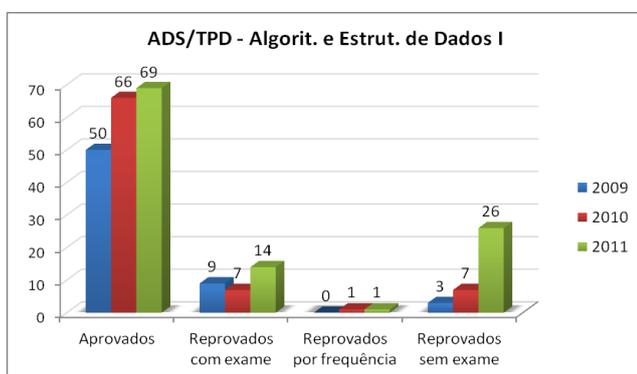


Figura 1. Gráfico de situações para os cursos de ADS e TPD

Apesar de ser em quantidade menor, o gráfico ilustrado na Figura 2 apresenta as estatísticas de alunos reprovados que frequentavam a disciplina de Algoritmos e Estruturas de Dados I do curso de Bacharelado em Ciências da Computação (BCC) no período analisado.

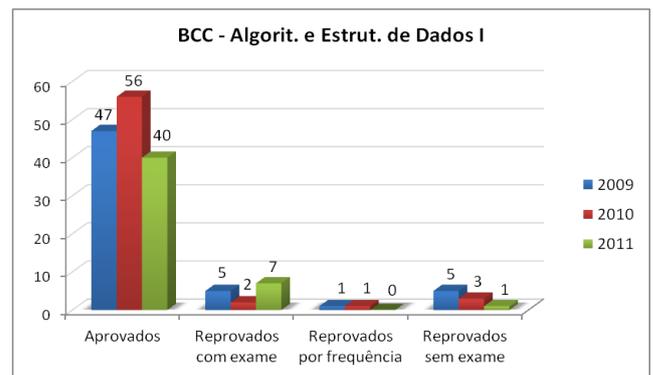


Figura 2. Gráfico de situações para o curso de BCC

Pode-se observar que o índice de reprovações é considerado alto, comparado a outras disciplinas do mesmo curso, entretanto reflete uma situação real da disciplina que é a tendência a dificuldades de aprendizado.

Na busca de soluções para esse problema recorrente nos cursos de tecnologia, diversas pesquisas enfocam tempo e recursos necessários em abstração, projeto e construção de ferramentas computacionais capazes de auxiliar o aluno nas fases iniciais de aprendizagem de algoritmos e lógica.

O principal desafio no processo de ensino de algoritmos e lógica se deve ao fato de que, na maioria das vezes, o conteúdo é aplicado a grupos heterogêneos de participantes, cada qual com seus talentos, modo de trabalhar e pensar, assim como métodos de aprendizagem diferentes. Uma alternativa para solucionar essa dificuldade está na possibilidade do próprio aluno gerir e conduzir boa parte da evolução de seu aprendizado.

Esse modo de aprender pode ser instituído por meio de um sistema de aprendizagem interativo, onde o participante, além de obter conteúdos e instruções de determinado assunto, é capaz de interagir com a aplicação, propondo soluções para resolver um determinado problema apresentado durante o processo de aprendizado.

A partir dessa ideia, os ambientes ou sistemas de aprendizagem interativos devem ser baseados em quatro princípios: o estudante deve construir conhecimento; o controle do sistema é feito, de forma mais significativa, pelo estudante; o sistema é individualizado para cada estudante; e o feedback é gerado em função da interação do estudante com o ambiente.

Outro assunto que motiva o tema proposto é o mercado de dispositivos móveis que cresce cada vez mais, correspondendo a um grande percentual da população mundial. Entre as plataformas disponíveis, destaca-se a do Google Android, por ser o primeiro ambiente de desenvolvimento de aplicativos móveis completamente livre e open-source, representando uma grande vantagem para sua evolução, uma vez que desenvolvedores podem contribuir com melhorias para a arquitetura e o sistema operacional.

Portanto, a tecnologia Google Android é um recurso muito importante e deve ser considerada nas pesquisas e sistemas de aprendizagem interativos, pois um aplicativo móvel permite ao aluno modelar e executar algoritmos tanto em aulas presenciais como a distância, assim como estar disponível a qualquer hora, local e dispositivo portátil, como um smartphone ou tablet, aumentando as oportunidades e maneiras de ser utilizado para fins de aprendizado.

Atualmente, há uma grande intensificação no desenvolvimento de metodologias de Ensino a Distância (EAD), descrevendo a interação entre professor e aluno durante o processo de ensino-aprendizagem. Porém, em sua grande maioria, a proposta do modelo de ensino é apenas a de transmitir, ao aprendiz, informações e conteúdos definidos pelo tutor por meio de lições pré-estabelecidas.

Sabendo-se da necessidade de metodologias de ensino a distância baseadas na geração de conteúdos em um sistema de aprendizagem, uma alternativa é fazer uso de conceitos da tecnologia adaptativa para realizar a implementação de aplicações mais dinâmicas, contribuindo com o aprendizado do aluno em uma determinada área de conhecimento.

Os resultados proporcionados pelo uso de tecnologia adaptativa são caracterizados por apresentar uma estrutura dinâmica, objetivando a automodificação provocada por interações com o meio externo, sendo ele real ou virtual. Esta habilidade de mudança comportamental é essencial para a construção de máquinas e programas de computadores capazes de evoluir e gerar novas situações com a própria experiência.

O principal diferencial da tecnologia adaptativa é tornar possível, de maneira razoavelmente simples, o reaproveitamento e a ampliação das capacidades de teorias e técnicas existentes e consolidadas, principalmente nas áreas de Inteligência Artificial e Teoria da Computação.

Este artigo aborda o desenvolvimento de um sistema de aprendizagem de algoritmos baseado na plataforma Google Android, com a finalidade de ser uma ferramenta de estudo para alunos das disciplinas de Algoritmos e Lógica de Programação.

Objetivando tornar a aplicação mais dinâmica, fez-se uso de conceitos da tecnologia adaptativa para possibilitar a geração dinâmica de problemas que desafiam o aluno a criar, a cada enunciado proposto, novas soluções computacionais, sem a necessidade de tal situação estar pré-estabelecida.

