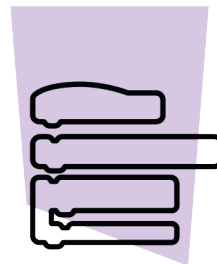
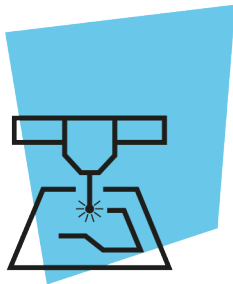
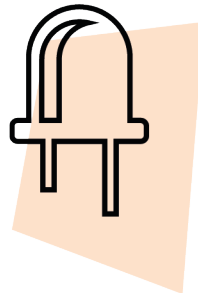
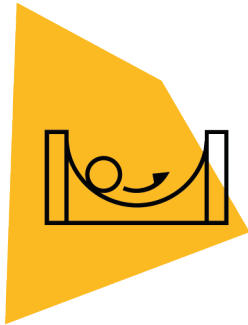
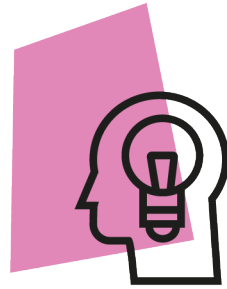


ROBÓTICA

Módulo 1



Softwares Arduino IDE e mBlock

AULA 05

GOVERNADOR DO ESTADO DO PARANÁ

Carlos Massa Ratinho Júnior

SECRETÁRIO DE ESTADO DA EDUCAÇÃO

Renato Feder

DIRETOR DE TECNOLOGIA E INOVAÇÃO

Andre Gustavo Souza Garbosa

COORDENADOR DE TECNOLOGIAS EDUCACIONAIS

Marcelo Gasparin

Produção de Conteúdo

Cleiton Rosa

Simone Sinara de Souza

Revisão Textual

Adilson Carlos Batista

Leitura Crítica e Normalização Bibliográfica

Ricardo Hasper

Projeto Gráfico e Diagramação

Edna do Rocio Becker

Ilustração

Jocelin Vianna (Educa Play)

2021

Aula 01	Por Que Robótica?
Aula 02	Tensão, Corrente e Resistência
Aula 03	Kit de Robótica
Aula 04	Arduino Uno R3
Aula 05	Softwares Arduino IDE e mBlock
Aula 06	Portas Digitais
Aula 07	Circuito Elétrico
Aula 08	LED e Resistor
Aula 09	Semáforo [Carros]
Aula 10	Semáforo [Cruzamento Carros]
Aula 11	Semáforo [Pedestres]
Aula 12	Semáforo [Cruzamento Carros + Pedestres]
Aula 13	Push Button
Aula 14	Feedbacks + Inventário I
Aula 15	Semáforo [Carros + Pedestres com Botão]
Aula 16	Display 7 Segmentos
Aula 17	Fonte DC + Plug P4
Aula 18	Portas PWM
Aula 19	LED Fade-In
Aula 20	LED Fade-Out
Aula 21	Super Máquina 80's
Aula 22	Super Máquina 2008
Aula 23	Potenciômetro
Aula 24	Buzzer Passivo
Aula 25	LED RGB
Aula 26	Arco-Íris
Aula 27	Sensor LDR
Aula 28	Feedbacks + Inventário II
Aula 29	Sensor de Temperatura
Aula 30	Sensor de Obstáculo IR
Aula 31	Controle Motor DC
Aula 32	Kit Chassi 2WD Robô
Aula 33	Seguidor de Linha
Aula 34	Sensor de Distância
Aula 35	Sensor de Estacionamento
Aula 36	Display LCD 16x2
Aula 37	Trena Digital
Aula 38	Robô Sumô [Estrutura]
Aula 39	Robô Sumô [Programação + Treinamento I]
Aula 40	Robô Sumô [Programação + Treinamento II]
Aula 41	Disputa de Sumôs
Aula 42	Feedbacks + Inventário III

Aula 04
Arduino Uno R3

Aula 05 Softwares Arduino IDE e mBlock

Aula 06
Portas Digitais

Sumário

Introdução	2
Objetivos desta Aula	2
Competências Gerais Previstas na BNCC	3
Habilidades do Século XXI a Serem Desenvolvidas	4
Roteiro da Aula	4
1. Contextualização	4
2. Conteúdo	5
2.1 Software Arduino IDE	5
2.1.1 Bibliotecas	11
2.2 Software mBlock	16
2.2.1 Variáveis	19
2.2.2 Crie um bloco	20
2.2.3 Extensão	23
2.3 Simulador Tinkercad	28
3. Feedback e Finalização	36
Referências	36



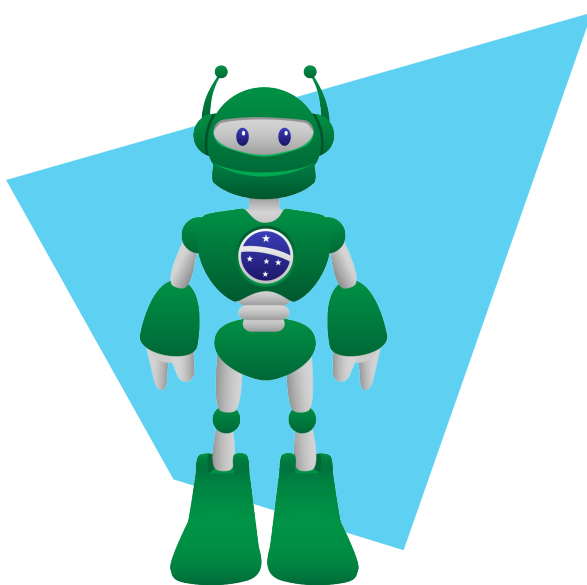
Introdução

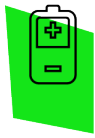
A palavra software deriva dos termos em inglês: soft (mole) e ware (mercadoria), e corresponde à parte lógica de computadores ou dispositivos móveis, ou seja, um conjunto de instruções (comandos) que deve ser seguido e executado por tais equipamentos. Nesta aula, conheceremos as características dos softwares Arduino IDE, mBlock e o simulador Tinkercad.



Objetivos desta Aula

- Conhecer os softwares Arduino IDE e mBlock e o simulador Tinkercad;
- Identificar as características de programação dos softwares Arduino IDE e mBlock.





Competências Gerais Previstas na BNCC

[CG02] - Exercitar a curiosidade intelectual e recorrer à abordagem própria das ciências, incluindo a investigação, a reflexão, a análise crítica, a imaginação e a criatividade, para investigar causas, elaborar e testar hipóteses, formular e resolver problemas e criar soluções (inclusive tecnológicas) com base nos conhecimentos das diferentes áreas.

[CG04] - Utilizar diferentes linguagens – verbal (oral ou visual-motora, como Libras, e escrita), corporal, visual, sonora e digital –, bem como conhecimentos das linguagens artística, matemática e científica, para se expressar e partilhar informações, experiências, ideias e sentimentos em diferentes contextos e produzir sentidos que levem ao entendimento mútuo.

[CG05] - Compreender, utilizar e criar tecnologias digitais de informação e comunicação de forma crítica, significativa, reflexiva e ética nas diversas práticas sociais (incluindo as escolares) para se comunicar, acessar e disseminar informações, produzir conhecimentos, resolver problemas e exercer protagonismo e autoria na vida pessoal e coletiva.

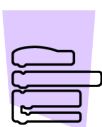
[CG09] - Exercitar a empatia, o diálogo, a resolução de conflitos e a cooperação, fazendo-se respeitar e promovendo o respeito ao outro e aos direitos humanos, com acolhimento e valorização da diversidade de indivíduos e de grupos sociais, seus saberes, identidades, culturas e potencialidades, sem preconceitos de qualquer natureza.

[CG10] - Agir pessoal e coletivamente com autonomia, responsabilidade, flexibilidade, resiliência e determinação, tomando decisões com base em princípios éticos, democráticos, inclusivos, sustentáveis e solidários.



Habilidades do Século XXI a Serem Desenvolvidas

- Pensamento crítico;
- Afinidade digital;
- Resiliência;
- Resolução de problemas;
- Colaboração;
- Comunicação.



Roteiro da aula

1 Contextualização (15min):

Quando pensamos em dispositivos móveis ou mesmo desktop, logo vem à mente seu formato, espessura, os botões que possuem e os dispositivos periféricos que interagem com eles, como, por exemplo: fone de ouvido, teclado, monitor, caixa de som, entre outros. Todas essas estruturas que pensamos e citamos faz parte da estrutura física desses dispositivos, também chamada de hardware. Se a estrutura física é o hardware, que estrutura é responsável pela parte lógica dessas ferramentas? Ou seja, que estrutura permite ao usuário ter acesso a diversas funcionalidades, como, por exemplo, sistemas operacionais, jogos, editores de texto, elaboração de planilhas, calendário, entre outras? Isso mesmo! O software. No ensino de robótica, você saberia dizer qual é a importância dos softwares?

2 Conteúdo (60min):

Conforme comentamos, o software é a parte lógica presente nos desktops e dispositivos móveis, ou seja, é responsável pela sequência de instruções (comandos) enviadas a esses equipamentos para que sejam executadas.

No caso da robótica, os softwares permitem programar protótipos ou robôs para que realizem determinadas funções. Nesta aula, vamos conhecer os dois softwares que serão utilizados para as atividades propostas ao longo deste módulo, sendo eles: Arduino IDE e mBlock e o simulador Tinkercad, nos quais você poderá montar circuitos elétricos e programá-los para os seus funcionamentos.

2.1 Software Arduino IDE

O Arduino IDE (do inglês, Integrated Development Environment), ou ambiente de desenvolvimento integrado do Arduino, é uma plataforma gratuita que utiliza uma linguagem de programação baseada em C/C++; isso significa que possui uma linguagem de nível médio e suporta mais de um estilo de programação. Este software se conecta ao hardware Arduino para fazer upload de programas e se comunicar com eles e é um ambiente compatível com os sistemas operacionais Windows, Mac OS X e Linux, podendo ser utilizado na versão off-line, quando estiver instalado no computador, ou na versão on-line, conhecida por Arduino Web Editor.

Para instalar a versão off-line do software Arduino IDE no desktop ou no dispositivo móvel, acesse o site oficial do Arduino, disponível no link <https://www.arduino.cc/en/Main/Software>, e clique sobre a IDE adequada a seu sistema operacional, conforme mostra a figura 1.



<https://www.arduino.cc/en/Main/Software>

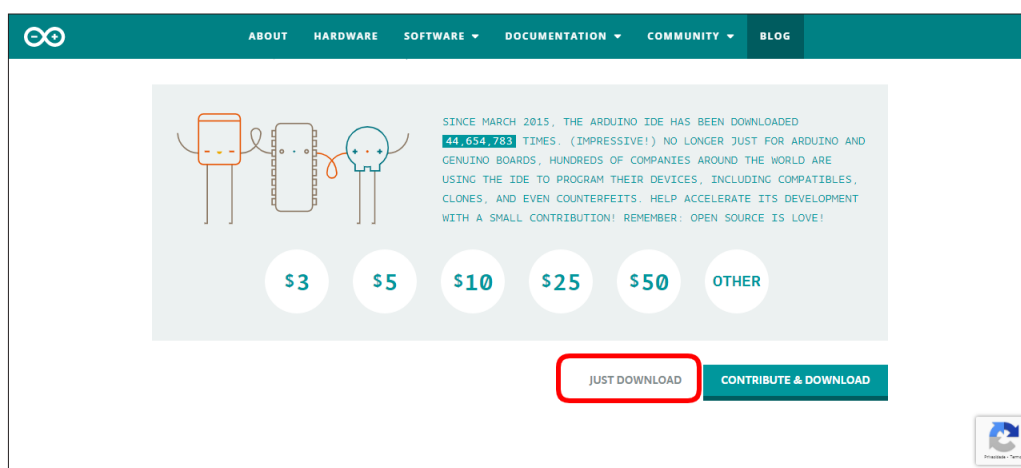
Figura 1 - Versões Arduino IDE para download



