

UM ESTUDO SOBRE A PROGRAMAÇÃO BASEADA EM BLOCOS

Natália Aragão Christ, Luiz Carlos Begosso

natalia.aragao.154@gmail.com, lbegosso@femanet.com.br

RESUMO: A programação baseada em blocos está se tornando a maneira pela qual os alunos estão sendo apresentados à programação de computadores e à ciência da computação. Neste artigo, discutimos o estado atual dos ambientes de programação baseados em blocos e revisamos a literatura sobre o aprendizado com ferramentas baseadas em blocos. A literatura mostrou que a programação inspirada em blocos pode ser um meio altamente eficaz de ensinar conceitos fundamentais de programação a estudantes de séries iniciais de cursos de programação de computadores.

PALAVRAS-CHAVE: Programação em blocos; educação de ciência da computação; algoritmos; programadores iniciantes.

ABSTRACT: Blocks-based programming is becoming the way that learners are being introduced to programming and computer science. In this paper, we discuss the current state of blocks-based programming environments and review what we know about learning with blocks-based tools. The literature has shown the blocks-inspired programming could be a highly effective means of teaching fundamental programming concepts to a wide student audience.

KEYWORDS: Block-based programming; Computer Science education; algorithms; naive programmers.

1. INTRODUÇÃO

Uma linguagem de programação visual (*Visual Programming Language* – VPL) permite ao usuário criar programas através de manipulação gráfica. A interação mais comum de modelos em VPLs são: arrastar blocos ao redor de uma tela; usar diagramas de fluxo, diagramas de estado e outros componentes; usar ícones ou representação

sem texto. A codificação baseada em blocos pode ser considerada como um modelo VPL e tornou-se altamente eficaz ao introduzir estudantes novatos em disciplinas introdutórias de programação de computadores em cursos da área de informática. Ao invés da tradicional programação baseada em texto, a codificação baseada em blocos envolve arrastar “blocos” de instruções dentro de um ambiente apropriado sem a preocupação com questões de sintaxe e qualidade do software. O presente trabalho tem por objetivo identificar e classificar ambientes de programação baseados em blocos como práticas pedagógicas para o ensino inicial de programação de computadores. Especificamente, pretende-se avaliar as características de tais ambientes e suas respectivas linguagens de programação baseadas em blocos.

2. LINGUAGENS DE PROGRAMAÇÃO BASEADA EM BLOCOS

Ferramentas baseadas em blocos foram elaboradas para produzirem recursos de ambientes de programação acessíveis e atraentes. A cor e a forma dos comandos, a organização dos blocos e o mecanismo de construção dos programas são facilmente navegáveis e exibidos de forma a contribuir para apoiar os estudantes novatos na escrita de programas. Neste sentido, foram estudadas as seguintes linguagens de programação em blocos: Alice, MIT App Inventor, Blockly Games, Code.org, Gameblox, Pencil Code, Microsoft MakeCode e Scratch. Nas próximas seções serão apresentados os ambientes de programação em blocos objeto desse trabalho.

2.1 ALICE

Construir ambientes tridimensionais na prática, não é uma tarefa muito fácil, principalmente para aqueles que querem aprender logo, pois têm pressa. O software Alice foi projetado justamente para facilitar o aprendizado dessa área tão crescente: a computação gráfica.

Alice (2019) é um ambiente de programação em 3D, principalmente, para iniciantes. Através de uma série de recursos e novidades, torna muito simples criar algo uma animação tridimensional nesse programa. Contar uma história, conceber um jogo ou um vídeo são algumas das possibilidades dessa ferramenta de iniciação a computação gráfica. O ambiente do programa Alice é atrativo: possui inúmeros elementos 3D de modelo, como pessoas, objetos, animais e veículos, que serão animados posteriormente pelo usuário.

Para facilitar o uso, a equipe de desenvolvedores do software Alice, preparou um material didático para facilitar o entendimento dos usuários. Os recursos incluem livros, aulas, programas de exemplos, bancos de testes, entre outros. A figura 1 ilustra a tela inicial de programação do Alice.