II. AS DIFICULDADES DE APRENDIZAGEM DE ALGORITMOS

Presente na literatura há diversos trabalhos que identificam justificativas para a dificuldade, quase que intrínseca, na tarefa de aprender os conceitos de lógica e programação.

Entre os argumentos propostos, deve-se considerar que o aprendizado de programação, assim como em outras áreas de conhecimento, é obtido por meio um processo lento e gradual, onde novos fundamentos e ideias, até então desconhecidos, devem ser transformados em algo familiar, facilitando a sua assimilação [1].

Ressalta-se, também, que uma das principais dificuldades reside na técnica de compreender e, em particular, aplicar as noções básicas, como estruturas de controle, para a criação de algoritmos capazes de resolver situações reais [2].

No âmbito de aprendizado de programação, uma possível causa de dificuldade se deve ao fato de que muitos dos alunos não apresentam interesse neste tipo de disciplina. Tal desmotivação pode ser explicada pelo grande volume de conhecimentos abstratos relacionados à atividade de programar, da mesma forma que muitas linguagens e tecnologias estão cada vez mais sofisticadas [3].

Devido às dificuldades enfrentadas pelos alunos, as disciplinas de Algoritmos e Lógica de Programação têm apresentado altos índices de desistência e reprovação. A evasão, além de distanciar o aluno da formação intelectual e profissional, incita a desconfiança sobre a qualidade dos cursos superiores, contribuindo por impedir a entrada de novos alunos, atrasando, conseqüentemente, o crescimento da área de computação [4].

Para obter sucesso no aprendizado de programação, deve-se realizar um treino intensivo em resolução de problemas e exigir uma precisão e atenção a detalhes muito mais elevada do que a requerida por grande parte de outras disciplinas [1], [5].

Além dos argumentos apresentados anteriormente, alguns trabalhos referem-se que, ao invés de haver uma dificuldade inerente ao aprendizado de algoritmos, há alunos que não possuem as aptidões necessárias para a área de programação, especificamente na resolução de problemas envolvendo lógica e matemática [6], [7], [8].

Entre outras, há várias causas para o insucesso em disciplinas de computação, como a dificuldade de abstração e compreensão, a falta de competências necessárias para resolver problemas, o uso inadequado de metodologias didáticas se comparado ao modelo de aprendizagem dos alunos, além de que as linguagens e tecnologias são detentoras de sintaxes complexas para aprendizes com pouca ou nenhuma experiência [9].

Muitos dos temas relacionados ao processo de aprendizagem são bastante questionáveis, já que o método de ensino abrange tanto alunos quanto professores, assim como as metodologias aplicadas na sala de aula e o certo grau de dificuldade inerente da área de tecnologia.

Há diversos outros motivos, de natureza didática, que podem ser considerados a origem da dificuldade de aprendizado, como o grande número de alunos por turma, dificuldade de o professor compreender a lógica formulada pelo aluno, níveis diferentes de experiência e ritmo entre os alunos, assim como a ausência de bons materiais para disciplinas introdutórias e a própria dificuldade da escolha do curso superior.

Alcançando um novo patamar, também podem ser analisados problemas de naturezas cognitivas, como ausência de perfil necessário para a resolução de problemas, e afetiva, como problemas de ordem pessoal que impedem a concentração durante explicações de conteúdo.

Analisando as dificuldades encontradas por alunos dos cursos de tecnologia e a fundamental importância dessa base inicial de conhecimento e considerando que os modos de pensar e aprender são pessoais e que não é possível nem viável ao tutor adequar-se às necessidades de cada aluno, tornam-se justificáveis a abstração, modelagem e implementação de um sistema de aprendizagem de algoritmos.

III. TECNOLOGIA ADAPTATIVA: ESTADO DA ARTE E APLICAÇÕES

A terminologia da palavra “adaptatividade” se aplica a diversos conceitos, porém, dentro do contexto de tecnologia e computação, refere-se à possibilidade de se automodificar, adaptando-se, de maneira instintiva, sem a necessidade de qualquer outro tipo de recurso.

A tecnologia adaptativa originou-se da notação científica de autômatos, onde a formalização é designada pelos conceitos de autômatos finitos, visto que a base das linguagens regulares é especificada por estados, transições e uma cadeia de entrada, definindo-se os conceitos e fundamentos.

O domínio da tecnologia adaptativa requer o conhecimento de três vertentes:

- Teoria: responsável por fornecer os fundamentos matemáticos;
- Ferramentas e ambientes: responsáveis por facilitar o desenvolvimento de aplicações adaptativas;
- Aplicações: responsáveis pela resolução eficiente de problemas das mais variadas áreas de interesse.

Entre os primeiros estudos realizados com tecnologia adaptativa, destaca-se o uso de conceitos de adaptatividade para a definição e construção de compiladores, aplicando mecanismos adaptativos na análise sintática e em geradores de reconhecimento sintático [10].

Devido ao avanço nas pesquisas, tornou-se possível a incorporação de funções de transdução sintática, apresentando um aperfeiçoamento de reconhecedores sintáticos, baseando-se nos autômatos de pilha estruturada [11]. Pouco tempo depois, por meio de um novo trabalho sobre transdutores adaptativos, definiu-se uma classe de máquinas de estados finitos armazenados em uma pilha e com memória organizada que demonstra a alteração dinâmica de sua configuração, de acordo com o aprendizado realizado nas transições, acrescentando um poder maior na representação dos modelos matemáticos propostos [12].

Com base nos trabalhos realizados anteriormente, novos estudos foram iniciados com o objetivo de aplicar a tecnologia adaptativa em projetos de sistemas reativos, ou seja, regidos basicamente por estímulos internos ou externos. Neste contexto, um formalismo para sistemas reativos é o statechart, que, em sua versão adaptativa, resultou na implementação da primeira ferramenta computacional para a construção e simulação de dispositivos adaptativos nomeada de Statecharts Adaptativos (STAD) [13].

Ao aplicar as técnicas de adaptatividade em um statechart, o formalismo de um sistema reativo começa a apresentar a

capacidade de modificar sua configuração em relação às entradas fornecidas ao sistema.

Continuando as pesquisas para a implementação de sistemas reativos adaptativos, uma evolução do STAD, adicionando recursos de sincronização de processos, baseados em redes de Petri, foi implementada em um sistema chamado de Statecharts Adaptativos Sincronizados (SAS) [14].