Fonte: site Arduino oficial

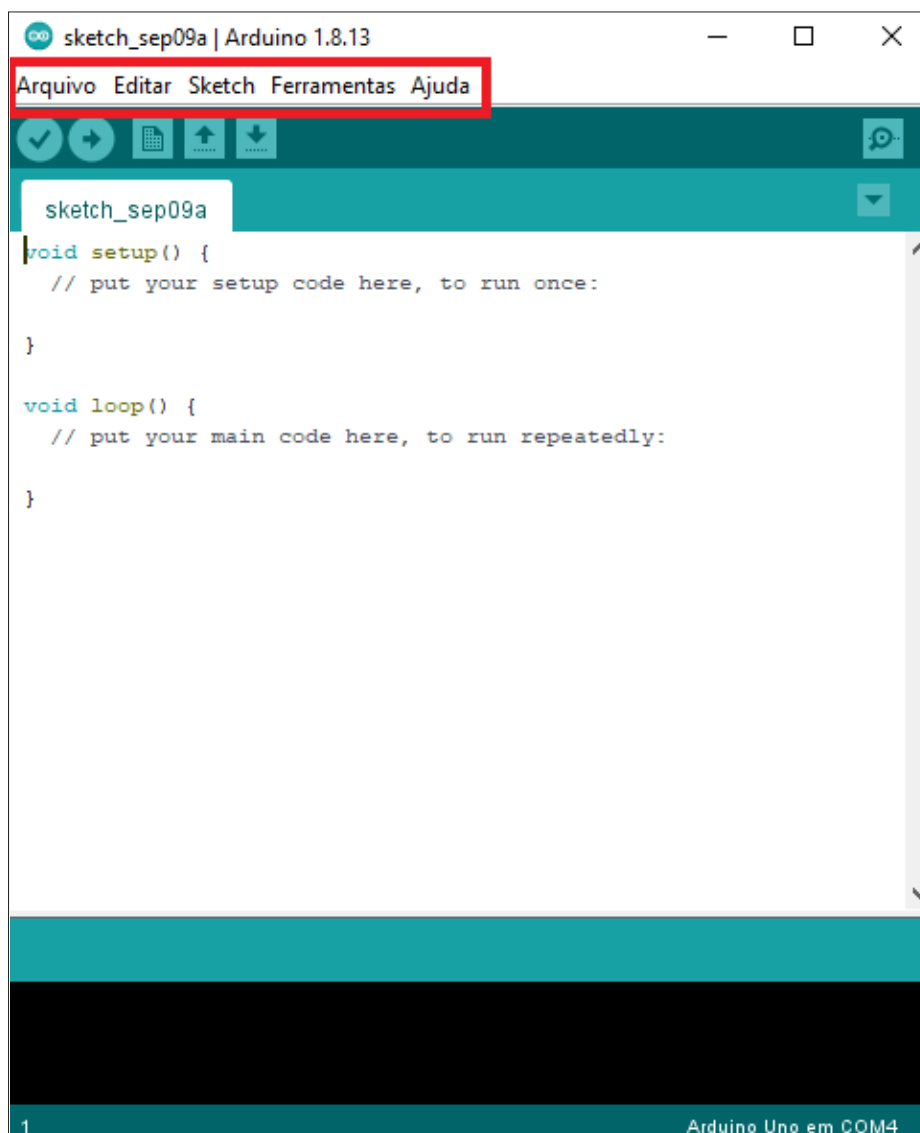
Ao clicar sobre o nome do sistema operacional, abrirá a página para que você possa realizar o download do software. Para tanto, clique no botão **JUST DOWNLOAD** (apenas baixar), como mostra a figura 2.

Figura 2 - Download do software Arduino IDE



Uma vez instalado o software no desktop ou outro dispositivo móvel, você terá acesso a todas as funcionalidades do IDE, dispostas em cinco categorias: **Arquivo**, **Editar**, **Sketch**, **Ferramentas** e **Ajuda**, apresentadas na figura 3.

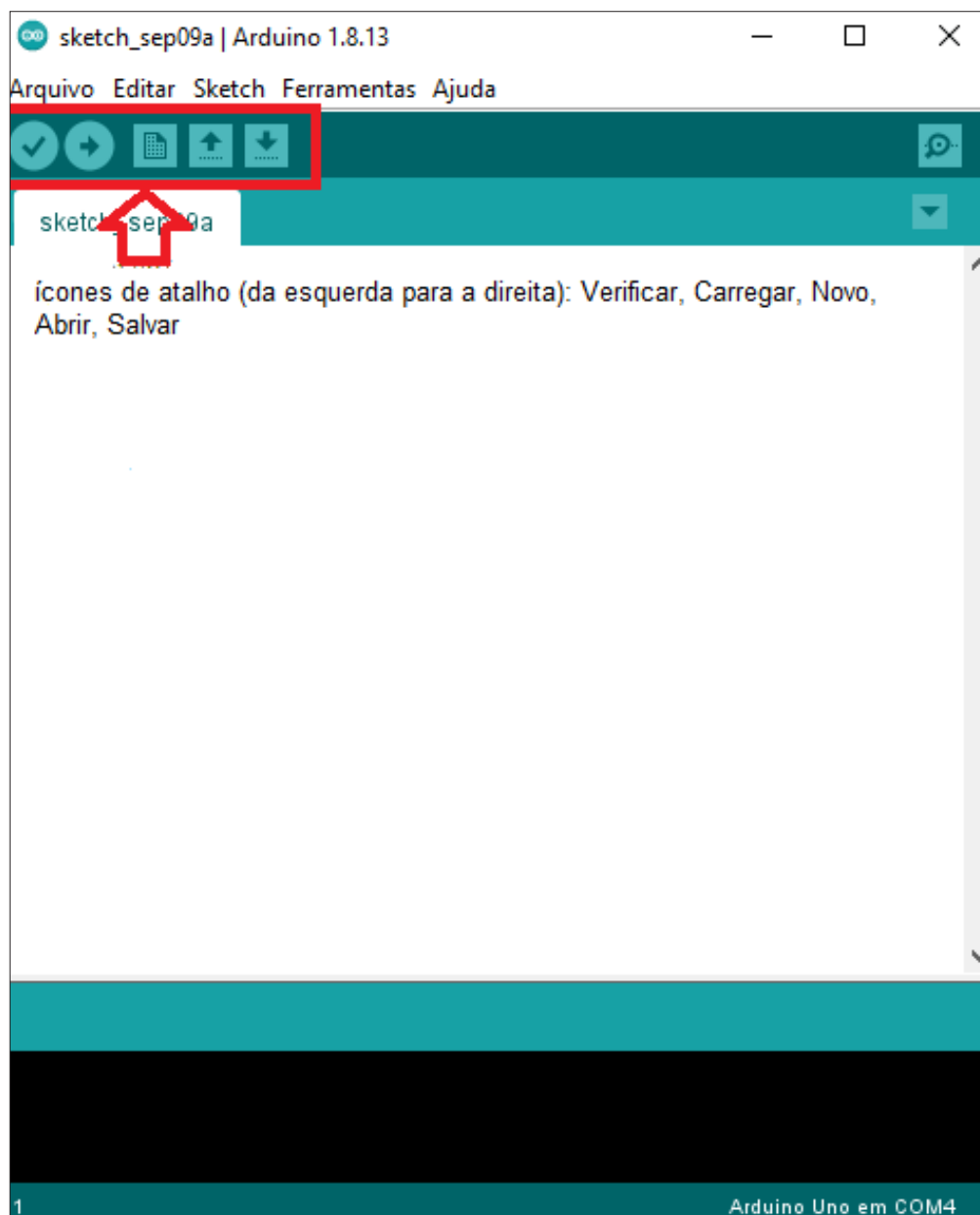
Figura 3 - Layout do software Arduino IDE, versão off-line



No menu **Sketch** são apresentados os *sketches* (esboços) elaborados pelo usuário, ou seja, todos os programas escritos no editor de texto do software Arduino IDE e salvos com a extensão de arquivo **.ino**.

Abaixo do menu principal, estão dispostos ícones de atalho, que permitem identificar erros de sintaxe no código-fonte (verificar), carregar o software na placa, criar uma nova aba, em branco, para a escrita de sketch, abrir um sketch já salvo no computador e salvar um sketch em desenvolvimento, figura 4.

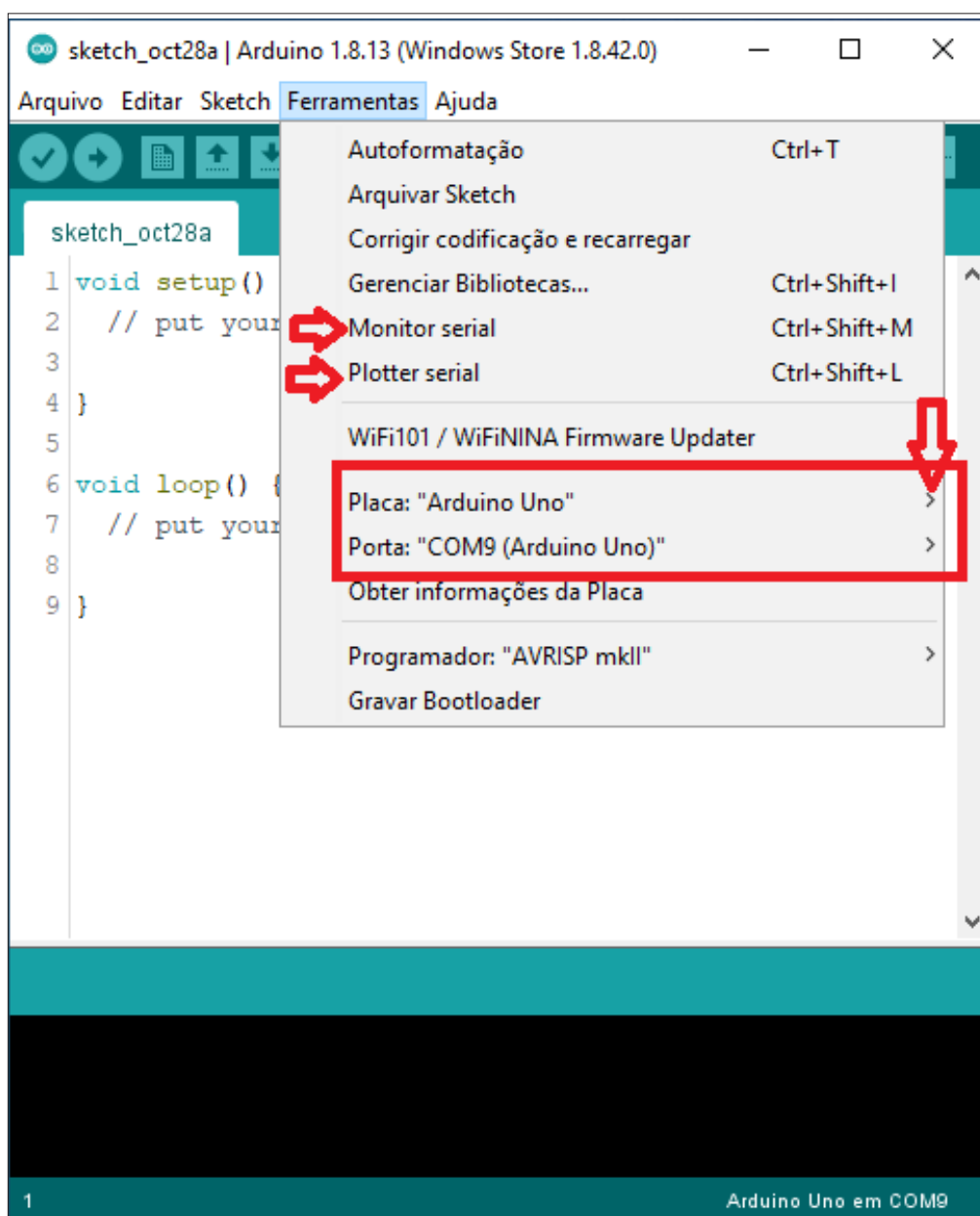
Figura 4 - Ícones de atalho de algumas funções do Arduino IDE



Para iniciar a programação, acesse o software Arduino IDE e conecte o computador, através de um cabo USB, à placa de Arduino Uno para que ocorra a comunicação entre ambos. No Arduino IDE, clique no menu **Ferramentas** e selecione a placa e a porta que concluirão a comunicação entre o software e a placa de prototipagem eletrônica, figura 5. Uma vez realizada a comunicação, escreva

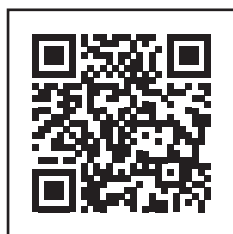
ou copie e cole o código-fonte de programação para as funções a serem realizadas. Compile o programa pressionando o botão **Verificar** (botão com sinal de tique) para averiguar se não há erros de sintaxe. Estando o código correto, o próximo passo é realizar a transferência do programa para a placa Arduino. Para tal, pressione o botão **Baixar** (botão com uma seta apontando para a direita), para iniciar o carregamento do programa para o Arduino Uno. A partir da transferência de dados, a placa Arduino comandará a realização das funções programadas.

