

INTERFACES GRÁFICAS COMO FERRAMENTA PARA AUXILIAR O ENSINO DE ABSTRAÇÃO NA DISCIPLINA DE PROGRAMAÇÃO ORIENTADA A OBJETOS

***Resumo:** Este trabalho tem como finalidade mostrar a importância de aplicação do conceito de abstração na disciplina de orientação a objetos. Conceito esse que é a base de todo o paradigma, e uma abordagem de metodologia de ensino trazendo o aprendizado baseado em desafios e problemas para construção de interfaces gráficas, permitindo o estudante uma aplicação concreta dos conhecimentos aprendidos em sala.*

***Palavras-chave:** Abstração. Interfaces. Metodologia.*

1 INTRODUÇÃO

Uma das justificativas dos índices de evasão em cursos de nível superior é a pouca compreensão dos estudantes sobre determinados conteúdos das disciplinas fundamentais do curso, desestimulando os alunos.

De acordo com (GOMES; HENRIQUES; MENDES.,2008) uma disciplina que os discentes sentem maior dificuldade, são disciplinas ligadas ao desenvolvimento e programação de sistemas de computação, a dificuldade apresenta-se desde o entendimento da lógica de programação e dos algoritmos, seja devido a carência na educação básica seja de matemática ou de inglês, ou na negligência dos docentes na construção do conhecimento de forma didática, apresentando ou não ferramentas auxiliares para estimular o aluno a compreender a proposta da disciplina e as aplicações da mesma em casos reais.

Outra grande dificuldade que surge, é a compreensão do conceito e aplicação de abstração, o que por sua vez dificulta ainda mais o aprendizado dos estudantes, principalmente em cursos que apresentam a disciplina de programação orientada a objetos, onde esse conceito mostra-se fundamental para a resolução dos problemas propostos.

Este artigo tem como objetivo destacar a importância do desenvolvimento de interfaces gráficas para auxiliar a compreensão e aplicação da abstração na disciplina de programação orientada a objetos, tendo como base uma pesquisa de caráter exploratório com diversos estudantes de cursos (que possuem essa disciplina) e professores (que ensinam a disciplina) do Instituto Federal de Educação, Ciência e Tecnologia do Ceará - IFCE.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 O conceito e os fundamentos da programação orientada a objetos:

O computador atualmente é indispensável em diversas áreas da engenharia, e o conhecimento de como programar permite ao profissional criar rotinas e programas para automatizar o trabalho e otimizar os resultados. O primeiro contato de um estudante com a

lógica e linguagem de programação, se dá através do paradigma de programação imperativa, isto é, todo o código baseia-se em etapas lógicas e sequenciais.

Entretanto se não for bem estruturado e organizado o código fonte do programa criado pode gerar um grande trabalho para o profissional, que terá problemas em realizar manutenções e correções do programa. Isto porque o paradigma de programação imperativa condensa todo o programa em um grande bloco de código. Além de dificultar a manutenibilidade do sistema desenvolvido, limita a possibilidade de reutilização do código, tornando assim o processo de desenvolvimento cansativo e desestimulante.

O paradigma de orientação a objetos criado pelo biólogo Alan Kay, permite ver o programa de uma maneira natural, dividindo o grande e denso bloco de código em pequenas partes que interagem entre si, que chamamos de classes, que por sua vez são descrições abstratas de conjuntos de elementos que farão parte do sistema desenvolvido.

Esses elementos que integram o sistema, chamamos de objetos, e os mesmos interagem entre si, trocando mensagens (informações, dados e solicitando ações ou comportamentos) uns com uns outros.

E com base nisto temos, o nome: Orientação a Objetos, onde a modelagem do problema e da solução se dá através da interação de objetos abstraídos da realidade, tornando o software natural, reutilizável (já que o fato de estar dividido em classes permite um reaproveito futuro de uma das classes), manutenível, e mais seguro, já que agora temos varias partes que compõem o software, ao invés de uma única, que pode devida a mínima interação pode alterar todo o funcionamento do programa.(BORATTI, 2007, p. 29-35).

