



*IMPA - INSTITUTO NACIONAL DE MATEMÁTICA PURA E APLICADA
PROFMAT - PROGRAMA DE MESTRADO PROFISSIONAL EM MATEMÁTICA
EM REDE NACIONAL.
TRABALHO DE CONCLUSÃO DE CURSO*

PROGRAMAÇÃO COM SCRATCH: DESENVOLVENDO RACIOCÍNIO ALGORÍTMICO

Rio de Janeiro
13 de julho de 2017

João Alvaro de Souza Baptista

PROGRAMAÇÃO COM SCRATCH: DESENVOLVENDO RACIOCÍNIO ALGORÍTMICO

Trabalho de conclusão de curso apresentado por João Alvaro de Souza Baptista ao Programa de mestrado profissional em matemática em rede nacional - PROFMAT - IMPA/RJ, como requisito para obtenção do grau de Mestre.

Linha de pesquisa: Educação Matemática - Programação.

Orientador: Paulo Cesar Pinto de Carvalho

Rio de Janeiro
13 de julho de 2017

Resumo

O ensino de lógica de programação para crianças e jovens já não é uma proposta nova. Esta ideia, no entanto, sofre com os mesmos obstáculos conhecidos na utilização de novas tecnologias no ensino, que são a falta de preparo dos profissionais de educação e uma melhor fundamentação das atividades propostas com estes recursos. O objetivo deste trabalho é apresentar uma proposta de intervenção didática onde, através da programação de jogos e animações, será ensinada lógica de programação, utilizando a linguagem Scratch.

O trabalho apresenta uma sequência didática dividida em 8 módulos, subdivididos em atividades, nas quais os estudantes produzem jogos ou animações, explorando os conceitos de programação e permitindo o aprendizado e desenvolvimento da lógica de programação. São também disponibilizados um manual de utilização da ferramenta do projeto e os links dos vídeos utilizados em cada uma das atividades propostas.

Espera-se que, de posse deste texto, profissionais de educação sejam motivados a iniciar, ou mesmo continuar, projetos semelhantes com estudantes, desenvolvendo o pensamento algorítmico em nossos jovens.

Palavras-chave: Programação, Scratch, Raciocínio Algorítmico.

Abstract

The teaching of programming logic for children and teenagers is no longer a new proposal, but this idea faces the same obstacles in the use of new technologies in teaching, namely, the lack of preparation of education professionals, and a better foundation for the activities to be carried out with these resources. The objective of this work is to present a didactic intervention proposal where, through the programming of games and animation, logic programming will be taught, using the Scratch language.

This work presents a didactic sequence divided into 8 modules, subdivided into activities, in which students will produce games or animations, exploring programming concepts, leading to the learning and development of programming. In this work, we also provide a manual for the use of the project tool, as well as the links of the videos used in each of the proposed activities.

It is expected that, with this text, education professionals will be able to initiate or even continue similar projects with more students, developing algorithmic thinking in our young people.

Key words: Programming, Scratch, Algorithmic thinking.

Dedico este trabalho a minha esposa Daiana Rosa, que além de uma grande companheira é um exemplo como profissional e principalmente como mãe. Dedico também a minha filha Isadora Rosa, que sem dúvida hoje é a minha maior motivação na vida, espero que tudo que faço sirva de exemplo para o seu caminho. E principalmente a minha mãe Conceição Aparecida (em memória) que é a grande responsável e incentivadora das minhas conquistas.

Agradecimento

Em primeiro lugar agradeço a Deus, que sempre me proporcionou as batalhas mais justas, para que eu pudesse aprender sempre, mesmo nas que fui derrotado.

Agradeço aos meus pais, Álvaro Baptista (em memória) e Conceição Aparecida (em memória) que me deram a vida e me proporcionaram condições para que hoje eu possa alcançar mais esta etapa na minha história.

Agradeço a minha esposa Daiana Amália Rosa Ferreira, a melhor companheira que a vida poderia ter me dado, por toda sua paciência e amor incondicional principalmente nessa etapa de nossa vida.

Agradeço a minha filha, que em pouco mais de 2 anos de vida, tem me trazido diversos ensinamentos.

Agradeço ao meu orientador, Paulo Cezar, por me apresentar ao tema deste trabalho, prestando auxílio e orientação durante este processo, me dando a oportunidade de conhecer e o professor que tanto admiro de longa data.

Agradeço aos professores do IMPA, pelos cursos dados durante este programa.

Agradeço a CAPES, que através de sua bolsa permitiu que eu durante o curso pudesse custear meu deslocamento até as aulas.

Agradeço a todos os meus amigos, pois sem vocês momentos difíceis provavelmente seriam muito mais difíceis.

Agradeço a cada um dos meus alunos que tenho e tive, que são os grandes motivadores do meu progresso.

SUMÁRIO

Lista de Tabelas	9
1 Introdução	10
1.1 Introdução	10
1.2 Problematizando	11
1.3 Proposta de intervenção didática	12
1.3.1 Criação do Clube de Programação Scratch	12
1.3.2 Objetivos Gerais	12
1.3.3 Objetivos Específicos	13
1.4 Metodologia	13
1.5 Organização do trabalho.	14
2 Embasamento teórico.	15
2.1 Construcionismo de Papert	15
2.2 Novas tecnologias na educação.	16
2.3 Programação como metodologia educacional.	17
3 O Scratch e outras linguagens em bloco	19
3.1 O Scratch	19
3.2 Scrath desenvolvendo habilidades e competências.	20
3.3 Possibilidades Pedagógicas.	20
3.4 Outros	21
3.4.1 CS-First.com.	21
3.4.2 Code.org.	22
3.4.3 APP-Inventor.	23
4 Sequência didática do Projeto.	24
4.1 Módulo 00 - Apresentação do projeto	25
4.2 Módulo 01 - Criando uma apresentação	27
4.3 Módulo 02 - Projeto corrida maluca.	27
4.4 Módulo 03 - Projeto jogo do labirinto.	28
4.5 Módulo 04 - Projeto jogo da plataforma.	29
4.6 Módulo 05 - Projeto jogo do gato e rato.	31
4.7 Módulo 06 - Projeto corrida no aquário.	32
4.8 Módulo 07 - Projeto rpg.	33
4.9 Módulo 08 - Projeto jogo Flying bat.	34
5 Análise dos resultados.	36
5.1 Análise dos módulos.	36

6 Considerações.	42
6.1 Trabalhos Futuros	42
6.2 Considerações Finais	43
A Tutorial Scratch	45
A.1 Conceitos Básicos	45
A.1.1 Tela inicial.	46
A.1.2 Iniciando e parando script	47
A.1.3 Atalhos de edição	47
A.1.4 Menu do scratch	48
A.1.5 Tela cheia	48
A.1.6 Palco	49
A.1.7 O Ator (Sprite)	52
A.1.8 Programação no Scratch - Conceitos básicos e comandos.	55
B Links para materiais de apoio ao projeto	66
B.1 Links para os vídeos	66
B.2 Links para projetos exemplo para módulos	67
Referências Bibliográficas	68

LISTA DE FIGURAS

3.1	Figura do site https://scratch.mit.edu/about/	19
3.2	Site oficial CS-First https://www.cs-first.com/en/home	21
3.3	Área da turma (clube)	21
3.4	Sequências de programação.	21
3.5	Guia do educador da sequência de criação de jogos.	22
3.6	Vídeo aula da Atividade 01 da sequência de criação de jogos.	22
3.7	Site oficial Code.org https://code.org	22
3.8	Vídeo motivacional Code.org.	22
3.9	Área de atividades do Code.org.	22
3.10	Site oficial APP-Inventor http://appinventor.mit.edu	23
3.11	Área de visualização do smartphone.	23
3.12	Tela de programação.	23
4.1	Site oficial - imagem de 02/04/2017	25
4.2	Primeira tela do cadastro.	26
4.3	Segunda tela do cadastro.	26
4.4	Terceira tela do cadastro.	26
4.5	Quarta tela do cadastro.	26
5.1	Exemplo de projeto.	37
5.2	Exemplo de projeto.	37
5.3	Exemplo de projeto.	37
5.4	Exemplo de projeto.	37
5.5	Exemplo de projeto.	38
5.6	Exemplo de projeto.	38
5.7	Exemplo de projeto.	38
5.8	Exemplo de projeto.	39
5.9	Exemplo de projeto.	39
5.10	Exemplo de projeto - Tela inicial.	40
5.11	Acessando planeta "Terra".	40
5.12	Acessando Saturno.	40
5.13	Exemplo de projeto.	41
6.1	Arduino V 1.6 para Linux.	42
6.2	Site Scratch para Arduino.	42
6.3	Programação de um Arduino feito em Scratch for Arduino.	42
A.1	Site oficial do scratch - imagem de 10/12/2016	45
A.2	Ambiente de programação da versão web: https://scratch.mit.edu	46
A.3	Botão iniciar e parar - imagem de 10/12/2016	47

A.4	Atalho de edição dos objetos no palco - imagem de 10/12/2016	47
A.5	Menu do Scratch - imagem de 10/12/2016	48
A.6	Escola de idioma.	48
A.7	Menu arquivo	48
A.8	Menu editar	48
A.9	Scratch em tela cheia - imagem de 19/12/2016	48
A.10	Dimensões do palco - imagem de 19/12/2016	49
A.11	Palco com eixo cartesiano - imagem de 19/12/2016	49
A.12	Palco com eixo cartesiano - imagem de 19/12/2016	50
A.13	Palco com eixo cartesiano - imagem de 19/12/2016	50
A.14	Biblioteca de palcos do Scratch - imagem de 19/12/2016	51
A.15	Ambiente para pintar um novo cenário. - imagem de 19/12/2016	51
A.16	Ator (Sprite) - imagem de 10/12/2016	52
A.17	Fantasias do ator - imagem de 10/12/2016	53
A.18	Área dos atores.	53
A.19	Configuração do ator.	53
A.20	Menu de inclusão de novos atores - imagem de 10/12/2016	54
A.21	Biblioteca de atores do Scratch - imagem de 19/12/2016	54
A.22	Ambiente para pintar atores. - imagem de 19/12/2016	54
A.23	Área de programação - imagem de 19/12/2016	55
A.24	Exemplo de algoritmo incorreto.	55
A.25	Exemplo de algoritmo correto.	55
A.26	Lista de categorias dos comandos - imagem de 19/12/2016	55
A.27	Opções de instrumento do comando tocar o tambor.	59
A.28	Tipos de instrumentos.	59
A.29	Ambiente de criação de Variáveis e/ou Lista. - imagem de 19/12/2016	60
A.30	Ambiente de criação de Variáveis e/ou Lista. - imagem de 19/12/2016	60
A.31	Mais opções do operador função. - imagem de 19/12/2016	64
A.32	Aparência básica da categoria mais blocos. - imagem de 19/12/2016	65
A.33	Criando um novo bloco. - imagem de 19/12/2016	65
A.34	Lista de dispositivos de extensão. - imagem de 19/12/2016	65

LISTA DE TABELAS

A.1	Tela inicial so Scratch.	46
A.2	Atalho para incio e fim de execução.	47
A.3	Atalho de edição dos objetos no palco	47
A.4	Atalho para execução em tela cheia.	48
A.5	Atalho do palco.	50
A.6	Atalho do ambiente de pintura do palco.	52
A.7	Atalho para comandos do info dos Atores.	53
A.8	Comandos de movimento	56
A.9	Comandos de aparênciâa.	57
A.10	Comandos de som	58
A.11	Comandos de Som	59
A.12	Comandos de variáveis.	60
A.13	Comandos de lista.	61
A.14	Comandos de eventos.	61
A.15	Comandos de controle.	62
A.16	Comandos de sensores.	63
A.17	Comandos de operadores.	64
B.1	Links para vídeos das atividades 1/2	66
B.2	Links para vídeos das atividades 2/2	67
B.3	Projetos exemplo.	67

CAPÍTULO 1

INTRODUÇÃO

1.1 Introdução

Ao final de cada período letivo, ano, semestre ou bimestre, cada professor, em particular os de matemática (disciplina mencionada neste trabalho), tem o hábito de fazer uma reflexão sobre o quanto proveitoso foi o seu trabalho para auxiliar os seus alunos na importante tarefa de ensinar. Quão perto dos objetivos eles foram capazes de chegar? Se e onde cada um falhou, e como podem melhorar? Todas essas reflexões fazem, estes docentes, buscar incansavelmente uma preparação e um aperfeiçoamento das suas atividades. O estudo dos conceitos construcionistas e o preparo da sequência das atividades, apresentados neste trabalho, surgiram da necessidade de encontrar novos métodos para auxiliar os alunos na tarefa de desenvolvimento intelectual e fazer com que a matemática cumpra o real papel na vida deles, na construção de cidadãos conscientes, criativos, capazes de reconhecer padrões e resolver problemas, tornando-os cidadãos ativos tanto no processo de ensino aprendizagem quanto na vida.

Este trabalho surge das angústias e necessidades dos professores; angústia por perceber que a matemática vem sendo tratada como a grande vilã da educação e que em geral os estudantes estão em um ciclo que parece sem saída. Ter um baixo rendimento por não gostar de matemática e não gostam de matemática, pois tem baixo rendimento. E a necessidade de trabalhar conceitos, práticas diferentes, e de forma consistente embasada de maneira, que efetivamente, vai trazer algo novo aos estudantes. Em um tempo onde estamos imersos na tecnologia e informação, o computador ainda é utilizado como um mero coadjuvante no processo de ensino aprendizagem. Há alguns casos em que ele participaativamente do processo, mas é feito por profissionais sem preparo, ou utilizado em atividades sem planejamento nem embasamento educacional. Por isso, a proposta deste trabalho não é incentivar o uso do computador nas aulas como uma simples ferramenta, muito menos propor a substituição de alguma disciplina pelo ensino de lógica de programação. A proposta é apresentar de forma estruturada e embasada em atividades de programação que possam ser utilizadas de maneira paralela ao currículo e atividades atuais para que com todas as suas qualidades, o aprendizado de programação, por parte dos nossos estudantes, possa trazer reais desenvolvimentos a eles.

Com base na teoria construcionista de Seymour Papert, auxiliado pelas plataformas de ensino de programação CS-First (Google) e Code.org, e utilizando a ferramenta Scratch, este trabalho propõe uma sequência didática completa para o ensino de lógica de programação às crianças, tendo como motivação a criação de jogos. Para o aprofundamento e a experimentação desses estudos, foi criado um projeto no IF Fluminense – campus Macaé, com alunos do ensino médio integrado ao técnico, denominado Clube de programação em Scratch, onde foi aplicada a sequência proposta. É também objetivo deste trabalho, apresentar uma breve análise sobre a sequência aplicada durante o projeto, e seus desdobramentos, para que possa servir de auxílio aos profissionais de educação que tenham interesse no assunto.

Espera-se que este trabalho possa trazer uma contribuição real para aos colegas educadores que como eu partilham de similar angústia e necessidades.

1.2 Problematizando

A educação brasileira é pautada nos Parâmetros Curriculares Nacionais do ensino fundamental e médio. E os Parâmetros Curriculares Nacionais para o Ensino Médio (PCNEM, 2000) norteiam uma reformulação do ensino médio no Brasil, com a necessidade de responder aos desafios impostos pela globalização, no qual o sucesso do indivíduo nesse mundo depende da habilidade de resolver problemas do dia a dia. A busca é por uma educação voltada para formação social, intelectual que forme cidadãos capazes de refletir criticamente, experimentar com autonomia, criativos e que trabalhem em grupo tornando-se parte do processo de desenvolvimento da sociedade.

Por conta disso, a matemática toma um papel importante na vida do estudante, sendo mais do que um conjunto de conceitos e regras, ela trabalha habilidades como criatividade, organização de raciocínio, e resolução de problemas. E tem a missão de estabelecer uma relação entre os conhecimentos da sala de aula e a vida, os PCNs dizem o seguinte sobre isso:

... a matemática contribui para o desenvolvimento de processos de pensamento e a aquisição de atitudes, cuja utilidade e alcance transcendem o âmbito da própria matemática, podendo formar no aluno a capacidade de resolver problemas genuínos, gerando hábitos de investigação, proporcionando confiança e desprendimento para analisar e enfrentar situações novas, propiciando a formação de uma visão ampla e científica da realidade, a percepção da beleza e da harmonia, o desenvolvimento da criatividade e de outras capacidades pessoais. [1]

Contudo, as últimas notícias sobre o PISA (Programme for International Student Assessment) não são nada animadoras, que é um programa internacional que tem como objetivo avaliar os sistemas educacionais mundiais por meio de testes de habilidades e de conhecimentos. Eles são feitos com estudantes de 15 anos, a faixa etária a qual se espera que o estudante tenha alcançado o término da sua educação básica. No Brasil, o PISA é coordenado pelo INEP Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira. Segundo a reportagem da Edição do dia seis de dezembro de dois mil e dezesseis do site G1, link: <http://g1.globo.com/jornal-nacional/noticia/2016/12/brasil-caiu-em-ranking-de-educacao-em-ciencias-leitura-e-matematica.html>, o Brasil caiu no ranking de educação em ciências, leitura e **matemática**. Na reportagem, o Brasil ocupa 65º posição em matemática. A discussão proposta não é sobre a posição, mas sim dos motivos que levam a matemática a ser, cada dia algo inalcançável pelos nossos estudantes, não conseguindo, assim, cumprir o seu papel de aprendizado.

O PISA nos dá uma informação importante, os nossos estudantes chegam ao ensino médio com verdadeiros abismos de conhecimento e principalmente com a incapacidade de importantes habilidades matemáticas como criatividade, poder de decisão e organização de raciocínio. Surge, então a reflexão sobre o ambiente escolar que estamos colocando os nossos alunos. A escola ainda é pautada na escrita, na transferência unilateral do conhecimento, onde o professor centraliza o conhecimento e o transmite falando ou escrevendo e os alunos ouvindo e copiando. Em seu livro a Máquina das Crianças, Seymour Papert [2] conta uma parábola sobre viagem no tempo, onde professores que viajam 100 anos para o futuro e veem a mesma forma de ensinar dos tempos atuais de Papert (década de 90) veremos um trecho para refletir:

"Os professores viajantes do tempo na minha parábola que não encontraram na sala moderna qualquer coisa que não reconhecessem teriam tido muitas surpresas se tivessem simplesmente ido para a casa com um ou dois dos estudantes, pois lá teriam descoberto que, com uma diligência e vivacidade que a escola raramente consegue gerar, muitos dos estudantes tornaram-se intensamente envolvidos em aprender as regras e estratégias do que pareceu, à primeira vista, ser um processo muito mais exigente do que qualquer tarefa de casa. Eles definiram a disciplina como "videogame", e o que estavam fazendo como "brincando"."

Nossos estudantes vivem em uma sociedade onde impera o mundo digital, são definidos, segundo Marc Prensky, "nativos digitais"[3] e em seu mundo, eles aprendem, ensinam uns aos outros e são diretores do conhecimento. Desacreditar nessa maneira de fazer o conhecimento, é persistir na escola pautada na forma atual, afastando cada vez mais os nossos estudantes da realidade e da própria escola e cabe ao professor ser orientador nesse processo.

1.3 Proposta de intervenção didática

Trabalhar com programação desde cedo, vem atender as necessidades desse novo perfil de sociedade considerada como uma sociedade da mídia e nossos alunos, segundo Marc Prensky, "nativos digitais"[3]. Esta sociedade, assim como as próximas, é somente consumidora de tecnologia e, por isso, deve ser preparada para que possa interagir, transformar e protagonizar, utilizando recursos tecnológicos de forma ativa. Programar, ajuda o indivíduo a agir de forma proativa na busca de solucionar situações problemas do dia a dia.