Figura 5 - Seleção de placa e porta no software Arduino IDE



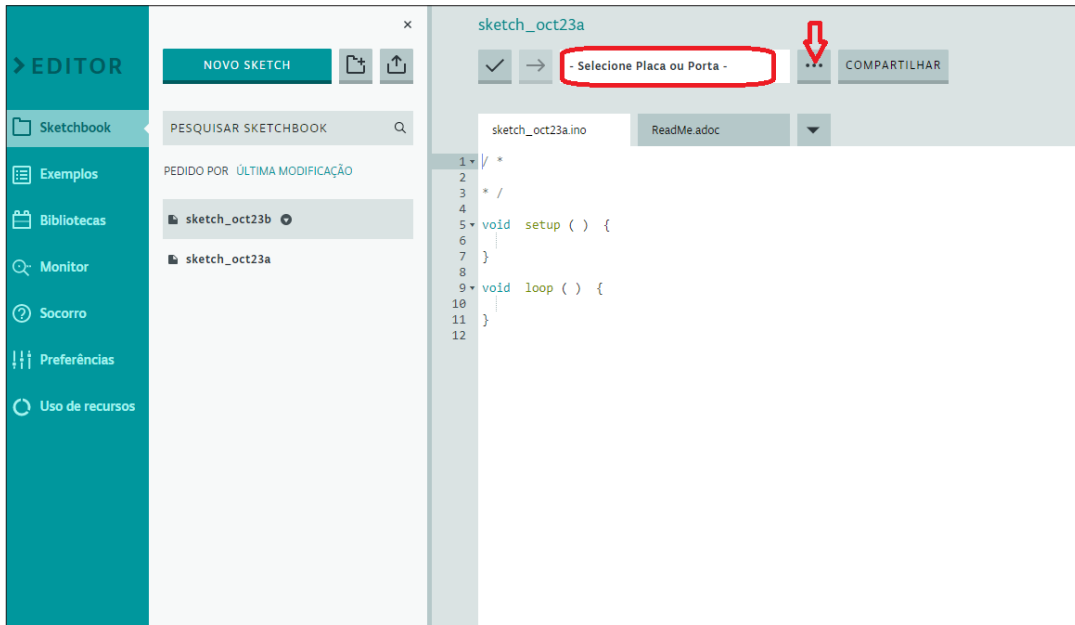
No menu **Ferramentas**, destacamos, ainda, outras duas funções: **Monitor serial** e **Plotter serial**, figura 5. A primeira função é uma interface que permite verificar o funcionamento de um programa sem que seja necessário a utilização de componente adicional, como, por exemplo, um display. Neste caso, a comunicação entre a placa Arduino e o software IDE é dita serial e ocorre através dos pinos Rx e Tx, da placa de prototipagem eletrônica com os pinos de dados do computador, via cabo USB (data+ e data-). Essa comunicação permite o envio e recebimento de dados entre o computador e a placa Arduino. Já a função **Plotter serial** corresponde a uma janela que transforma os dados recebidos em forma de gráficos. Ambas funções permitem escolher a velocidade de envio e recebimento de dados entre a placa Arduino Uno e o software IDE.

Caso você não deseje ou não possa instalar a versão off-line do Arduino IDE em seu dispositivo móvel ou desktop, você tem a opção de utilizar este software na versão on-line, conforme já relatamos anteriormente. Para fazer uso desta versão, você deverá realizar um cadastro no site oficial do Arduino, disponível no link <https://create.arduino.cc/editor>. Com o cadastro realizado, você será direcionado à página do Arduino Web Editor, a qual, apesar de apresentar layout diferente da versão off-line, possui as mesmas funcionalidades, ou seja, permite a escrita de programas (sketches), verifica possíveis erros no código-fonte, carrega o software na placa Arduino Uno, cria novo espaço para escrita de sketch, abre aqueles salvos no computador e abre os novos, figura 6.



<https://create.arduino.cc/editor>

Figura 6 - Layout do software Arduino IDE, versão on-line



Para iniciar a programação, caso seja a primeira vez que esteja utilizando a versão on-line, será necessária a instalação de um plug-in no desktop ou dispositivo móvel para que ocorra, com auxílio de um cabo USB, a comunicação entre o Arduino Web Editor e a placa de prototipagem eletrônica. Além disso, você deverá selecionar, no software, a placa Arduino e a porta (figura 6) que permitirão a comunicação entre a versão on-line e o Arduino Uno. Realizada a comunicação, escreva ou copie e cole o código-fonte de programação para as funções a serem realizadas. Compile o programa pressionando o botão **Verificar** (botão com sinal de tique) para averiguar se não há erros de sintaxe. Estando o código correto, o próximo passo é realizar a transferência do programa para a placa Arduino. Pressione o botão **Baixar** (botão com uma seta apontando para a direita), para realizar o carregamento do programa para o Arduino Uno R3. A partir da transferência de dados, a placa Arduino comandará a realização das funções programadas.

2.1.1 Bibliotecas

Uma forma de estender as funcionalidades do software Arduino IDE é através do uso de bibliotecas. Bibliotecas são trechos de um software que fornecem funcionalidades específicas a um programa, podemos citar, como exemplo, a biblioteca **Ultrasonic**, a qual oferece

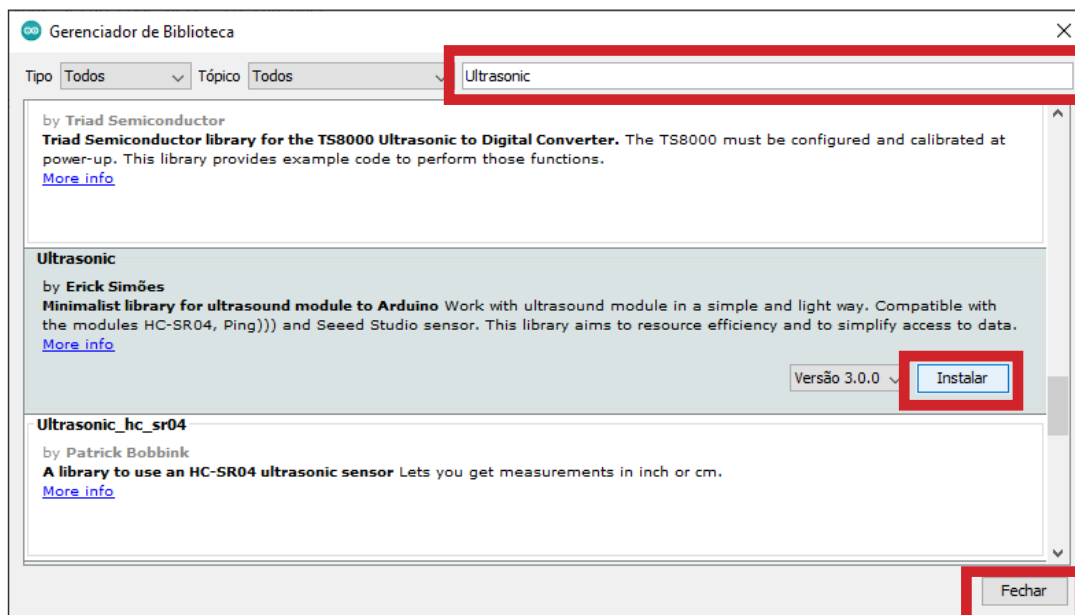
todos os comandos necessários para um Sensor de Ultrassônico fornecer a medida de distância. Para tanto, basta o programador inserir esta biblioteca em seu programa, simplificando assim a elaboração de aplicações.

A instalação de uma biblioteca no software Arduino IDE é simples e pode ser feita no modo off-line e on-line.

Na versão off-line, existe um ambiente dentro do Arduino IDE que gerencia a instalação de bibliotecas, você pode acessá-lo através do menu **Sketch > Incluir Biblioteca** e, então, escolher uma das duas opções de instalação: gerenciar bibliotecas ou adicionar biblioteca.

a) Gerenciar Bibliotecas - clicando nessa opção, abrirá uma janela onde você poderá pesquisar, através da barra de pesquisa, a biblioteca desejada pelo nome ou algum termo de referência. Uma vez localizada a biblioteca desejada, clique no botão **Instalar**. Aguarde a instalação, e finalize clicando no botão **Fechar**, como mostra a figura 7.

Figura 7 - Gerenciador de Biblioteca do software Arduino IDE versão off-line

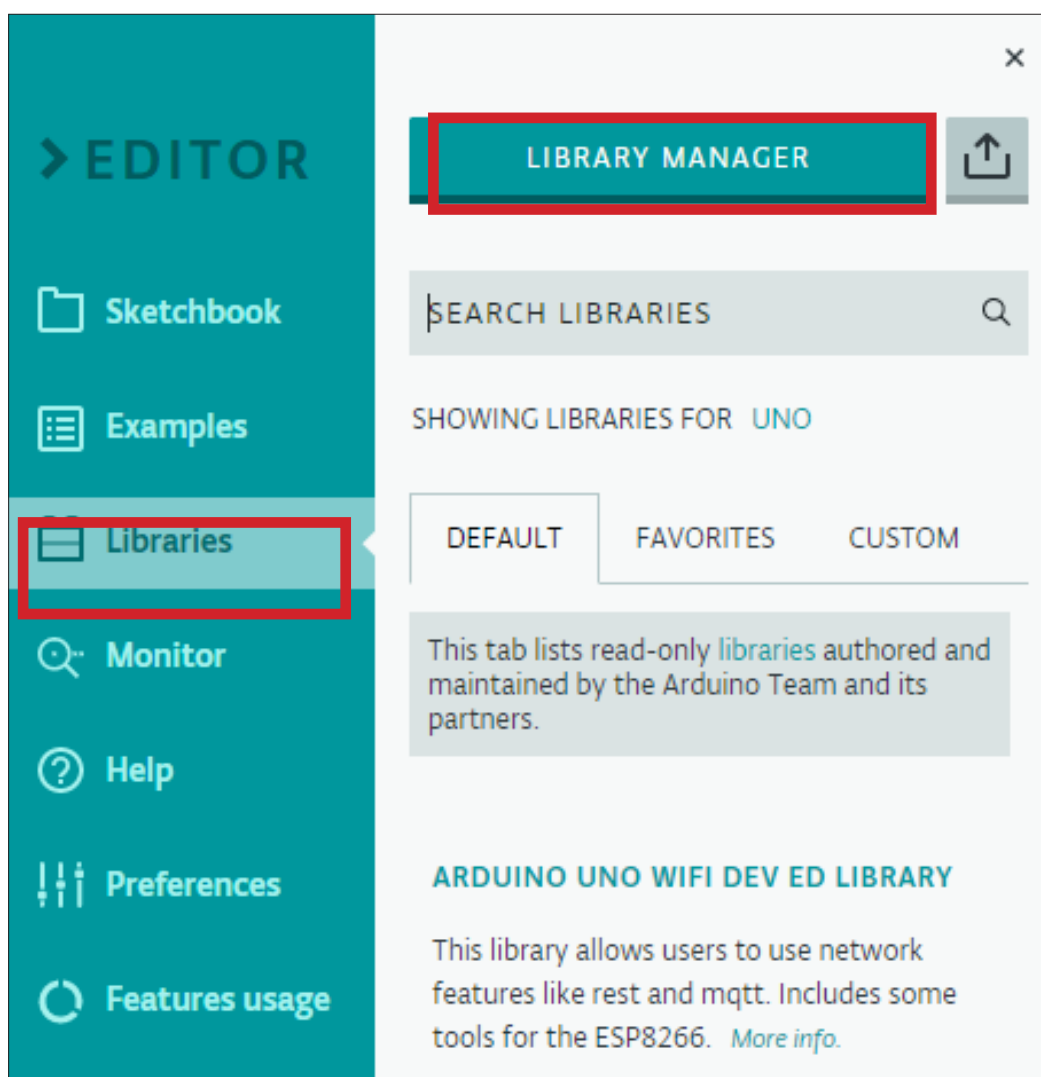


b) Adicionar Biblioteca .ZIP - escolhendo essa opção, você poderá instalar uma biblioteca baixada, através da internet, no formato

compactada **.zip**. Ao clicar sobre ela, abrirá uma janela para você selecionar a biblioteca no local de seu computador em que foi realizado o download e, então, clique em **Abrir**, concluindo assim, a instalação.