2.2 Abstração

Abstrair em programação orientada a objetos, consiste em ignorar detalhes irrelevantes ao problema, simplificando o problema, considerando apenas o necessário para a solução do problema.

Abstração é o processo utilizado na análise de determinada situação, através do qual se observa a realidade, tendo-se por objetivo a determinação dos aspectos e fenômenos considerados essenciais, excluindo-se todos os aspectos considerados irrelevantes ou secundários. (BORATTI, 2007, p.15).

A falta de conhecimento desse conceito, prejudica a implementação de uma modelagem orientada a objetos, sendo considerado por muitos autores como a base da POO.

Deve-se pensar em como as entidades que serão criadas se relacionaram com as demais, preocupando-se na funcionalidade e não na implementação, excluindo informações repetitivas e inúteis para o problema em si.

2.3 A importância dos conceitos de orientação a objetos

O conhecimento de orientação a objetos não são restritos apenas para o desenvolvimento de sistemas de computação, mas também auxiliam a elaboração de em projetos em diversas áreas, facilitando o desenvolvimento e criação de modelos e protótipos.

Como exemplo temos o BIM (Building Information Modeling) que é um software de modelagem de construção, que apresenta uma abordagem orientada a objetos, onde elementos

individuais abstraídos da realidade, que podem receber um significado semântico, interagem entre si.(AHN et al., 2010)

Diversos outros estudos mostram a orientação a objetos como poderosa ferramenta para modelagem, como a modelagem de um sistema de controle industrial (BONFÈ; FANTUZZIS., 2003), modelagem para o sistema de transporte público (MAKA; CUPEK; ROSNER., 2011), modelagem da avaliação de riscos e falhas de equipamentos elétricos (DMITRIEV et al;.,2017), modelagem da logística de um e-commerce (JIANAN; WEILI., 2010), entre outros.

3 RESULTADOS DA PESQUISA

3.1 As dificuldades principais dos estudantes em POO:

Na pesquisa realizada com 32 estudantes as respostas foram semelhantes tanto para os estudantes de cursos técnicos como para os estudantes de curso superior, os estudantes tinham dificuldades para compreender os fundamentos da orientação a objeto, e de que forma iriam modelar e trabalhar com os dados. Poucos dos estudantes entrevistados souberam responder o significado de abstração, e não conseguiam enxergar aplicações da orientação a objetos.

O aluno entrevistado chamado de A1 por questões éticas, relatou que teve dificuldades em assimilar os conceitos, principalmente pelo fato de não conseguir enxergar de forma prática aquilo que estava sendo produzido, e teve dificuldades de entender o resultado mostrado na interface console.

3.2 Avaliações dos professores sobre as dificuldades dos alunos

Foram entrevistados 4 professores. Os professores quando questionados sobre os desafios para o ensino de POO, responderam que maior dificuldade é ensinar o conceito de abstração para os alunos, e convencê-los de que é fundamental tal conceito para modelagem de uma situação.

De acordo com o docente entrevistado E1, é complicado transmitir ao aluno que POO, utiliza a programação de forma estruturada, entretanto organiza-se de maneira diferente, a forma que o problema será modelado, será outra. E com uma boa capacidade de abstração, esse processo torna-se simples.

Já o entrevistado E2 afirma que o tempo para explicar os conceitos de POO é curto, em especial os conceitos abstração e o desenvolvimento da capacidade de abstrair, visto que muitos que chegam na disciplina ainda carregam dificuldades da disciplina de lógica de programação, então parte do tempo é sacrificado para ensinar conteúdos que já deveriam saber ao ingressar em POO.

4 DESENVOLVIMENTO DE INTERFACES GRÁFICAS

4.1 Motivação

A motivação é desenvolver de forma gráfica e interativa, simples aplicações da abstração e com base nisso modelar o desenvolvimento do sistema, utilizando o paradigma da orientação a objetos.