Os conceitos da tecnologia adaptativa também têm sido abordados em pesquisas na área de ensino a distância, definindo propostas para sistemas baseados no modelo adaptativo de Architectural Modelling Box for Enterprise Redesign, também conhecido como AMBER-Adp [15]. O modelo AMBER é uma ferramenta de apoio que visa auxiliar o projeto e desenvolvimento de sistemas distribuídos e, por meio das técnicas fornecidas pela adaptatividade, seu poder de expressão é aumentado, possibilitando a modelagem natural de sistemas com regras que se modificam dinamicamente.

A referência [16] propõe um gerador de ambiente, também chamado de meta-ambiente, que possibilita a geração automática de ambientes para projetos baseados em aplicações adaptativas. Por meio deste trabalho, foram documentados o método para definição de dispositivos adaptativos, a arquitetura geral de um ambiente para projeto de aplicações adaptativas e a arquitetura para um gerador de ambientes para a modelagem de aplicações utilizando um dispositivo adaptativo específico.

Desde então, tem-se estudado e aplicado os fundamentos de adaptatividade, com a finalidade de criar mecanismos que, por meio de um dispositivo e conjunto de regras, podem modificar o comportamento de aplicações, objetivando desenvolver softwares mais dinâmicos e eficientes na resolução de diversos problemas existentes na computação.

A aplicação e uso de dispositivos adaptativos abrangem muitas áreas, tais como:

- Educação: uso de fundamentos de adaptatividade para a modelagem e implementação de softwares de apoio ao ensino e aplicativos educacionais;
- Inteligência artificial: modelos adaptativos utilizados para representar e manipular conhecimento, capazes de aprender por meio de informações pré-definidas e de um algoritmo de tomada de decisões;
- Segurança e privacidade: criptografia, controle de acesso, classificação de dados, reconhecimento de padrões, entre outros tópicos;
- Robótica: uma vasta área de aplicação da tecnologia adaptativa, intensificando seu uso em protocolos de roteamento, redes móveis sem fio e estudos ligados à navegação robótica autônoma;
- Jogos e simuladores: mecanismos adaptativos que podem ser acoplados a jogos e simuladores, fornecendo realismo e situações mais dinâmicas para um determinado contexto.

Conforme demonstrado por meio desta breve introdução, a tecnologia adaptativa objetiva o estudo de técnicas computacionais, tanto de modelagem quanto de desenvolvimento, para a construção de modelos ou dispositivos com características automodificáveis, dificilmente encontradas quando utilizada grande parte de métodos que buscam atender a mesma finalidade.

IV. AMBIENTES DE APRENDIZAGEM BASEADOS EM MOBILE LEARNING

O uso de ambientes de ensino, especialmente os baseados em ensino a distância, começou, há pouco tempo, a ingressar para a terceira onda tecnológica, chamada de Mobile Learning ou Aprendizagem Móvel.

Essa modalidade de aprendizado caracteriza-se pelo uso de equipamentos portáteis, como smartphones e tablets, em um contexto de “computação pervasiva”, amparado pela mobilidade global, conectividade ubíqua, independência do dispositivo e ambiente computacional do usuário disponível em quaisquer locais e horários [17].

Além das perspectivas citadas anteriormente, é possível relacionar algumas características-chaves para facilitar o entendimento e contribuir com a definição de Mobile Learning, destacando-se:

- Prover acesso a conteúdos didáticos educacionais em qualquer local e a qualquer momento, por meio de recursos de conectividade do dispositivo portátil utilizado;
- Expandir os limites internos e externos da sala de aula ou da empresa, de forma ubíqua, ou seja, integrando a computação com as ações e comportamentos naturais das pessoas;
- Fornecer os meios necessários para o desenvolvimento de metodologias inovadoras de ensino e treinamento, por meio de novos recursos da computação e portabilidade.

Portanto, de maneira resumida, pode-se conceituar Mobile Learning como qualquer tipo de aprendizado que faz uso de dispositivos móveis como ferramentas de ensino a qualquer contexto educacional, seja ele na área acadêmica ou organizacional.

A Figura 3 ilustra a evolução dos sistemas de aprendizagem a distância, definindo Mobile Learning como sendo uma extensão de E-Learning e, conseqüentemente, do Distance Learning [18].

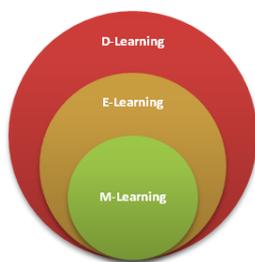


Figura 3. Evolução dos sistemas de aprendizagem a distância [18]

Partindo dessa mesma classificação de sistemas ou metodologias de ensino a distância, pode-se afirmar que o objetivo do Mobile Learning é fazer com que o ambiente de E-Learning seja implementado em dispositivos computacionais móveis, valendo-se de diversos recursos até então inexistentes ou, por muitas vezes, inexplorados.

Essa abordagem visa prover ubiquidade e um processo de aprendizagem significativo para o aprendiz por meio da criação de um contexto mais dinâmico e motivador, fazendo-se uso de conteúdos de multimídia e de uma alta interatividade [19].

Há diversas ações já iniciadas nas instituições de ensino do Brasil com o objetivo de introduzir os primeiros indícios da

computação móvel e, com isso, definir as bases principais para novos estágios relacionados à computação pervasiva e ubíqua. Por esse e outros motivos citados anteriormente, motivou-se a elaboração deste trabalho, como alternativa para fornecer novos meios de ensino-aprendizagem de disciplinas de computação.

V. ESTUDO DE CASO

O processo de ensino utilizando dispositivos computacionais teve início no final da década de 1950, quando o psicólogo americano B. F. Skinner, fundamentado na teoria comportamentalista, propusera uma metodologia de ensino, chamando-a de “máquina de ensinar”. Nessa metodologia, um conjunto de conhecimentos a ser ensinado se divide em módulos sequenciais onde o aluno deve responder, corretamente, as questões propostas para avançar de etapa durante o processo de ensino-aprendizagem.

Por meio desse método, foram desenvolvidas as primeiras aplicações de computador específicas para o ensino, sendo conhecidas como Computer Aided Instruction (CAI) ou Instrução Assistida ou Auxiliada por Computador.