Acreditando na programação como uma importante ferramenta didática, idealizei o clube de programação Scratch, utilizando como base as diversas fontes bibliográficas que pesquisei e a plataforma do Google CS-First "www.cs-first.com" com suas sequências didáticas.

1.3.1 Criação do Clube de Programação Scratch

Na busca de uma aula mais significativa para os alunos e na possibilidade de desenvolver suas habilidades, considerando a potencialidade do uso de tecnologia no processo ensino aprendizagem, pretendo por meios dos estudos da utilização de lógica de programação, potencializar as competências e habilidades como raciocínio estruturado, criatividade, resolução de problemas, autonomia e trabalho em equipe. Os próprios PCNs falam do computador como instrumento mais relevante ao impacto da tecnologia, exigindo da matemática um redirecionamento que favoreça o desenvolvimento de habilidades e procedimentos para que os indivíduos possam reconhecer e orientar-se nesse mundo em constante evolução. O domínio de uma linguagem de programação, ou simplesmente da lógica de programação, tem um papel muito importante no desenvolvimento matemático do indivíduo e é isso que pretendo demonstrar ao longo deste trabalho, os métodos e seus benefícios.

"... O impacto da tecnologia na vida de cada indivíduo vai exigir competências que vão além do simples lidar com as máquinas. A velocidade do surgimento e renovação de saberes e de formas de fazer em todas as atividades humanas tornarão rapidamente ultrapassadas a maior parte das competências adquiridas por uma pessoa ao início de sua vida profissional. O trabalho ganha então uma nova exigência, que é a de aprender continuamente em um processo não mais solitário. O indivíduo, imerso em um mar de informações, se liga a outras pessoas, que, juntas, complementar-se-ão em um exercício coletivo de memória, imaginação, percepção, raciocínios e competências para a produção e transmissão de conhecimentos." [1]

1.3.2 Objetivos Gerais

A proposta deste trabalho é falar sobre uma metodologia diferente, que tem como objetivo auxiliar os estudantes no desenvolvimento das habilidades e competências matemáticas. A ideia de ensinar lógica de programação, utilizando a linguagem Scratch, surge como uma forma de desenvolver habilidades como criatividade, raciocínio estruturado e trabalho em equipe, segundo o próprio site do software.

Falar de uma metodologia dessas, utilizando a tecnologia não é necessariamente um tema novo na educação, poderia inclusive dizer que é bem recorrente entre os educadores e pesquisadores. Como falarei da linguagem Scratch, este trabalho irá se basear na teoria construcionista de Seymour Papert [2]

1.3.3 Objetivos Específicos

- Analisar o comportamento dos alunos nas resoluções de problemas computacionais;
- Permitir e incentivar resoluções de problemas, utilizando a ferramenta Scrath;
- Trazer para a sala de aula propostas de jogos e buscar com os alunos os conceitos necessários em programação para criação dos jogos propostos;
- Colaborar para solidificação de conhecimentos em algoritmo dos estudantes e;
- Criação de um aplicativo/jogo/animação educacional que possa ser disponibilizado na rede para auxílio de outros projetos futuros.

1.4 Metodologia

Este trabalho será baseado em pesquisa sobre o método construcionista de ensino segundo Papert, confecção e execução de sequências didáticas a serem aplicadas com os alunos de ensino médio do Instituto Federal de Educação Ciência e Tecnologia Fluminense - IFF - campus Macaé.

Para aplicação das sequências, foi criado um projeto de extensão no instituto, formando uma turma, o clube de programação, que possibilitou a exploração das sequências e avaliação das suas aplicações. O projeto foi executado em etapas da seguinte maneira:

- Apresentação de uma oficina com duração de 3 horas no laboratório do instituto. Nos primeiros 30 minutos foram apresentados a ideia do projeto e a ferramenta que seria utilizada, o Scratch na versão Web. Durante, aproximadamente, 1 hora foram apresentadas algumas programações com o scratch, e foi pedido aos alunos que executassem as programações. Foram feitas propostas de atividades curtas, no restante da aula, com o intuito de familiarizar os estudantes a ferramenta.
- Após a oficina, foram listados os alunos que apresentaram interesse na participação do projeto para o cadastramento na ferramenta Moodle mantida no site <http://www.matemaniacos.com.br/moodle> que serve de apoio aos encontros presenciais.
Nesta plataforma foi criada uma sala virtual com o curso separado em módulos, com sequências de atividades para desenvolvimento da lógica de programação e tendo como objetivo a cada módulo a produção de um projeto de jogo como descrito na sequência deste trabalho.
- Durante o transcorrer do projeto, houve pelo menos 1 encontro semanal no laboratório de informática do instituto para o desenvolvimento dos módulos da plataforma Moodle, contando com a presença do professor que teve papel de orientador dos alunos na execução do módulo que cada um estivesse.
- Os alunos do curso não têm a obrigação de fazer os módulos da maneira em que estão listados na plataforma, pois um módulo não é pré-requisito ao outro, ou seja, o conhecimento de um não é necessário para o próximo. Permitindo que o estudante execute cada módulo, na sua ordem de preferência. Porém, somente quando terminaram os 8 módulos do curso, foi proposto a criação de alguma aplicação, jogo ou animação com o Scratch. Foram incentivados a fazer pelo menos 1 jogo ou animação, com um tema educacional que pudesse servir como ferramenta didática para alguma área do conhecimento, e de alguma série do ensino fundamental ou médio.

1.5 Organização do trabalho.

Este TCC está organizado em 6 capítulos e 2 apêndices:

Capítulo 1: No primeiro capítulo, apresento uma introdução do trabalho com a problematização que justifica a escolha do tema, uma proposta de intervenção com objetivos gerais e específicos a serem alcançados e a metodologia que será utilizada durante a pesquisa.

Capítulo 2: O segundo capítulo apresenta o embasamento teórico pesquisado durante o estudo do tema. Ele estará baseado em referenciais teóricos pesquisados na ferramenta scholar.google.com e banco de TCCs do Profmat, contemplando temas como o uso de novas tecnologias na educação, construcionismo de Papert, uso de lógica de programação no ensino e resolução de problemas.

Capítulo 3: O terceiro capítulo faz uma apresentação da linguagem Scratch, que é a linguagem utilizada na sequência didática proposta pelo projeto, com suas possibilidades pedagógicas, uma breve apresentação do APP-Inventor, que também tem origem no MIT e utiliza o mesmo tipo de programação em bloco que o Scratch, porém voltado para os smartphones e uma breve apresentação de dois importantes projetos mundiais de incentivo ao ensino de lógica de programação para crianças.

Capítulo 4: No quarto capítulo, faço uma apresentação da sequência didática, com detalhamento das atividades aplicadas durante os encontros do projeto do Clube de Programação.

Capítulo 5: No quinto capítulo, exponho uma análise dos resultados do projeto. Neste será apresentado uma breve análise módulo a módulo sobre o comportamento dos alunos durante as atividades, apresentando alguns exemplos criados, durante as mesmas.

Capítulo 6: No sexto capítulo, exponho minhas considerações sobre o tema e projeto. Apresentando também os desdobramentos e propostas de trabalhos futuros.

Apêndice A: Um tutorial dos comandos básicos do Scratch.

Apêndice B: Links dos materiais, vídeos e projetos no Scratch, utilizados durante a aplicação do projeto do Clube de Programação.

CAPÍTULO 2

EMBASAMENTO TEÓRICO.

2.1 Construcionismo de Papert

O construcionismo é uma filosofia educacional criada por Seymour Papert, que defende que a aprendizagem é mais efetiva quando o estudante constrói alguma coisa, para que ele ou outros possam utilizá-la. Esta construção pode ser uma frase, uma apresentação, uma vídeo aula, a criação de um jogo de computador entre outros. O construcionismo nega uma “verdade” educacional de que para uma melhor aprendizagem deve-se aperfeiçoar a instrução, ou seja, – se a escola não é perfeita, ela deve simplesmente ensinar melhor. Porém, vale dizer que não é proposta do construcionismo colocar em dúvida o valor da instrução, retirando apenas o seu protagonismo no processo de ensino aprendizagem.

A meta do construcionista é ensinar de forma a produzir uma maior aprendizagem a partir do mínimo de ensino. É evidente que não se pode atingir isto, reduzindo sistematicamente a quantidade de ensino e deixando todo o resto inalterado. Seymour Papert cita em seu livro, ”A máquina das Crianças”, que a principal mudança necessária ao construcionista se assemelha a um provérbio popular que diz: “se o homem tem fome, você pode dar-lhe um peixe, mas é melhor dar-lhe uma vara e ensiná-lo a pescar”. O construcionismo trabalha com a suposição de que as crianças farão melhor se descobrirem por si só o conhecimento específico que elas precisam, e a educação informal pode ajudar certificando-se que elas sejam apoiadas moral, psicológica, material e intelectualmente nos seus esforços. Assim, temos no construcionismo o tipo de conhecimento que as crianças mais precisam, é aquele que as levarão desenvolver um maior conhecimento. Voltando ao provérbio, devemos sim ensinar a pescar, mas para isso precisamos de uma boa vara que permita aqueles que não sabem pescar, a melhor condição possível para aprendê-la. A essa analogia, Papert justifica a presença do computador com softwares adequados como ferramenta fundamental ao processo de aprendizagem. Outro fator importante na pescaria é um local fértil em peixes, onde tenhamos a certeza de que haverá peixes para treinarmos a pescaria, e baseado nessa analogia Papert cita a importância da criação de um ambiente de aprendizagem próprio, como por exemplo laboratórios de química, física, matemática e outras disciplinas, para que a criança em um micromundo viva o seu aprendizado.

Um outro pilar do construcionismo é a defesa de que a aprendizagem efetiva ocorre quando a criança constrói algo, que ela ou outros irão utilizar. Essa construção pode ser qualquer coisa relacionada aquilo que ela está aprendendo, desde uma frase, uma apresentação, criação de uma música, um vídeo, um jogo de computador e outros, estimulando que a criança tente explicar com suas próprias palavras aquilo que está estudando. Este fato se baseia na ideia de que podemos ler uma página ou um exercício dezenas de vezes e ainda assim esquecê-lo em pouco tempo, ao passo que, na tentativa da construção de algum trabalho que explique aquele conteúdo para si e para os outros, a criança será levada a investigação e exploração do que está estudando, tornando-se agente do próprio conhecimento em uma aprendizagem mais significativa.

Como exemplo, utilizei uma oficina do projeto de Programação com Scratch. A ideia era trabalhar conceitos de física e matemática, como plano cartesiano, velocidade, aceleração, vetor e operações com números inteiros entre outros. Todos estes temas já são tratados nas aulas ditas "tradicionais", muitas vezes temidos pelos alunos e não assimilados nem despertando interesse deles, pois não veem significado. Sob o olhar do construcionismo, montei uma oficina sobre **movimento dos personagens**. A proposta era trabalhar movimento com as setas do teclado fazendo um personagem andar para direita, para esquerda e dar saltos. Nos movimentos laterais, pedi que criassem uma variável V de velocidade, que com os operadores de adição, alterariam sua posição fazendo $x = (x \text{ atual}) + v$ para andar para a direita e $x = (x \text{ atual}) - V$ para andar para a esquerda. Com as limitações do Scratch, os alunos foram levados a utilizar o operador de multiplicação para fazer o $-V$ multiplicando a variável por -1 . Outra tarefa proposta era de fazer o personagem dar um salto, e nesta atividade o personagem enquanto não tocar o solo teria efeito de queda. Para isso também seria utilizado uma variável, porém agora seria modificada a coordenada y que passaria a ser " $y = (y \text{ atual}) + (\text{variável de gravidade})$ ". Com todo um ambiente amigável que foi criado por conta do Scratch, os alunos tiveram acesso aos conceitos como vetores, operações com números inteiros de forma mais significativa.

Portanto, pode-se dizer que o construcionismo é pautado no protagonismo da criança no processo de ensino aprendizagem, onde o educador ganha um papel de orientador e fica responsável por proporcionar as condições necessárias ao desenvolvimento da criança, traçando objetivos e propondo sequências com atividades que permitam, gradativamente, a construção e solidificação do conhecimento.

2.2 Novas tecnologias na educação.

A escola de hoje possui poucas diferenças com a escola do século passado, e muitas vezes parece caminhar na mesma direção para o próximo século. A Educação, ao contrário de outras áreas, não tem apresentado grandes mudanças, mesmo que a sociedade tenha mudado drasticamente, ela clama por uma transformação nesta área. O mundo evoluiu tecnologicamente assim como a escola, mas de fato, esta tecnologia não tem uma participação efetiva no processo educacional. Nossos estudantes, nativos digitais, veem a escola como mera transmissora de conteúdo, um local lento, desinteressante e completamente fora da realidade. Os estudantes, quando fora da escola, mostram para a escola como eles aprendem, mas esta parece não querer ver, nem entender esse processo. A velocidade de troca de informação entre os jovens, ao criar canais de conteúdos visuais, e que muitas vezes ensinam sobre um assunto do qual dominam, e são seguidos por outras centenas de jovens, que aprendem e reproduzem esse conhecimento. Logo, tornando-se diretores do seu próprio aprendizado, reforça a importância para a escola em ter-se uma prática de uso dessas novas tecnologias em benefício da educação. Este processo não é de substituição da escola ou do professor, mas torna, cada vez mais evidente a necessidade de uma mudança na maneira de ensinar, pois mais do que ensinar coisas prontas, devemos ensinar a aprender, ensinar a criar e para isso, as novas tecnologias dariam um suporte necessário a esse processo. Muda o papel do professor, de mero transmissor de conhecimento, para mediador do aprendizado. Vejamos um artigo da Revista Brasileira da educação profissional tecnológica [4] em defesa do uso de recursos:

"O movimento em defesa do uso dos recursos tecnológicos no contexto escolar advém dos desafios das mudanças qualitativas que tanto se espera que sejam efetivadas nas práticas escolares, na perspectiva de que se altere o atual panorama em que ocorre processo ensino-aprendizagem, historicamente pautado na ênfase da transmissão e memorização dos conteúdos escolares."

Desse modo, a influência tecnológica docente está relacionada a habilidade do professor em escolher, entre as inúmeras possibilidades de uso dos recursos tecnológicos, aqueles que melhor se adequam ao objetivo de aprendizagem pretendido. É necessário ter sempre um planejamento prévio e organizado dessas atividades, para que elas não percam suas reais potencialidades.

2.3 Programação como metodologia educacional.

Para falar sobre programação como metodologia educacional, vamos tocar em dois aspectos muito importantes e que merecem destaque nas escolas. O primeiro é o protagonismo e a autoria dos alunos quando utilizam as chamadas Tecnologias Digitais de Informação e Comunicação TDIC e o segundo, é a homogeneidade das atividades propostas para um público completamente heterogêneo que a escola recebe.

É fato que nossos estudante tem grandes familiaridade com recursos os digitais e, segundo Marc Prensk [3], a interação das crianças e jovens com as tecnologias acarretou em uma modificação no modo de pensar e processar as informações, nas novas gerações. Para Prensk, a geração de nativos digitais tem o contato com as diferentes tecnologias intrínsecas ao seu crescimento. Mark Surman (2013), diretor-executivo da Mozilla Foundation, alertou para a necessidade de promover uma transformação na atual geração, o de consumidora para o de criadora de tecnologia. Falando abertamente sobre a necessidade de buscar fluência digital, não apenas para navegar ou utilizar ferramentas, mas também para criá-las.

No site do Scratch (scratch.mit.edu), ferramenta que dá origem a esse trabalho de pesquisa, temos uma citação que dá alguns bons motivos para utilização de linguagem de programação como metodologia de ensino:

”...O Scratch ajuda jovens a aprender a pensar de forma criativa, a raciocinar de forma sistemática e a trabalhar colaborativamente — habilidades essenciais para a vida no século XXI.”

De fato, essas habilidades e competências já são contempladas no ensino tradicional, porém sem sucesso pois, como discutido anteriormente, o espaço escolar não acompanha a vida do estudante. É exigido que o cidadão do século XXI tenha essas habilidades e a escola não consegue fazer com que isso aconteça, e a mesma não percebe que a programação de computadores é capaz de proporcionar ao aluno, desde cedo, a oportunidade de se colocar em situações reais, onde o estudante precisa refletir, testar, argumentar, comprovar, enfim assumir postura proativa, autoral e dinâmica deste processo de ensino e aprendizagem. Ronaldo Machado da Silva [5] destaca que o trabalho com a linguagem de programação vai muito além de se formar programadores, uma vez que:

”Aquilo que a criança aprendeu porque fez, após ter explorado, investigado e descoberto por si própria, além de contribuir para o desenvolvimento de suas estruturas cognitivas, reveste-se de um significado especial que ajuda a reter e transferir com muito mais facilidade aquilo que foi aprendido. Está imbuído na filosofia do Logo, como concebeu Papert, a ideia que a aquisição de um conhecimento não se dá em função do desenvolvimento, mas principalmente na maneira pelas quais as pessoas se relacionam com o meio, ou seja, as condições que este oferece para exercitar o pensamento qualitativo. Acredita na necessidade da pessoa controlar sua aprendizagem, poder reconhecer e escolher entre várias possibilidades de pensamento estruturado. (Silva, 2007, p. 2).”

Um outro fator importante sobre a utilização da programação como metodologia de ensino em áreas diversas, está no auxílio que ela pode dar ao estudante em como trabalhar a questão do erro. Ele consegue perceber a importância de aproveitar os seus erros de forma construtiva, para rever, refletir e procurar soluções, fazendo com que o conhecimento atual possa ser enriquecido, alcançando níveis superiores de um determinado assunto. Papert ainda descreve alguns requisitos do perfil de um cidadão do século XXI no seguinte trecho:

”Os cidadãos do futuro precisam lidar com desafios, enfrentar um problema inesperado para o qual não há uma explicação preestabelecida. Precisamos adquirir habilidades necessárias para participar da construção do novo ou então

nos resignarmos a uma vida de dependência. A verdadeira habilidade competitiva é a habilidade de aprender. Não devemos aprender a dar respostas certas ou erradas, temos de aprender a solucionar problemas. (Papert, apud Martins, 2012, p. 18)”

Tais habilidades independem da linguagem que será utilizada, porém devemos tomar o cuidado, pois podem acabar afastando nossos alunos ao invés de aproveitar os potenciais da programação. É por conta disso que, a linguagem escolhida para desenvolvimento deste trabalho e projeto é o Scratch.

CAPÍTULO 3

O SCRATCH E OUTRAS LINGUAGENS EM BLOCO

3.1 O Scratch

The screenshot shows the official Scratch website at <https://scratch.mit.edu/about/>. The page has a blue header bar with the Scratch logo, navigation links (Criar, Explorar, Discutir, Sobre, Ajuda), a search bar, and user options (Inscreva-se, Entrar). Below the header, the main content area features a large title "Sobre o Scratch". A text block explains that Scratch allows users to program their own games, animations, and interactive stories, and share them online. It highlights the project's educational goals of teaching creative thinking, systematic reasoning, and collaborative work. Below this, there are links for "INFORMAÇÕES PARA OS PAIS" and "INFORMAÇÃO PARA OS EDUCADORES". On the right side, there is a video player showing a Scratch logo animation with a play button, a progress bar at 01:37, and "HD" and "vimeo" buttons. At the bottom, there are two sections: "Quem usa o Scratch?" with a thumbnail image of children using computers, and "Aprenda a Programar, Programe para Aprender" with a link.

Figura 3.1: Figura do site <https://scratch.mit.edu/about/>.