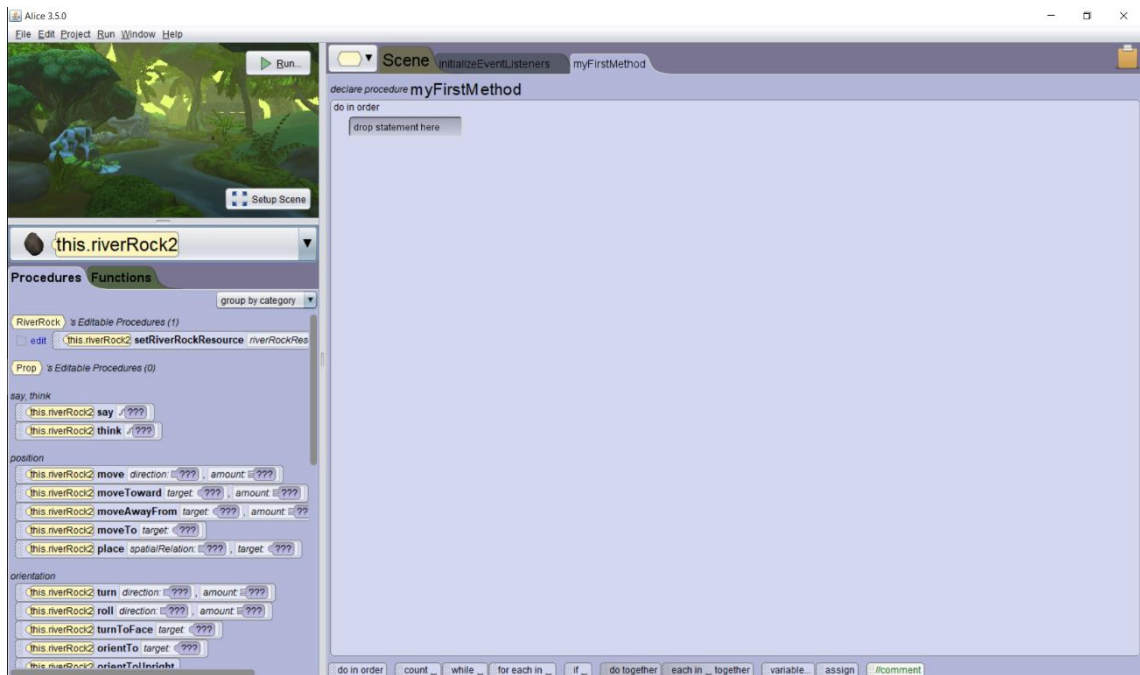


Figura 1 – Tela de programação

2.3 MIT APP INVENTOR

O App Inventor é um software web criado pela universidade americana *Massachusetts Institute of Technology* (MIT) que permite desenvolver aplicativos Android usando um navegador da Web e um telefone ou emulador conectados (CORDEIRO, 2019).

É possível criar aplicativos selecionando componentes para o mesmo e montando blocos que especificam como os componentes devem se comportar.

Toda criação do aplicativo é feita de forma visual, juntando peças com peças como se fosse um quebra-cabeça.

A interface do aplicativo aparece na tela à medida em que os comandos são executados, para que o projeto possa ser testado conforme cria-se.

Quando terminar o projeto, é possível empacotar tudo e produzir um aplicativo executável para instalar em outros celulares. A figura 2 ilustra a tela principal do App Inventor.

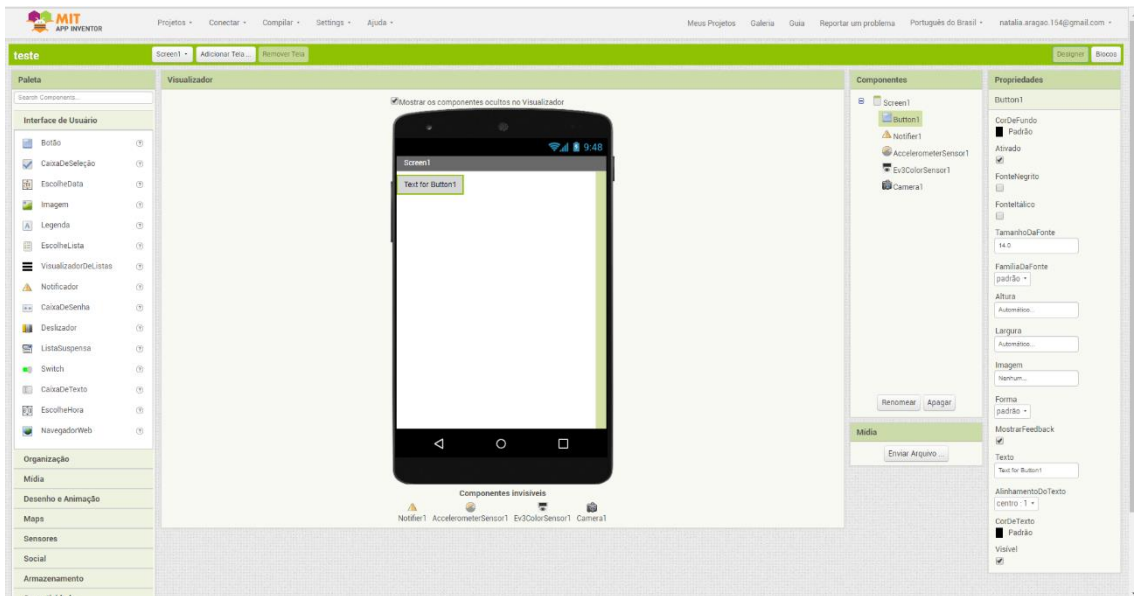


Figura 2 – Tela principal

A figura 3 apresenta a tela de programação com as estruturas disponíveis para a elaboração do código fonte.

