

EASYCODE: UM AMBIENTE DE TECNOLOGIAS DE APOIO PARA O ENSINO DE LÓGICA DE PROGRAMAÇÃO

Daniela F. Feitosa ¹, Eduardo N. B. Neves ¹, Emmerson S. R. Silva ¹, Jucimar B. Souza ¹

¹Instituto Federal de Educação, Ciência e Tecnologia do Amazonas
Campus Manaus Centro (IFAM-CMC)

{danielaferreira1133, eddunic, emmsr2004, jucibs}@gmail.com

Abstract. *This article presents the scientific contribution aimed at the EasyCode software, developed by IFAM-CMC students in 2017, aiming to adapt it to students who have difficulty applying the programming logic of the imperative paradigm in the resolution of computational problems. The application is being developed on the web platform and will consist of visual programming based on programming blocks through Google's Blockly library, flowcharts and application of the cognitive domain of the Revised Bloom Taxonomy [ANDERSON, L. W. and KRATHWOHL, D. R. and AIRASIAN, P. W. and WITTROCK, M. C. 2001] in a future gamification in order to stimulate the assimilation of concepts of logic. The technologies that will be used in the teaching-learning process will allow programming concepts to be understood in a playful way.*

Resumo. *Este artigo apresenta a contribuição científica objetivada para o software EasyCode, desenvolvido por alunos do IFAM-CMC em 2017, visando adequá-lo aos estudantes que têm dificuldade em aplicar a lógica de programação do paradigma imperativo na resolução de problemas computacionais. A aplicação está sendo desenvolvida na plataforma web e consistirá na programação visual com base em blocos de programação por meio da biblioteca Blockly da Google, fluxogramas e na aplicação do domínio cognitivo da Taxonomia de Bloom Revisada [ANDERSON, L. W. and KRATHWOHL, D. R. and AIRASIAN, P. W. and WITTROCK, M. C. 2001] em uma futura gamificação a fim de estimular a assimilação dos conceitos da lógica. As tecnologias ao serem empregadas no processo de ensino-aprendizagem permitirão que conceitos da programação sejam compreendidos de maneira lúdica.*

1. Introdução

As dificuldades que inviabilizam a compreensão da programação variam desde a abordagem utilizada para o ensino até o problema na natureza cognitiva do aluno. Segundo Gomes *et al.* (2008), dentre os possíveis fatores deste problema destacam-se o elevado nível de abstração da programação e a falta de motivação do próprio aluno.

Em um estudo de caso baseado na Taxonomia de Bloom e no Scratch, realizado por Araújo *et al.* (2013), percebeu-se a necessidade de incentivo ao aprendizado e do uso de outros recursos visuais como base metodológica. Portanto, modificações na abordagem utilizada durante as aulas de lógica podem torná-las mais interativas e despertar

gradativamente o interesse do aluno, apesar das possíveis dificuldades enfrentadas [SILVA, E. S. R. and TAKANO, E. M. and SOUZA, J. B. and OLIVEIRA, M. V. 2017].

O *software* EasyCode desenvolvido por Silva *et al.* (2017) dispõe de um ambiente de programação visual baseado em blocos e desenvolvido em Java Swing para ser empregado no aprendizado de lógica de programação, veja Figura 1. Este *software* apresenta limitações, como: só é executado localmente em uma máquina (requer *download* do programa), converte o algoritmo apenas para a linguagem C e, além da má organização dos blocos, apresenta uma interface confusa e pouco atrativa. Objetivase neste trabalho, uma nova fase de desenvolvimento voltada à *web* com conversão para multilinguagens e com a implementação de novas tecnologias que ampliem as funcionalidades para o público-alvo.

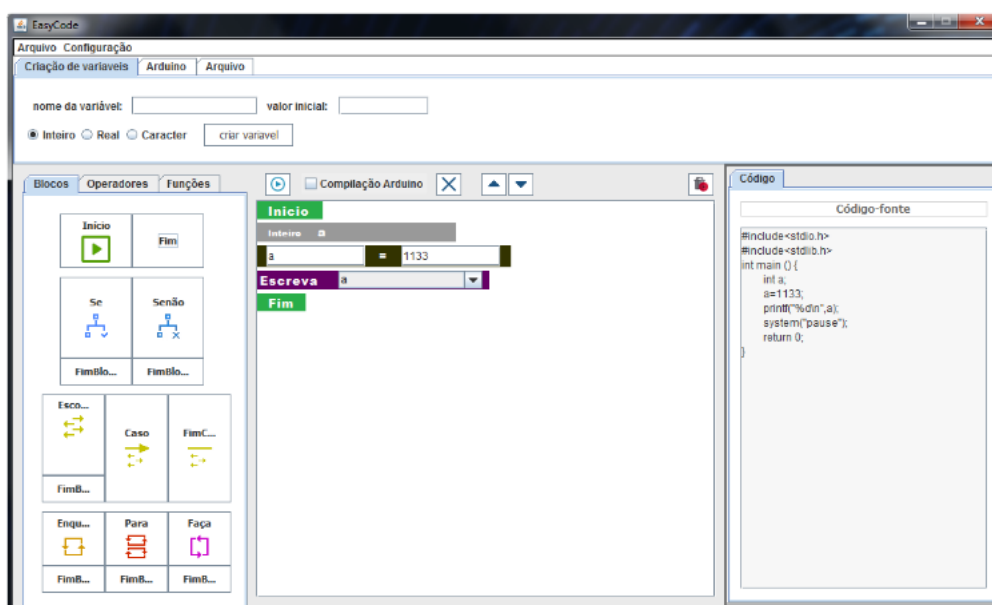


Figura 1. Layout da tela principal do ambiente EasyCode.