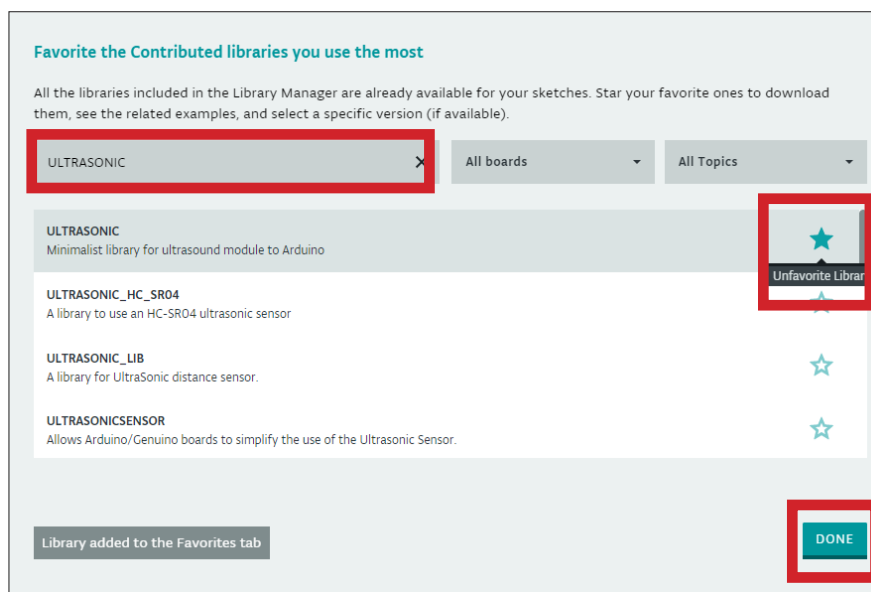
Na versão on-line do software Arduino IDE, clique em **Libraries**, localizado no menu lateral da página do software. Na sequência, clique no botão **LIBRARY MANAGER**, figura 8.

Figura 8 – Gerenciador de Biblioteca do software Arduino IDE versão on-line



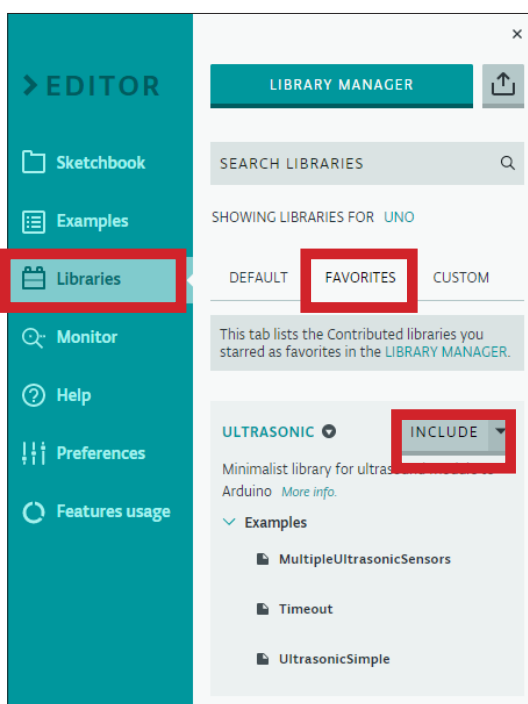
Ao clicar no botão **Library Manager**, abrirá o Gerenciador de Bibliotecas, onde você poderá, através da barra de pesquisa, buscar pelo nome a biblioteca desejada e marcá-la como favorita, clicando sobre a estrela ao seu lado e a destacando na cor azul. A seguir, clique no botão **DONE**, figura 9.

Figura 9 - Instalando biblioteca no Arduino IDE versão on-line



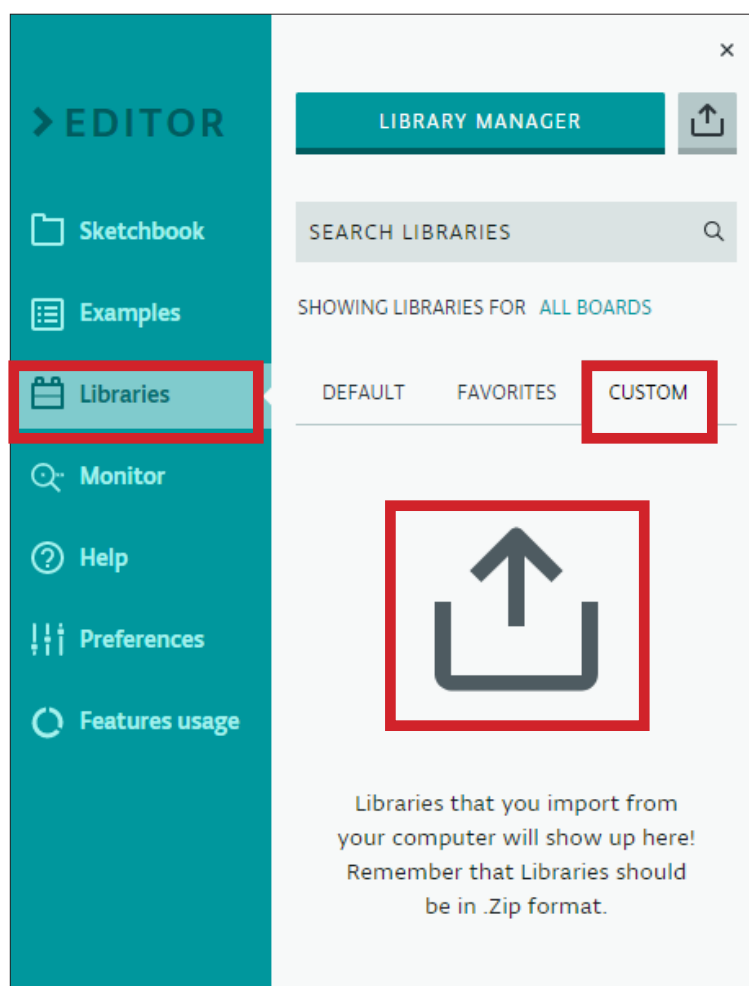
Após favoritar a biblioteca desejada, ela estará acessível e aparecerá dentro do menu lateral **Libraries**, na aba **Favorites**. Para incluí-la, clique no botão **INCLUDE**, figura 10.

Figura 10 - Instalando biblioteca no Arduino IDE versão on-line



Também é possível importar uma biblioteca no formato **.zip** de seu computador. Para tanto, no menu lateral **Libraries**, clique sobre a aba **CUSTOM** e, então, sobre o ícone de **importação**, conforme mostra a figura 11.

Figura 11 – Importando uma biblioteca no Arduino IDE versão on-line



Fonte: site Arduino oficial

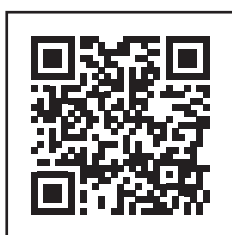
Ao clicar no ícone **Import**, abrirá um tooltip para localizar em seu computador a biblioteca a ser instalada (no formato.zip). Finalize clicando em **Abrir**. Após a instalação, a biblioteca estará acessível no menu lateral **Libraries**, dentro da aba **Favorites**. Para incluí-la clique no botão **INCLUDE**, figura 10.

Além da programação por codificação realizada pelo software Arduino IDE, você também aprenderá a programar em blocos. Para esta linguagem de programação, utilizaremos o software mBlock. Vamos conhecê-lo!

2.2 Software mBlock

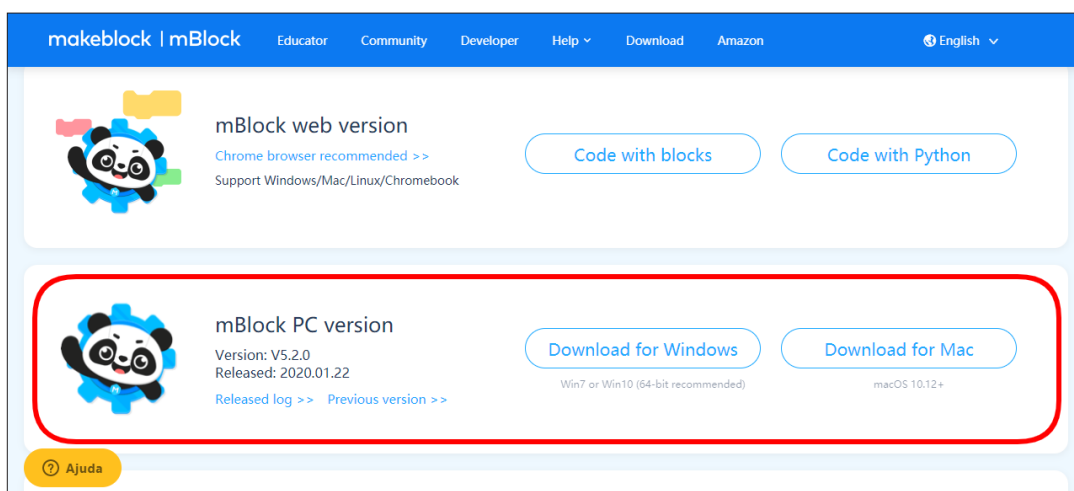
O mBlock é uma plataforma que suporta linguagens de programação gráfica e textual, permitindo, assim, programar arrastando e soltando blocos de construção. Da mesma forma que o software Arduino IDE, o mBlock se conecta ao hardware Arduino para fazer upload de programas e se comunicar com eles. Este software encontra-se disponível nas versões para desktop, on-line e aplicativos para Android e iOS.

Para instalar a versão off-line do software no desktop ou no dispositivo móvel, acesse o site oficial do mBlock, disponível no link <http://www.mblock.cc/en-us/download>, clique sobre o instalador compatível com o sistema operacional (Windows ou Mac) de seu computador e siga as orientações para a instalação do software, figura 12.



<http://www.mblock.cc/en-us/download>

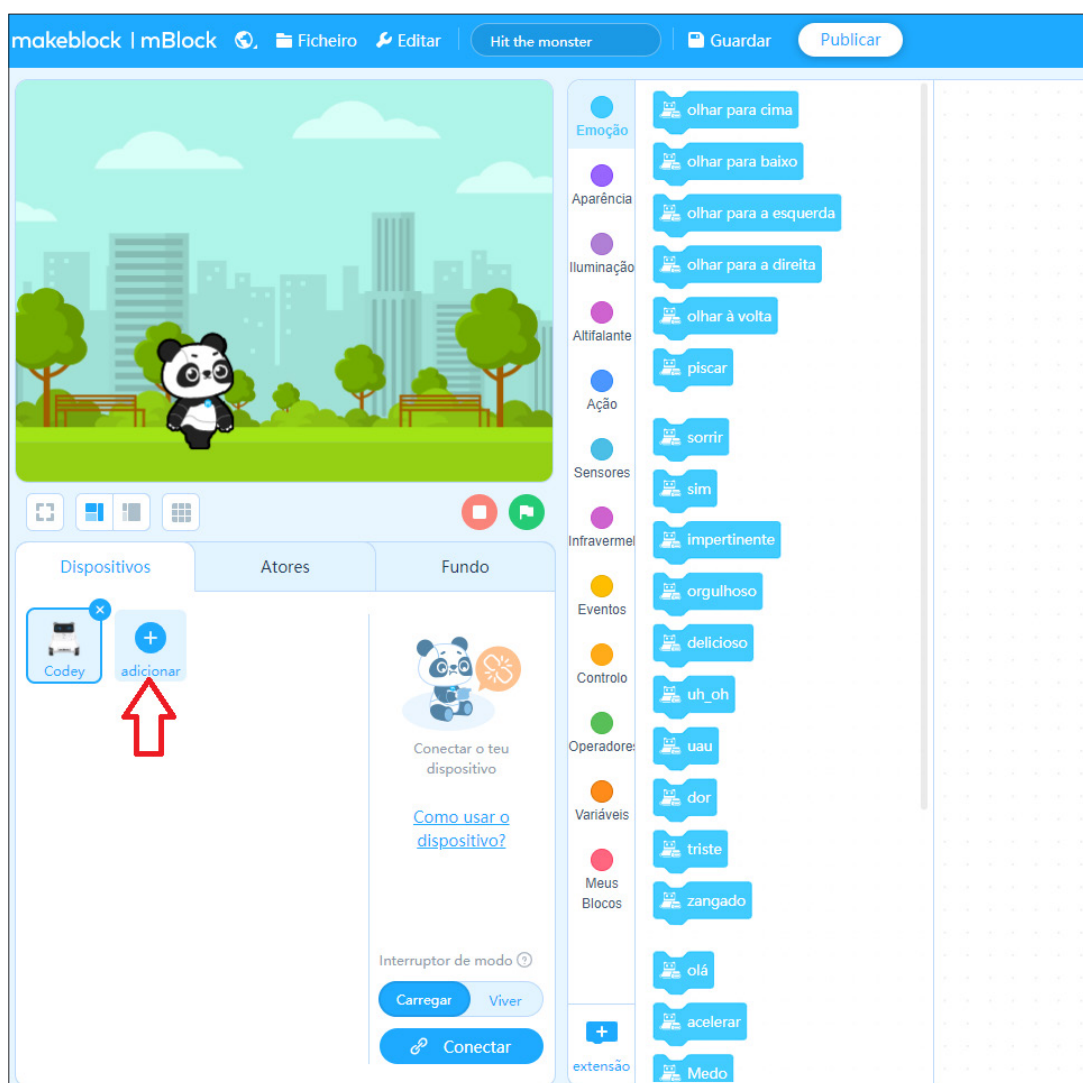
Figura 12 – Opções de download do software mBlock