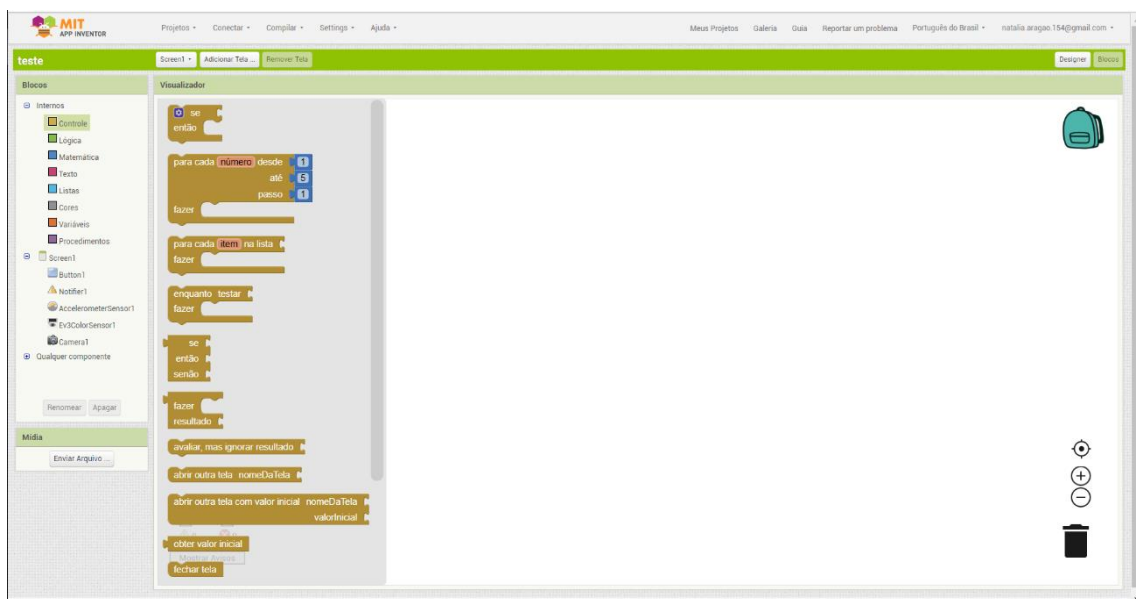


Figura 3 – Tela de programação

2.4 BLOCKLY GAMES

Blockly Games (2019) foi desenvolvido para crianças que não tiveram experiência anterior com programação de computadores. No final desses jogos, os jogadores estão prontos para usar as linguagens convencionais baseados em texto. A figura 4 ilustra a tela principal do Blockly.

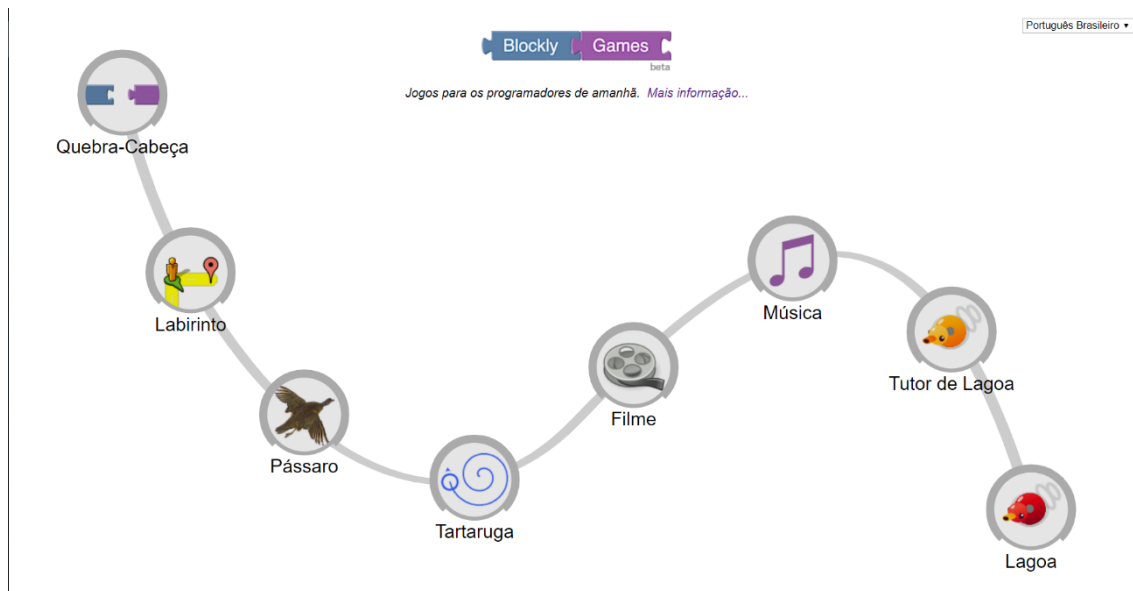


Figura 4 – Tela Inicial

Blockly Games é composto por uma série de jogos educativos que ensinam programação. Cada jogo representa uma fase, que estão listadas a seguir:

Quebra-cabeça é uma rápida introdução às formas de Blockly e como as peças se encaixam.

Labirinto é uma introdução à loops e condicionais. Começa de forma simples, mas cada nível é mais difícil que o anterior.

Pássaro é um mergulho profundo em condicionais. O controle de fluxo é explorado em condições cada vez mais complexas.

Tartaruga é uma explicação mais detalhada em loops. Use loops aninhados para pintar um quadro. Em seguida, publicar a sua arte ao Reddit para o mundo ver.

Filme é uma introdução a equações matemáticas. Usar a matemática para animar um filme. Em seguida, publicar o filme em Reddit para o mundo ver.

Tutor de lagoa apresenta a programação baseada em texto. Com níveis que alternam entre a programação em blocos e um editor de Java Script.

Lagoa é um concurso aberto para programar o pato mais inteligente. Pode-se usar a programação em blocos ou Java Script.

Blockly Games é um projeto do Google para incentivar os programadores iniciantes a desenvolverem habilidades de programação. Os jogos são projetados para serem de autoaprendizagem.

2.5 CODE.ORG

A plataforma **Code.org**, visa oferecer o acesso à tecnologia e a programação para crianças e adolescentes do mundo inteiro com a ajuda de atividades online ou offline. As atividades propostas estão disponíveis para qualquer nível de conhecimento, do iniciante ao *expert*, para crianças a partir dos 4 anos de idade.