2. Metodologia

Utilizar-se-á o domínio cognitivo da Taxonomia de Bloom Revisada que consiste nas categorias lembrar, entender, aplicar, analisar, avaliar e criar [FERRAZ, A. P. C. M. and BELHOT, R. V. 2010] para que ocorra a integração de abordagens de ensino que se adequem aos tipos de aprendizagem. Esta será a base metodológica para a gamificação que será implementada no *software* EasyCode, que contará com uma sessão de desafios onde o discente poderá revisar o conteúdo de lógica de programação imperativa (inicialmente as estruturas lógicas), codificar e gerar fluxogramas ou a montagem de blocos (veja Figura 2), testar os códigos produzidos e avaliá-los, além de produzir e planejar técnicas de resoluções diversas.

Ademais, a taxonomia em conjunto com pesquisas quanti-qualitativas serão aplicadas para alunos do primeiro ano do ensino médio técnico em informática no IFAM-CMC para avaliar o nível de desenvoltura na resolução de problemas computacionais a partir do *software* EasyCode.

Como ferramentas de desenvolvimento do *website* estão sendo aplicados os *frameworks* Maven, Hibernate e Bootstrap; as bibliotecas JQuery e Blockly; o padrão de arquitetura MVC; a metodologia ágil SCRUM; o controle de versão pelo GitHub; e a API JDBC. No *front-end*, além do Bootstrap, utiliza-se a linguagem HTML5, JavaScript e as folhas de estilo em CSS3. O padrão XML está voltado para armazenar os blocos de programação, os processos de fluxogramas e o código-fonte gerado pela interação entre usuário e *software*.

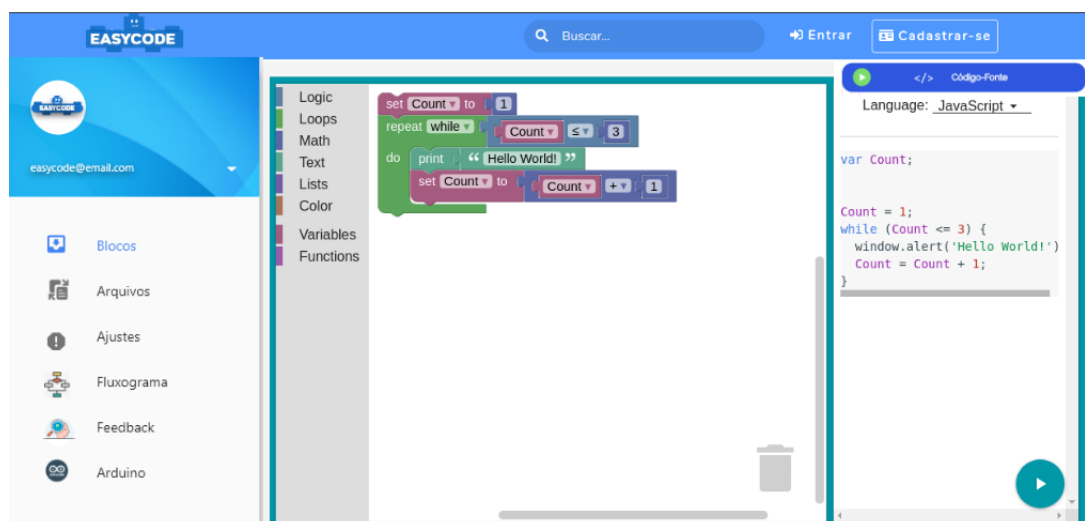


Figura 2. Protótipo da tela de montagem de blocos do EasyCode.

3. Resultados Parciais e Discussão



Figura 3. Gráficos.

Mapeamento e análise do *software* na versão de 2017 para a remodelagem foram concluídos a partir da esquematização das telas em um manual e da diagramação da estrutura do projeto; verificação de falhas, prototipação da nova versão; produção do questionário; avanço na implementação *web* a partir da transição para independência do Java Swing.

4. Considerações Parciais

O EasyCode é um ambiente que proporciona os recursos da programação e da robótica para permitir que alunos com dificuldades em compreender a lógica de programação possam aprender a programar de maneira simples e prática. Espera-se que as mudanças propostas colaborem com o processo de ensino-aprendizagem, por exemplo, permitindo que professores possam demonstrar de forma mais lúdica conceitos da programação e também que ocorra a adequação da abordagem de ensino aos tipos de aprendizagem.

A visualização por meio de fluxogramas dos passos para se construir o algoritmo permitirá o entendimento da aplicação da codificação de maneira simples. A implantação

da conversão entre multilinguagens de programação e da biblioteca Blockly da Google permitirá que a ferramenta se torne mais atrativa para qualquer estudante.

Os alvitre para a continuidade do projeto em questão possuem a intenção de aumentar a usabilidade do referido por meio da eficácia do planejado. Através dos meios acessíveis impostos, pretende-se aumentar o alcance de uso do *software* por intermédio da implementação em plataforma *web*.

Referências

- ANDERSON, L. W. and KRATHWOHL, D. R. and AIRASIAN, P. W. and WITTROCK, M. C. (2001). A taxonomy for learning, teaching and assessing: A revision of bloom's taxonomy of educational objectives. *Nova York: Addison Wesley Longman*, page 336.
- FERRAZ, A. P. C. M. and BELHOT, R. V. (2010). Taxonomia de bloom: revisão teórica e apresentação das adequações do instrumento para definição de objetivos instrucionais. *Gest. Prod., São Carlos*, 17(n. 2):421–431.
- SILVA, E. S. R. and TAKANO, E. M. and SOUZA, J. B. and OLIVEIRA, M. V. (2017). Easycode: Um ambiente de programação visual para o ensino de lógica de programação. *Recife: 7º Simpósio de Hipertexto e Tecnologias na Educação*.