Visto que o aluno aprende melhor ao interagir com a interface gráfica do que com a interface console, devido a possibilidade de organização, visualização e manipulação dos dados, além de motivar ainda mais o aprendizado.(PAMBOUKIAN et al, 2011, p. 8).

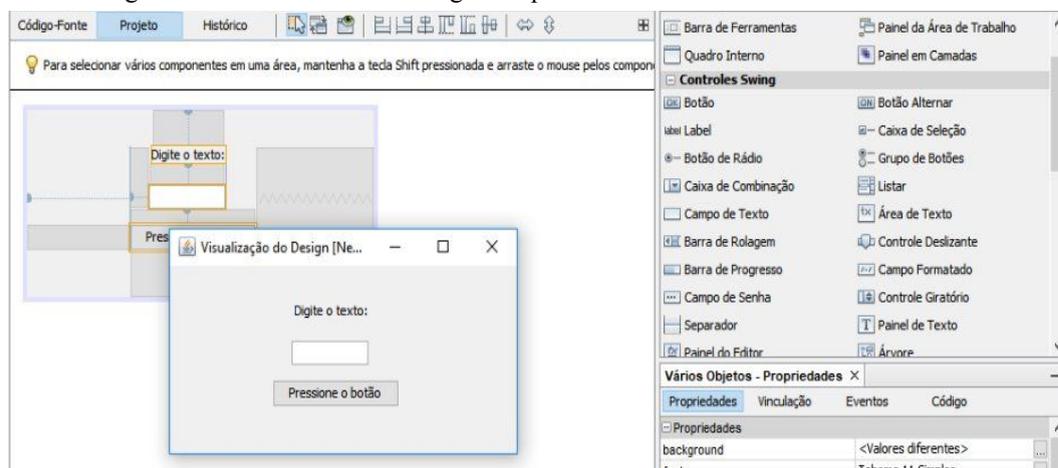
4.2 Aprendizagem baseada em problemas

O conceito de aprendizagem baseada em problemas (*Problem Based Learning* - PBL) é uma metodologia que busca trazer desafios ao aluno com objetivo de construir o conhecimento de forma mais ativa, concreta e contextualizada, permitindo ao estudante visualizar as aplicações do conceito ensinado pelo professor. (MAMEDE et al, 2001).

Com base nisso podemos aplicar essa metodologia de ensino para reforçar os conceitos apresentados na disciplina de orientação a objetos, projetando interfaces gráficas. Grande parte dos IDE's (Ambientes de Desenvolvimento Integrado) auxiliam na correção de sintaxe de código e criação rápida de código, este último possibilita o usuário fazer coisas simples como abrir um menu de componentes para seleção. Basta o aluno arrastar o componente até uma área de trabalho fornecida pelo próprio IDE, preocupando-se assim somente com a parte do código em si, e tratando cada componente da maneira mais isolada possível, aplicando assim o conceito de abstração como base para modelagem da interface.

A seguir na figura 1, temos um ambiente na própria ferramenta para auxiliar na criação e organização de componentes para interfaces gráficas, o software é o IDE NetBeans desenvolvido para a Oracle e distribuído de forma gratuita, com grande quantidade de recursos para a linguagem Java, que é uma das mais utilizadas para o ensino de orientação a objetos. Além do NetBeans, existem diversos outros ambientes de desenvolvimento integrado, ficando a critério do instrutor a escolha da ferramenta.