O Scratch é uma linguagem de programação educacional, criada por *Seymour Papert* [2] a partir da linguagem Logo. Foi desenvolvida pela equipe Lifelong Kindergarten, coordenada por Mitchel Resnick do MIT - Massachusetts Institute of Technology, em 2007. O Scratch possui código aberto e gratuito podendo ser facilmente instalado em diversos sistemas ou utilizado na versão web no site oficial (<https://scratch.mit.edu>).

Com uma aparência amigável, o Scratch permite que crianças e jovens produzam facilmente seus próprios jogos, suas histórias e animações, incentivando também o compartilhamento de tudo que foi produzido com a comunidade.

O Scratch possui um estilo de programação próprio, que é conhecido como "programação em bloco", onde os comandos são blocos de encaixar e os algoritmos, são sequências de blocos encaixados. Podemos dizer, que o Scratch apresenta uma interface bem intuitiva, o que facilita o usuário na pro-

gramação até mesmo aqueles que nunca tiveram uma experiência com qualquer tipo de programação, em geral, não tem grandes dificuldade de programar em Scratch.

Mesmo com sua simplicidade, é possível criar diversos tipos de projetos com o Scratch como apresentações simples, desenhos animados, jogos e outros. Ele também conta com a possibilidade de trabalhar com outras interfaces, por exemplo Arduíno, Lego e outros, permitindo ao usuário programações de automação e robótica.

3.2 Scrath desenvolvendo habilidades e competências.

De acordo com o "*Programming Concepts and skills supported in Scratch*", as competências para a resolução de problemas e a concepção de projetos se desenvolvem com o raciocínio lógico, com a depuração de problemas, com a produção de ideias e com a atenção permanente e perseverante. Podemos, propiciar aos estudantes atividades desafiadoras, que serão executadas através de uma proposta de projeto, que mostre a necessidade da produção de algorítmos que os resolvam.

O trabalho com linguagens de programação, no nosso caso o Scratch, colabora com a criação autoral e o entendimento dos procedimentos por meio das respostas do computador. O aluno é posto em um ciclo de ações "imaginação - execução - reflexão - depuração", o que certamente favorece ao processo de construção do conhecimento. Portanto, nosso papel como educador, passa de transmissor de conhecimento para proposito de atividades desafiadoras aos nossos estudantes.

Dentro deste ciclo de ações, estamos fazendo com que o mesmo tenha ganhos em habilidades e competências indispensáveis ao cidadão do século XXI. Durante a imaginação do projeto, estamos trabalhando a criatividade do estudante tanto na construção gráfica do ambiente, que será criado no Scratch, como na organização dos algorítmos necessários para a execução do projeto. Na fase de execução, muitas vezes iremos colocar os estudantes em situações de erro, onde serão trabalhados habilidades de resolução de problemas. É muito importante que o aluno seja incentivado a testar cada passo de sua programação, para uma reflexão sobre a resposta que o computador deu a uma tarefa exercida. E por fim, a depuração, onde o estudante pode ser apresentado a diferentes algorítmos que, executem a mesma ação pedida, proporcionando uma boa noção sobre a eficiência e diferentes leituras sobre resolução de um mesmo problema.

Passando por todo esse processo, nossos estudantes são colocados em situações que cobram muito mais deles do que a forma padronizada que a escola propõe. E pensando no cidadão ideal para os dias de hoje, aprender programação se torna importantíssimo para a construção deste indivíduo, sendo o Scratch uma excelente ferramenta para isto, visto sua interface amigável e intuitiva.

3.3 Possibilidades Pedagógicas.

Durante a pesquisa e principalmente durante a execução do projeto com os alunos do IFF, campus Macaé, diversas possibilidades pedagógicas para a utilização desta ferramenta (Scratch) foram pensadas, algumas delas aplicadas e avaliadas, dando origem as sequências didáticas expostas no anexo A. O Scratch é uma excelente ferramenta para a busca de uma proposta de ensino significativo, em que o aluno, construtor do seu conhecimento, facilmente conecta seus resultados as aplicações na sua vida. Como a base do projeto foi o projeto do Google CS-First, as possibilidades citadas aqui serão as adaptações feitas com as sequências do CS-First.

Dentre as possibilidades pedagógicas, destacamos a criação de animações e a produção de jogos em diversos formatos tipo Quizes, tabuleiros, etc.

Nas animações, os alunos poderão criar histórias animadas, com diálogos, interações entre personagens, efeitos de som, etc. Estas animações são ideais para iniciar o trabalho de programação com o Scratch, visto que utilizam recursos básicos do programa e são de fácil construção. Possibilitando a prática da observação e análise rápida dos resultados a cada execução. As animações também são excelentes práticas pedagógicas com o objetivo de formar no aluno um raciocínio criativo e crítico e desenvolver a habilidade de trabalhar em grupo.

A produção de jogos tem um apelo muito grande entre crianças e jovens. A oportunidade de produzir um jogo de sua autoria é extremamente estimulante ao aluno. Acredito que, no momento da produção de jogos, o estudante tem um desenvolvimento real da linguagem de programação e das habilidades e competências proposta pelo "Programming Concepts and skills supported in Scratch". Esse recurso deve ser trabalhado desde de propostas de criação dos jogos mais simples, utilizando poucos recursos, aos mais complexos, fazendo com que o estudante possa compreender e solidificar a lógica dos comandos e rotinas de programação.

A produção de jogos educativos, utiliza um ambiente onde os estudantes tem maior intimidade, que é dos games, pode também ser trabalhado com temas das diversas disciplinas, propiciando uma interdisciplinaridade e conquistando mais os estudantes. Trazendo para o projeto, sempre que necessário, professores das outra áreas para que possam dar apoio aos conceitos durante a criação do(s) jogo(s).

3.4 Outros

3.4.1 CS-First.com.

O CS-First é um site mantido pelo Google com o objetivo de auxiliar educadores do mundo inteiro no ensino de linguagem de programação para crianças.

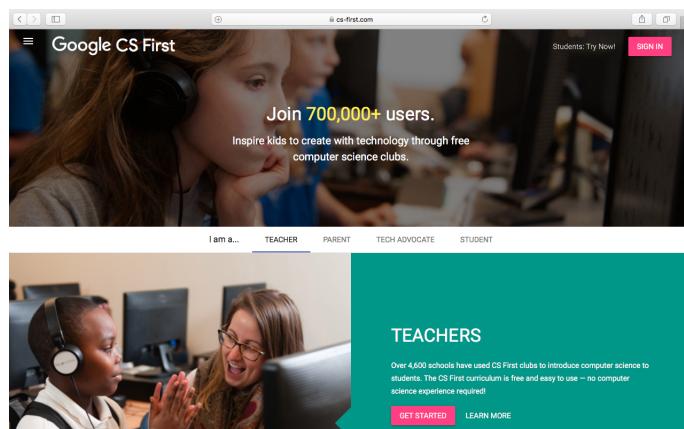


Figura 3.2: Site oficial CS-First <https://www.cs-first.com/en/home>.

Para acessar a plataforma, é necessário que o educador possua uma conta no Google. Com a conta de educador, o professor terá acesso ao suporte completo com guias de atividades, vídeos e materiais para orientação. O CS-First permite e incentiva que o educador crie um ambiente virtual para trabalhar suas turmas com as sequências existentes na plataforma sendo todas as atividades no CS-First baseadas no Scratch.

Figura 3.3: Área da turma (clube).

Figura 3.4: Sequências de programação.

Figura 3.5: Guia do educador da sequência de criação de jogos.

3.4.2 Code.org.

Acreditando na importância de ensinar linguagem de programação desde cedo, o projeto Code.org disponibiliza uma plataforma para auxiliar os educadores e os jovens na tarefa de aprender lógica de programação. Também utilizando a linguagem de programação em bloco, o Code.org possui diversas atividades separadas em grau de aprendizado, que vai do iniciante ao avançado com atividades e guias para auxiliar educadores na aplicação. E a plataforma ainda disponibiliza um treinamento para aplicação das atividades para os educadores.

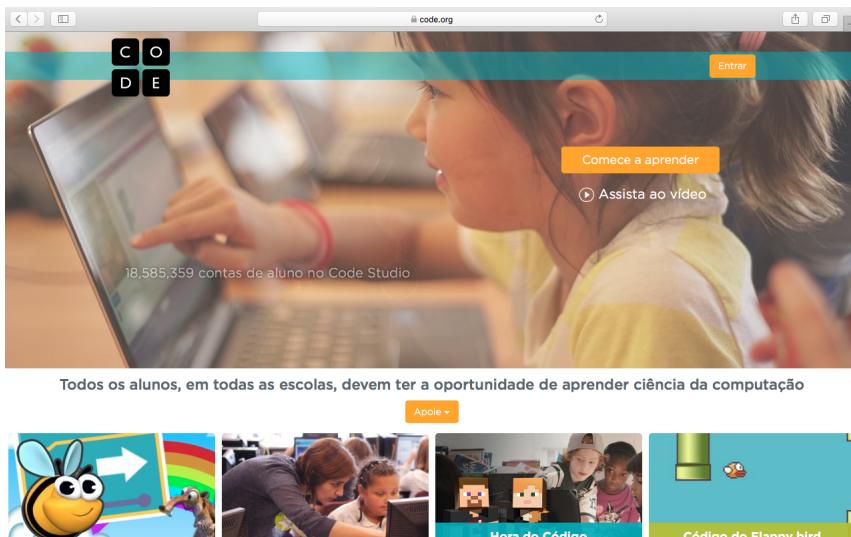


Figura 3.7: Site oficial Code.org <https://code.org>.



Figura 3.8: Vídeo motivacional Code.org.

Figura 3.6: Vídeo aula da Atividade 01 da sequência de criação de jogos.

Figura 3.9: Área de atividades do Code.org.

3.4.3 APP-Inventor.

Também desenvolvido pelo MIT, o APP-Inventor é uma linguagem de programação em bloco feita para criar aplicativos para Smartphones Android. No APP-Inventor, não temos o apelo gráfico existente no Scratch, mas temos o uso do smartphone tão presente na vida dos nossos jovens como uma boa motivação para o uso desta ferramenta.

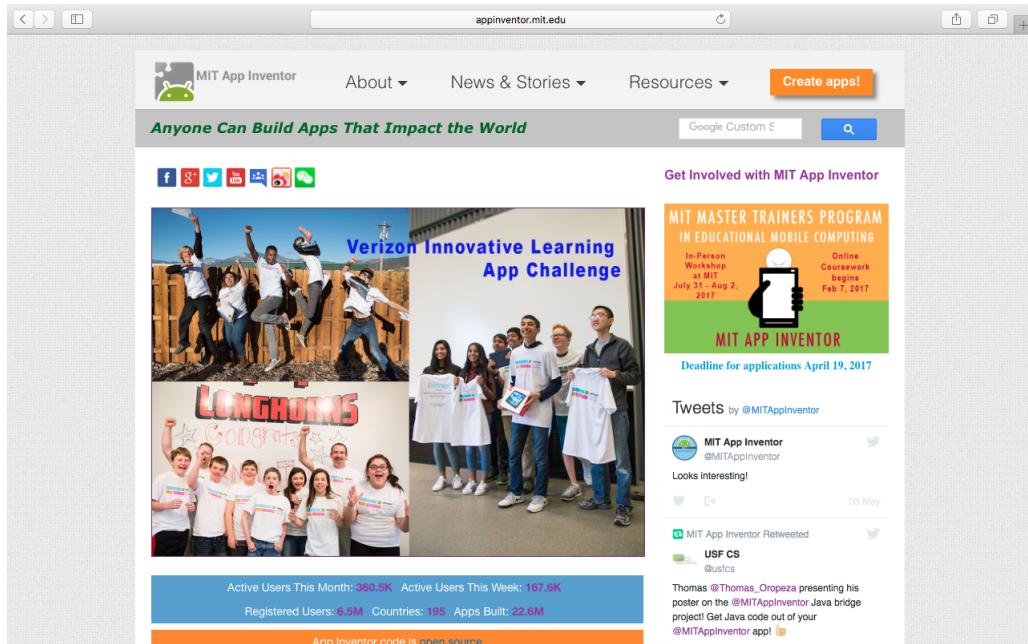


Figura 3.10: Site oficial APP-Inventor <http://appinventor.mit.edu>.

A programação do APP-Inventor é dividida na área de programação por uma tela onde podemos visualizar a tela de um celular.

Figura 3.11: Área de visualização do smartphone.

Figura 3.12: Tela de programação.

CAPÍTULO 4

SEQUÊNCIA DIDÁTICA DO PROJETO.

Neste capítulo serão expostas as sequências didática utilizadas no projeto Programação com Scratch, aplicadas com alunos do instituto Federal de Educação Ciência e Tecnologia Fluminense - IFF - campus Macaé. Ela tem como base as adaptações da metodologia e sequências de aula do projeto mantido pelo Google CS-FIRST <https://www.cs-first.com>. A proposta é ensinar programação utilizando como motivação a confecção de jogos.

Esta sequência foi dividida em 9 módulos, o primeiro de apresentação do projeto e da ferramenta Scratch, e os oito seguintes com pequenos projetos de jogos completos, tendo como objetivo principal introduzir a lógica de programação no universo desse jovens.

Além desta sequência e das orientações expostas neste trabalho, todo material em vídeo e projetos em Scratch utilizados ao transcorrer do projeto, estão disponível na rede e seus links estarão no Apêndice B para que possa ser utilizado como consulta, inspiração ou até mesmo na execução de projetos similares.

A seguir serão expostos cada módulo com suas respectivas orientações e objetivos. Porém, listarei inicialmente as primeiras, que denomino **orientações gerais**, pois são comuns a todos os módulos do projeto.

Orientações Gerais

- A cada novo encontro, certifique-se de que os computadores que serão utilizados tenham internet para utilizar o Scratch na versão web ou tenham a versão offline instalada nas máquinas;
- É interessante que cada estudante possua um fone de ouvido, de boa qualidade para que possa acompanhar vídeos, caso sejam utilizados, como no projeto original, para que possam utilizar, sem prejudicar aos outros estudantes, recursos de áudio oferecidos pelo Scratch em seus projetos. É importante que os alunos estejam livres para utilizar a criatividade durante a execução dos seus projetos;
- É importante que o orientador tenha modelos dos projetos que irá propor nos encontros para ilustrar os módulos. Ele deve dominar os conceitos utilizados em cada um e em cada atividade, conseguindo assim, executar com êxito o papel de orientador durante as oficinas;
- É interessante que o professor incentive os estudantes a compartilharem seus projetos. O Scratch possui uma comunidade onde ficam diversos projetos, que servem de motivação para outros, e este é também o intuito deste, trabalhar colaborativamente, contribuindo para que outros com ideias ou até mesmo com projetos finalizados, possuam alguma motivação educacional, ou serem utilizados como ferramenta didática;

- A cada atividade do módulo peça que os estudantes executem o projeto na busca de erros ou imperfeições. Isso faz com que eles criem o hábito de corrigir pequenos problemas, que são mais fáceis de resolver se comparado a grandes problemas.

4.1 Módulo 00 - Apresentação do projeto

Este módulo tem como objetivo apresentar o projeto e a ferramenta que será utilizada, o Scratch.

Atividade 01: Apresentação dos participantes.

Orientação: No início dessa atividade, o professor deve se apresentar mostrando um pouco sobre suas expectativas para o projeto e abrir um momento para que os alunos façam sua apresentação, falando um pouco de si e das suas expectativas. Após as apresentações deve acontecer uma rodada de perguntas para que os alunos possam tirar possíveis dúvidas sobre o projeto.

Objetivo: O objetivo dessa atividade é criar um ambiente de diálogo e interação entre os alunos e o professor, mostrando suas afinidades e complementos.

Atividade 02: Criação da conta no site do Scratch (<https://scratch.mit.edu>).

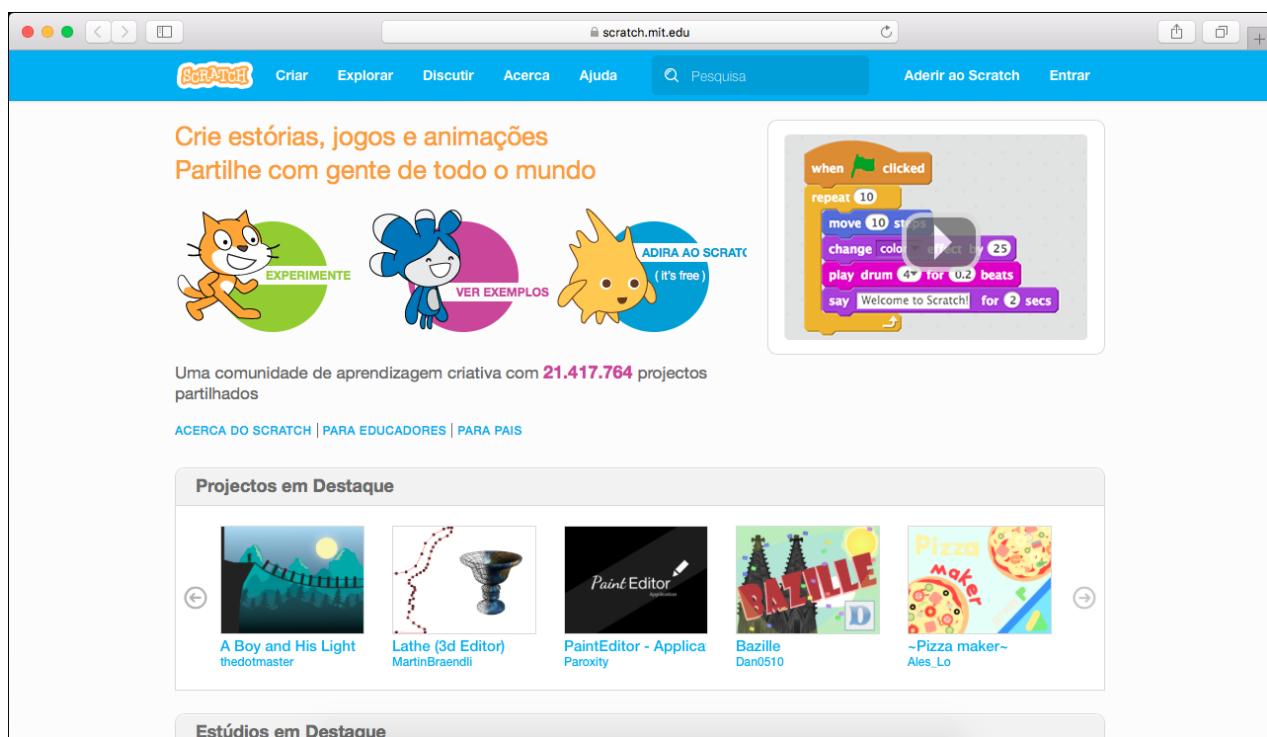


Figura 4.1: Site oficial - imagem de 02/04/2017

Para criação da conta o aluno deverá clicar em "aderir ao scratch" e preencher as telas a seguir:



Figura 4.2: Primeira tela do cadastramento.



Figura 4.3: Segunda tela do cadastramento.



Figura 4.4: Terceira tela do cadastramento.



Figura 4.5: Quarta tela do cadastro.

Orientação: O papel do professor nesse momento será de auxiliar os alunos na criação de uma conta como usuário no site do Scratch. Após esta etapa, faça uma navegação de reconhecimento pelo site, deixando que os alunos o vasculhem durante um tempo e discuta com eles o que encontraram de interessante.

Agora peça para que cliquem no botão criar, que dará acesso ao ambiente de programação online do Scratch, ambiente este, em que todo o projeto acontecerá. Quando todos estiverem com o ambiente de criação aberto, é hora de mostrar aos alunos mais sobre o Scratch. Faça um tour pelas funções do programa, você pode utilizar o tutorial, que está no apêndice deste trabalho para auxiliá-lo nos comandos. Mostre como manipular os personagens, palcos, importando da biblioteca ou, até mesmo, criando com a ferramenta de desenho do Scratch.