Chamado de A Hora do Código do Brasil, o projeto do Code.org pretende ajudar a desmistificar a ideia de que a programação é algo difícil, e permitir que pais, professores e alunos de todo o país conheçam a programação de uma forma mais descontraída. O espaço também tem recursos para educadores poderem aprender e ensinar ciência da computação a seus alunos (PROGRAMAÊ, 2019). A figura 5 apresenta a tela de programação com as estruturas disponíveis para a elaboração de um projeto do Code.org.

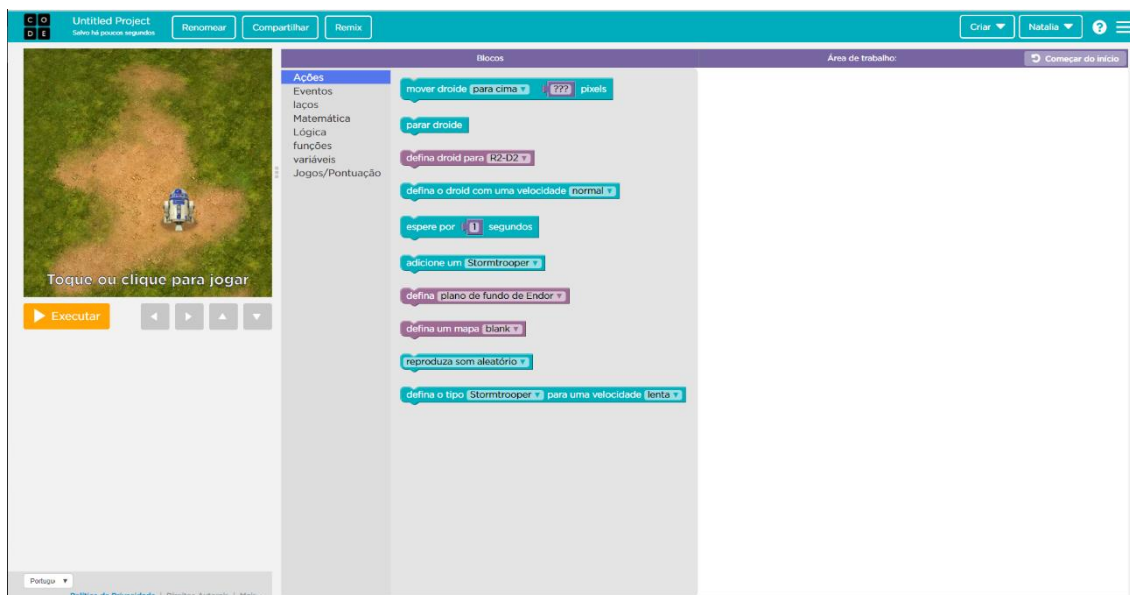


Figura 5 – Criação de um projeto

A figura 6 apresenta a tela da primeira fase do projeto A Hora do Código.

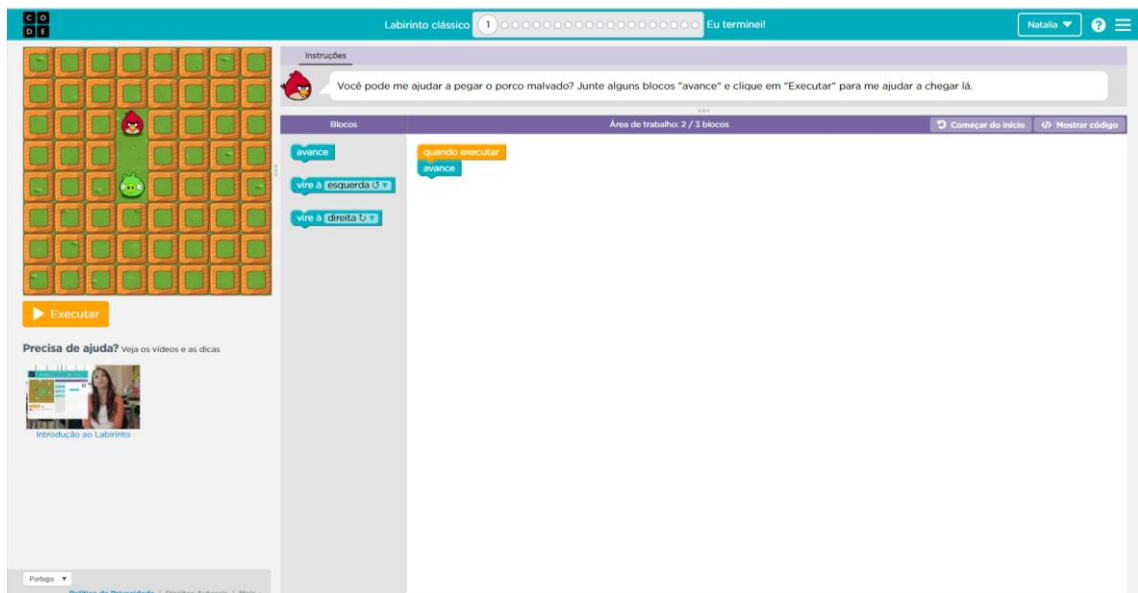


Figura 6 – Hora do código

2.6 GAMEBLOX

Gameblox (2019) é um ambiente de programação baseado em blocos para criar jogos on-line para a Web e dispositivos móveis. Ele está sendo desenvolvido no MIT STEP Lab e é usado em cursos da edX, incluindo Introdução ao Design de Jogos. É um editor de jogos que usa uma linguagem de programação baseada em blocos para permitir que qualquer um faça jogos. É gratuito e não é necessário fazer *downloads*. A figura 7 apresenta a tela de programação com as estruturas disponíveis para a criação de jogos.