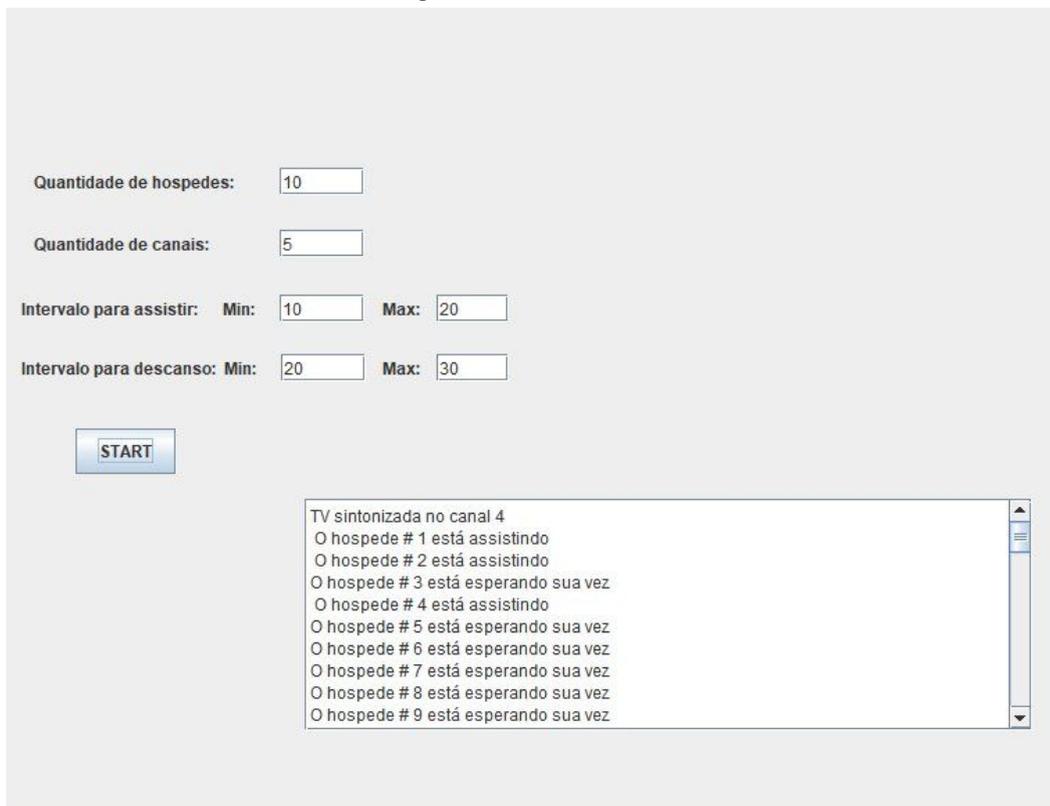
Figura 1 - Auxiliar de interface gráfica para desenvolvimento Java da IDE Netbeans.



Fonte: O autor (2018).

Cabe então ao professor ou responsável pela supervisão e orientação do aluno na criação de interface gráfica, criar propostas de soluções de problemas que demandem uma interface gráfica, podendo inclusive solicitar aos alunos para implementarem interfaces que possam auxiliar resoluções de problemas em outras disciplinas promovendo assim a interdisciplinaridade, como por exemplo a imagem abaixo mostra uma interface gráfica desenvolvida por um dos alunos entrevistados para visualização de threads e processos, o aluno desenvolveu tal interface para a disciplina de Sistema Operacional Básico.

Figura 2 - Interface para acompanhamento e visualização de processos para disciplina de Sistema Operacional Básico.



Fonte: Captura de tela da interface desenvolvida pelo estudante entrevistado(2018).

4.3 Resultados

Dos 32 estudantes, 10 foram selecionados e tiveram como desafio a criação de interfaces gráficas e conseguiram compreender melhor não somente o conceito e o fundamento da abstração como também alguns outros pilares da orientação a objetos, especialmente o encapsulamento, que consiste em isolar um objeto dos demais, tornando o mesmo independente e caso necessário alterar algum dado, não compromete o sistema de forma crítica. Segundo A2, foi somente aplicando interface gráfica que ele conseguiu compreender uma real utilidade para a disciplina.

Entretanto segundo os professores, aplicar interfaces em sala ainda não é completamente viável para todas as turmas, isto devido a deficiências que os alunos trazem

da disciplina de lógica e linguagem de programação e que ainda acabam sendo corrigidas e tratadas na disciplina de programação orientada a objetos.