Nos sistemas de ensino baseados no modelo CAI, o aprendiz segue uma série finita e predeterminada de passos. A cada etapa, novos conhecimentos são adquiridos, testados e, se correspondendo ao resultado correto, um novo conjunto de informações é transmitido; caso contrário, o conhecimento ainda não assimilado é novamente apresentado e o teste é refeito. Esse ciclo se repete até que o estudante responda corretamente ao teste, resultado este que comprova a aquisição do conhecimento exposto pelo sistema CAI.

Ao longo do tempo, das pesquisas realizadas e da constante evolução das tecnologias computacionais existentes, diversos sistemas de apoio à aprendizagem de programação foram desenvolvidos, como representações gráficas de algoritmos, sistemas tutores inteligentes, ambientes de aprendizagem a distância, companheiros de aprendizagem, entre outros.

Sabendo acerca dos problemas existentes e das dificuldades enfrentadas por alunos durante o aprendizado de conceitos fundamentais, torna-se viável a utilização de sistemas de aprendizagem em diversas disciplinas lecionadas nos cursos de computação, destacando-se, nesse cenário, Algoritmos e Lógica de Programação.

A proposta e um dos objetivos principais para a realização deste trabalho é, além de contribuir de forma teórica para os ambientes computacionais especializados na área de educação, o de implementar um sistema de aprendizagem de algoritmos baseado na plataforma móvel Google Android, disponibilizando um novo recurso de ensino-aprendizagem a ser utilizado para auxiliar e motivar os alunos dentro e fora de instituições de ensino e empresas.

Dessa forma, o projeto computacional, proposto e desenvolvido neste trabalho, contém, como premissas e requisitos iniciais, características relevantes presentes em um sistema de aprendizagem, resultando em uma aplicação capaz de fornecer um ambiente e metodologia para o estudo de algoritmos. Portanto, com o uso dessa ferramenta, cria-se a possibilidade de os alunos adquirirem e compreenderem as diversas etapas de construção de um algoritmo, além de permitir, a partir de sua implementação, testes, validações e possíveis correções em suas próprias soluções algorítmicas para um determinado problema sugerido pelo sistema tutor.

A. Arquitetura do sistema de aprendizagem de algoritmos

Durante a etapa de análise e modelagem do sistema de aprendizagem de algoritmos, cinco componentes básicos, herdados dos métodos de construção de sistemas CAI, foram identificados:

- Interface: camada responsável por intercambiar ou trocar informações entre o sistema de aprendizagem e o aluno. É por meio da interface que ocorre a interação do aprendiz com os conhecimentos expostos pelo sistema;
- Controlador de eventos: camada responsável por gerenciar e efetuar a troca de informações entre os demais módulos da arquitetura;
- Modelo do aluno: camada responsável por gerenciar o conhecimento do aluno, registrando informações sobre seus acertos e erros, assim como quais conteúdos já foram assimilados;
- Base de domínio: camada responsável por conter e descrever os conhecimentos de um especialista na área de domínio do sistema, contribuindo para a evolução do modelo do aluno;
- Modelo pedagógico: camada responsável por gerenciar as instruções ou regras de ensino, assim como executar um diagnóstico baseado no conhecimento do aluno para decidir quais as estratégias de ensino serão adotadas durante o processo de aprendizagem.

A Figura 4 ilustra a arquitetura simplificada do sistema de aprendizagem de algoritmos proposto neste trabalho, destacando os cinco componentes básicos de uma aplicação de ensino: interface, controlador de eventos, modelo do aluno, base de domínio e modelo pedagógico.

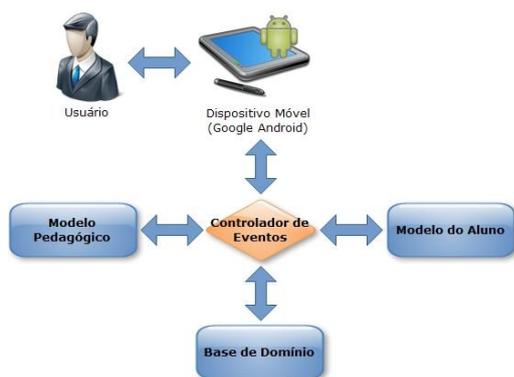


Figura 4. Arquitetura simplificada do sistema de aprendizagem de algoritmos

Apesar da simplicidade da arquitetura adotada, conforme ilustrada na Figura 4, as camadas definidas são suficientes para uma abordagem eficaz e eficiente dentro do contexto de um sistema de aprendizagem.

A base de domínio, como citado anteriormente, é responsável pelos conhecimentos da área de ensino de um sistema de aprendizagem. Portanto, para o ensino de algoritmos, essa camada deve prover as estruturas dos comandos disponíveis assim como a correta e melhor forma de utilizá-los.

A Tabela I relaciona todas as instruções ou comandos existentes que podem ser utilizados para a criação de soluções algorítmicas no sistema de aprendizagem proposto neste trabalho. A tipagem de dados, utilizada na definição de

constantes e variáveis, é restrita aos valores “booleano”, “inteiro”, “real” e “texto”, enquanto os operadores condicionais são: igual a (==), diferente de (!=), menor que (<), maior que (>), menor ou igual a (<=) e maior ou igual a (>=).

TABELA I. COMANDOS DISPONÍVEIS NO SISTEMA DE APRENDIZAGEM DE ALGORITMOS

Comando	Padrão para utilização
Declaração de Algoritmo	ALGORITMO <nome do programa>
Início de Algoritmo	INICIO:
Fim de Algoritmo	FIM.
Declaração de Área de Constantes	CONSTANTES:
Declaração de Constante	<nome> = <valor> : <tipo>;
Declaração de Área de Variáveis	VARIAVEIS:
Declaração de Variáveis	<nome> : <tipo>;
Atribuição de Valor	<variável> = <valor>;
Estrutura Condicional Se... Então... Senão...	SE (<condição>) ENTAO ... SENAO ... FIM-SE.
Estrutura de Repetição Enquanto... Faça...	ENQUANTO (<condição>) FAÇA ... FIM-ENQUANTO.
Leitura de Valor	LEIA (<variável>);
Escrita de Valor	ESCREVA (<constante ou variável>);

Após a definição dos comandos a serem disponibilizados na aplicação, tornou-se possível elaborar um diagrama de classes para representar os tipos existentes no sistema de aprendizagem de algoritmos, conforme ilustrado na Figura 5.