Faça um tour pelos comandos, mencionando de forma rápida a ideia de linguagem em bloco. Coloque alguns comandos na área de programação e execute-os mostrando suas funções. É importante incentivar que os alunos usem o comando "ajuda" sempre que estiverem com dúvida sobre a função de algum comando. Neste momento, você pode começar a pedir que repitam algumas programações e deixá-los livres para experimentar os comandos.

Objetivo: Esta atividade tem como finalidade apresentar a linguagem em bloco aos alunos, os comandos e a forma com que eles são utilizados para programar. Ao final, verifique se seus alunos entenderam o conceito de linguagem em bloco e a ideia de sequência lógica na execução de uma programação.

4.2 Módulo 01 - Criando uma apresentação

Neste módulo, os alunos vão trabalhar com os comandos de saída, utilizando a aba de comandos de aparência, e vão também manipular atores e palco, fazendo uma apresentação ou um diálogo entre dois personagens.

Atividade 01: Apresentação

Orientação: No início desta atividade é recomendado que os alunos deletem o personagem padrão para que façam toda a construção do cenário e personagem ou personagens que irão utilizar no projeto. Aproveite este momento de inclusão, para conversar com eles sobre as opções e programações possíveis para cada um desses objetos. Para a apresentação, é necessário que o orientador exiba os comandos de aparência, mostrando como fazer o ator falar ou pensar alguma coisa. Dê um tempo para que eles experimentem os comandos. Passado o reconhecimento, proponha a atividade do módulo, pedindo que façam com que o personagem escolhido por eles apresente cada um deles. Uma sugestão é a criação de um questionário que sirva de modelo do que pode ser apresentado. Outra proposta é fazer com que dois personagens conversem utilizando o controle de espera entre as frases.

Objetivo: Nesta atividade o aluno deverá manipular os comandos de aparência e ser capaz de fazer uma pequena apresentação no Scratch ou uma conversa.

4.3 Módulo 02 - Projeto corrida maluca.

A proposta deste módulo é um jogo de corrida. Para isso será necessário criar, ou importar, panos de fundo para as pistas, incluir personagens e trabalhar com fantasias, para troca de corredores. Vamos trabalhar com comandos de movimento e alguns efeitos.

Atividade 01: Criando as pistas.

Orientação: Peça que os alunos acessem o site do Scratch e façam login abrindo o programa. Feito isto, mostre-lhes como pode ser criado ou incluído um circuito. Utilize os panos de fundo para criar ou incluir outros circuitos do jogo.

Agora deixe que os alunos façam seus circuitos. Peça que utilizem os panos de fundo para criar mais de uma opção para pista, incentive que elas sejam criadas utilizando as ferramentas de desenho, mas os deixe livres para escolher importar algo, caso desejem.

Objetivo: O aluno deverá ser capaz de utilizar as ferramentas de desenho, de criar desenhos simples, de modificar desenhos existentes e de importar imagens prontas da internet para utilizar como panos de fundo dos projetos.

Atividade 02: Adicionando corredores.

Orientação: Com a área de programação aberta, inclua um novo ator que será o "Jogador 01" no nosso jogo de corrida, aproveite este momento para mostrar as configurações que podem ser feitas nos atores. Acesse a aba fantasia e fale sobre o conceito delas para criação de animação. Logo depois inclua outras fantasias para este ator, que servirão como personagens que o jogador 01 poderá escolher para jogar. Agora peça aos alunos que incluam um novo ator e novas fantasias que serão utilizadas pelo jogador 01. Mostre que eles podem "CLONAR" o ator para que este seja utilizado pelo jogador 02.

Objetivo: O aluno deverá ser capaz de incluir novos personagens "atores" e fantasias ou até mesmo criar seus próprios atores com a ferramenta de desenho.

Atividade 03: Criando comandos de movimento.

Orientação: Com a área de programação aberta, clique em um dos jogadores e explore os comandos de movimento. Você pode falar um pouco do sistema de posição cartesiana que é utilizado pelo Scratch, fale também sobre os limites da tela em x e em y. Esta é uma boa hora para apresentar os comandos de evento e movimento, que serão utilizados. Peça que tentem dar movimento ao personagem utilizando o que foi ensinado, e depois assistam o vídeo da atividade.

Depois, solicite aos alunos que criem os comandos de movimento para os seus dois jogadores, de maneira que seja possível a participação de duas pessoas. Peça que testem os jogadores, relatando os sucessos e os possíveis Bugs.

Objetivo: O aluno deve entender o sistema de posição e as diferenças entre os comandos de movimento como “mova” ou “mude x” ou “mude y”. No momento do teste, os alunos devem perceber que dois comandos não conseguem ser executados simultaneamente o que será resolvido na próxima atividade.

Atividade 04: Aprimorando comandos de movimento.

Orientação: Caso os alunos tenham percebido um bug, quando dois comandos são executados simultaneamente, e um comando interrompe o outro, trabalhe a parti daí. Caso não, motive-os a perceberem este problema afim de levá-los a buscar uma solução.

Para solucionar o bug, iremos trabalhar com o comando de “repita até que” e o operador lógico “não”. A ideia é que algo aconteça até que o botão não esteja sendo mais pressionado, permitindo que mais de um botão seja interpretado e mais de uma ação executada. Antes de pedir que os alunos assistam o vídeo da atividade, explique sobre o controle de repetição que será usado e sobre os operadores lógicos. Em seguida, peça que refaçam os movimentos dos jogadores e os testem.

Objetivo: O aluno deverá ser capaz de decidir quando utilizar um comando de repetição com um teste lógico de verdadeiro ou falso.

Atividade 05: Finalizando o projeto.

Orientação: Para finalizar o projeto, criaremos controles que mudam o personagem de cada jogador e um botão para alterar as pistas “panos de fundo”. No seu projeto exemplo feito pelo professor orientador, será criado um botão que ao ser clicado altera as pistas que podem ser utilizadas. Feito isto, crie uma maneira de trocar o corredor de cada jogador, por exemplo, a tecla 1 para alterar a fantasia do jogador 01 e a tecla 2 para alterar a fantasia do jogador 02. Solicite que os alunos façam o mesmo. Ao finalizar o projeto, peça que o salvem e, lhes mostre como compartilhar um projeto na comunidade Scratch.

Objetivo: O aluno deverá ser capaz de executar ações a partir de comandos de tecla e compartilhar projetos com a comunidade do Scratch.

4.4 Módulo 03 - Projeto jogo do labirinto.

Motivado pelo velho jogo do *Pacman*, o módulo 03, procurará produzir um jogo de labirinto, em que o jogador deverá conduzir o personagem até um ponto final. Neste módulo, o aluno irá aprender o conceito de condicional, utilizando o controle “se ... então ... ” e uma maneira diferente de dar movimento ao personagem.

Atividade 01: Criando o ambiente do jogo e incluindo personagens.

Orientação: Você pode iniciar esta atividade com um vídeo de jogos de labirinto ou com algum jogo com o mesmo tema feito no Scratch, isso dará ideias aos alunos sobre a criação do ambiente para o jogo e na escolha dos personagens.

A proposta de movimentação deste módulo é por meio do mouse. Nesta atividade, o aluno deve incluir o personagem que irá jogar e outros personagens (objetos) que servirão de obstáculos no palco. Feito isto, ele deve programar o movimento do personagem principal, através do clique do mouse, na aba de comandos evento.

Objetivo: Nesta atividade, o aluno irá conhecer outro método de movimento de personagem.

Atividade 02: Criando uma condição para derrota.

Orientação: O aluno irá trabalhar com o controle de condição “se... então...” criando uma condição que será utilizada quando o personagem esbarrar em algum obstáculo, fazendo com que ele volte para a posição inicial. Peça ao aluno que faça a programação, depois, devem perceber que o personagem segue o mouse mesmo quando toca o obstáculo. Mostre como resolver este Bug com o comando para parar todos os personagens.

Objetivo: O aluno deverá entender o funcionamento do controle “se ... então ...”, quando utilizá-lo e o sistema cartesiano de coordenadas.

Atividade 03: Criando uma condição para vitória.

Orientação: O aluno deverá criar uma condição de vitória. No vídeo desta atividade foi colocado um objeto e a condição de tocá-lo para vencer o jogo. Você também pode pensar em efeitos que o personagem fará caso vença o jogo, por exemplo, emitir um som ou girar na tela. Crie alguns efeitos para exemplo e peça que os alunos criem uma condição para vitória.

Objetivo: Esta atividade também tem objetivo de trabalhar o controle condicional. Incentive que os alunos utilizem comandos de condições diferentes do usado anteriormente podendo ser algum sensor, algum teste lógico, etc.

Atividade 04: Finalizando o projeto.

Orientação: Devemos colocar elementos que aumentem a dificuldade do jogo, objetos girando na tela, movimentando-se de um lado para o outro, fazendo movimentos repetitivos que aumentem a dificuldade. Utilize essa atividade para trabalhar os controles de repetição. Peça que os alunos incluam alguma dificuldade no seu jogo.

Objetivo: Após esta atividade, o aluno deverá dominar o conceito de repetição.

4.5 Módulo 04 - Projeto jogo da plataforma.

No módulo 04, os alunos irão criar um jogo com um personagem que irá subir andares, plataformas ou uma escada, visando alcançar um objeto que está no alto. Será utilizado conceito de gravidade em que o personagem não fique ”voando” na tela para causar efeito de salto. Utilize um jogo de exemplo para iniciar a ideia.

Atividade 01: Criando o cenário e efeito de pulo.

Orientação: O aluno deverá criar o palco onde acontecerá o jogo, incentive-os a utilizar a ferramenta de desenho, criando os próprios palcos. Lembre-se de orientá-los a colocar uma mesma cor para o chão e para as plataformas que forem sendo elaboradas, pois a mesma cor será utilizada posteriormente. É interessante que plataformas ou objetos, que servirão para dar apoio ao personagem, sejam atores e não parte do palco, para que possam ser programados no futuro. Feito isto, inicie a ideia do pulo, fazendo com que o personagem suba ao clicar em uma tecla escolhida. E para que o personagem volte ao chão, utilize o controle condicional. Crie um teste para a cor que foi escolhida para o chão, fazendo com que o personagem caia até atingir a cor determinada.

Objetivo: Estimular a criatividade do aluno e explorar os controles condicionais e eventos iniciados pelo teclado, que são formas de interação entre o usuário e o jogo. Veja se os alunos estão dominando tais conceitos, peça que façam o personagem executar outras ações por comandos de tecla como emitir som, por exemplo, ou quando o personagem saltar.

Atividade 02: Fazendo o personagem andar .

Orientação: O aluno deve fazer com que o personagem tenha movimentos de andar para direita e esquerda utilizando o teclado. Enfatize o movimento como no módulo 02, onde o personagem deve executar a ação até que a tecla não esteja sendo pressionada. Nesta atividade, seja mais detalhista, peça que o aluno utilize comandos de virar antes de andar. Introduza a questão da fantasia, que dá animação ao personagem, se possível.

Objetivo: Espera-se que o aluno tenha domínio de comandos de movimento, utilizando o controle de repetição. Ao final dela, deve-se ser capaz de fazer pequenas animações utilizando comandos de aparência.

Atividade 03: Dando objetivo ao jogo.

Orientação: O aluno deve utilizar o controle de condição para finalizar o jogo quando o alvo for atingido. Oriente-os, explicando que quando o alvo for alcançado, o personagem deve voltar para a posição inicial, utilizando o comando que trabalha com o par ordenado (x,y). O aluno também pode, por meio do mesmo, determinar o ponto de partida do personagem sempre que o jogo for iniciado.

Objetivo:Trabalhar o controle condicional ”se... então ...”e determinação de posição, utilizando coordenadas cartesianas.

Atividade 04: Colocando dificuldades no jogo.

Orientação: O aluno deverá programar eventos que aumentem a dificuldade do jogo. Ele pode fazer com que a plataforma ou o patamar, que o personagem vai subir, fique mudando de posição de maneira repetitiva, deslizando na tela ou rodando. Faça com que o aluno utilize comandos de repetição para executar esta ação. Outra ideia é trabalhar com comandos de aparência e fazer com que objetos mudem de tamanho, aumentando ao serem tocados e diminuindo quando não tocados. Utilize o controle ”se...então ... senão”para esta ação.

Objetivo: Estimular a criatividade do aluno e fazer com que ele domine controles de repetição, e condicionais, sendo capazes de executar ações repetitivas e estabelecer ações caso condições sejam verdadeiras ou falsas.

Atividade 05: Finalizando o jogo .

Orientação: Inicie a atividade motivando a busca por um bug muito comum. Quando os personagens precisam subir as plataformas. Quando utilizamos o comando ”se ... então ...” para o personagem tocar em algo, ele funciona quando o personagem toca com qualquer parte do corpo, permitindo que o mesmo suba por ”baixo da plataforma” ou atravessando paredes. Para corrigir este bug, utilize o sensor de cores, pois o controle condicional testa se as duas cores específicas estão se tocando e mude a cor do pé do personagem, fazendo com que ele só fique de pé na plataforma caso seu pé toque a mesma.

Objetivo: Trabalhar resolução de problemas nas programações utilizando criatividade para solucionar problemas.

4.6 Módulo 05 - Projeto jogo do gato e rato.

O Projeto do Gato e o Rato nos dá uma ideia de um jogo de fuga. A proposta é que um ou mais personagens fiquem executando movimentos aleatório na tela, e utilizando o ponteiro do mouse como guia o jogador fará um personagem fugir. Durante este módulo, o aluno irá trabalhar os conceitos de controle condicionais, controle de repetição e movimento por meio de perseguição ao cursor do mouse.

Atividade 01: Incluindo palco, personagens e dando movimento de perseguição .

Orientação: Após mostrar a ideia do jogo, peça aos alunos que construam o seu ambiente, com um cenário e dois personagens, o que irá perseguir e o que irá fugir. Como o personagem perseguidor ficará andando durante a execução do jogo, introduza o conceito do controle ”sempre” para movimentos.

Objetivo: Dominar os controles de movimento, repetição infinita e trabalhar com geradores de números aleatórios para criar dificuldade em jogos.

Atividade 02: Dando movimento ao jogador.

Orientação: Inicie a atividade pedindo que os alunos façam o personagem perseguir o ponteiro do mouse utilizando os comandos de movimento e o controle de looping. Para que o personagem não fique tremendo quando tocar o ponteiro do mouse utilize o controle condicional com operador de negação como descrito no vídeo da atividade.

Objetivo: Dar ao aluno mais habilidade com comandos de movimento e controles condicionais.

Atividade 03: Trabalhando com variável cronômetro.

Orientação: Apresente a variável cronômetro em sensores e faça com que o personagem perseguidor fale o tempo passado sempre que tocar no jogador. Peça que utilizem a ferramenta para duplicar personagem, aumentando o número de perseguidores, mostre que ela preserva a programação do personagem duplicado.

Objetivo: Conhecer o sensor/variável cronômetro e a ferramenta de duplicação e sua funcionalidade.

Atividade 04: Manipulando personagens.

Orientação: Os alunos irão utilizar a ferramenta de redução e aumento para configurar a aparência dos personagens. Apresente as ferramentas de som e peça para que os alunos o incluam sempre que alguma coisa acontecer.

Objetivo: Trabalhar criatividade e conhecer comandos de som.

Atividade 05: Tornando o jogo para dois.

Orientação: Uma possibilidade para este jogo é que um dos perseguidores seja controlado pelo teclado. Peça que os alunos deem controle a um dos perseguidores, utilizando o teclado e executando comando de movimento semelhante ao módulo 02.

Objetivo: Utilizar comandos de movimento por teclado na criação de um jogo para mais de um jogador.

Atividade 06: Criando efeitos no jogador.

Orientação: Utilizando os comandos de aparência, dê um efeito ao jogador sempre que tocado.

Objetivo: Utilizar comandos condicionais, de aparência e controles condicionais.

Atividade 07: Usando o cronômetro para dar dificuldade ao jogo.

Orientação: A variável cronômetro pode ser utilizada para dar dificuldade ao jogo, como por exemplo, ela pode ser responsável pela velocidade do perseguidor. Peça que os alunos coloquem a variável no lugar da quantidade de passos no mova 10 passos. Eles podem também utilizar os operadores aritméticos no auxílio desta operação.

Objetivo: Incentivar a criatividade e trabalhar com operadores aritméticos. Os alunos deverão ser capazes de utilizar algum recurso do Scratch na busca de tornar o jogo mais atraente.

Atividade 08: Finalizando o jogo.

Orientação: Nesta última atividade vamos criar um marcador de pontos utilizando uma variável. Aproveite este momento para explicar o conceito de variável para seus alunos e peça que eles pensem em alguma forma de criar uma pontuação. Por exemplo, ela pode ter uma relação com o cronômetro.

Objetivo: Entender o conceito de variável e trabalhar com o controle condicional.

4.7 Módulo 06 - Projeto corrida no aquário.

Este é um jogo para dois jogadores, eles usarão as teclas para dar largada e controlar direção dos personagens. A proposta é que seja criado um objeto para ser a linha de chegada e aumente a dificuldade para as próximas rodadas.

Atividade 01: Criando o ambiente do jogo .

Orientação: Os alunos irão criar o ambiente do jogo modificando palco, incluindo os personagens e algum objeto que possa servir como "linha de chegada", recomende aos alunos que use a ferramenta de desenho para criar tal objeto. Já o personagem deve iniciar de um ponto fixo e andar para frente até que toque na linha de chegada, fazendo com que volte ao ponto de partida.

Objetivo: Trabalhar com coordenadas cartesianas, controle de repetição e incentivar a criatividade.

Atividade 02: Continuando o movimento.

Orientação: Trabalhar com comandos que mudem a direção do personagem. Com uma ideia similar ao módulo 02, faça com que o personagem mude a direção.

Objetivo: Trabalhar com movimento e controle ”até que...”.

Atividade 03: Utilizando a ferramenta de clone.

Orientação: Nesta atividade vamos colocar no jogo um adversário que terá movimento aleatório similar ao do módulo 05, com objetivo de atrapalhar nossos jogadores. Para colocar mais personagens na tela, vamos utilizar o controle de criar clones. Apresente esta ferramenta e o seu conceito aos alunos e peça que eles criem os clones e esconda o personagem clonado da tela.

Objetivo: Trabalhar com controle de clones e controle de aparência.

Atividade 04: Criando sistema de pontuação.

Orientação: Utilizando variável, faça com que o personagem some um ponto sempre que tocar no objeto de chegada. Agora que o jogo está contando os pontos, o aluno pode criar uma condição para não ganhar ou perder pontos, esta condição poderia ser tocar no personagem clonado fazendo com que o jogador volte para o início e perca pontos. Uma ideia interessante é utilizar a pontuação para dar algum tipo de dificuldade ao jogo, como uma relação entre os pontos e a quantidade de clones ou a velocidade deles. Peça que os alunos utilizem este recurso.

Objetivo: Trabalhar a criatividade do aluno, trabalhar com variáveis, controles condicionais e operadores lógicos.

Atividade 05: Incluindo efeitos no jogo e incluindo outro personagem.

Orientação: Peça aos alunos que incluam algum efeito de aparência ou de som no jogo, dando personalidade ao jogo. Depois, peça que eles incluam um segundo jogador com suas programações e variáveis de pontos.

Objetivo: Trabalhar criatividade e controles condicionais para efeitos.

Atividade 06: Finalizando o jogo.