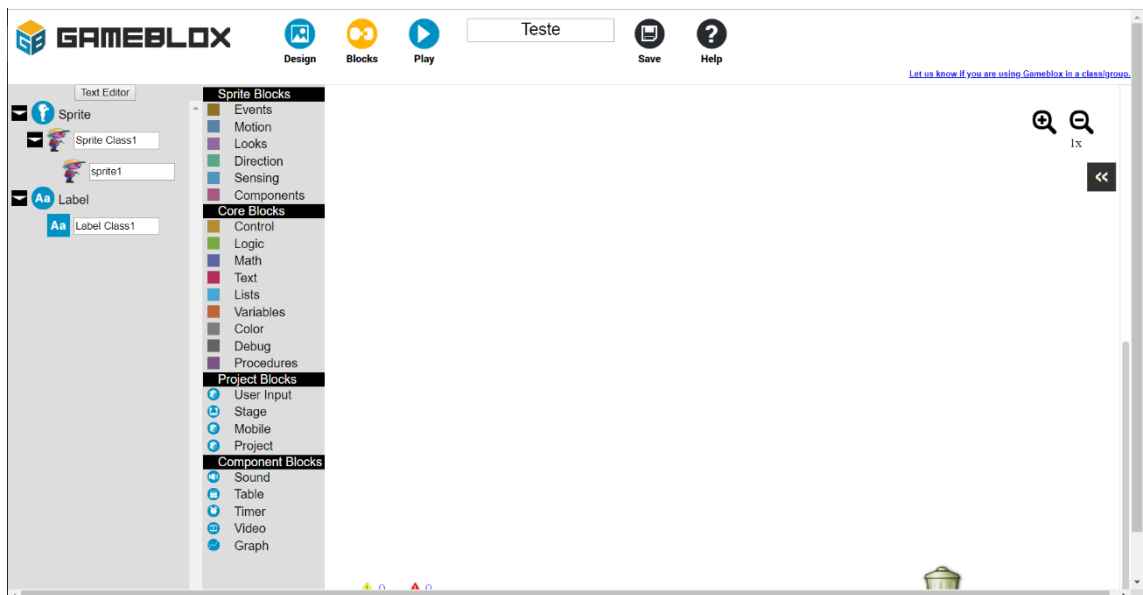


Figura 7 – Criação de projeto

2.7 PENCIL CODE

O Pencil Code (2019) é um site de programação colaborativo para desenhar arte, tocar música, criar jogos e para experimentar funções matemáticas, geometria, gráficos, páginas da web, simulações e algoritmos. Os programas estão abertos para que todos possam ver e copiar. A linguagem Coffeescript foi utilizada para desenvolver o Pencil Code.

O Pencil Code também pode ser usado para explorar e aprender Java script, HTML e CSS: quando estiver pronto, basta encontrar o botão "engrenagem" para ajustar os idiomas. A figura 8 apresenta a tela de programação para a elaboração de projetos.

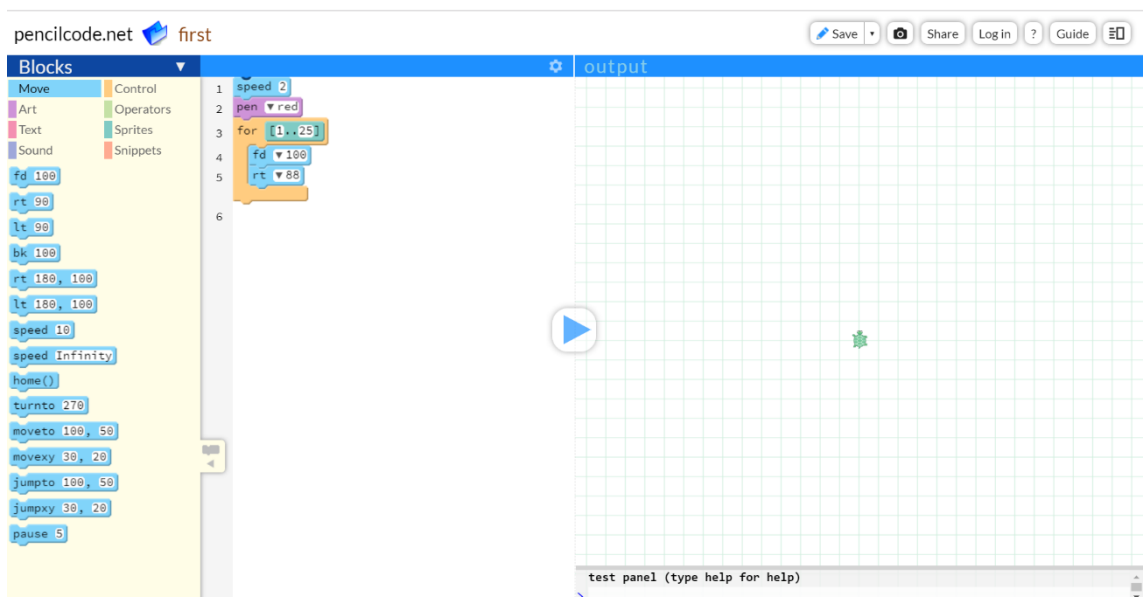


Figura 8 – Tela de programação

2.8 MIROSOFT MAKECODE

O Microsoft MakeCode (2019) é uma plataforma gratuita de software livre para criar experiências envolventes de aprendizagem em ciência da computação que pavimentam um caminho de progressão para a programação do mundo real.