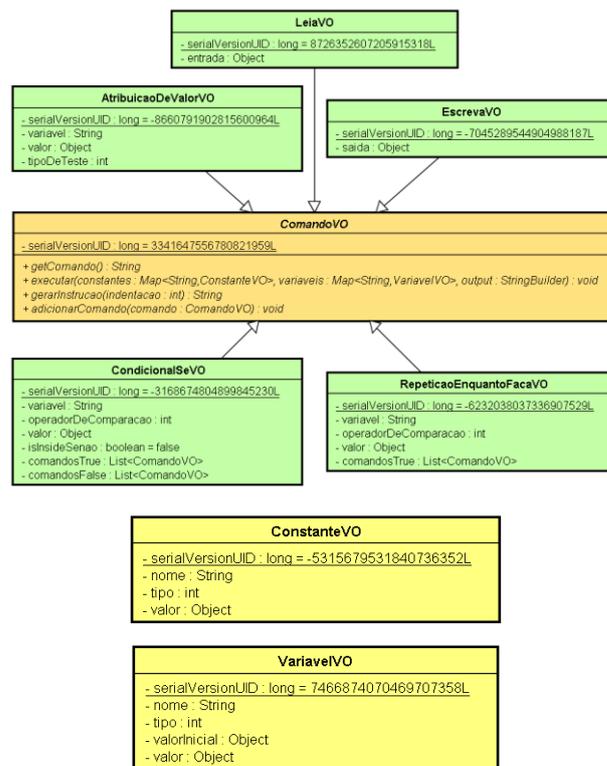


Figura 5. Diagrama de classes elaborado para representar os comandos disponíveis no sistema de aprendizagem de algoritmos

Concluída a etapa de análise e modelagem do projeto, iniciou-se a construção do sistema de aprendizagem de algoritmos em uma estrutura de projeto Java, implementando

os dispositivos adaptativos a partir dos conceitos estudados, com a finalidade de tornar a aplicação e o processo de aprendizagem mais dinâmico.

Em seguida, um projeto baseado na plataforma Google Android fora integrado aos componentes desenvolvidos anteriormente, sem quaisquer mudanças, comprovando a facilidade ao modificar e evoluir o software a partir da arquitetura utilizada.

B. Interação e fluxo de processos

A interação e o fluxo de processos do sistema de aprendizagem de algoritmos podem ser divididos em três etapas principais, cada qual com suas funcionalidades e responsabilidades, visando contribuir com a aquisição dos conhecimentos necessários durante o processo de ensino-aprendizagem.

Por meio da interface móvel, desenvolvida em um projeto Java baseado na plataforma Google Android, o usuário tem acesso aos recursos providos pela aplicação de ensino de algoritmos, auxiliando-o nas atividades de pensar, modelar e implementar soluções para os problemas gerados pelos dispositivos adaptativos.

O sistema de aprendizagem gerencia, pela camada de modelo do aluno, os níveis de conhecimento do aprendiz na disciplina de algoritmos, sendo possível dividi-los em:

- Nível 1: Implementação de algoritmos estruturados e sequenciais.
- Nível 2: Implementação de algoritmos estruturados e sequenciais, acrescentando conceitos de estrutura condicional (Se... Então... Senão...).
- Nível 3: Implementação de algoritmos estruturados e sequenciais, acrescentando conceitos de estrutura condicional e de repetição (Enquanto... Faça...).

A escolha da arquitetura e a aplicação de conceitos de componentização durante o desenvolvimento do projeto contribuem para uma melhor manutenibilidade, reusabilidade e novas implementações, já que, grande parte do sistema, é resultado da integração de diversos componentes fracamente acoplados.

Portanto, caso novos níveis ou módulos necessitem ser adequados ou acrescentados ao sistema de aprendizagem de algoritmos, basta realizar a sua implementação nas camadas de base de domínio e modelo pedagógico.

A primeira etapa, presente na interação e fluxo de processos, faz uso do primeiro dispositivo adaptativo implementado no sistema de aprendizagem e objetiva ser responsável pela geração dinâmica de enunciados de algoritmos. Por meio da recuperação de informações do modelo de aluno e com algumas configurações iniciais armazenadas em um arquivo de propriedades, o dispositivo adaptativo elabora um enunciado, solicitando, ao aluno, que implemente uma solução para determinado problema criado, estabelecendo-se a situação inicial do processo de ensino-aprendizagem.

Na segunda etapa, após a geração do enunciado, o aluno inicia a implementação do algoritmo, com o objetivo de fornecer uma solução por meio dos comandos disponíveis na aplicação. A interação entre o aluno e o sistema de aprendizagem resulta em uma fonte de estímulos utilizada para validar os comandos construídos pelo aprendiz, além de ser

possível identificar erros durante cada passo do desenvolvimento do algoritmo.

A terceira e última etapa tem por finalidade realizar a validação completa do algoritmo implementado pelo aluno, comparando-o com o enunciado proposto no início do processo de ensino. Com o resultado da validação, fornecido pelo segundo dispositivo adaptativo e amparado por um conjunto de regras estabelecidas para o ensino e a construção de algoritmos, é possível obter informações sobre acertos e erros, além de etapas ou blocos pendentes de implementação.

A Figura 6 ilustra a interação e o fluxo de processos do sistema de aprendizagem, podendo-se notar, claramente, as três etapas citadas, além de permitir um rápido entendimento de cada elemento e seus relacionamentos dentro do contexto de uma aplicação de ensino de algoritmos.

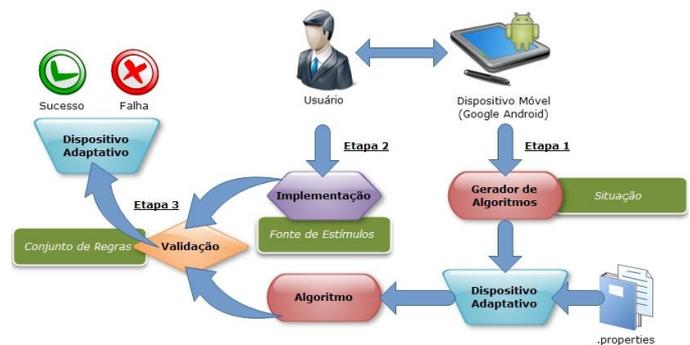


Figura 6. Interação e fluxo de processos do sistema de aprendizagem de algoritmos

C. Tecnologias e abordagens adotadas para a implementação

Para realizar a implementação do sistema de aprendizagem de algoritmos, com as funcionalidades e requisitos citados, um conjunto de tecnologias e abordagens foram pesquisadas e utilizadas durante o desenvolvimento da aplicação de ensino.