Orientação: Utilizar a linha de chegada para dar o efeito de troca de cenário. Peça para que os alunos adicionem novos panos de fundo, eles podem utilizar os panos da biblioteca ou podem também importar ou utilizar a ferramenta de desenho. Criando um controle condicional para que o pano de fundo seja trocado sempre que algum jogador tocar o objeto de chegada.

Objetivo: Trabalhar com controles condicionais para troca de cenário.

4.8 Módulo 07 - Projeto rpg.

Este projeto não consiste de um jogo propriamente dito, mas de técnicas que permitam a construção de um jogo estilo RPG e que possam ser utilizadas em outros tipos de jogos. Inicie este módulo apresentando a proposta com um projeto modelo.

Atividade 01: Início do projeto

Orientação: Na primeira atividade, dê um tempo para que os alunos criem um roteiro do jogo, com personagens e cenários completos, usando objetos, sons e efeitos. Em seguida, peça que deem movimento ao personagem utilizando o comando de setas e configurando o personagem e as animações .

Objetivo: Trabalhar a criatividade e comandos de movimento.

Atividade 02: Efeito de entrar e mudar de cenário.

Orientação: Trabalhar o efeito de mudança de cenário sempre que o personagem "entrar" em algum lugar. Nesta atividade, os alunos irão utilizar o recurso de emitir sinal ao programa pelo comando "enviar mensagem" em eventos.

Objetivo: Dominar o conceito de troca de panos de fundo e objetos do cenário e dominar o conceito de envio de sinal ao programa.

Atividade 03: Efeito de saída do cenário .

Orientação: Os alunos irão executar tarefa semelhante a atividade 02 para que o personagem volte ao cenário anterior. Peça que estabeleçam um critério para que o personagem volte ao cenário anterior ou avance para outro cenário.

Objetivo: Trabalhar o conceito de troca de cenários e dominar os recursos necessários.

Atividade 04: Finalizando o jogo.

Orientação: Os alunos irão aprender uma maneira de criar telas de abertura ou transição utilizando personagens para este recurso. Utilize o vídeo de exemplo do módulo e outros projetos similares como exemplo da atividade e deixe seus alunos trabalharem a criatividade.

Objetivo: Trabalhar a criatividade

4.9 Módulo 08 - Projeto jogo Flying bat.

Baseado em um jogo famoso "Flappybird", este módulo versa a criação de um jogo Flying Bat. O módulo 08 pode ser considerado o mais complexo do curso, ele é indicado a ser o último, pois os alunos já terão domínio dos conceitos de movimento, de variável e outros necessários para o processo de produção do projeto.

Atividade 01: Iniciando o jogo.

Orientação: O primeiro passo é a construção dos objetos do jogo que serão: o fundo, o personagem e outros. Recomenda-se que utilize o vídeo exemplo ou faça um com seus alunos para que eles entendam como usar as ferramentas de desenho. É importante que tudo que for criado tenha a mesma cor na borda para programações futuras.

Objetivo: Desenvolver criatividade e domínio da ferramenta de desenho.

Atividade 02: Dando movimento ao cenário.

Orientação: Os objetos que foram criados irão ganhar movimento, dando ideia de que o personagem está andando para frente. Mais uma vez é interessante que utilize o vídeo ou faça um exemplo com os alunos antes de pedir que os mesmos o produzam.

Objetivo: O aluno deverá dominar efeito de movimento em cenário.

Atividade 03: Fazendo o personagem voar.

Orientação: O personagem deve subir quando uma tecla for pressionada e cair quando esta tecla não for pressionada. Para isso, oriente a utilização do controle "se...então...senão".

Objetivo: Dominar o controle condicional.

Atividade 04: Criando condições para vitória e derrota.

Orientação: Utilizando as bordas dos desenhos feitos, crie uma condição para derrota. E peça para que seus alunos pensem em um objetivo no jogo para criar uma condição para vitória.

Objetivo: Trabalhar com controles condicionais.

Atividade 05: Finalizando o projeto.

Orientação: Para finalizar o projeto, peça que os alunos criem algum efeito para quando o personagem perder o jogo. Outro recurso que pode ser feito é um sistema de pontuação por variável, que pode ser relacionado com uma próxima fase ou com o aumento da dificuldade.

Objetivo: Trabalhar com efeitos e variáveis de forma criativa.

CAPÍTULO 5

ANÁLISE DOS RESULTADOS.

5.1 Análise dos módulos.

Antes de iniciar um relato sobre os módulos, lembro que a turma criada foi formada com alunos de ensino médio do instituto federal fluminense de cursos técnico com formação em Meio Ambiente, Automação industrial, Eletromecânica e Eletrônica, em sua maioria, estudantes do primeiro e segundo ano do ensino médio. Ao selecionar o grupo que faria parte do Clube, escolhi alunos que tinham certa facilidade com a área de informática, mas mantive um foco maior nos das minhas turmas regulares de matemática, em que percebia a necessidade do desenvolvimento das habilidades e competências propostas pela aplicação desta sequência. Em uma análise geral, pude perceber nos alunos que tinham facilidade em programação uma maior desenvoltura durante as atividades, permitindo-se ir além do que era pedido e em geral os resultados de seus jogos continham efeitos ou objetos bem diferentes dos exemplos dados. Já nos alunos com certa dificuldade em matemática pude perceber o mesmo durante as atividades, porém era nítido que o ambiente em que estavam e o propósito pelo qual estavam aprendendo os estimulavam terminar suas tarefas e permitia até incrementar seus módulos atuais com conceitos aprendidos nos anteriores. Ressalto também, que não houve caso de alunos que não conseguissem cumprir algum módulo e a menos do tempo de execução, todos tiveram êxito nos módulos feitos. Outro fator importante, que deve ser considerado, é que na data de produção do capítulo em questão, houve um atraso no calendário escolar do IF Fluminense - campus Macaé, logo, este projeto ainda encontra-se em andamento, tendo os módulos concluídos. Porém a proposta final, que é a criação de um jogo com a participação de todos os alunos do Clube, ainda está em andamento. Sendo assim, vamos a um breve relato sobre cada módulo:

- **Módulo 00:**

Os alunos foram apresentados oficialmente ao projeto e ambientados a plataforma Moodle e ao Scratch. Foi um módulo de criação de usuários, tanto na plataforma Moodle quanto no site do Scratch. Feito isso, demos uma passada geral no programa, falando sobre programação em bloco, os comandos e fazendo com que eles experimentassem execuções aleatórias. Alguns alunos, que já conheciam a ferramenta, tentaram construir alguma aplicação, mas não tivemos nenhuma atividade de construção formal. E no geral, os alunos tiveram boa receptividade ao projeto e ao Scratch.

- **Módulo 01:**

Os alunos foram apresentados aos comandos de aparência, mais precisamente os comandos (DIGA, PENSE) e suas variações. Como prática a atividade pedia uma apresentação onde fosse exposto as expectativas sobre o projeto. Os alunos tiveram que fazer, ao final, uma apresentação em datashow dos projetos finalizados. Neste módulo, os alunos fizeram de forma muito

descontraída e rápida a atividade proposta e voltaram a experimentar o Scratch como no módulo anterior. Abaixo temos dois exemplos de projetos feitos, no exemplo 5.1, temos uma apresentação simples onde o personagem do gato faz a apresentação. Já no exemplo 5.2 os dois personagens na tela fazem uma conversa sincronizada.



Figura 5.1: Exemplo de projeto.

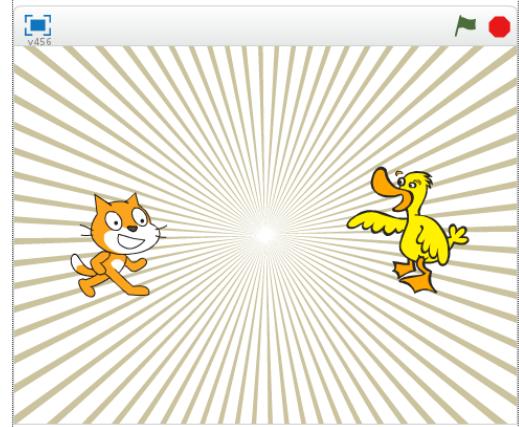


Figura 5.2: Exemplo de projeto.

• Módulo 02:

Foi o primeiro com uma sequência efetiva de atividades até a criação completa de um jogo. Nele o papel do orientador foi muito importante, principalmente para controlar a ansiedade dos alunos em tentar fazer o jogo sem seguir os passos propostos. Aqueles que já tinham algum conhecimento ou, que depois dos primeiros encontros, foram pesquisar mais sobre o Scratch conseguiram progredir nas atividades do módulo antes mesmo de ver alguns vídeos. Não tive grandes questionamentos sobre os conceitos empregados durante o módulo. Nas figuras abaixo, temos dois projetos do módulo.

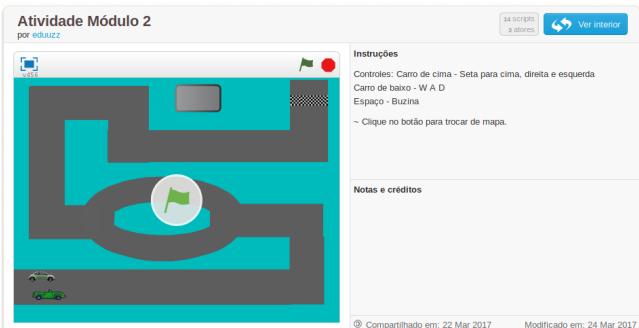


Figura 5.3: Exemplo de projeto.

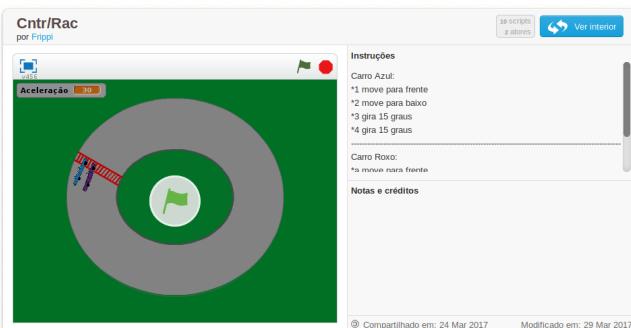


Figura 5.4: Exemplo de projeto.

- **Módulo 03:**

Os alunos começaram a ter uma postura mais ativa e independente nas atividades. O jogo propõe o movimento guiado pelo mouse e alguns alunos decidiram fazer movimento como no módulo anterior, pois acharam que a mecânica do jogo ficaria melhor desta maneira. Procurei não coibir nenhuma alteração proposta, porém incentivei que seguissem as orientações dos vídeos. Foi o primeiro módulo em que a postura ativa dos alunos foi além de aprender e tive o primeiro caso em que um aluno tomou o meu lugar em alguns momentos, tornando-se o orientador dos colegas com dificuldades ou erros apresentados durante os testes. Abaixo, dois exemplos de projetos, o primeiro (Figura 5.5) o aluno apresentou um labirinto feito por personagens estáticos e outros animados. Já no projeto da Figura 5.6 as bolas que servem de obstáculo são todas animadas e a dificuldade do jogo aumenta conforme o tempo.



Figura 5.5: Exemplo de projeto.

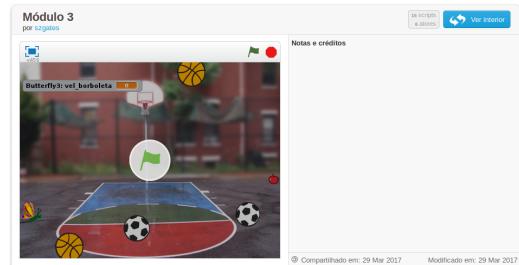


Figura 5.6: Exemplo de projeto.

- **Módulo 04:**

Acredito que todos os alunos e eu, tenhamos nos deparado com o primeiro grande problema que não sabemos resolver. No jogo da plataforma, era proposto que um personagem usando plataformas suspensas, alcançasse algo. Descobrimos que bugs muito comuns nesses jogos: quando os personagens ficam sobrepostos ou presos em outros objetos ou no próprio cenário. Como no caso do jogo exemplo, uma plataforma que faz movimento de subir e descer em looping, não permitindo que o personagem ficasse parado sobre ela, fazendo o personagem cair. Esta atividade foi feita de forma relativamente rápida, fazendo com que grande parte do encontro fosse dedicado a busca da solução desse bug que não foi descoberta. Apenas um aluno decidiu que seu jogo estava finalizado, tendo os outros tomado a decisão de buscar a solução para o problema.

Abaixo, a imagem do projeto dado como finalizado, onde podemos ver o bug, que é o fato do personagem do cachorro atravessar um patamar marrom na tela.



Figura 5.7: Exemplo de projeto.

- **Módulo 05**

Apresentou uma dificuldade diferente dos outros, que foi na questão da quantidade de atividades. Já com um certo conhecimento por parte dos alunos, minhas intervenções foram principalmente para que mantivessem o foco nas atividades.

No exemplo, temos a inclusão de mais uma personagem para dar um grau de dificuldade diferente no jogo.

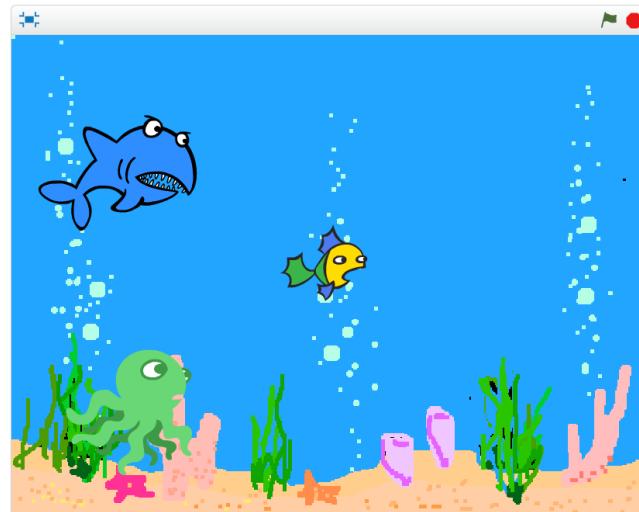


Figura 5.8: Exemplo de projeto.

- **Módulo 06**

Possuiu dois efeitos que despertaram muito interesse nos alunos. Um deles foi a estratégia de colocar uma barra no final da tela, fazendo com que o personagem voltasse ao início da tela do jogo, esta pode ser utilizada para mudanças de cenários nos jogos. E sem sombra de dúvida, o efeito dos clones foi o que lhes chamou mais a atenção e despertou muita criatividade durante a atividade.

No exemplo, o início de um projeto do jogo proposto, onde o aluno cria um jogo com a mesma lógica, porém prefere personagens que tenham mais trajes para utilizar os controles de aparência e dar animação ao movimento dos personagens.



Figura 5.9: Exemplo de projeto.

- **Módulo 07**

Os recursos trabalhados foram muito pedidos desde o início das atividades do Clube, muito pelo fato de efeitos como mudança de cenário para criação de histórias darem uma possibilidade maior de jogos que poderiam ser criados. Percebi que foram as atividades em que os alunos mais se envolveram, questionaram e pesquisaram. Mesmo com um número pequeno de tarefas, o módulo teve uma duração tão grande quanto módulos com uma maior quantidade de atividades.

No exemplo, temos um interessante projeto, onde o personagem espacial interage com outros cenários dos planetas, e da espaçonave.

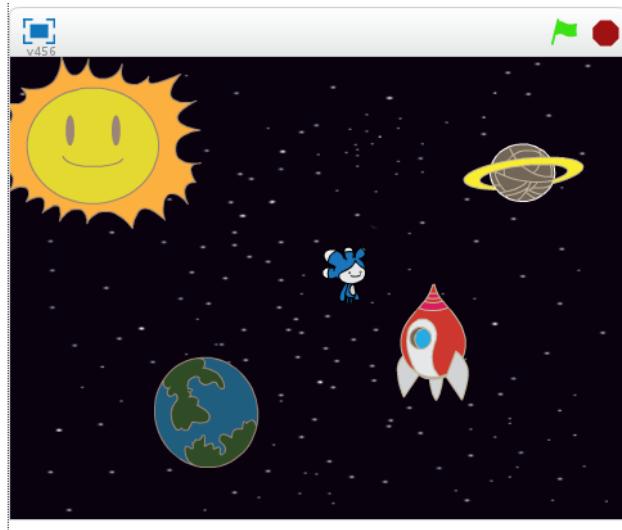


Figura 5.10: Exemplo de projeto - Tela inicial.

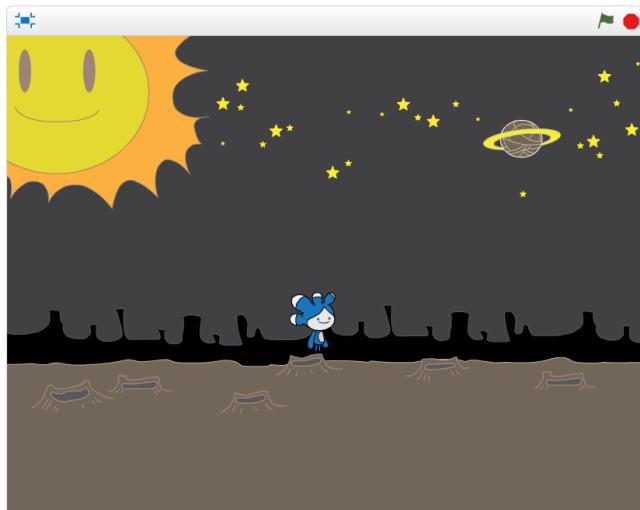


Figura 5.11: Acessando planeta "Terra".

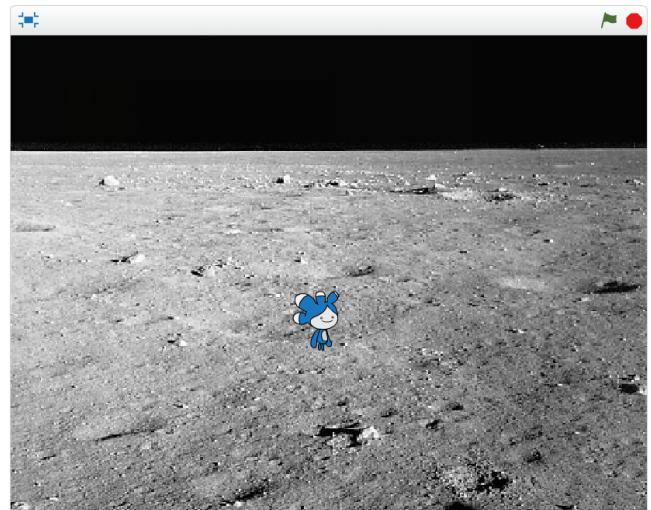


Figura 5.12: Acessando Saturno.

- **Módulo 08**

Este é o módulo com o efeito visual mais interessante de todos. A ideia de ”fazer o cenário andar”, mesmo com um grau de dificuldade mais alto, prendeu bastante a atenção dos alunos durante a sua realização. Este módulo foi programado para ser o último, pois usa praticamente todos os conceitos vistos antes e alguns novos, o que fez com que minhas intervenções fossem mais frequentes.

Neste módulo, acredito que pela dificuldade dos recursos utilizados não houve muita criatividade por parte dos alunos, havendo diferenças apenas nos cenários devido ao recurso utilizado para desenhar as telas. Abaixo, a tela do jogo.