Fonte: site mBlock oficial

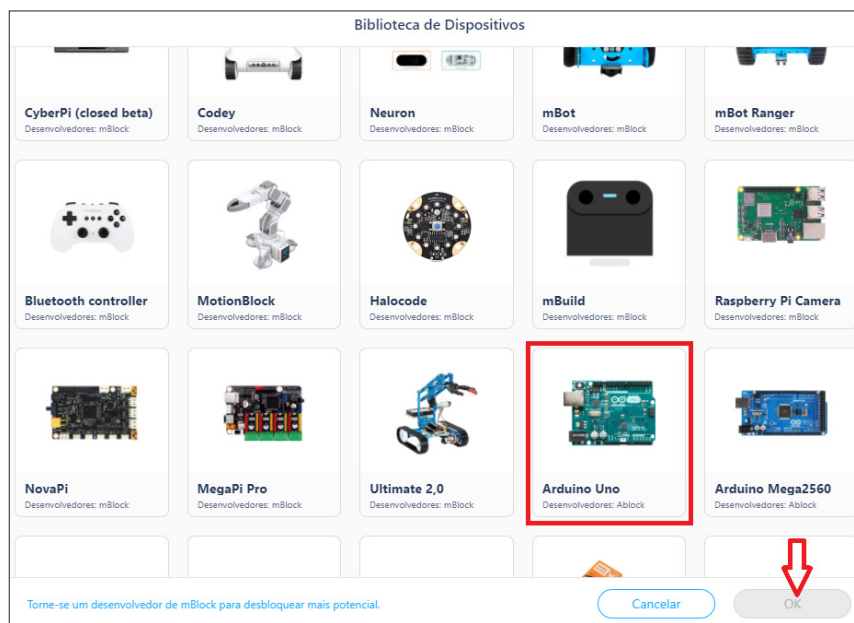
Uma vez instalado o software mBlock, conecte esse à placa Arduino Uno clicando no ícone **Adicionar**, localizado no campo **Dispositivos**, conforme mostra a figura 13.

Figura 13 - Layout do software mBlock



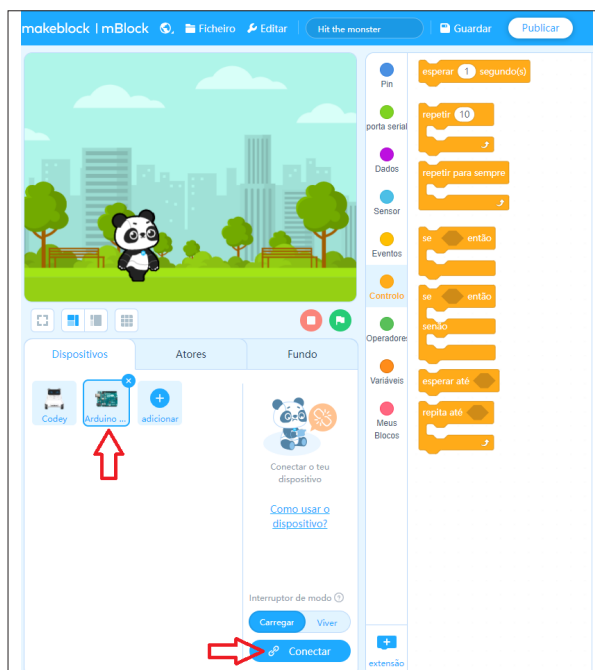
Na sequência, selecione a placa de Arduino Uno na biblioteca de dispositivos do mBlock e clique no botão **OK**, como demonstra a figura 14.

Figura 14 - Seleccionando a placa de Arduino Uno



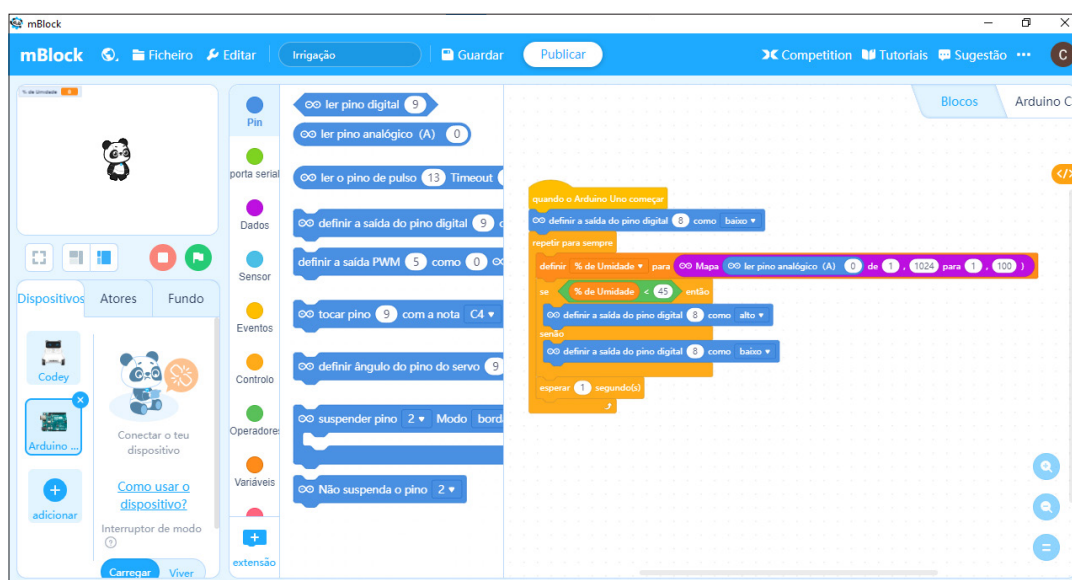
Feito a seleção, o Arduino Uno é visualizado no campo **Dispositivos** do mBlock, sendo possível iniciar a programação em blocos, figura 15.

Figura 15 - Conectando Arduino Uno ao mBlock



A seguir, monte os blocos, arrastando e soltando, de acordo com a programação das funções desejadas, como mostra a figura 16.

Figura 16 - Programação em blocos



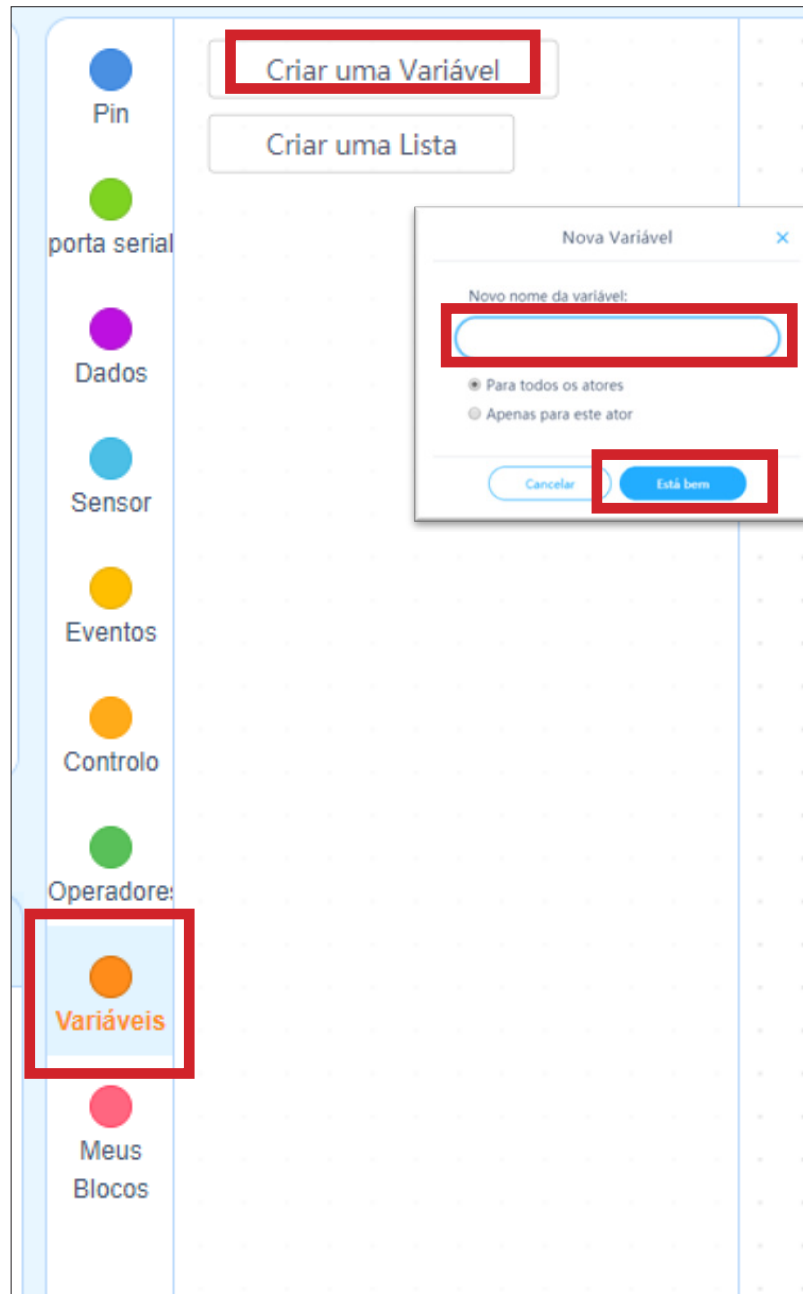
2.2.1 Variáveis:

No universo da programação é muito comum o uso de variáveis para armazenar informações ou valores na memória da placa Controladora. Para programar no Arduino, não é diferente.

Quando optar em programar através de blocos, como exemplo o mBlock, é necessário criar um novo bloco com a função de uma variável.

Para criar o bloco variável, basta clicar na aba lateral denominada **Variáveis**, na cor laranja e, então, clicar em **Criar uma Variável**, abrirá uma pequena janela onde você deverá informar um nome para a variável, feito isso, clique em **Está Bem**, figura 17. Alguns blocos, na cor laranja, referente ao uso da variável, serão gerados e poderão ser acessados e utilizados no nosso programa, através da aba **Variáveis**.

Figura 17 - Criando uma variável

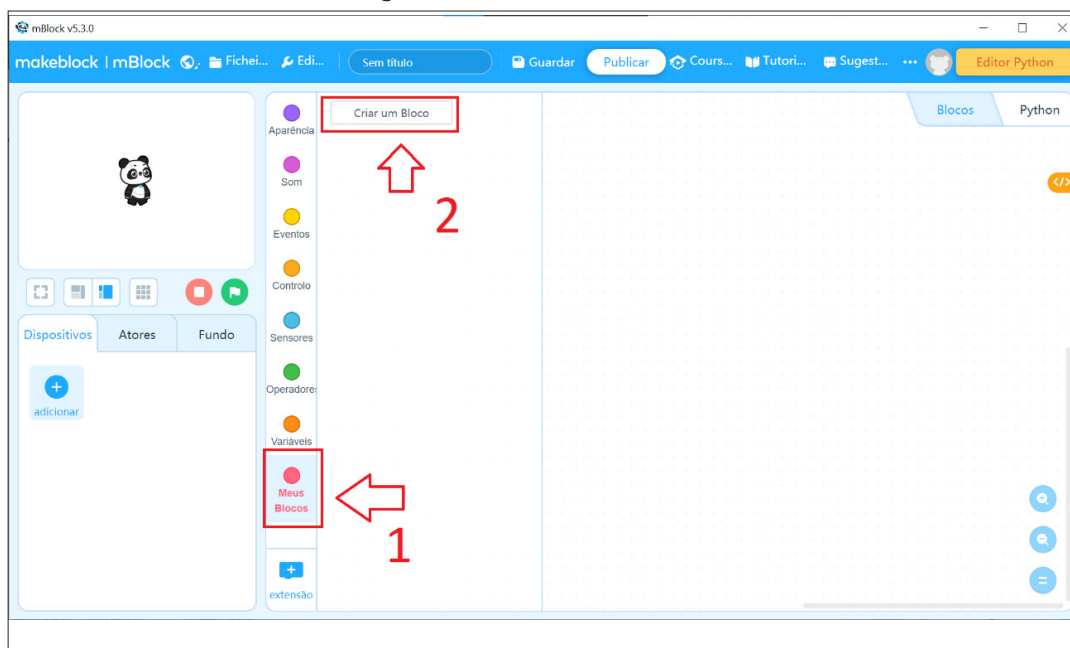


2.2.2 Crie um Bloco

Se você precisar de um bloco que não pode ser encontrado em qualquer uma das categorias de blocos disponíveis, você poderá criá-lo.