Iniciando pela escolha da linguagem de programação, optou-se pela tecnologia Java, por ser utilizada em todos os principais segmentos da indústria, estando presente em uma gama de dispositivos, computadores e redes. Com uma comunidade diversificada entre analistas, projetistas e desenvolvedores, Java é a tecnologia que tem sido a principal escolha do mercado de TI na área de desenvolvimento de aplicações Web, sistemas distribuídos, programação em redes de computadores e, mais recentemente, aplicações móveis.

O uso da moderna plataforma do Google Android, como tecnologia escolhida para o desenvolvimento deste projeto, se deve ao fato de que, por meio dela, torna-se possível a implementação de aplicações móveis que podem ser integradas a diversos recursos de maneira simplificada, utilizando a linguagem de programação Java e um ambiente de desenvolvimento de alto nível e produtividade. Além de que, cada vez mais, o uso de dispositivos móveis, como smartphones e tablets, vem crescendo, criando novas oportunidades de abordagem, entre elas, a integração das áreas de educação, aprendizado eletrônico e computação.

Devido à necessidade de que as instruções implementadas pelo usuário precisam ser validadas conforme os padrões de comandos estabelecidos para a criação de algoritmos, fez-se uso de Regular Expressions (RE) ou Expressões Regulares, um importante recurso que tem por finalidade realizar a validação e manipulação, de maneira precisa e flexível, de informações baseadas em cadeias de caracteres.

Portanto, para cada comando implementando, uma validação por meio de RE é realizada, autenticando com sucesso, ou não, a instrução elaborada pelo aluno. Assim que a instrução é validada e aceita, o comando é decomposto e uma instância do objeto Java é armazenada na lista de instruções do algoritmo que está sendo construído pelo aprendiz.

Além da portabilidade e integração de diversas tecnologias presentes no projeto, buscou-se, ainda, desenvolver um sistema de aprendizagem mais dinâmico a partir da geração de enunciados de algoritmos. Para isso, foram utilizados os conceitos da tecnologia adaptativa, possibilitando a criação de dispositivos adaptativos capazes de elaborar novos enunciados e situações para estudos de algoritmos e lógica de programação. Esses recursos automatizáveis também foram implementados no processo de validação e identificação de acertos e erros da solução proposta pelo aluno.

A construção de sistemas computacionais extensos, como ocorre no contexto de aplicações de ensino, tende a resultar em projetos maiores e mais complexos, dificultando alterações futuras e novas implementações. Para resolver esse e outros problemas recorrentes em projetos de software, novas abordagens surgiram no final da década de 1990 no ramo de Engenharia de Software, destacando-se a Component-Based Software Engineering (CBSE) ou Engenharia de Software Baseada em Componentes.

A engenharia de componentes foi adotada como metodologia de desenvolvimento do sistema proposto neste trabalho, sendo utilizada para definir um processo de análise, construção e integração de componentes independentes não firmemente acoplados. A CBSE tornou-se uma importante abordagem de desenvolvimento, contribuindo para a reusabilidade de diferentes partes já implementadas, com a finalidade de acelerar e garantir maior qualidade no processo de construção de software.

D. Imagens do sistema de aprendizagem de algoritmos

A Figura 7 ilustra a geração dinâmica, realizada pelo dispositivo adaptativo, de um enunciado de algoritmo, conforme definido na descrição da primeira etapa de interação com o sistema de aprendizagem.

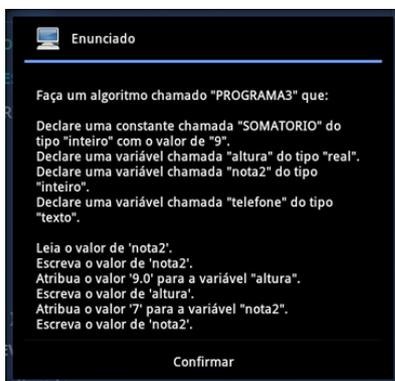


Figura 7. Enunciado de algoritmo gerado dinamicamente (primeira etapa) pelo dispositivo adaptativo

A Figura 8 ilustra a etapa de implementação de uma solução algorítmica pelo aluno, como possível resposta para atender o enunciado proposto.



Figura 8. Implementação da solução algorítmica (segunda etapa) realizada pelo aluno-aprendiz

A Figura 9 ilustra o resultado da validação realizada pelo dispositivo adaptativo, exibindo, nesse caso, uma mensagem de sucesso. Caso houvessem erros de implementação, o dispositivo adaptativo alertaria quais pontos estão em desacordo com o que fora solicitado no enunciado, cabendo, ao aprendiz, refatorar ou refazer a solução proposta.

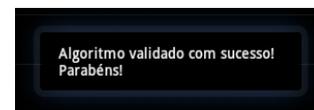


Figura 9. Resultado da validação (terceira etapa) realizada pelo dispositivo adaptativo

E. Experimentos acerca deste projeto

Com a finalidade de contribuir com o desenvolvimento deste projeto, assim como transmitir os conhecimentos gerados e acrescentar novos valores a este trabalho, uma atividade acadêmica fora realizada com grupos de alunos que frequentam a disciplina de Algoritmos e Estruturas de Dados I dos cursos de ADS e BCC.

Durante o evento, ocorrido nas dependências da instituição FEMA no ano de 2012, buscou-se apresentar os objetivos deste artigo, além de permitir o uso e, conseqüentemente, testes em um ambiente real de ensino com a aplicação totalmente desenvolvida, possibilitando o contato dos alunos com o sistema de aprendizagem de algoritmos.

Após uma introdução acerca dos conceitos envolvidos neste trabalho, assim como da utilização e testes do projeto implementado, alguns questionamentos foram avaliados pelos alunos presentes, com o objetivo de extrair indicadores sobre as opiniões de cada participante.