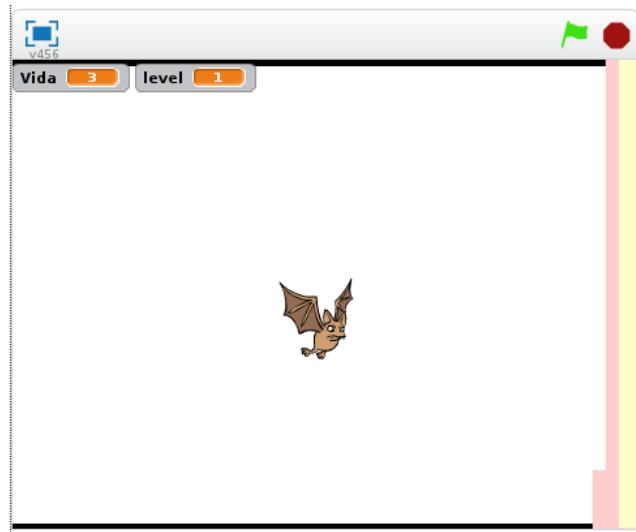


Figura 5.13: Exemplo de projeto.

CAPÍTULO 6

CONSIDERAÇÕES.

6.1 Trabalhos Futuros.

Durante as atividades do clube, diversas propostas foram levantadas pelos alunos e discutidas nas oficinas. Dessa proposta surgiram embriões de trabalhos futuros, dos quais destaco os seguintes:

- Por se tratar de uma escola técnica, os alunos que fazem parte do projeto em sua maioria tem algum tipo de contato com disciplinas ligadas a automação industrial e são, desde cedo, apresentados a placas controladoras como Arduino, Raspberry e similares. Normalmente a programação dessas placas, em particular o Arduino, é feita pela linguagem padrão delas que é muito semelhante a linguagem C.

Abaixo temos um exemplo de programa feito no programa Arduino V 1.6 para Linux.

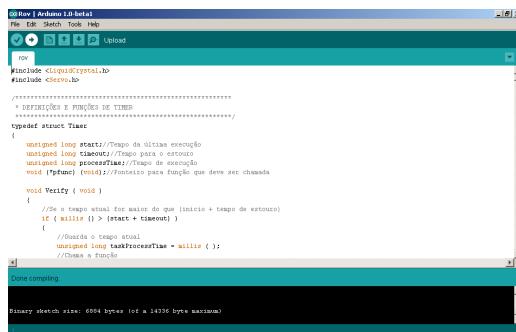


Figura 6.1: Arduino V 1.6 para Linux.

O Scratch possui uma versão feita para programar Arduino que pode ser encontrada em (<http://s4a.cat/>).

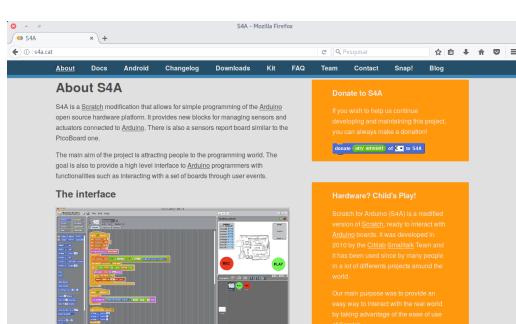


Figura 6.2: Site Scratch para Arduino.

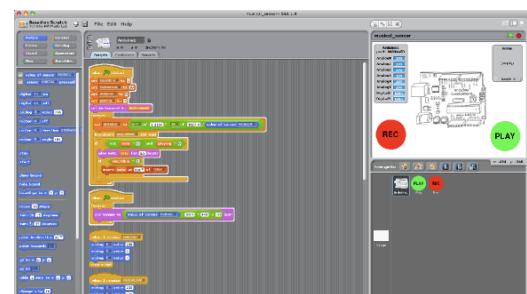


Figura 6.3: Programação de um Arduino feito em Scratch for Arduino.

Utilizando a programação em bloco, e um Arduino, um grupo de alunos do projeto iniciou uma pesquisa para produzir um projeto de automação. A ideia, até a presente data, é construir uma lixeira automática que fique andando pelo instituto trabalhando também a questão do meio ambiente que é outro curso técnico com aluno no clube.

- Outra proposta, é continuar desenvolvendo jogos, mas na tentativa de uma versão digital de algum jogo físico. Para esta proposta, os alunos do clube, que tem interesse nessa tarefa, fazem reuniões onde alguns jogos são trazidos para jogar, estudar os manuais e as regras, buscando entender o funcionamento do jogo, e pesquisando algoritmos que permitam a criação de uma versão digital, utilizando o Scratch como plataforma.
- O módulo da apresentação, não só inspirou um projeto de produção de um curta metragem animado, como também foi motivador de alguns alunos para produção de trabalhos de algumas disciplinas que pediam uma apresentação oral. Alunos que fazem parte do projeto produziram animações para apresentar seus trabalhos de algumas disciplinas da sua grade.

6.2 Considerações Finais

O Projeto do Clube de Programação em Scratch, que foi onde a sequência didática exposta neste trabalho foi executada, é um projeto que tem como objetivo desenvolver a linguagem de programação com jovens do Instituto Federal Fluminense - IFF - campus Macaé. Mesmo sendo alunos de curso técnico e alguns deles têm ou terão programação em sua vida escolar, o projeto teve objetivo de trabalhar o desenvolvimento das habilidades e competências e não necessariamente a linguagem de programação. As sequências de cada módulo tinha o objetivo de ensinar conceitos de programação através da confecção de pequenos jogos, importante para os participantes. E ao final foi feita uma proposta maior, em grupo ou individual, onde os alunos deveriam criar um jogo, preferencialmente com uma proposta educativa que pudesse ser disponibilizado na rede para auxiliar professores da educação básica nas suas atividades didáticas.

Nas sequências dos módulos sempre foram trabalhados os vídeos de exemplo, para que os alunos tivessem uma ideia do tema que era tratado, o professor teve papel de orientador, as aulas foram o mínimo expositivas possível e os alunos estimulados a seguir a sequência, sempre foi dado muita abertura e incentivo a criatividade para que os mesmos não reproduzissem simplesmente o que estavam assistindo. Sobre esse aspecto, o resultado pode ser considerado muito positivo, os alunos, em geral, usavam o vídeo apenas para saber qual era o mecanismo do jogo dado, e na maioria das vezes se permitiam mudar seus projetos em relação ao que era proposto.

A proposta da sequência foi uma adaptação baseada na sequência feita pelo projeto do Google (CS-First). Foram feitas adaptações nos vídeos, que foram regravados por mim em português, incluindo ou excluindo passos na sequência de cada módulo quando julguei necessário. Minha avaliação como professor é que no início, os estudantes tiveram dificuldades em seguir as sequências na ordem proposta, tanto devido ao manuseio da plataforma Moodle utilizada no projeto, quanto pelo ímpeto jovem de não seguir a ordem cronológica das coisas. Sobre a sequência utilizada, todo material está disponibilizado neste trabalho. Tanto este trabalho quanto os vídeos utilizados (com links nos apêndices), ficarão na rede para que sirvam de modelo ou de ferramenta de aplicação de projeto similar. Também vale ressaltar que o CS-First do Google possui um material completo à disposição de quem estiver interessado em utilizá-la. Durante o transcorrer do projeto, conheci outras ferramentas que também podem ser utilizadas com a mesma finalidade. Uma delas é o **APP-Inventor** do mesmo MIT (<http://appinventor.mit.edu/explore/>) que também é uma linguagem em bloco, mas que faz programas para Smartphones Android e o **Code.org** (<https://code.org>) que também possui sequências didáticas de atividades e possibilidade de criação de salas virtuais para clubes semelhantes.

A avaliação dos estudantes foi feita durante a execução de cada módulo, levando em conta não somente o projeto finalizado, mas o transcorrer do módulo. A proposta é que o estudante siga seu tempo de aprendizado, não sendo obrigado a terminá-lo em um encontro. Sobre este aspecto, os estudantes foram bem homogêneos, pois terminaram seus módulos no máximo em duas aulas. Rodas

de discussão sobre o jogo proposto era levantada e também nesse momento era possível avaliar o interesse dos estudantes no tema e a criatividade com que iriam produzir o projeto.

Durante a execução das oficinas, percebi claramente que os participantes não viam aquela prática como uma aula, a maioria via como diversão na criação de jogos. Inicialmente pensei que fosse por conta da motivação usada em cada oficina ou por conta da simplicidade do Scratch, pois sabendo que são estudantes de cursos técnicos e alguns deles já foram apresentados a linguagens de programação, como C ou C++ e outras, acreditei que aquela linguagem simples parecia lúdica demais para eles. Mas essa constatação foi quebrada pela existência de alunos que não fazem curso técnico no eixo industrial e esses viam a aula da mesma maneira. Digo que isso facilitou muito o processo inclusive quando tivemos que falar de conceitos de programação mais complicados.

Ao final desse projeto e com a autorização dos participantes, pretendo publicar todos os projetos feitos, incluindo o "projeto final" no meu site (<http://blog.matemaniacos.com.br>) para que possam ser utilizados também em outros projetos.

Minha consideração como professor de matemática é que o tempo de pesquisa, construção e aplicação dessa sequência me fizeram repensar muito sobre minha atividade em sala de aula. Repensar a forma com que desenvolvemos o aprendizado dos conteúdos matemáticos, a maneira de avaliar nossos alunos e principalmente os objetivos que desejamos alcançar quando entramos em sala para ensinar nossa disciplina, seja ela qual for no currículo escolar. Sei que a informática não é realidade em todas as escolas pelo nosso país, mas durante minha pesquisa vi, inclusive, que existem recursos físicos das atividades utilizando peças de madeira, jogos de tabuleiro e outros tão criativas e possíveis de atingir diversas realidades escolares, as quais ainda pretendo me debruçar e aprender mais sobre. Conhecer essa proposta me fez refletir em outras formas de atingir meus alunos e conseguir cumprir meu papel de educador.

APÊNDICE A

TUTORIAL SCRATCH

A.1 Conceitos Básicos

Este tutorial será feito baseado na versão online do Scratch que pode ser acessada no site oficial, <https://scratch.mit.edu>, Logo após acessar o site, o usuário pode clicar em **CRIAR**:



Figura A.1: Site oficial do scratch - imagem de 10/12/2016

A.1.1 Tela inicial.

Abaixo, temos a tela inicial de programação.

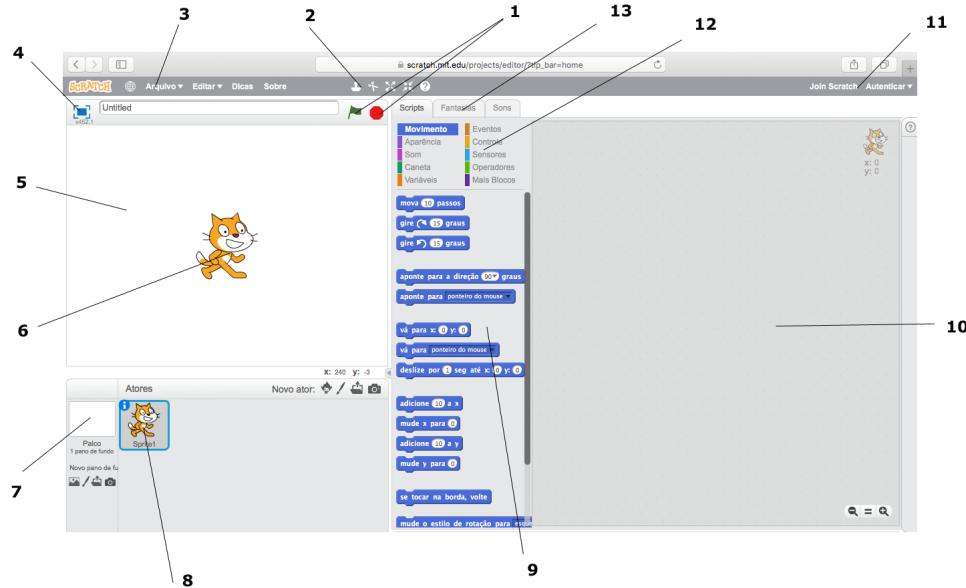


Figura A.2: Ambiente de programação da versão web: <https://scratch.mit.edu>

Na tela inicial do Scratch podemos ver:

1	Botão para iniciar (bandeira verde) e parar (círculo vermelho) o script
2	Botões para editar objeto no palco; duplicar, apagar, aumentar e diminuir
3	Menu do programa.
4	Atalho para tela cheia
5	Palco
6	Ator (Sprite)
7	Configuração, programação e escolha dos palcos
8	Lista de atores (sprite) e acesso as informações dos atores.
9	Paleta de blocos de comandos.
10	Área de programação
11	Atalho para conexão à comunidade do Scratch
12	Categoria de comandos.
13	Paleta de acesso aos scripts / Fantasias / Sons.

Tabela A.1: Tela inicial so Scratch.

A.1.2 Iniciando e parando script



Figura A.3: Botão iniciar e parar - imagem de 10/12/2016

	Inicia o script de programação
	Para o script de programação

Tabela A.2: Atalho para inicio e fim de execução.

A.1.3 Atalhos de edição



Figura A.4: Atalho de edição dos objetos no palco - imagem de 10/12/2016

	Duplicar objeto.
	Apagar objeto
	Aumentar objeto
	Reduzir objeto
	Ajuda do bloco.

Tabela A.3: Atalho de edição dos objetos no palco

A.1.4 Menu do scratch



Figura A.5: Menu do Scratch - imagem de 10/12/2016

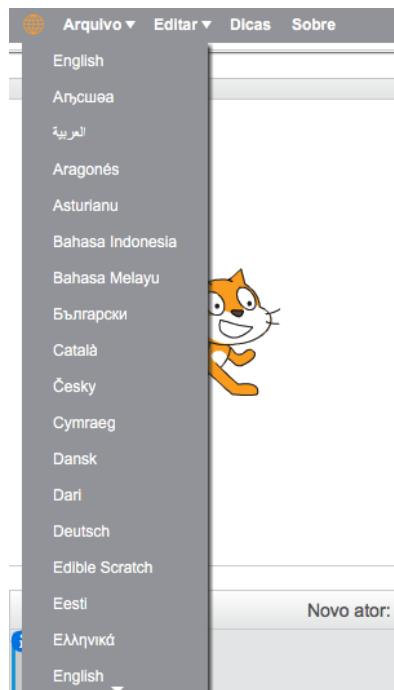


Figura A.6: Escola de idioma.

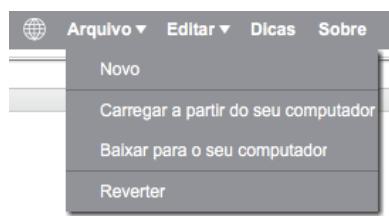


Figura A.7: Menu arquivo

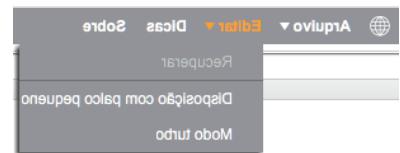


Figura A.8: Menu editar

A.1.5 Tela cheia



Botão para colocar o Scratch em modo de execução em tela cheia.

Tabela A.4: Atalho para execução em tela cheia.

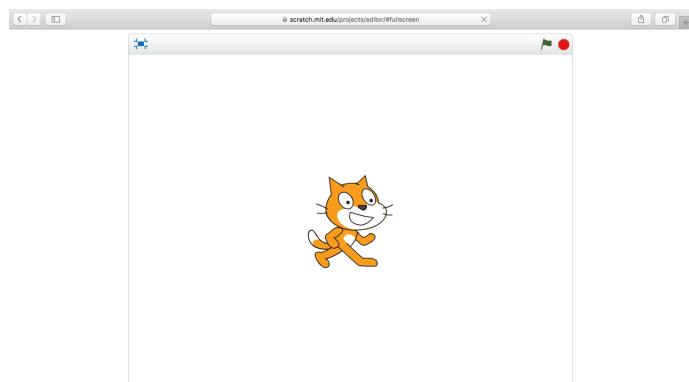


Figura A.9: Scratch em tela cheia - imagem de 19/12/2016

A.1.6 Palco

O palco é o local onde ocorre a execução dos eventos programados no Scratch como movimentação dos objetos, animação, as variáveis e resultados de algoritmos. O palco é um retângulo com dimensões de 480 unidades de largura e 360 de altura conforme a figura abaixo.

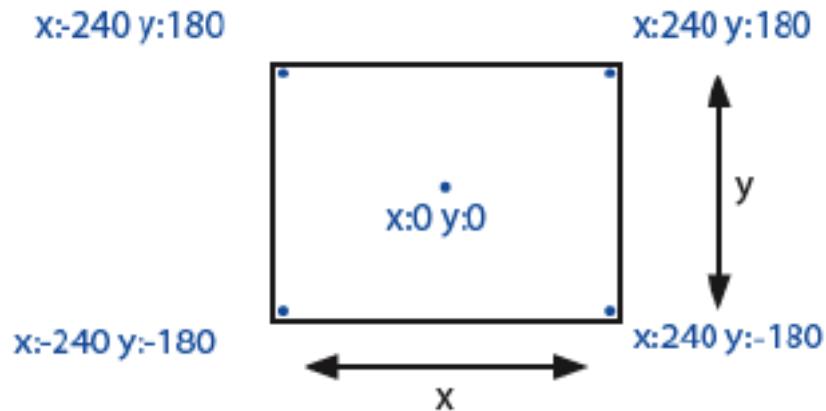


Figura A.10: Dimensões do palco - imagem de 19/12/2016

Ao palco podemos associar um plano cartesiano $X \circ Y$

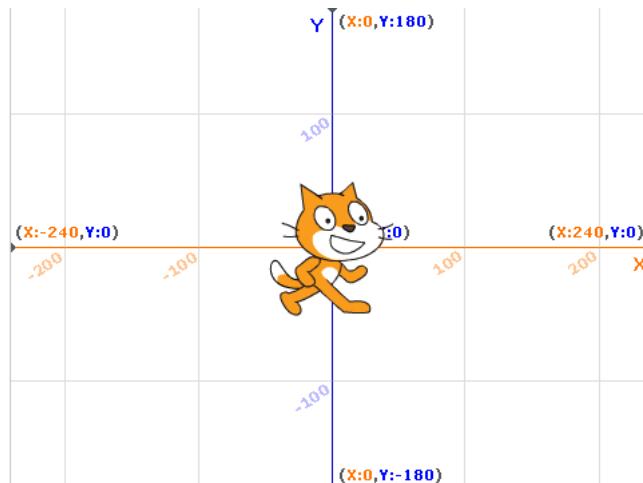


Figura A.11: Palco com eixo cartesiano - imagem de 19/12/2016

Para efetuar programações no palco, basta clicar com o mouse no placô e acessar o script de programação do palco.