Com o MakeCode é possível ter acesso a diversas ferramentas que ajudam na aprendizagem, como o simulador interativo que fornece aos alunos um feedback imediato sobre como seu programa está sendo executado e facilita o teste e a depuração do código.

O MakeCode é um editor de blocos, onde os alunos iniciantes em codificação podem começar com blocos coloridos, arrastar e soltar em seu espaço de trabalho para construir seus programas.

O software também possui um editor Java Script, que quando estiverem prontos, os alunos podem mudar para um editor de Java Script completo com trechos de código, dicas de ferramentas e detecção de erros para ajudá-los. As figuras 9 e 10 apresentam, respectivamente, os projetos disponíveis para aprendizagem e a tela para a programação para o desenvolvimento do projeto.

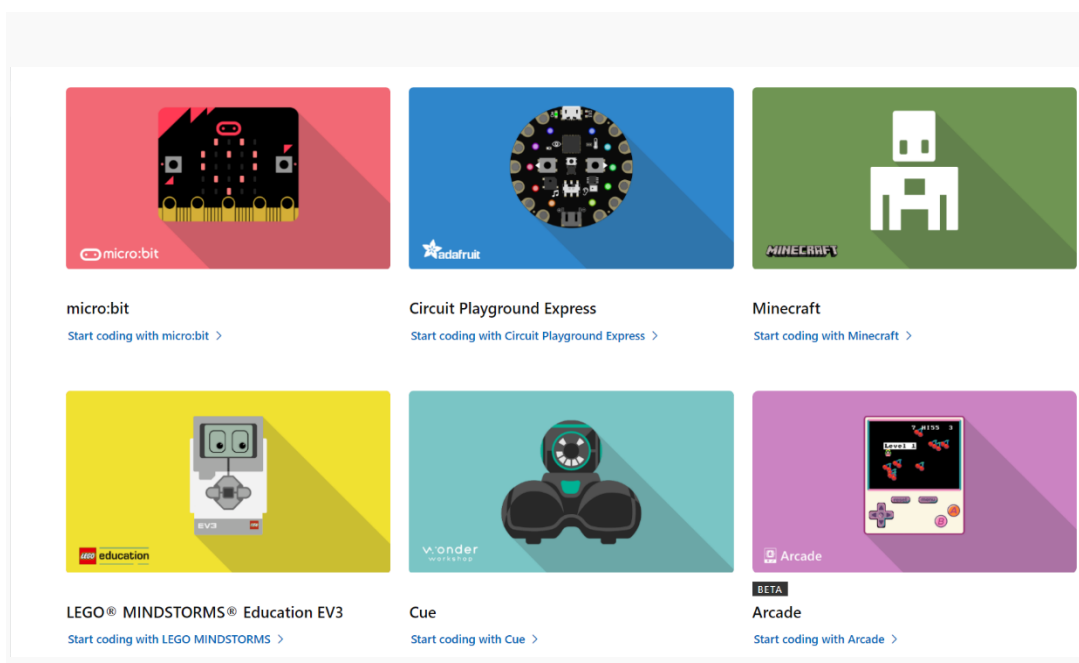


Figura 9 – Projetos

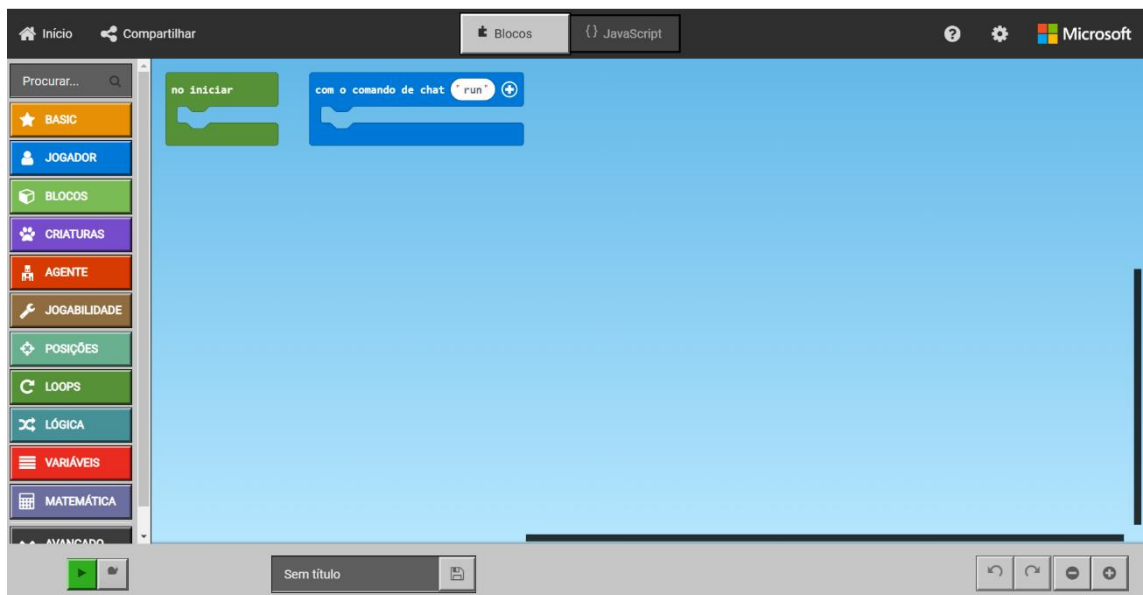


Figura 10 – Desenvolvimento do projeto

2.9 SCRATCH

O Scratch (2019) é um programa desenvolvido pelo Instituto de Tecnologia de *Massachusetts* (MIT, experiente no desenvolvimento de ferramentas educativas para crianças na idade escolar) e pelo grupo KIDS da Universidade de Califórnia, Los Angeles.