Os tópicos questionados abordam o uso de dispositivos móveis no processo de aprendizagem, destacando-se a qualidade de ensino, aumento do acesso à educação e a importância de sistemas de aprendizagem nos domínios social, acadêmico e profissional. Os questionamentos, no total de 14, são:

1. Facilidade ao utilizar o dispositivo móvel (smartphone e/ou tablet);
2. Facilidade ao utilizar o sistema de aprendizagem de algoritmos;

3. Ficou motivado a concluir as tarefas propostas pelo sistema de aprendizagem de algoritmos;
4. Utilizaria o sistema de aprendizagem de algoritmos com a finalidade de estudar para trabalhos e provas;
5. A geração dinâmica de problemas e novas situações contribuem para o desenvolvimento intelectual do aluno-aprendiz;
6. Facilidade ao interpretar o problema proposto;
7. Facilidade ao pensar e modelar uma solução para o problema proposto;
8. Facilidade ao solucionar o problema proposto;
9. O uso de dispositivos móveis aumenta a qualidade do processo de ensino-aprendizagem;
10. O uso de aplicações móveis para o ensino aumenta o acesso à educação e à formação profissional;
11. Utilizaria fora das instituições de ensino e/ou empresas um sistema de aprendizagem baseado em Mobile Learning;
12. Por meio de dispositivos e aplicações móveis, os objetivos do processo de ensino-aprendizagem podem ser alcançados;
13. Recomendaria Mobile Learning como uma metodologia de estudo para amigos ou colegas;
14. Professores, alunos, instituições de ensino e/ou empresas devem adotar soluções baseadas em Mobile Learning para a formação social, acadêmica e profissional.

Por meio dos questionamentos, apresentados a um grupo de 23 alunos do curso de ADS, foi possível extrair as estatísticas apresentadas na Tabela II.

TABELA II. ESTATÍSTICAS EXTRAÍDAS DOS QUESTIONAMENTOS AVALIADOS POR ALUNOS DO CURSO DE ADS

Questionamento	Concordam plenamente	Concordam	Não concordam	Incerto
1	17	6	0	0
2	17	6	0	0
3	18	5	0	0
4	18	5	0	0
5	18	5	0	0
6	15	8	0	0
7	11	12	0	0
8	12	11	0	0
9	17	5	0	1
10	17	6	0	0
11	16	7	0	0
12	13	10	0	0
13	17	6	0	0
14	15	8	0	0

Os mesmos questionamentos foram apresentados a um grupo de 27 alunos do curso de BCC, a partir dos quais foi possível extrair as estatísticas apresentadas na Tabela III.

TABELA III. ESTATÍSTICAS EXTRAÍDAS DOS QUESTIONAMENTOS AVALIADOS POR ALUNOS DO CURSO DE BCC

Questionamento	Concordam plenamente	Concordam	Não concordam	Incerto
1	15	12	0	0
2	19	8	0	0
3	11	16	0	0
4	23	4	0	0
5	21	6	0	0
6	17	10	0	0
7	15	11	0	1
8	19	7	0	1
9	13	14	0	0
10	18	9	0	0
11	17	10	0	0
12	12	15	0	0
13	20	7	0	0
14	16	11	0	0

Ao visualizar as estatísticas extraídas das avaliações, pode-se notar o elevado nível de concordância com os questionamentos definidos, provando-se a real importância de trabalhos na área de sistemas de aprendizagem baseados em Mobile Learning.

Outro ponto que merece o devido destaque é a interação e motivação criada pelos alunos ao fazer uso do sistema de aprendizagem de algoritmos, sendo possível observar um grande desejo em continuar utilizando-o, como parte do processo e metodologia de ensino-aprendizagem fornecido pela instituição FEMA, para fornecer uma nova forma de estender e fomentar os estudos além dos limites das aulas presenciais.

A partir de tal atividade acadêmica, as estatísticas e produtos gerados também se integram a este trabalho, tornando-se parte do conjunto de resultados obtidos por meio da realização deste artigo e projeto.

VI. CONCLUSÃO

O aprendizado de algoritmos é um dos grandes desafios na área de ensino de computação, visto que o índice de dificuldades encontradas pelos alunos é bastante elevado. Portanto, torna-se importante o desenvolvimento de sistemas e ambientes que busquem promover o interesse do aluno, assim como facilitar a compreensão dos conceitos de programação.

A utilização de ambientes computacionais de ensino tem crescido e se tornado uma metodologia comum em universidades e centros de estudo, auxiliando alunos e professores no processo de aprendizagem. Nas áreas de computação e tecnologia, o aprendizado de algoritmos e lógica de programação é essencial para a carreira do estudante, contribuindo para que este consiga obter os fundamentos básicos, tornando-o apto a avaliar, pensar, modelar e desenvolver suas próprias soluções algorítmicas para problemas existentes no mundo atual.

Por meio do estudo e projeto realizados, conclui-se que o uso de ferramentas e recursos adequados ao ensino de determinado assunto é importante, permitindo que o aluno participe de experiências acadêmicas que nem sempre são possíveis em sala de aula, além de proporcionar diversas situações novas e dinâmicas a cada etapa do estudo.

Como produtos desta pesquisa, a proposta deste trabalho foi a de desenvolver os estudos teóricos com a finalidade de adquirir os conhecimentos necessários acerca da tecnologia adaptativa aplicados à área de educação, assim como realizar a

implementação de um sistema de aprendizagem de algoritmos baseado na plataforma Google Android.

A partir dessa abordagem inicial, buscou-se construir uma nova ferramenta de ensino, contribuindo com os estudos de alunos de disciplinas iniciais, como Algoritmos e Lógica de Programação, de cursos de computação, com o objetivo de reduzir o índice de dificuldades encontradas, além de incentivar os estudos após as aulas presenciais.

Espera-se que este trabalho venha a contribuir para a pesquisa na área de sistemas adaptativos, além de ampliar o interesse pelo seu uso no contexto de ambientes de aprendizagem, tornando-os dinâmicos e adaptáveis em relação ao processo de ensino-aprendizagem.

A. Trabalhos futuros

Como trabalhos futuros, almeja-se continuar as pesquisas e o desenvolvimento da aplicação, com a finalidade de melhorar e evoluir o sistema de aprendizagem de algoritmos apresentado neste projeto.