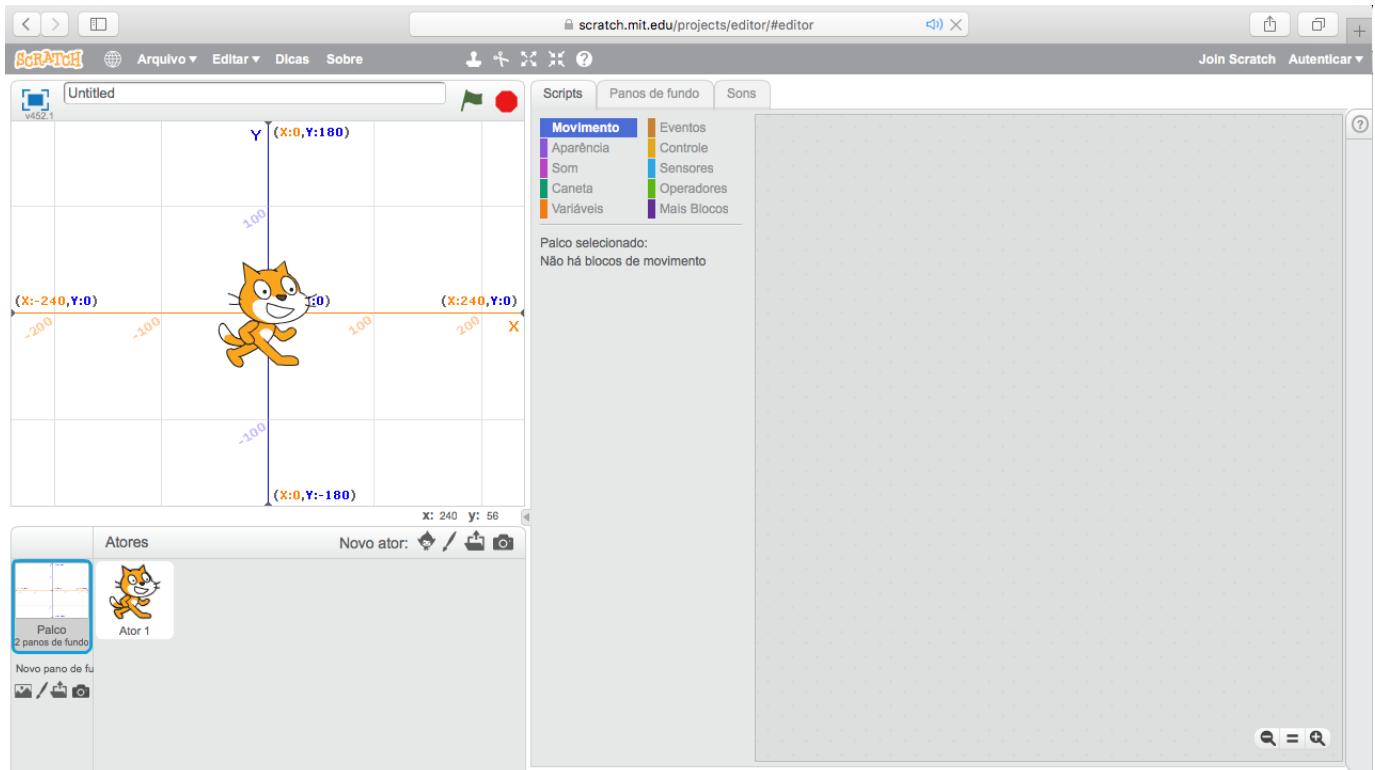


Figura A.12: Palco com eixo cartesiano - imagem de 19/12/2016

O palco pode ser substituído utilizando o menu:

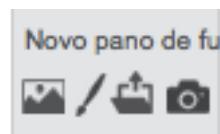


Figura A.13: Palco com eixo cartesiano - imagem de 19/12/2016

	Carregar novo palco da biblioteca do scratch.
	Desenhar novo palco
	Carregar novo palco do arquivo
	Carregar palco de imagem da câmera.

Tabela A.5: Atalho do palco.

Um novo palco pode ser carregado da biblioteca do Scratch, construído com a ferramenta de desenho, importado de arquivo externo ou de uma foto da câmera.

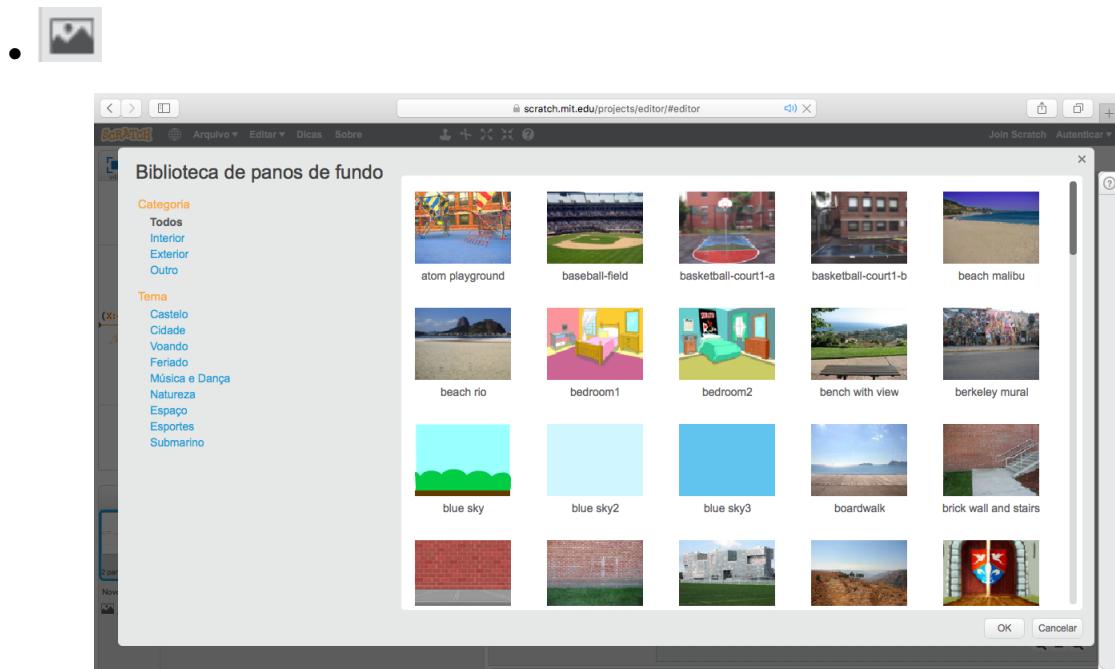


Figura A.14: Biblioteca de palcos do Scratch - imagem de 19/12/2016

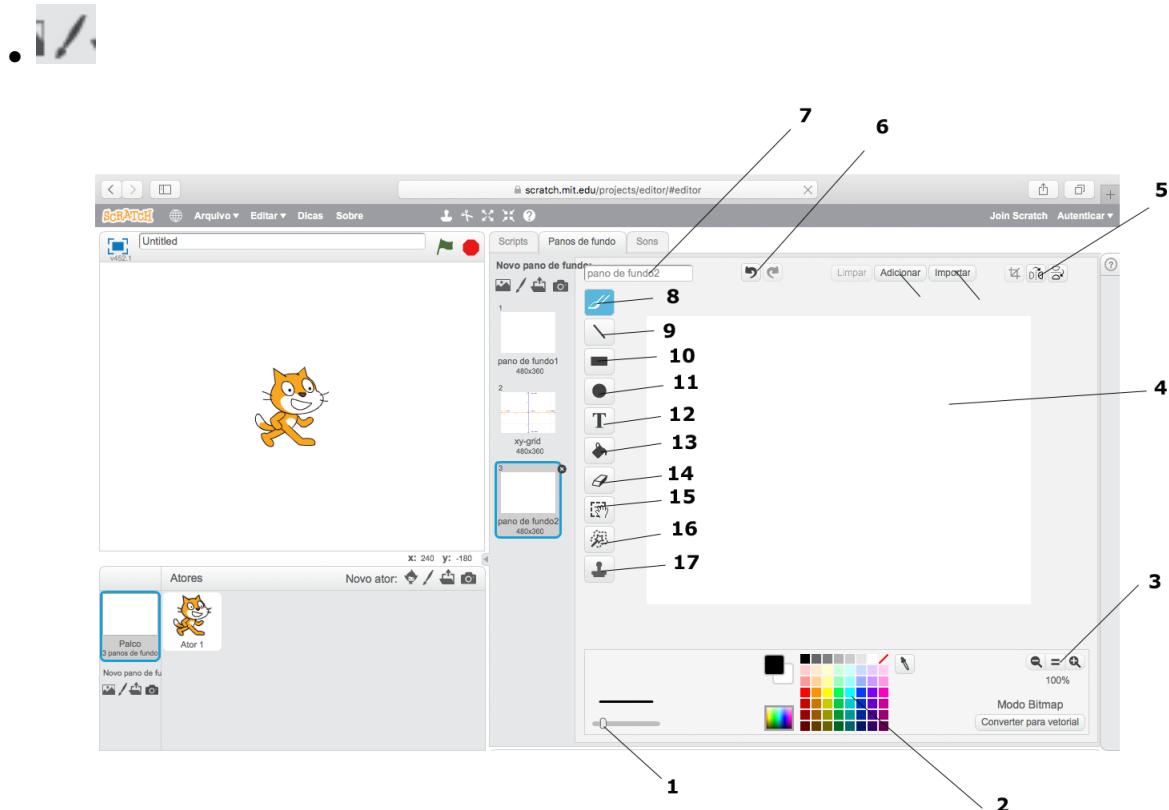


Figura A.15: Ambiente para pintar um novo cenário. - imagem de 19/12/2016

1	Expressura da linha.
2	palheta de cores.
3	Zoom
4	Área do palco.
5	Modo de espelhamento.
6	Voltar/Avançar.
7	Nome do Placo
8	Pincel
9	Linha
10	Retangulo/Quadrado
11	Elipse/Círculo.
12	Texto
13	Preencher com cor
14	Borracha
15	Selecionar
16	Remover plano de fundo
17	Duplicar.

Tabela A.6: Atalho do ambiente de pintura do palco.

A.1.7 O Ator (Sprite)



Figura A.16: Ator (Sprite) - imagem de 10/12/2016

Os atores ou Sprites são os personagens do Scratch. Eles podem ser adicionados a partir da biblioteca de atores ou importados de arquivo gif.

O Scratch é capaz de fazer animação quadro a quadro utilizando as fantasias (trajes) para dar ideia de movimento. No exemplo abaixo o gato possui duas fantasias trazendo a ideia de que o gato está andando.



Figura A.17: Fantasias do ator - imagem de 10/12/2016

É possível mexer nas configurações clicando no do ator.



Figura A.18: Área dos atores.



Figura A.19: Configuração do ator.

Na imagem A.28 temos como configurar:

- O nome do ator "Ator 1", que será a referência do ator na programação.
- O estilo de rotação, permitindo que o ator faça giros, apenas movimentos horizontais ou não permita rotação.

	Permite rotação.
	Apenas movimentos horizontais.
	Nenhuma rotação

Tabela A.7: Atalho para comandos do info dos Atores..

- Permitir ou não que o ator seja arrastado pelo mouse.
- Ocultar ou mostrar o ator.

A inclusão de novos atores é feita utilizando o menu:

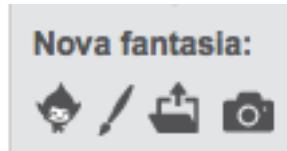


Figura A.20: Menu de inclusão de novos atores - imagem de 10/12/2016

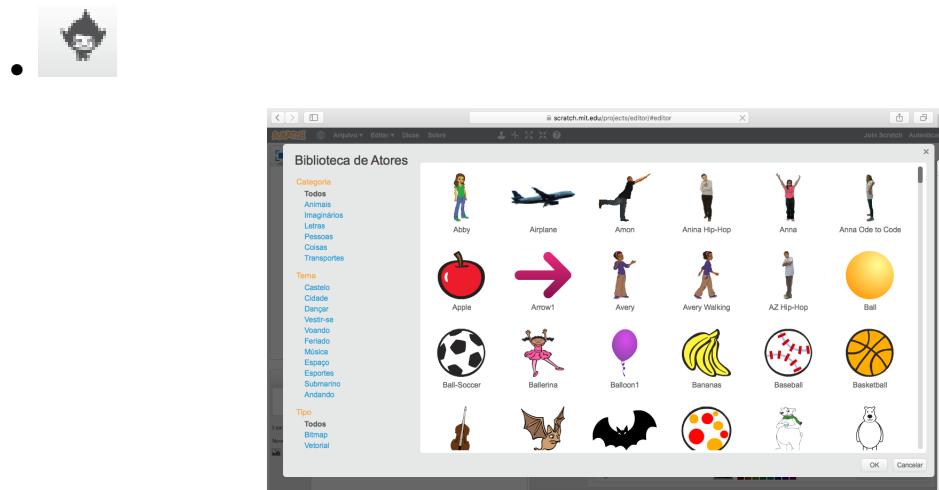


Figura A.21: Biblioteca de atores do Scratch - imagem de 19/12/2016

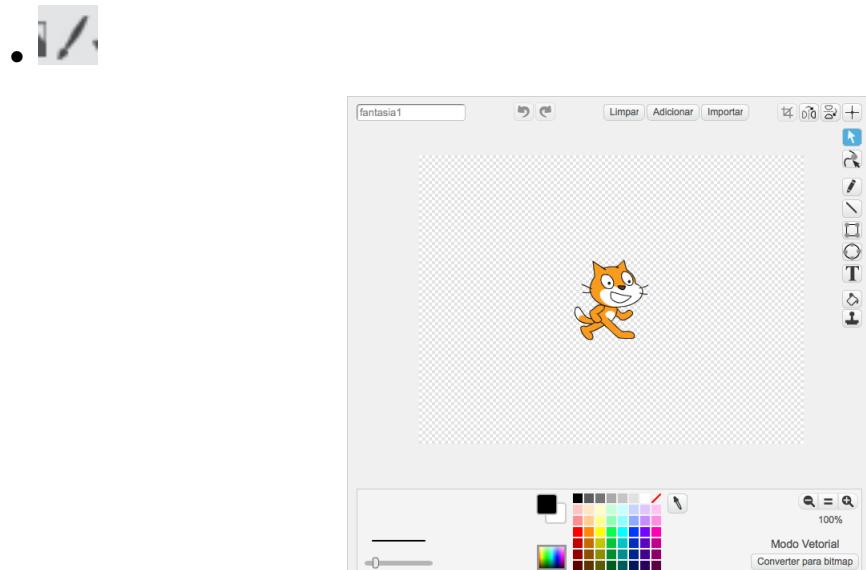


Figura A.22: Ambiente para pintar atores. - imagem de 19/12/2016

A.1.8 Programação no Scratch - Conceitos básicos e comandos.

A programação no Scratch é feita pela ligação dos blocos de comando na área de programação.

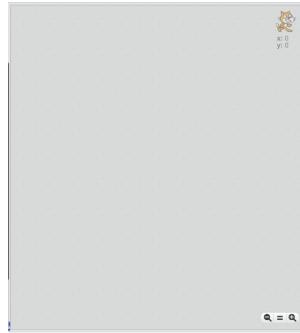


Figura A.23: Área de programação - imagem de 19/12/2016

Para a construção do algoritmo de programação no Scratch, é necessário que haja ligação entre os blocos de comandos. Abaixo temos dois exemplos de algoritmos.

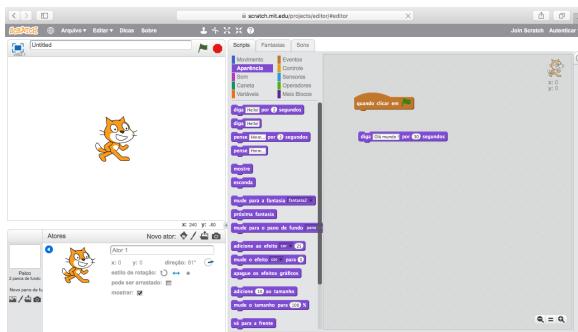


Figura A.24: Exemplo de algoritmo incorreto.

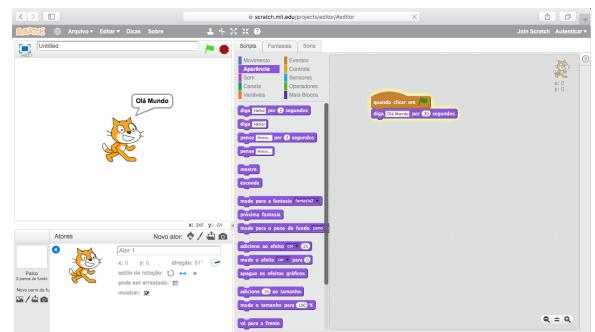


Figura A.25: Exemplo de algoritmo correto.

Os comandos são separados por categorias.

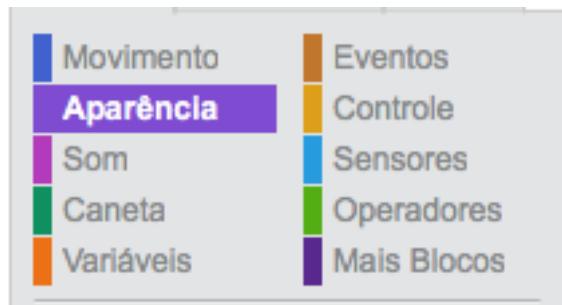


Figura A.26: Lista de categorias dos comandos - imagem de 19/12/2016

- Comandos de movimento.

	Mova o ator um determinado número de passos
	Gire para a direita
	Gire para a esquerda
	Estabelece a direção do ator atual.
	Orienta o ator na direção do mouse ou de outro ator
	Vá para a posição no palco.
	Vá para a localização do ponteiro do mouse ou de outro ator.
	Deslize ao longo do tempo para a posição (x, y)
	Paleta de blocos de comandos.
	Adicione esta quantidade a coordenada x
	Defina a posição x de um ator
	Adicione esta quantidade a coordenada y
	Defina a posição y de um ator
	Se tocar na borda do palco, volta.
	Informa a posição x de um ator.
	Informa a posição y de um ator.
	Informa a direção atual de um ator.

Tabela A.8: Comandos de movimento

fonte:<https://scratch.mit.edu/>

- Comandos de aparência.

	Mostra texto em um balão de fala durante um número de segundos.
	Mostra um texto em um balão de fala.
	Exibe palavra em um balão de pensamento por um número de segundos.
	Exibe palavra em um balão de pensamento.
	Faz o ator aparecer no palco.
	Faz o ator desaparecer no palco.
	Mude de fantasia para alterar o aspecto de um ator.
	Muda para a proxima fantasia da lista de fantasias do ator.
	Muda o palco especificado.
	Altera o efeito gráfico aplicado ao ator.
	Atribui um número específico ao efeito gráfico aplicado a um ator.
	Limpa todos os efeitos gráficos aplicados a um ator.
	Altera a quantidade especificada ao tamanho do ator.
	Altera o tamanho do ator a uma porcentagem do original.
	Desloca o ator para a frente de todos os outros atores.
	Recue um certo número de camadas.
	Relata o númeor da fantasia atual do ator.
	Informa o nome do pano de fundo atual do palco.
	Informa a porcentagem do tamanho original do ator.

Tabela A.9: Comandos de aparência.

fonte:<https://scratch.mit.edu/>

- Comandos de som.

	Toca um som.
	Reproduz um som e espera ate que o som termine.
	Interrompe todos os sons.
	Toca um som de tambor durante um determinado número de batidas.
	Descansa (interrompe o jogo) por determinado núemro de batidas.
	Reproduz uma nota musical para um determinado número de batidas.
	Define o tipo de instrumento que o ator usa para os blocos tocarem a nota.
	Altera o volume do som do ator em uma quantidade especificada.
	Atribui um dado valor ao volume de som de um ator.
	Relata o volume do som do ator.
	Altera o tempo do ator em quantidade determinada.
	Adicione esta quantidade a coordenada y
	Relata o tempo de um ator em batidas por minuto.

Tabela A.10: Comandos de som

fonte:<https://scratch.mit.edu/>

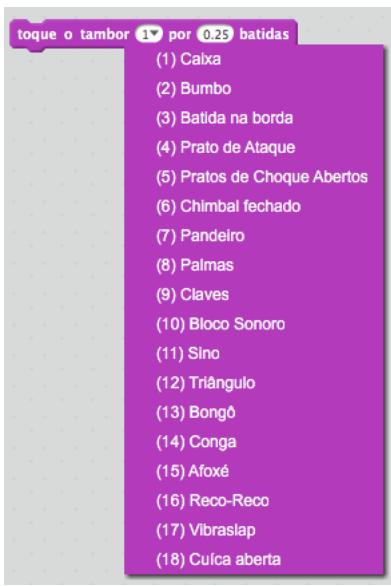


Figura A.27: Opções de instrumento do comando tocar o tambor.