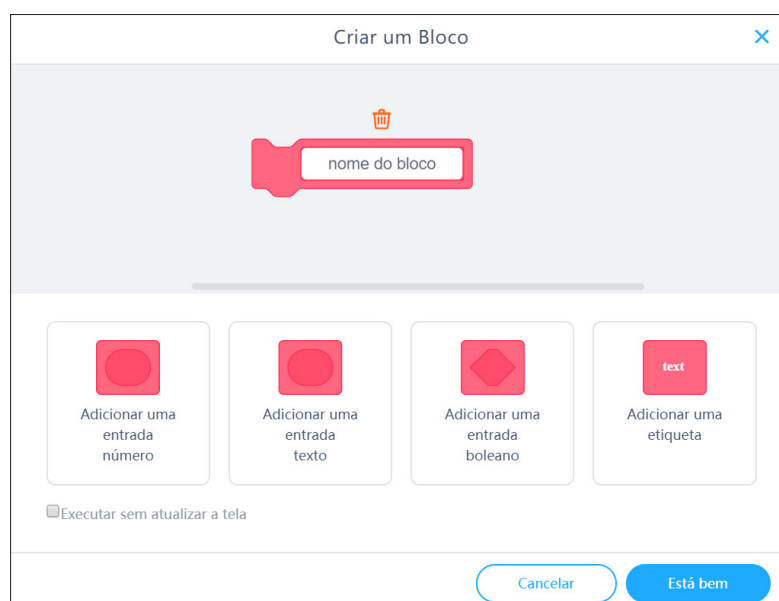
Clique em **“Meus Blocos > Criar um Bloco”**, figura 18.

Figura 18 - Criando um bloco



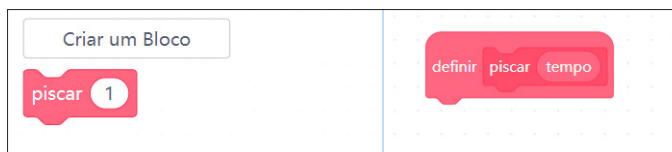
A seguir, defina a aparência do seu bloco, figura 19.

Figura 19 - Definindo a aparência do bloco



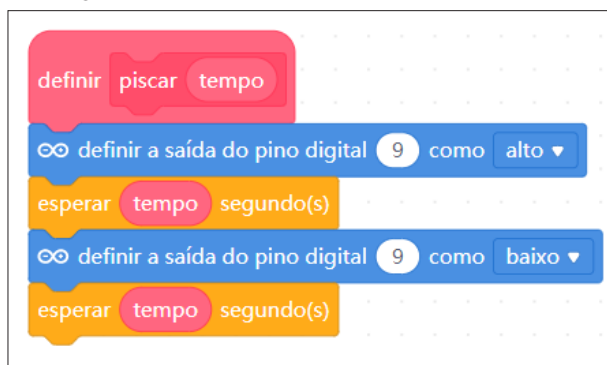
Você pode adicionar caixas de entrada e etiquetas para o seu bloco, conforme necessário. Neste exemplo, crie um bloco chamado piscar com uma entrada de número chamada tempo, figura 20.

Figura 20 - Bloco criado



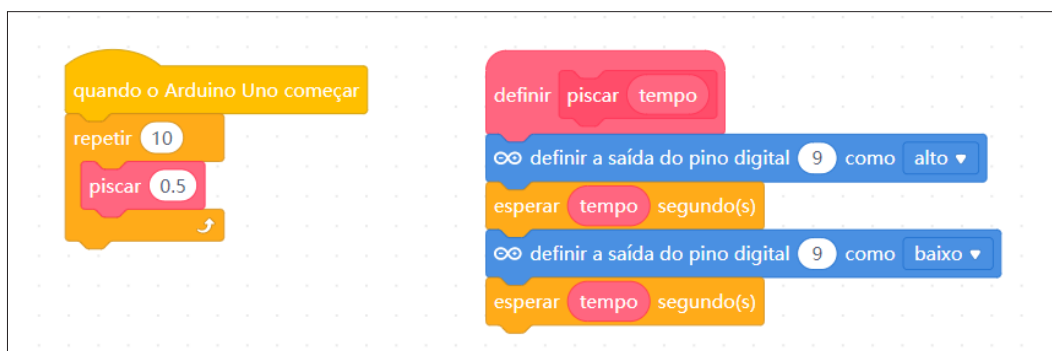
Defina a função do seu bloco encaixando os comandos nas sequências que eles deverão ser executados, figura 21.

Figura 21 - Definindo a função do bloco



Monte seu programa para testar seu bloco, figura 22.

Figura 22 - Utilizando o bloco na programação



Seu programa utilizará 10 vezes o bloco **piscar**, que consiste em fazer o LED instalado na porta digital 9 do Arduino, piscar 10 vezes com intervalos de tempo definido (meio segundo).

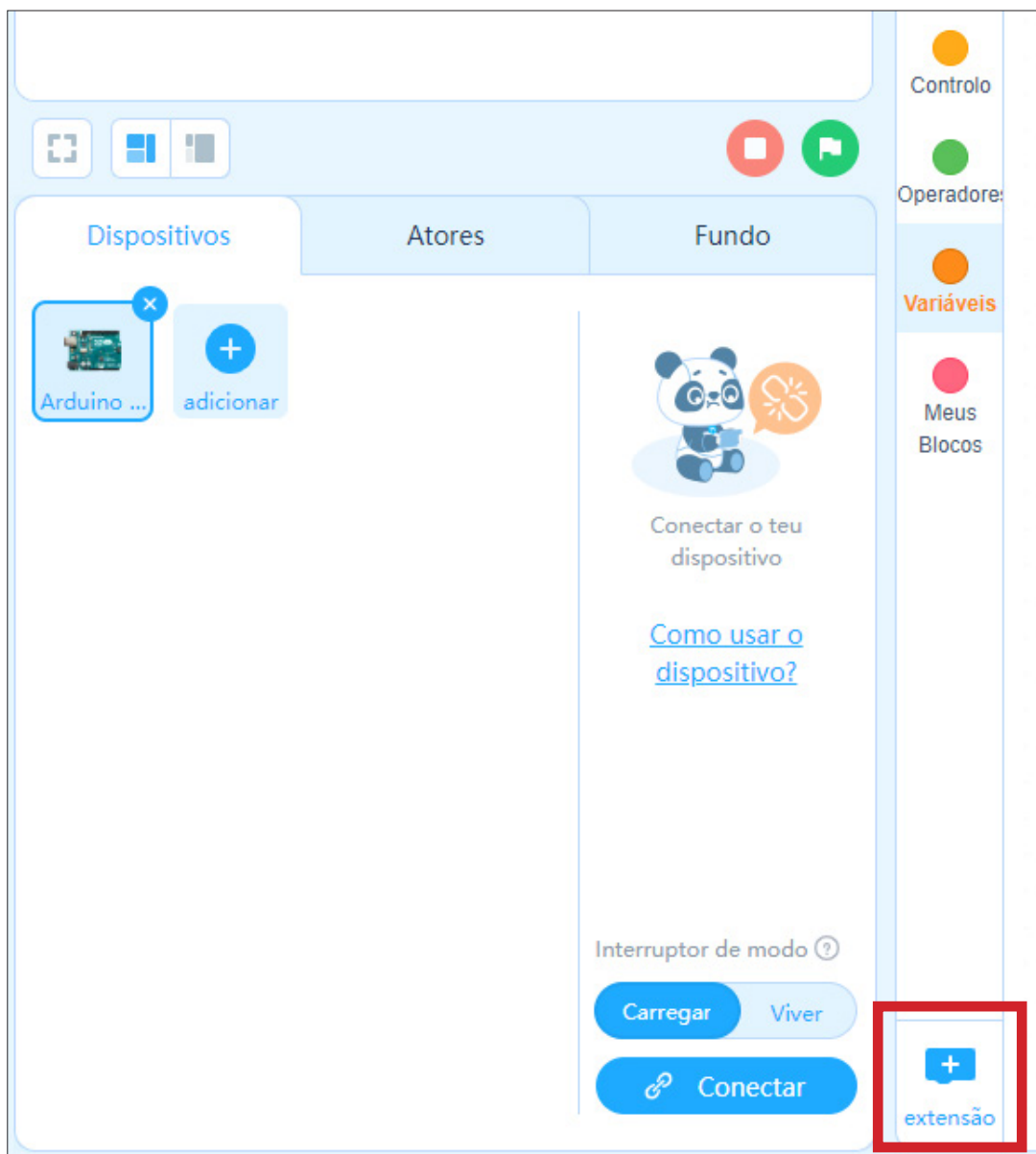
2.2.3 Extensão

Assim como no Arduino IDE, o mBlock também permite trabalhar com o uso de bibliotecas, nele é chamado de **Extensão**.

Extensões podem ser baixadas pelo Software mBlock tanto na versão on-line quanto na off-line, seguindo o mesmo procedimento detalhado abaixo.

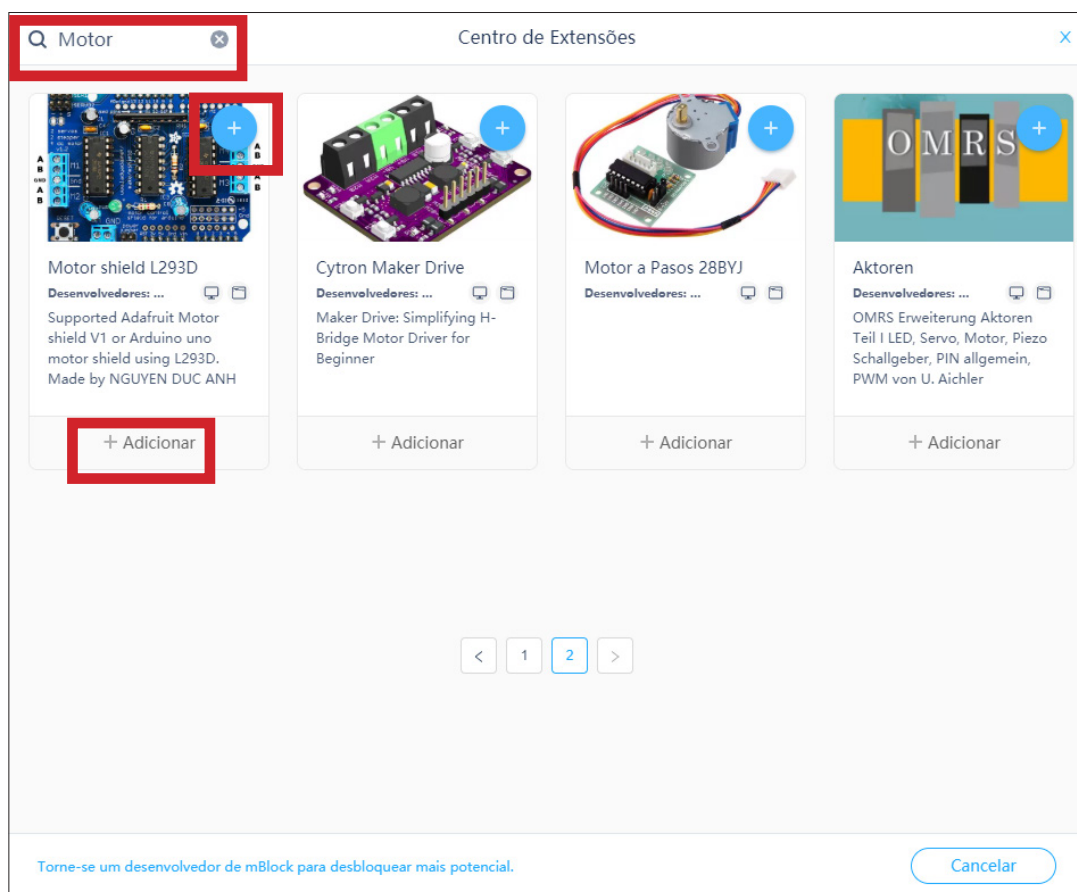
Para baixar uma extensão, você deverá clicar sobre o botão **adicionar extensão**, localizado na parte inferior da barra lateral, figura 23.