5 CONCLUSÃO

Esse estudo atingiu seu objetivo em conseguir auxiliar com a ferramenta de criação de interfaces o ensino de abstração, assim os alunos apenas deveriam se preocupar com funcionalidades e ignorar aquilo que não fosse necessário a implementação do sistema.

Entretanto algumas deficiências se dão devido as disciplinas anteriores que tratam a programação e o desenvolvimento de sistemas como algo muito complexo e que distante do real, além de não se preocupar em uma abordagem que permita o aluno enxergar que existem diferentes paradigmas de programação e eles podem cooperar juntos em prol de uma mesma finalidade. Assim muitos estudantes, entendem que orientação a objetos é algo completamente diferente do que já conheciam e além de algumas dificuldades com o aprendizado do paradigma em si, apresentam dificuldades na construção de blocos de códigos dentro das classes.

REFERÊNCIAS BIBLIOGRÁFICAS

AHN, Seungjun; PARK, Moonseo; LEE, Hynsoo; YANG, Youngjun. Object-oriented modeling of construction operations for schedule-cost integrated planning, based on BIM. In: International Conference on Computing in Civil and Building Engineering. 2010.

BONFÈ, Marcelo; FANTUZZIS, Cesare. Design and Verification of Mechatronic Object-Oriented Models for Industrial Control Systems. In: Emerging Technologies and Factory Automation. 2003.

BORATTI, Isaias Camilo. Programação Orientada a Objetos em Java. / Isaias Camilo Boratti - Florianópolis: Visual Books. 2007.

DMITRIEV, Stepan A.; KOKORIN, Evgenii L.; VOLOBUEV, Aleksandr V.; KORELINA, Anna A. The Object-Oriented Model of the Electrical Equipment Failures Risks Assessment. In: 58th International Scientific Conference on Power and Electrical Engineering of Riga Technical University (RTUCON).2017.

GOMES, A.; HENRIQUES, J.; MENDES, A. J. Uma proposta para ajudar alunos com dificuldades na aprendizagem inicial de programação de computadores. In: *Educação, Formação & Tecnologias*; vol.1(1), pp. 93-103. 2008.

JIANAN, Xu; WEILI, Yao. E-Commerce Logistics Model Research Based On Object-Oriented Petri Net. In: 2010 International Forum on Information Technology And Applications. 2010.

MAKA, Aldona; CUPEK, Rafal; ROSNER, Jakub. OPC UA Object Oriented Model for Public Transportation System. In: UKSim 5th European Symposium on Computer Modeling and Simulation. 2011.

MAMEDE, Silvia; PENAFORTE, Júlio; SCHMIDT, Henk; CAPRARA, Andrea; TOMAZ, José Batista; SÁ, Henrique. Aprendizagem baseada em problemas: anatomia de uma nova abordagem educacional. 2001.

PAMBOUKIAN, Sergio Vicente Denser; CYMROT, Raquel; ZAMBONI, Lincoln Cesar; HU, Osvaldo Ramos Tsan; Barros, Edson de Almeida Rego. Ensino de programação em cursos de engenharia: interfaces console X interfaces gráficas. In: XXXIX Congresso Brasileiro de Educação em Engenharia - Anais do XXXIX COBENGE. 2011.

GRAPHIC INTERFACES AS A TOOL TO AUXILIATE ABSTRACTION TEACHING IN THE OBJECT-ORIENTED PROGRAMMING DISCIPLINE

***Abstract:** This paper has the purpose of showing the importance of applying the concept of abstraction in the discipline of object orientation. This concept is the base for the whole paradigm, and an approach to teaching methodology bringing the learning based on challenges and problems for the construction of graphical interfaces, allowing the student a concrete application of the knowledge learned in the classroom.*

***Key-words:** Abstraction. Interfaces. Methodology.*