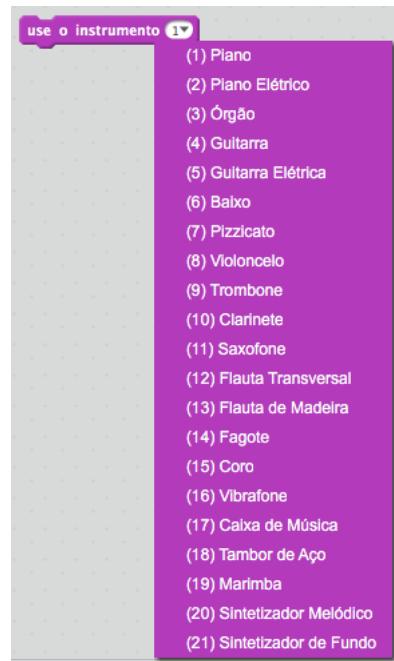


Figura A.28: Tipos de instrumentos.

• Comandos de caneta.

	Apaga do palco todas as marcas de caneta e todos os carimbos.
	Carimba no palco a imagem do ator.
	Usa a caneta do ator para que ela risque a medida que ele se move.
	Levanta a caneta do ator, para que ela não risque à medida que ele se move.
	Estabelece a cor da caneta com base em uma amostra feita usando a pipeta.
	Altera o valor especificado à cor da caneta.
	Estabelece a cor da caneta de acordo com o valor especificado.
	Altera o valor especificado à intensidade da caneta.
	Atribui um dado valor à intensidade da caneta.
	Altera a espessura da caneta.
	Estabelece a espessura da caneta.

Tabela A.11: Comandos de Som

fonte:<https://scratch.mit.edu/>

- **Variáveis / Lista**

O ambiente de variável inicialmente aparece vazio e podemos criar uma variável e/ou uma lista.

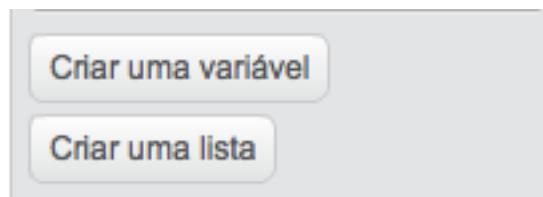


Figura A.29: Ambiente de criação de Variáveis e/ou Lista. - imagem de 19/12/2016

– **Variáveis:** Espaço da memória reservado para guardar uma informação.

VARIAVEIS	Informa o valor da variável.
mude VARIABEL para 0	Atribui a variável o valor especificado.
adicone a VARIABEL 1	Altera o valor da variável em um valor especificado.
mostre variável VARIABEL	Mostra no monitor as variáveis no palco.
esconda a variável VARIABEL	Esconde o monitor da variável de modo que não seja visível no palco.

Tabela A.12: Comandos de variáveis.

fonte:<https://scratch.mit.edu/>

– **Lista:** É um espaço na memória do computador capaz de armazenar uma lista de informações.



Figura A.30: Ambiente de criação de Variáveis e/ou Lista. - imagem de 19/12/2016

	Informa todos os itens da lista
	Adiciona item especificado ao final da lista.
	Exclui um ou todos os itens da lista.
	Insere novo item na posição especificada da lista.
	Substitui o valor de um item da lista pelo valor especificado.
	Coloca o item na posição especificada da lista.
	Informa quantos itens há na lista.
	Tem o valor "VERDADEIRO" se a lista contiver o valor especificado.
	Mostra o monitor da lista no palco.
	Esconde o monitor da lista que está no palco.

Tabela A.13: Comandos de lista.

fonte:<https://scratch.mit.edu/>

- Comandos de eventos.

	Executa o script após clicar na bandeira verde.
	Executa um script quando a tecla especificada é pressionada.
	Executa um script após clicar no ator.
	Estabelece a direção do ator atual.
	Executa um script quando o pano de fundo muda para um dado cenário.
	Executa um script quando o atributo selecionado ultrapassar um valor especificado.
	Executa um script quando recebe a mensagem de difusão especificada.
	Envia uma mensagem a todos os atores e espera. (x, y)

Tabela A.14: Comandos de eventos.

fonte:<https://scratch.mit.edu/>

- Comandos de controle.

	Espera um número especificado de segundos, continuando depois para o próximo bloco.
	Executa os blocos envolvidos um determinado número de vezes.
	Executa repetidamente e para sempre os comandos no seu interior.
	Se a condição for VERDADEIRA, executa os blocos envolvidos.
	Se a condição for VERDADEIRA, executa os comandos envolvidos pela porção SE; se não for, executa os comandos envolvidos pela porção SENÃO.
	Espera até que a condição seja verdadeira, depois executa os comandos que se seguem.
	Repete os blocos que se seguem até que a condição seja VERDADEIRA.
	Interrompe todos os scripts em todos os atores.
	Diz a um clone o que fazer quando criado.
	Cria um clone (duplicação temporária) do ator especificado.
	Exclui um clone atual de um ator

Tabela A.15: Comandos de controle.

fonte:<https://scratch.mit.edu/>

- Comandos de sensores.

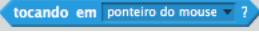
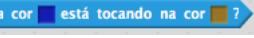
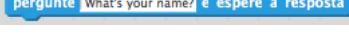
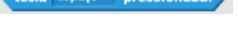
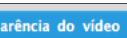
 tocando em ponteiro do mouse ?	Tem o valor verdadeiro se o ator estiver tocando no ator especificado, na borda ou no cursor do mouse.
 tocando na cor ?	Tem o valor verdadeiro se o ator está tocando na cor especificada.
 a cor está tocando na cor ?	Tem o valor verdadeiro se a primeira cor estiver tocando a segunda cor.
 distância até ponteiro do mouse	Informa a distância até o ator especificado ou até o cursor do mouse.
 pergunte What's your name? e espere a resposta	Faz uma pergunta na tela e armazena a entrada do teclado em resposta.
 resposta	Relata de entrada de teclado do uso mais recente de pergunte... e espere
 tecla espaço pressionada?	Tem valor verdadeiro se a tecla especificada estiver sendo pressionada.
 mouse pressionado?	Tem o valor verdadeiro se o botão do mouse estiver pressionado.
 posição x do mouse	Tem o valor da coordenada x da posição do ponteiro do mouse.
 posição y do mouse	Tem o valor da coordenada y da posição do ponteiro do mouse.
 ruido	Informa a intensidade (entre 1 e 100) dos sons capturados pelo microfone do computador.
 movimento do vídeo em este ator	Deteta quanto de movimento ou direção há na imagem do vídeo atualmente.
 vídeo ligado	Liga a câmera de vídeo.
 mude a transparência do vídeo para %	Estabelece o grau de transparência do vídeo.
 cronômetro	Informa o valor do cronômetro em segundos.
 zere o cronômetro	Zera o cronômetro.
 posição x de Ator 1	Informa um atributo do ator ou do palco.
 minuto atual	Tem o valor da hora atual.
 número de dias desde 2000	Informa o número de dias decorridos desde 2000.
 nome de usuário	Informa o nome de usuário do usuário que está usando o projeto.

Tabela A.16: Comandos de sensores.

fonte:<https://scratch.mit.edu/>

- Comandos de operadores.

	Adiciona dois números.
	Subtrai o primeiro número pelo segundo.
	Multiplica dois números.
	Divide o primeiro número pelo segundo.
número aleatório entre 1 e 10	Escolhe um valor inteiro aleatório num intervalo especificado.
	Tem o valor verdadeiro se o primeiro valor for menor que o segundo.
	Informa verdadeiro se os dois valores forem iguais.
	Tem o valor verdadeiro se o primeiro valor for maior que o segundo.
	Tem o valor verdadeiro se ambas as condições forem verdadeiras.
	Tem o valor verdadeiro se alguma das condições for verdadeira.
	Tem o valor verdadeiro se a condição for falsa e tem o valor falso se a condição for verdadeira.
junta hello com world	Concatena (combina, junta) cadeias de caracteres.
letra 1 de world	Apresenta o caractere na posição especificada da cadeia de caracteres.
tamanho de world	Informa o número de caracteres em uma cadeia de caracteres.
resto de por	Tem o valor do resto da divisão do primeiro número pelo segundo número.
arredondamento de	Tem o valor do número inteiro mais próximo do número dado.
raiz quadrada de 9	Calcula uma função.

Tabela A.17: Comandos de operadores.

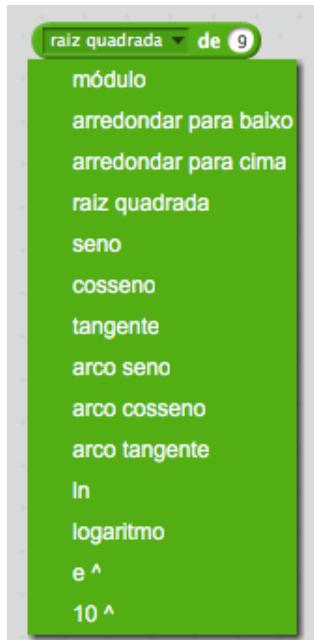


Figura A.31: Mais opções do operador função. - imagem de 19/12/2016

fonte:<https://scratch.mit.edu/>

- **Mais blocos.** Esta função permite criar uma rotina ou um comando não contido nas categorias de comandos. Há também a possibilidade de habilitar uma extensão, permitindo conexão com dispositivos físicos ou a serviços web.

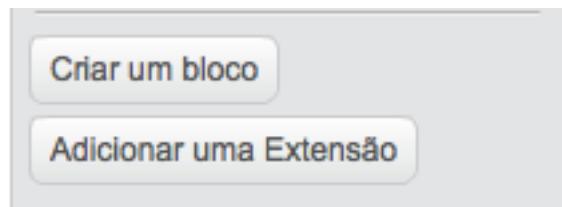


Figura A.32: Aparência básica da categoria mais blocos. - imagem de 19/12/2016

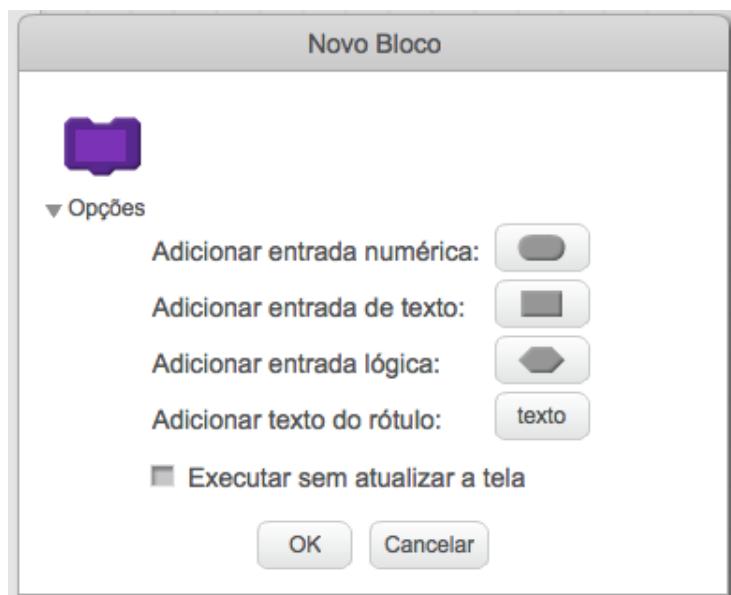


Figura A.33: Criando um novo bloco. - imagem de 19/12/2016

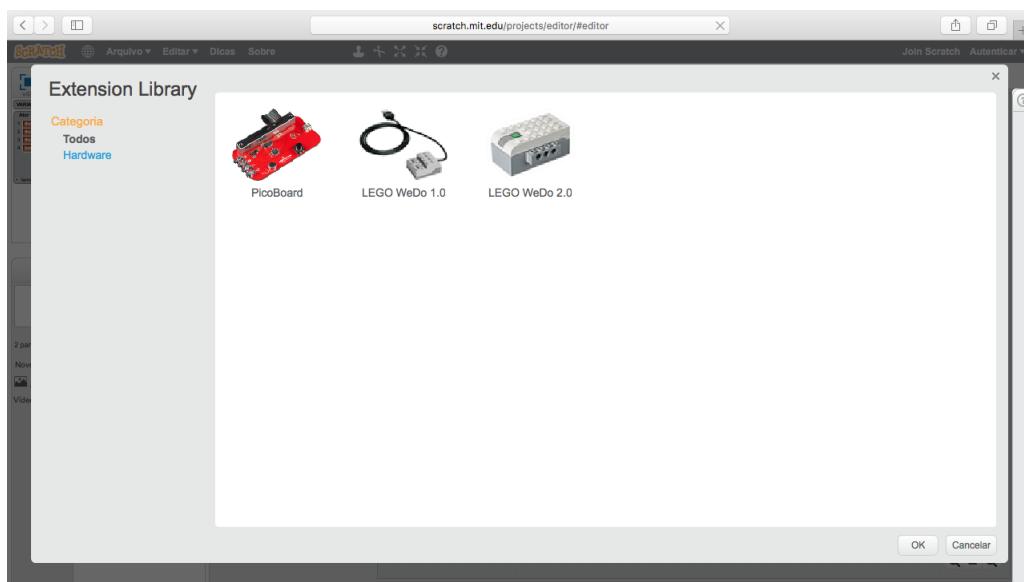


Figura A.34: Lista de dispositivos de extensão. - imagem de 19/12/2016

APÊNDICE B

LINKS PARA MATERIAIS DE APOIO AO PROJETO

B.1 Links para os vídeos

Módulo	Link
0- Apresentação	https://youtu.be/fiTz1Zf8tOI?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAJZ
1 - Atividade 01	https://youtu.be/9g5WuwdcEEo?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAJZ
2 - Apresentação	https://youtu.be/_jtS7i0fEsw?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAJZ
2 - Atividade 01	https://youtu.be/kNbLo-Xjtgw?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAJZ
2 - Atividade 02	https://youtu.be/kqhh1OjKQl4?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAJZ
2- Atividade 03	https://youtu.be/mUL9a1KbfEc?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAJZ
2 - Atividade 04	https://youtu.be/lMeh2gZluT0?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAJZ
2 - Atividade 05	https://youtu.be/9g9w-OyjxaY?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAJZ
3 - Apresentação	https://youtu.be/DkEFTFigDz4?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAJZ
3 - Atividade 01	https://youtu.be/kVqpKPDt_gs?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAJZ
3 - Atividade 02	https://youtu.be/E-A6FxWD9w0?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAJZ
3 - Atividade 03	https://youtu.be/eZ6lIjLQhLM?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAJZ
3 - Atividade 04	https://youtu.be/fS-YOPTC33I?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAJZ
4 - Apresentação	https://youtu.be/x26UCNE-qyg?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAJZ
4 - Atividade 01	https://youtu.be/LooQBf2qAHg?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAJZ
4 - Atividade 02	https://youtu.be/54v0TV6Q-io?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAJZ
4 - Atividade 03	https://youtu.be/FJaVtI8Wrt0?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAJZ
4 - Atividade 04	https://youtu.be/Bwtu6w8cKUU?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAJZ
4 - Atividade 05	https://youtu.be/czaxJogST6s?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAJZ
4 - Atividade 06	https://youtu.be/M07wBB48qEo?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAJZ

Tabela B.1: Links para vídeos das atividades 1/2

Módulo	Link
5 - Apresentação	https://youtu.be/6AeBZoMFIDk?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAjZ
5 - Atividade 01	https://youtu.be/eYdyDxyA-A?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAjZ
5 - Atividade 02	https://youtu.be/GeVmo2nP1Y?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAjZ
5 - Atividade 03	https://youtu.be/JREGA8-ZqZs?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAjZ
5 - Atividade 04	https://youtu.be/dYpouw0nBTg?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAjZ
5 - Atividade 05	https://youtu.be/ikg-hPvYSY?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAjZ
5 - Atividade 06	https://youtu.be/1gHdfQRVMR0?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAjZ
5 - Atividade 07	https://youtu.be/hodBVvJy-gI?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAjZ
5 - Atividade 08	https://youtu.be/SGEVWU6hFls?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAjZ
6 - Apresentação	https://youtu.be/rSbccPCsFrM?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAjZ
6 - Atividade 01	https://youtu.be/Y0wtrokKh2k?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAjZ
6 - Atividade 02	https://youtu.be/zMPxvpTBKKs?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAjZ
6 - Atividade 03	https://youtu.be/Sw9KwjafCOw?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAjZ
6 - Atividade 04	https://youtu.be/AfvBhnpAREs?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAjZ
6 - Atividade 05	https://youtu.be/x2a20jtFPOg?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAjZ
6 - Atividade 06	https://youtu.be/Bg8eH9gZEnQ?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAjZ
7 - Apresentação	https://youtu.be/hplq2Kbf9ME?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAjZ
7 - Atividade 01	https://youtu.be/WQSdEq0LxQo?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAjZ
7 - Atividade 02	https://youtu.be/cnnlhRFK4Sc?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAjZ
7 - Atividade 03	https://youtu.be/WLSkNnOQfeI?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAjZ
7 - Atividade 04	https://youtu.be/9fz-thTRG3c?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAjZ
8 - Apresentação	https://youtu.be/meUu5Rd2woU?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAjZ
8 - Atividade 01	https://youtu.be/GBoDup2TYzE?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAjZ
8 - Atividade 02	https://youtu.be/w2u1TJAZF6c?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAjZ
8 - Atividade 03	https://youtu.be/7OJ4DWPKRiQ?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAjZ
8 - Atividade 04	https://youtu.be/X1A8jXsG-hA?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAjZ
8 - Atividade 05	https://youtu.be/9RYMy_5OojI?list=PLH30CwiU7gvBCc_Tnjl82Ezt31klRQAjZ

Tabela B.2: Links para vídeos das atividades 2/2

B.2 Links para projetos exemplo para módulos

Módulo	Link
1	scratch.mit.edu/projects/embed/154937141
2	scratch.mit.edu/projects/embed/151168936
3	scratch.mit.edu/projects/embed/151917497
4	scratch.mit.edu/projects/embed/151989165
5	scratch.mit.edu/projects/embed/152532672
6	scratch.mit.edu/projects/embed/153231074
7	scratch.mit.edu/projects/embed/153263952
8	scratch.mit.edu/projects/embed/153343950

Tabela B.3: Projetos exemplo.

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] MEC Brasil. Parâmetros curriculares nacionais para o ensino médio: Ciências da natureza, matemática e suas tecnologia. *Secretaria de Educação Média e Tecnológica, Brasília, 2000c*. Disponível em:[j http://portal.mec.gov.br/seb/arquivos/pdf/ciencian.pdf](http://portal.mec.gov.br/seb/arquivos/pdf/ciencian.pdf); Acessado em, 01, 2017.
- [2] Seymour Papert. A máquina das crianças. *Porto Alegre: Artmed*, 1994.
- [3] Marc Prensky. Digital natives, digital immigrants part 1. *On the horizon*, 9(5):1–6, 2001.
- [4] Andréia Paula Ferreira de Araújo, Pricila Rodrigues de Souza, and Jando Abraão de Miranda Silva. Uso do scratch no processo de aprendizagem em sala de aula: Relato de experiências de alunos do mestrado profissional de ensino tecnológico/ifam. *Revista Brasileira da Educação Profissional e Tecnológica*, 1(7):125–136, 2016.
- [5] Ronaldo Machado da silva. O uso da linguagem logo na educação infantil. artigo acadêmico - universidade federal de mato grosso (ufmt). Disponível em:[j http://docplayer.com.br/16572888-O-uso-da-linguagem-logo-na-educacao-infantil.html](http://docplayer.com.br/16572888-O-uso-da-linguagem-logo-na-educacao-infantil.html); Acessado em, 10, 2017.