Entre alguns dos interessantes pontos a serem estudados, destaca-se a integração do sistema de aprendizagem com um ambiente acadêmico existente, para que as informações sobre a evolução do aluno possam ser enviadas a um servidor, fornecendo novas estatísticas para o professor acompanhar os níveis de aprendizado de uma maneira mais simplificada.

REFERÊNCIAS

[1] E. W. Dijkstra, "On the cruelty of really teaching computing science", *Communications of ACM*, Vol. 32, pp. 1398-1404, December 1989, in press.

[2] A. J. Gomes, "Ambiente de suporte à aprendizagem de conceitos básicos de programação", Dissertação (Mestrado), Centro de Informática e Sistemas da Universidade de Coimbra, Instituto Superior de Engenharia de Coimbra, Coimbra, Portugal, 2001.

[3] E. S. Almeida, E. B. Costa, J. D. H. Silva, K. S. Paes, A. A. M. Almeida, "AMBAP: Um ambiente de apoio ao aprendizado de programação", X Workshop sobre Educação em Computação, Florianópolis, Santa Catarina, Brasil, Anais do WEI 2002 / SBC 2002, 2002, in press.

[4] M. C. Rodrigues Jr., "Experiências positivas para o ensino de algoritmos", IV Escola Regional de Computação, Feira de Santana, UFES, Bahia – Sergipe, Brasil, 2004, in press.

[5] D. N. Perkins, S. Schwartz, R. Simmons, "Instructional strategies for the problems of novice programmers", R. E. Mayer (ed.), *Teaching and Learning Computer Programming*, pp. 153-178, Hillsdale, NJ: Lawrence Erlbaum Associates, 1988, in press.

[6] P. Byrne, G. Lyons, "The effect of student attributes on success in programming", *Proceedings of the 6th Annual Conference on Innovation and Technology in Computer Science Education*, ITiCSE, United Kingdom, pp. 49-52, 2001, in press.

[7] J. C. R. P. Júnior, C. E. Rapkiewicz, "O processo de ensino e aprendizagem de algoritmos e programação: uma visão crítica da literatura", III Workshop de Educação em Computação e Informática do Estado de Minas Gerais, WEIMIG 2004, Belo Horizonte, Minas Gerais, Brasil, 2004, in press.

[8] J. C. R. P. Júnior, C. E. Rapkiewicz, C. Delgado, J. A. M. Xexeo, "Ensino de algoritmos e programação: uma experiência ao nível médio",

XIII Workshop de Educação em Computação, WEI 2005, São Leopoldo, Rio Grande do Sul, Brasil, 2005, in press.

- [9] T. Jenkins, "On the difficulty of learning to program", *Proceedings of 3rd Annual LTSN_ICS Conference*, Loughborough University, United Kingdom, The Higher Education Academy, pp. 53-58, August 27-29, 2002, in press.
- [10] J. J. Neto, M. E. S. Magalhães, "Um Gerador Automático de Reconhecedores Sintáticos para o SPD", VIII SEMISH – Seminário de Software e Hardware, pp. 213-228, Florianópolis, Santa Catarina, Brasil, 1981, in press.
- [11] J. J. Neto, "Geração automática de analisadores sintáticos para o SPD: evolução e estado da arte", *Anais do VI Congresso Nacional de Matemática Aplicada e Computação*, ITA, São José dos Campos, São Paulo, Brasil, 1983, in press.
- [12] J. J. Neto, "Uma Solução Adaptativa para Reconhecedores Sintáticos", *Anais Escola Politécnica, Universidade de São Paulo – Engenharia de Eletricidade – Série B*, vol. 1, pp. 645-657, São Paulo, Brasil, 1988, in press.
- [13] J. R. Almeida, "STAD: Uma ferramenta para representação e simulação de sistemas através de statecharts adaptativos", Tese (Doutorado), Escola Politécnica, Universidade de São Paulo, São Paulo, Brasil, 1995.
- [14] J. M. N. Santos, "Um formalismo adaptativo com mecanismo de sincronização para aplicações concorrentes", Dissertação (Mestrado), Escola Politécnica, Universidade de São Paulo, São Paulo, Brasil, 1997.
- [15] A. R. Camolesi, J. J. Neto, "Modelagem AMBER-Adp de um ambiente para gerenciamento de ensino a distância", *Anais do XIII Simpósio Brasileiro de Informática na Educação*, SBIE 2002, pp. 401-409, São Leopoldo, Rio Grande do Sul, Brasil, 2002, in press.
- [16] A. R. Camolesi, "Proposta de um gerador de ambientes para a modelagem de aplicações usando tecnologia adaptativa", Dissertação (Doutorado), Escola Politécnica, Universidade de São Paulo, São Paulo, Brasil, 2007.
- [17] A. Yasmin, "Uso de computadores de mão no contexto do sub-projeto Ambiente de Execução direcionado à Pervasive Computing – EXEHDA", Universidade Católica de Pelotas, Pelotas, Rio Grande do Sul, Brasil, April 2004.
- [18] T. Georgiev, E. Georgieva, A. Smrikarov, "M-Learning: A New Stage of E-Learning", *International Conference on Computer Systems and Technologies – CompSysTech 2004*, Bulgaria, Rousse, June 2004, in press.
- [19] L. M. R. Tarouco, M. C. J. M. Fabre, A. R. S. Grando, M. L. P. Konrath, "Objetos de Aprendizagem para M-Learning", *Congresso Nacional de Tecnologia da Informação e Comunicação*, Florianópolis, Santa Catarina, Brasil, 2004, in press.



Guilherme de Cleve Farto é graduado em Bacharelado em Ciência da Computação pela Fundação Educacional do Município de Assis (2010) e pós-graduado em Engenharia de Componentes utilizando Java pela TNT Educacional e Faculdades Integradas de Ourinhos (2011). Atualmente é analista e desenvolvedor de sistemas Java - Próxima - Software e Serviços, empresa do grupo TOTVS. Também é professor universitário do curso de Bacharelado em Ciência da Computação, junto ao Instituto Municipal de Ensino Superior de Assis (IMESA) e Fundação Educacional do Município de Assis (FEMA), em Assis/SP. Tem experiência na área de Ciência da Computação, com ênfase em desenvolvimento de sistemas e novas tecnologias, atuando principalmente nos seguintes temas: Java, Bancos de Dados, XML, Web Services, Computação em Nuvem, Computação Física com Arduino, Google Android, Google App Engine e integração de sistemas EAI.