Scratch é um novo contexto de programação visual e multimídia baseado em Squeak. Está destinado à criação e promoção de sequencias animada para a aprendizagem de programação de forma simples e eficiente. Oferece uma interface intuitiva e muito fácil de compreender. No Scratch é possível trabalhar com imagens, fotos, música, criar desenhos, mudar aparência, fazendo com que os objetos interajam, tornando a programação inteiramente visual. Ele recupera o modelo construtivista do Logo e dos E-Toys Squeak.

O público alvo do Scratch são crianças do ensino fundamental, permitindo a construção de animações, trabalhando também, numa abordagem interdisciplinar, ou seja, utilizar conceitos das disciplinas escolares para montar projetos específicos e permitir que as crianças aprendam de forma criativa e saborosa. A figura 11 apresenta a tela de programação com as estruturas disponíveis para a elaboração do código.

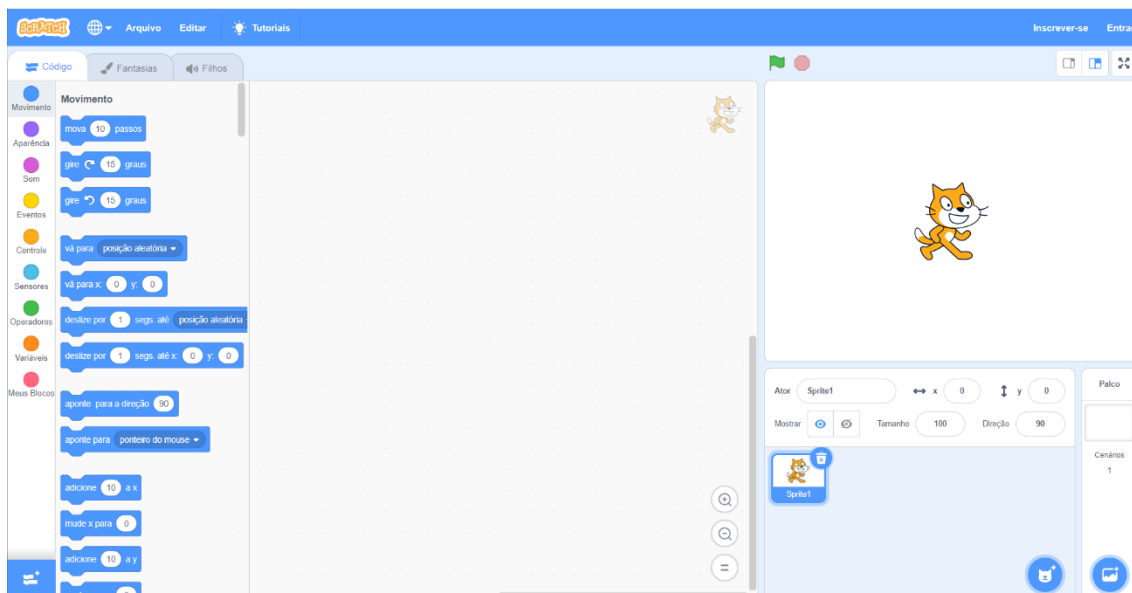


Figura 11 – Criação do projeto

3. METODOLOGIA

A metodologia utilizada nesta pesquisa centra-se numa abordagem de natureza descritiva, no qual foi realizado um levantamento bibliográfico seguido de avaliações dos ambientes e linguagens de programação baseadas em blocos.

A pesquisa descritiva busca essencialmente a enumeração e a ordenação de dados, sem o objetivo de comprovar ou refutar hipóteses exploratórias, sempre possibilitando uma nova pesquisa explicativa, fundamentada na experimentação. Envolve o uso de técnicas padronizadas de coleta de dados como questionário e observação sistemática. Assume, em geral, a forma de levantamento. Neste tipo de pesquisa os pesquisadores “procuram caracterizar e identificar opiniões, atitudes ou crenças de um determinado grupo ou população” (CASARIN, 2012, p. 41).

Para o desenvolvimento deste projeto foram definidas quatro etapas para o estudo das linguagens, são elas: identificar o estado da arte na área de linguagens de programação baseada em blocos; identificar o estado da arte na área de ambientes de programação em blocos; instalar e estudar as linguagens obtidas; e avaliar as linguagens e seus respectivos ambiente de programação.

Após os estudos dos referidos ambientes considerou-se identificar e classificar as principais características de tais linguagens e, para isso, estabeleceu-se os seguintes critérios: exigência de conhecimento prévio de programação?; interface facilita a

interação com os usuários?; qual é o idioma do código para programação?; existe disponibilidade de documentação para o aprendizado?; o ambiente enfatiza práticas de programação?; e, por último, o sistema favorece o aprendizado/criação? Para este último critério é importante diferenciar aprendizado de criação. Nesse contexto, a aprendizagem estabelece ligações entre certos estímulos e respostas, enquanto a criação atua no ato ou efeito de criar, de usar a imaginação e a criatividade.