Figura 23 - Adicionar uma Extensão ao mBlock



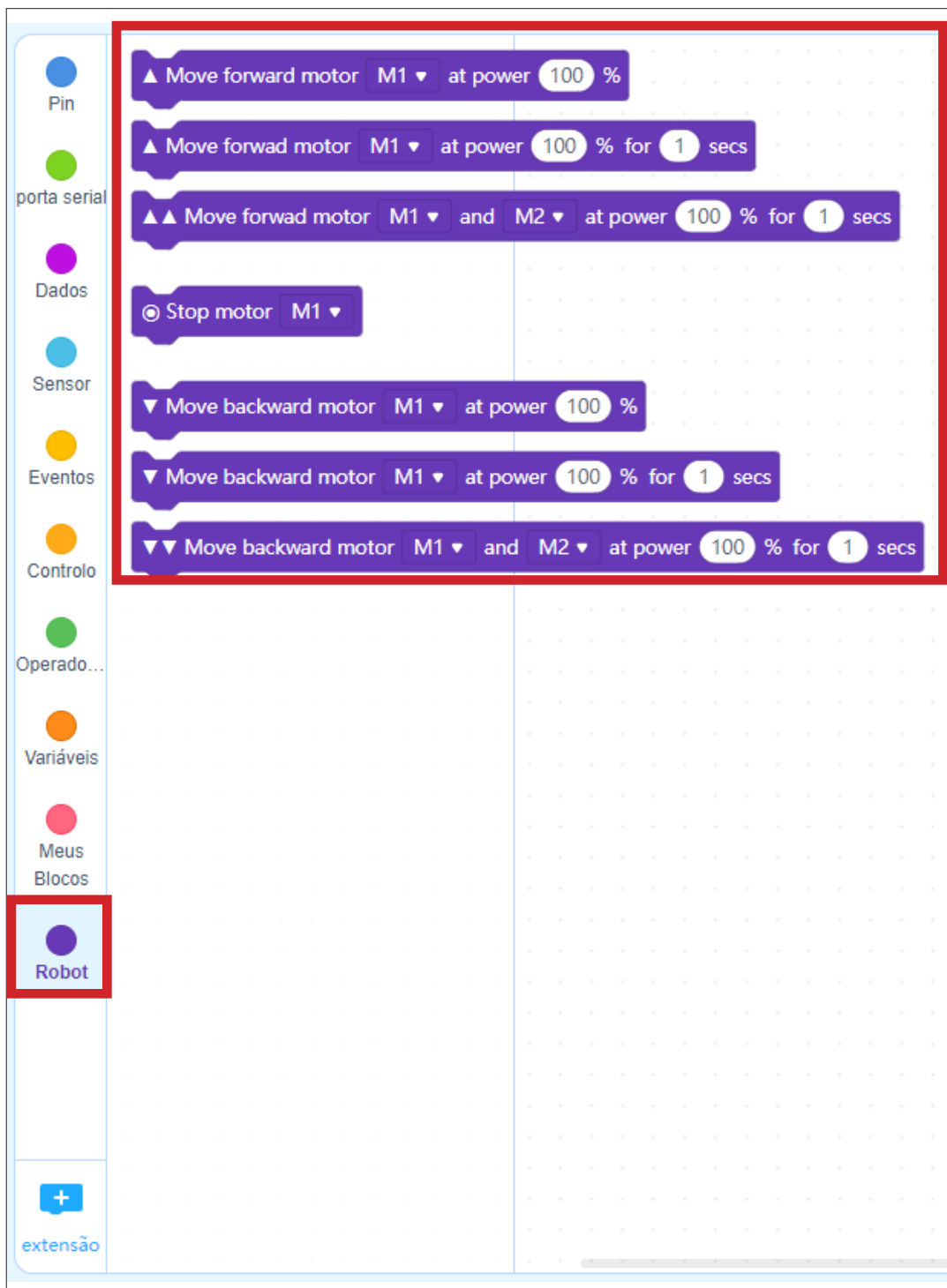
Abrirá o Centro de Extensões onde você poderá pesquisar pela extensão desejada. Quando uma extensão ainda não estiver baixada, estará indicado por um ícone com o símbolo “+” no seu canto superior direito, clique sobre ele para baixá-la e, então, será habilitado o botão “**Adicionar**”, localizado abaixo da extensão, figura 24.

Figura 24 - Centro de Extensão do mBlock



Após adicionar a extensão, surgirá uma nova aba no painel lateral, referente aos comandos da extensão que você adicionou, como indicado no exemplo da figura 25, que poderão ser usados em sua programação.

Figura 25 - Nova aba da extensão adicionada

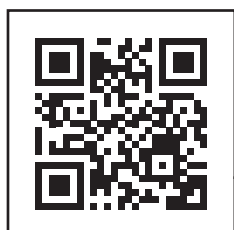


Assim que os blocos estiverem montados, clique no botão **Conectar**, figura 26, para transferir os comandos criados no mBlock para o Arduino Uno. Ao clicar neste botão, abrirá um Tooltip solicitando a confirmação da conexão entre os dois dispositivos, conforme mostra a figura 26. Após a transferência dos dados, as funções programadas serão executadas.

Figura 26 - Confirmar conexão entre mBlock e Arduino Uno



Outra forma de você utilizar o software mBlock é na versão on-line, disponível no link <https://ide.mblock.cc/>. Ao acessar a versão on-line, você terá acesso aos blocos de programação, com os quais poderá arrastar e soltar os comandos, conforme as funções desejadas.



<https://ide.mblock.cc/>

Uma vez finalizada a programação em blocos, clique sobre o botão **Conectar** para que seja feita a conexão do mBlock com a placa de Arduino Uno. Ao clicar sobre este botão, aparecerá um *Tooltip* solicitando que sejam instalados e ativados o driver e o controlador de dispositivo mLink para o funcionamento do mBlock Web, figura 27, permitindo, assim, a conexão entre os dois dispositivos.

Figura 27 - Instalar dispositivo mBlock Web



Fonte: site mBlock oficial

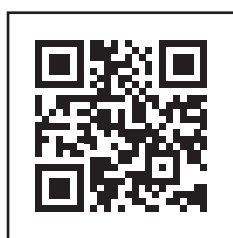
Executada a transferência do código para o dispositivo Arduino Uno, inicia-se o funcionamento das ações programadas.

Com a utilização destes dois softwares, poderemos programar vários comandos para que sejam executados nas diferentes atividades propostas ao longo deste módulo de robótica.

2.3 Simulador Tinkercad

O Tinkercad é uma plataforma on-line gratuita de modelagem em 3D, a qual permite a simulação de circuitos elétricos e programação de seu funcionamento.

Para acessar a plataforma Tinkercad, digite na barra de navegação o link: <https://www.tinkercad.com/>. Você será direcionado à página inicial da plataforma, figura 28. Nesta página, você poderá se inscrever na plataforma ou simplesmente acessá-la através de seu login e senha de e-mail do domínio Google conta **@escola.pr.gov.br**. Para esta forma de acesso, basta clicar no botão **Entrar**, conforme mostra a figura 28.



<https://www.tinkercad.com/>

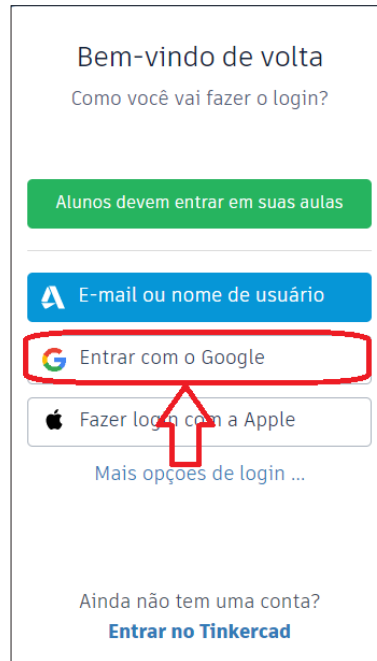
Figura 28 - Layout do simulador Tinkercad



Fonte: site Tinkercad oficial.

Na tela de login, clique na opção: **Entrar com o Google**, figura 29.

Figura 29 – Tela de login do Tinkercad



Digite o seu e-mail do domínio Google conta @**escola.pr.gov.br** e clique no botão **Próxima**, figura 30.

Figura 30 – Inserindo e-mail de domínio Google @escola



Na página que se abre, digite a senha do seu e-mail do domínio Google conta **@escola.pr.gov.br** e clique no botão **Próxima**, como mostra a figura 31.

Figura 31 – Inserindo senha do e-mail @escola



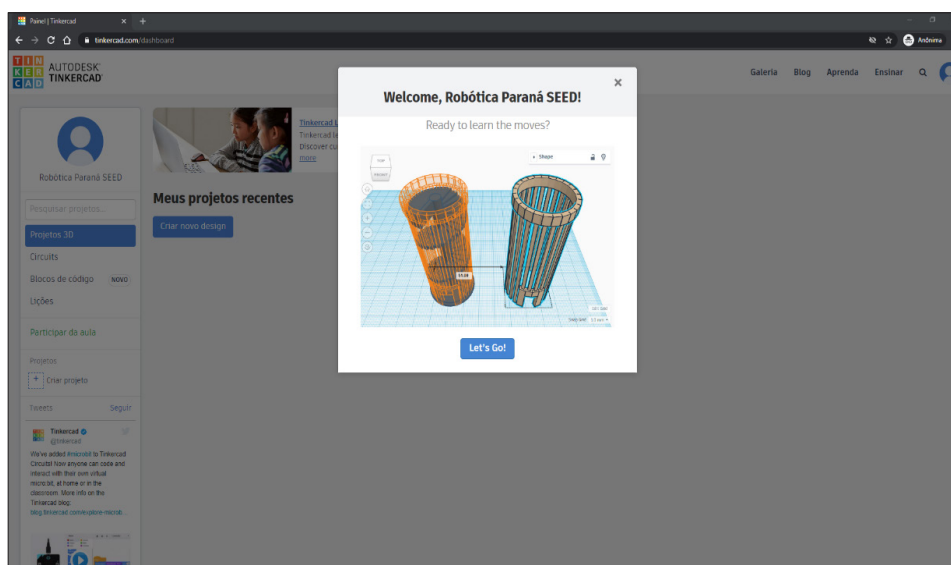
Finalize seu login, clicando no botão **Continuar**, figura 32.

Figura 32 – Finalizando login na plataforma Tinkercad



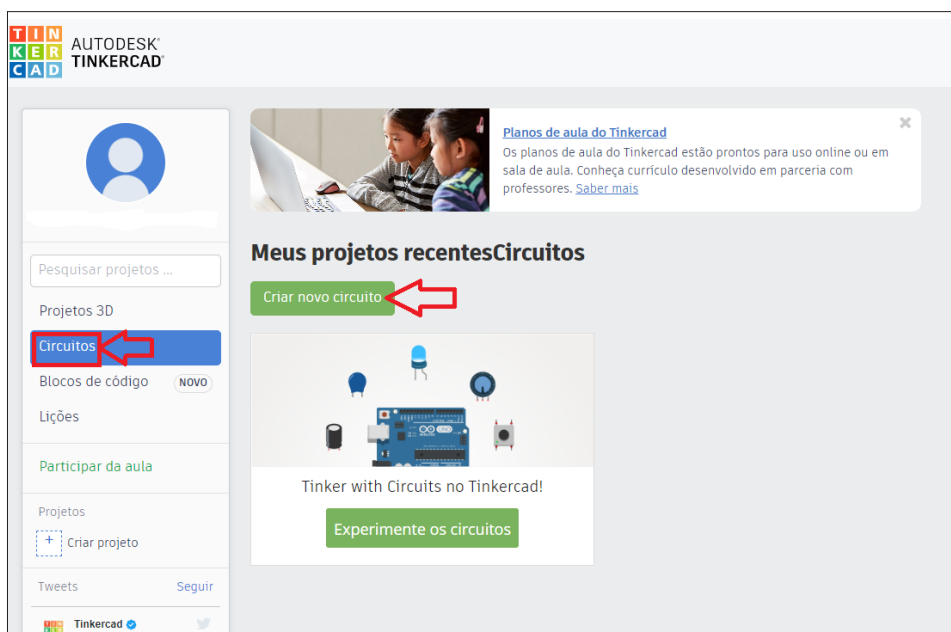
Ao acessar a plataforma, pode ser que apareça algum tutorial explicando como utilizá-la. Você tem a opção de segui-lo, clicando no botão **Let's Go!** ou optar em fechá-lo, clicando no **X**, localizado no canto superior direito do tutorial, figura 33.

