Avaliação Heurística de Ferramentas de Programação em Blocos

Franciely Alves de Souza
Universidade Federal Rural de Pernambuco
Recife, Pernambuco, Brasil
francielyalves835@gmail.com

Taciana Pontual Falcão
Universidade Federal Rural de Pernambuco
Recife, Pernambuco, Brasil
taciana.pontual@ufrpe.br

Rafael Ferreira Mello
Universidade Federal Rural de Pernambuco
Recife, Pernambuco, Brasil
rafael.mello@ufrpe.br

O ensino de programação na escola é algo que vem se expandindo rápido com o passar dos anos. A linguagem de programação em blocos foi desenvolvida especialmente para crianças, pois usa uma interface gráfica de simples compreensão, e utiliza apenas blocos parecidos com brinquedos de montar como o Lego. Pensando no público que essas ferramentas possuem e no seu objetivo, é essencial que as interfaces sejam claras, fáceis de usar, organizadas e atraentes. Neste trabalho, avaliamos a interface de ferramentas que buscam estimular o aprendizado de programação em blocos, sendo elas: Scratch[1], Scratch[2] Jr e Tinkercard[3], usando a avaliação heurística proposta por [1] que tem como finalidade avaliar interfaces para localizar falhas de usabilidade.

A usabilidade foca em princípios de boas práticas de design de interface. Um sistema precisa ser simples e fácil de utilizar, possibilitando que o usuário obtenha sucesso na execução de suas tarefas. Segundo [2] a essência da usabilidade é o acordo entre interface, usuário, tarefa e ambiente. As 10 heurísticas de Nielsen foram criadas para auxiliar no processo de avaliação de interfaces.

A avaliação heurística foi realizada por uma inspetora mestranda em informática aplicada, que possui experiências com o uso de ferramentas no ensino de programação. Nas interfaces das três ferramentas citadas, foram encontrados 19 problemas de usabilidade[4], sendo 6 problemas no Scratch, 9 no Scratch Jr e 4 no Tinkercard. Entre eles existem problemas de gravidade catastrófica, onde os usuários não podem reverter ações realizadas, dessa forma não possuem liberdade e controle e as ferramentas não apresentam funcionalidades de prevenção de erros.

Foram propostas soluções para apontar as necessidades de atualizações nas ferramentas, que proporcionariam uma melhor usabilidade. No Scratch para o problema de ajuda e documentação, onde a tela é estática e não apresenta nenhuma ajuda, tutorial ou auxílio para orientar sobre a utilização das funcionalidades, a solução proposta é**:** adicionar signos metalinguísticos dinâmicos (balões) explicativos no sistema, para orientar os usuários.

No Scratch Jr, para o problema de liberdade e controle do usuário, onde não há acesso a uma lixeira para recuperar projetos já excluídos (caso um projeto seja excluindo indesejavelmente não existe possibilidade de recuperá-lo), a solução proposta é: adicionar uma lixeira para arquivar os projetos deletados, dando a oportunidade para o usuário resgatá-los se necessário. Já que a exclusão indesejável de um projeto pode causar frustação para os usuários.

A ferramenta Tinkercard apresenta problemas de ajuda aos usuários, para reconhecer diagnosticar e se recuperar de erros. A mensagem de erro ao tentar efetuar o login, não é apresentada de forma clara para que todos os usuários possam compreender facilmente, pois, ao digitar de forma incorreta seu nome de login, senha ou ambos, a mensagem de erro apresentada ("o nome de usuários e a senha não coincidem"), leva a entender que as informações precisam ser idênticas, desta forma causando dúvidas. Como solução para esse problema foi proposto que é necessário seguir o padrão de mensagem de erro que é vista em outras ferramentas, onde a mensagem apresentada é: "Login ou senha incorretos. Digite novamente".

Como trabalhos futuros, a inspetora propõe aplicar o conjunto de heurísticas para outras ferramentas de programação em blocos e também em IDEs de programação como: Eclipse, NetBeans, Pycharm e Android Studio. Conseguir avaliar as ferramentas antes da utilizar é fundamental para analisar suas limitações e a viabilidade de uso, para então escolher uma ferramenta que possa proporcionar ao usuário uma melhor interação.

REFERÊNCIAS

[1] NIELSEN, Jakob; MOLICH, Rolf. Heuristic evaluation of user interfaces, CHI'90 Conferência. Seattle, WA: 1990, p. 249-256.

[2] CYBIS, Walter. BETIOL, Adriana Holtz. FAUST, Richard. Ergonomia e Usabilidade: conhecimentos, métodos e aplicações. 2. Ed. São Paulo: Novatec Editora, 2010.Conference Name:ACM Woodstock conference

Fica permitido ao(s) autor(es) ou a terceiros a reprodução ou distribuição, em parte ou no todo, do material extraído dessa obra, de forma verbatim, adaptada ou remixada, bem como a criação ou produção a partir do conteúdo dessa obra, para fins não comerciais, desde que sejam atribuídos os devidos créditos à criação original, sob os termos da licença CC BY-NC 4.0.
EduComp'21, Abril 26–30, 2021, Jataí, Goiás, Brasil (On-line)
©2021 Copyright mantido pelo(s) autor(es). Direitos de publicação licenciados à Sociedade Brasileira de Computação (SBC).

[1] Link de acesso ao Scratch: http://scratch.mit.edu/
[2] Link de acesso ao Scratch Jr: https://jfo8000.github.io/ScratchJr-Desktop/
[3] Link de acesso ao Tinkercar: https://www.tinkercad.com/login
[4] Link de acesso ao arquivo que contém todas as telas inspecionadas: https://drive.google.com/file/d/1cHYm5M3ux8nqqbp9tkAdGTGGA1bTvN1b/view?usp=sharing