4. CONCLUSÃO

O presente trabalho teve como objetivo principal identificar e classificar ambientes de programação baseados em blocos como práticas pedagógicas para o ensino inicial de programação de computadores. A análise dos ambientes foi realizada sobre: Alice, MIT App Inventor, Blockly Games, Code.org, Gameblox, Pencil Code, Microsoft MakeCode e Scratch. A partir dos critérios estabelecidos na seção anterior, construiu-se a Tabela 1 onde foi possível cruzar as características dos ambientes de programação em blocos com os respectivos critérios.

Tabela 1 – Resultado dos critérios avaliados

Linguagem	Critérios Avaliados				
	Conhecimento prévio em programação	Interface facilita a interação	Idioma do código	Documentação para aprendizado	Aprendizado/criação
Alice	Não	Não	Inglês	Sim	Criação
AppInventor	Não	Sim	Inglês/Português	Sim	Criação
Blockly	Não	Sim	Todos idiomas	Sim	Aprendizado
Code.org	Não	Sim	Inglês/Português	Sim	Aprendizado
Gameblox	Não	Sim	Inglês	Sim	Criação
PencilCode	Não	Sim	Inglês/Português	Não	Criação
MakeCode	Não	Sim	Inglês/Português	Sim	Criação
Scratch	Não	Sim	Inglês/Português	Sim	Criação

Ao final desse artigo, tendo avaliado as ferramentas citadas de acordo com os critérios estabelecidos, pode-se concluir que há um predomínio de linguagens baseadas em blocos que favorecem o desenvolvimento de jogos, o que é muito bom, pois a área de jogos tende a despertar o interesse em crianças e jovens.

Nenhuma das linguagens estudadas requer conhecimento prévio de programação, visto que é um ambiente para ensino de estudantes novatos. A maioria das linguagens já está em português, sendo muito eficiente quando se trata de ensinar crianças.

Este trabalho apresentou ferramentas visuais para o ensino de programação, onde cada uma delas contribui com uma forma de aprendizado.

A ferramenta Alice, ensina programação orientada a objetos e programação em ambiente gráfico virtual 3D.

AppInventor auxilia na construção de aplicações móveis na plataforma Android usando um construtor de interface gráfica de usuário baseada na web.

Blockly utiliza blocos gráficos conectáveis para representar os conceitos de código como repetição, condição, *loops* entre outros.

Code.org é uma plataforma que disponibiliza recursos para aprender e ensinar ciência da computação, através de jogos educacionais digitais, de forma gratuita.

Gameblox é uma ferramenta projetada para o design de jogos, fornece um ambiente para criar jogos divertidos incorporando algumas tarefas de programação mais complexas como, por exemplo, conteúdos de física tais como colisões e gravidade dentro dos blocos.

No Pencil Code é possível elaborar rotinas com blocos de ações, a base da lógica de programação. Ele enfatiza a criatividade, para desenhar ou criar músicas a partir do pensamento matemático.

O Microsoft MakeCode abrange conceitos básicos da programação que ajudam a desenvolver a criatividade, raciocínio computacional e habilidades de colaboração, através das ferramentas disponibilizadas.

Finalmente, a plataforma Scratch, é um ambiente de programação visual e multimídia que ensina a produzir animação com ou sem som e ajuda na aprendizagem de sequências de programação.

5. REFERÊNCIAS

ALICE. Página do software Alice. Disponível em: <https://www.alice.org/>. Acesso em 03 de dez. 2019.

APPINVENTOR. Página do software Appinventor. Disponível em: <https://appinventor.mit.edu/>. Acesso em 03 de dez. 2019.

BATISTA, Esteic Janaina Santos. Uma Análise de Ambientes de Programação em Blocos Conclusão de Curso. Universidade Federal do Mato Grosso do Sul.

BLOCKLY. Página do software Blockly. Disponível em: <https://blockly-games.appspot.com/?lang=pt-br>. Acesso em 03 de dez. 2019.

BROWN, Neil C.C.; MÖNIG, Jens; BAU, Anthony; WEINTROP, David. Future Directions of Block-based Programming. In: Proceeding of SIGCSE '16 Proceedings of the 47th ACM Technical Symposium on Computing Science Education, 2016. Memphis, Tennessee, USA, march, 2016. P.315- 316.

CASARIN, Helen de Castro Silva; CASARIN, Samuel José. Pesquisa Científica da teoria à prática. Curitiba: Ed. IBPEX, 2012.

CODE.ORG. Página do software Code.org. Disponível em: <https://code.org/>. Acesso em 03 de dez. 2019.

CORDEIRO, Filipe. App Inventor: Guia de Criação de Apps. Disponível em: <https://www.androidpro.com.br/blog/desenvolvimento-android/app-inventor/>. Acesso em 05 de dez. 2019

GAMEBLOX. Página do software Gameblox. Disponível em: <https://gameblox.org/>. Acesso em 03 de dez. 2019.

PENCILCODE. Página do software PencilCode. Disponível em: <https://pencilcode.net/>. Acesso em 03 de dez. 2019.

PROGRAMAÊ. Página oficial. Disponível em: <http://programae.org.br/hora-do-codigo-conheca-o-projeto-que-incentiva-a-programacao/>. Acesso em 05 de dez. 2019.

MAKECODE. Página do software MakeCode. Disponível em: <https://www.microsoft.com/en-us/makecode>. Acesso em 03 de dez. 2019.

SCRATCH. MIT Media Lab. 2018. Página do software Scratch. Disponível em: <https://scratch.mit.edu/statistics/>. Acesso em 03 de dez. 2019.