Figura 33 - Página inicial da plataforma Tinkercad



No menu lateral, clique em **Circuitos** e em **Criar novo circuito**, como mostra a figura 34. Você será direcionado à página que permite a criação de circuitos elétricos.

Figura 34 - Acessando campo para criação de novo circuito



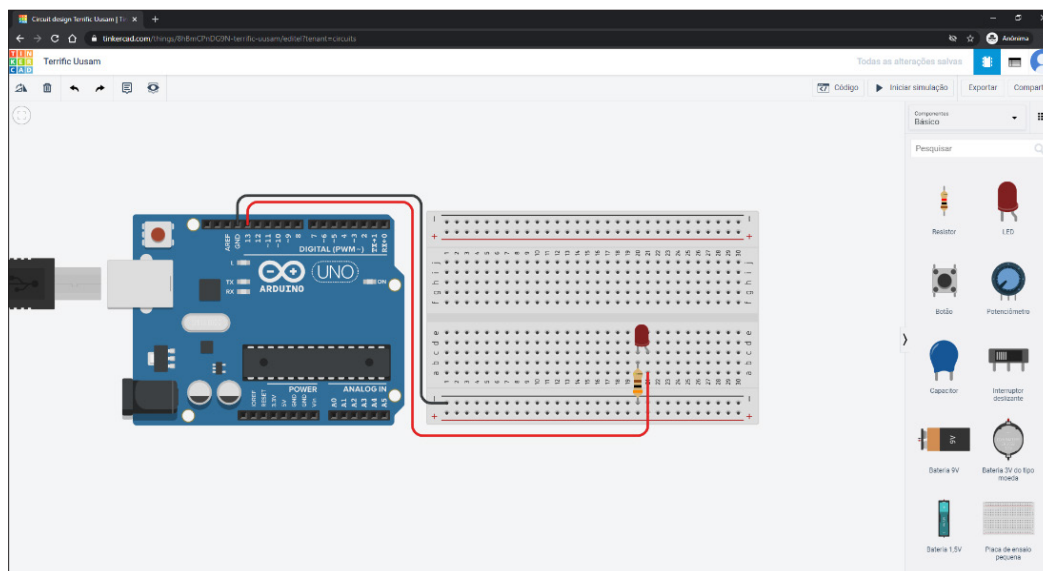
Para montar algum experimento, basta clicar e arrastar os componentes eletrônicos, presentes no menu lateral direito da página, para o centro da tela, figura 35.

Figura 35 - Selecionando e arrastando componentes eletrônicos



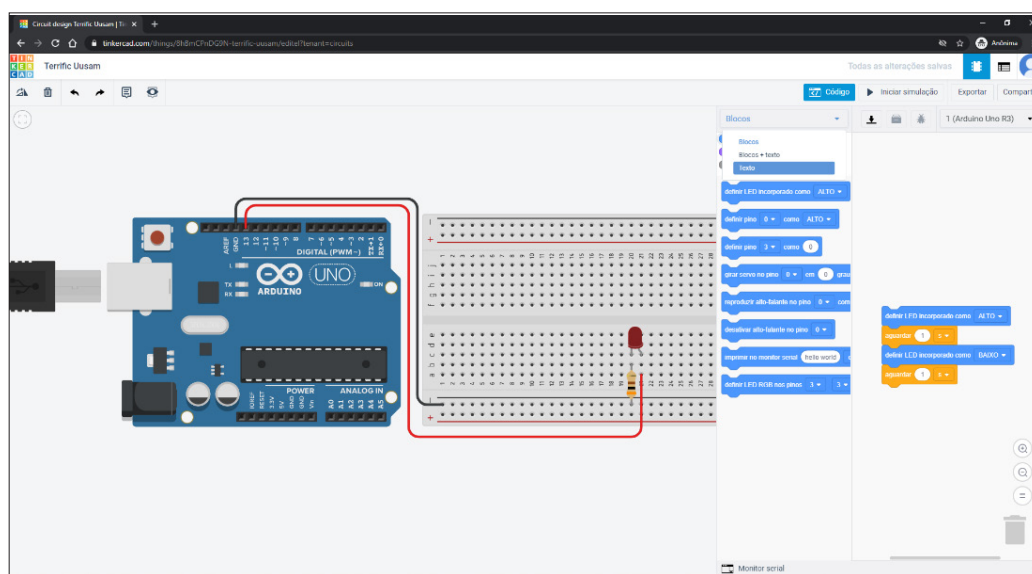
Uma vez seleccionados os componentes eletrônicos que participarão do circuito elétrico, faça as ligações desses com a placa do Arduino, figura 36.

Figura 36 - Ligação dos componentes eletrônicos com a placa Arduino



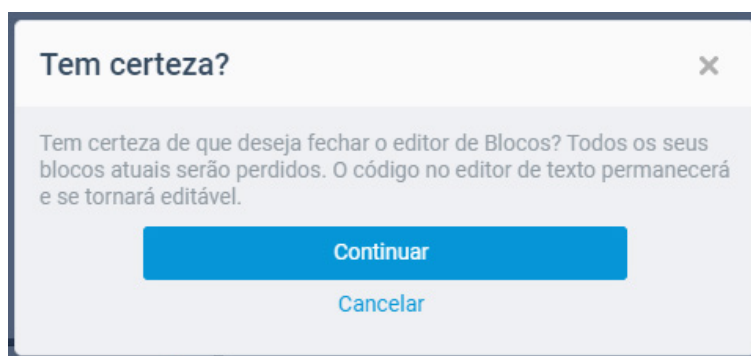
Para programar o funcionamento do circuito, clique no botão **Código** e, logo na caixa abaixo, escolha a programação que deseja fazer, entre as opções: **Blocos**, **Blocos + texto** ou **Texto**. Recomendamos a opção **Texto** pela semelhança com a programação realizada no software Arduino IDE, figura 37.

Figura 37 - Programando o funcionamento do circuito elétrico



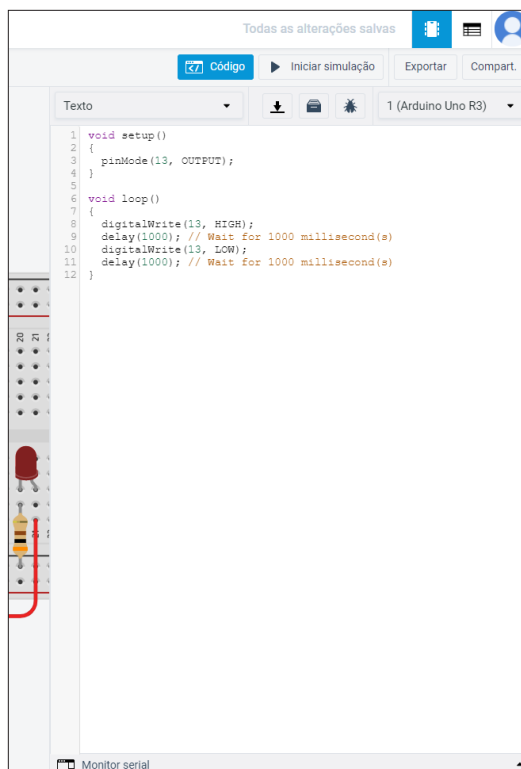
Ao selecionar a opção **Texto**, aparecerá uma mensagem avisando que ele fechará os blocos e abrirá o editor de texto. Clique no botão **Continuar** para confirmar a mensagem, figura 38.

Figura 38 – Confirmando a programação por texto



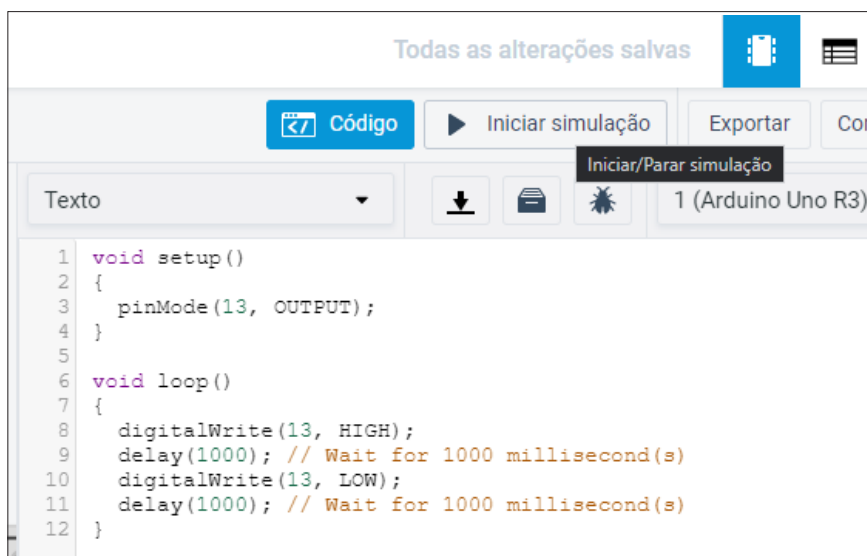
Após confirmar a programação por texto, você terá acesso ao editor e poderá colocar o seu código do Arduino, figura 39.

Figura 39 – Inserindo programação por código



Uma vez inserido o código de programação, você poderá simular o funcionamento do circuito para verificar se o código está correto. Para tanto, clique no botão **Iniciar/Parar simulação**, como mostra a figura 40.

Figura 40 – Iniciar simulação para verificação do código de programação



A simulação pode ser interrompida a qualquer momento, basta clicar no botão **Parar simulação**, conforme demonstra a figura 41. Durante a simulação não é permitido alterar a montagem dos componentes. Caso precise modificar a disposição dos componentes no circuito, interrompa a simulação.

Figura 41 – Encerrando o processo de simulação



Fonte: site Tinkercad oficial

3 Feedback e Finalização (15min):

a. Você e seus colegas se adaptaram com os ambientes de programação e já imaginam as possibilidades de utilização? Compartilhem suas impressões e as novas ideias.

b. Reúna todos os componentes utilizados nesta aula e os organize novamente, junto aos demais, no kit de robótica.



Referências

ARDUINO. Site oficial. **Ambiente de Programação do Arduino**. Disponível em: <https://create.arduino.cc/editor>. Acesso em: 15 out. 2021.

ARDUINO. Site oficial. **Downloads**. Disponível em: www.arduino.cc/en/Main/Software. Acesso em: 15 out. 2021.

BRASIL. Ministério da Educação. **Base Nacional Comum Curricular**. Brasília, 2018. Disponível em: http://basenacionalcomum.mec.gov.br/images/BNCC_EI_EF_110518_versaofinal_site.pdf. Acesso em: 18 out. 2021.

MAKEBLOCK. mBlock. **Download mBlock**. Disponível em: <https://mblock.makeblock.com/en-us/download/>. Acesso em: 15 out. 2021.

MAKEBLOCK. mBlock. **Programação em blocos**. Disponível em: <https://ide.mblock.cc/>. Acesso em: 15 out. 2021.

TINKERCAD. Autodesk. **Da mente ao projeto em minutos**. Disponível em: <https://www.tinkercad.com/>. Acesso em: 23 set. 2021.

DIRETORIA DE TECNOLOGIAS E INOVAÇÃO (DTI)
COORDENAÇÃO DE TECNOLOGIAS EDUCACIONAIS (CTE)

EQUIPE ROBÓTICA PARANÁ

Adilson Carlos Batista
Cleiton Rosa
Darice Alessandra Deckmann Zanardini
Edna do Rocio Becker
Marcelo Gasparin
Michelle dos Santos
Ricardo Hasper
Simone Sinara de Souza

Os materiais, aulas e projetos da “Robótica Paraná”, foram produzidos pela Coordenação de Tecnologias Educacionais (CTE), da Diretoria de Tecnologia e Inovação (DTI), da Secretaria de Estado da Educação e do Esporte do Paraná (Seed), com o objetivo de subsidiar as práticas docentes com os estudantes por meio da Robótica.

Este material foi produzido para uso didático-pedagógico exclusivo em sala de aula.



Este trabalho está licenciado com uma Licença
Creative Commons – CC BY-NC-SA
[Atribuição - NãoComercial - Compartilha Igual 4.0](https://creativecommons.org/licenses/by-nc-sa/4.0/)

