

UNIVERSIDADE CATÓLICA DE PERNAMBUCO

Hemilly Monara Rodrigues da Silva
João Vitor Lima da Silva
Karolayne Teixeira da Silva
Lucas Gabriel Dias da Luz
Samara Regina Rodrigues da Silva

Hikas

O Jogo Musical Programável

Recife/2020

Sumário:

- Introdução
- A Importância da Educação Musical
- Público-Alvo
- O Projeto
- Como Funciona?
- Tecnologias e Diferenciais
- Aplicativo Hikas
- Protótipo da Versão Completa
- Como acessar o projeto completo?

INTRODUÇÃO

Hikas é um jogo musical que estimula de diversas formas o aprendizado musical e sonoro. Com seus modos diversificados e interativos, Hikas procura se manter divertido e não monótono juntando em um só lugar várias táticas de ensino e jogo musical.

Com seu estilo dinâmico surge a possibilidade do contínuo aprendizado em diversos âmbitos. Nele, além de se aprender a tocar uma música, há a estimulação da memória, da criatividade e da precisão. Além de outros benefícios que vem junto com a educação musical básica.

A IMPORTÂNCIA DA EDUCAÇÃO MUSICAL

- A música ajuda no desenvolvimento neurológico da criança

Uma pesquisa realizada pela Universidade de Vermont nos EUA, constatou depois de analisar tomografias de 232 crianças entre 6 e 18 anos, que as crianças que estudavam música desde a infância apresentavam ganhos positivos e duradouros em relação ao desenvolvimento cerebral.

- O aprendizado musical modifica fisicamente o cérebro, principalmente na primeira infância (0 a 6 anos) e os ganhos se mantêm por toda a vida.

Com os estudos avançados da Neurociência, hoje temos informações seguras que o período entre 0 e 6 anos é de extrema importância para o desenvolvimento infantil. É a fase de estruturação do Sistema Nervoso, onde comportamentos são aprendidos e consolidados.

- Todos podem desenvolver a inteligência musical.

Howard Gardner, professor da Faculdade de Harvard, ganhador de diversos prêmios, afirma que o que leva as pessoas a desenvolverem a inteligência (inclusive a musical) é: O ambiente que vivem, experiências significativas que proporcionem a vontade de praticar, a educação e as oportunidades que surgem ao longo da vida.

- A música é interdisciplinar.

Desde uma brincadeira de roda, ou tocar um instrumento musical, nas aulas de música, o cérebro é desafiado a utilizar diversas coordenações ao mesmo tempo (cantar/dançar, cantar/tocar/dançar, dançar/tocar, etc.) Essas informações são assimiladas

pelo cérebro através dos sentidos, colaborando de forma significativa nas aquisições linguísticas, sociais e cognitivas, lembrando que todos os aspectos do desenvolvimento (motor, intelectual e afetivo) estão interligados e um exerce influência sobre o outro.

- Praticar música ensina a importância da disciplina.

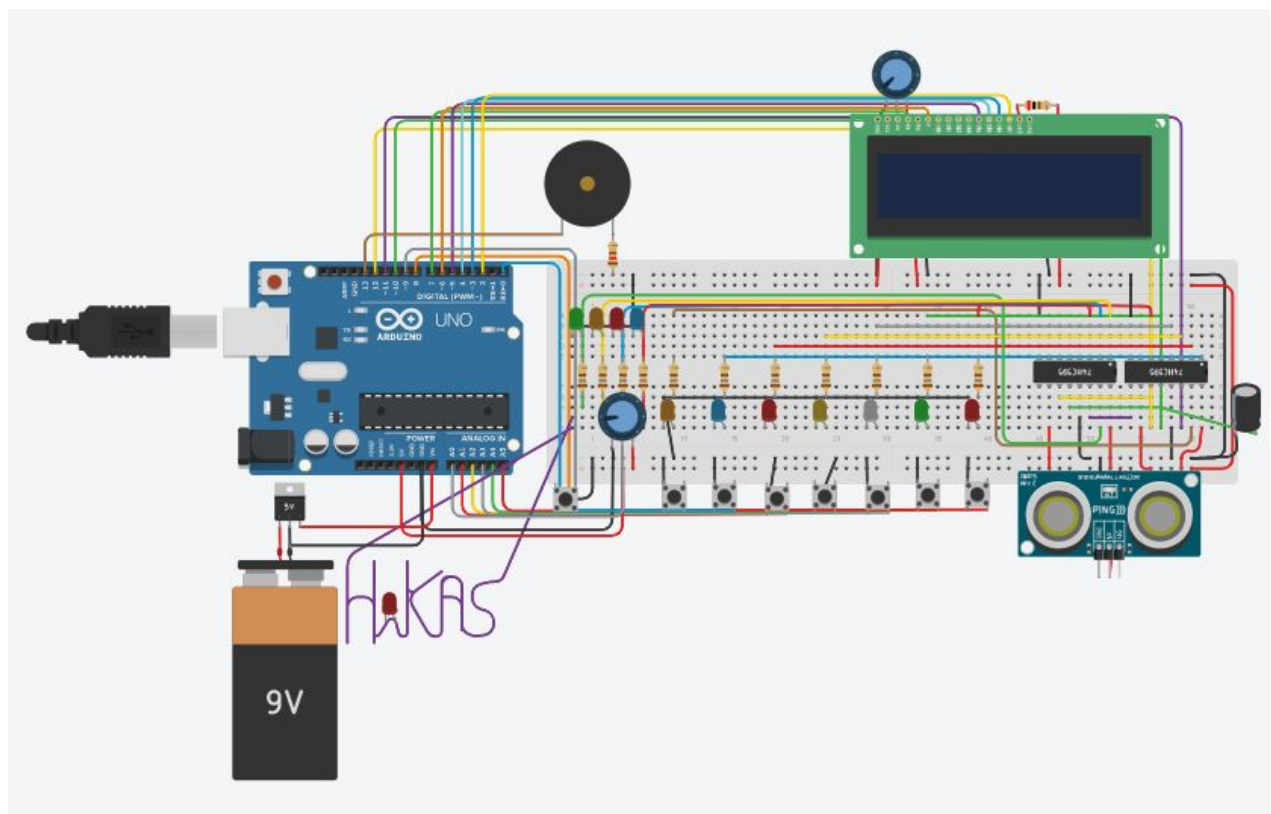
A música é uma linguagem matemática, mas ao mesmo tempo afetiva sempre inserida em um contexto social. Por ser tão poderosa e influente, através dela podemos aprender qualquer conteúdo, pois é uma linguagem que traz a motivação necessária para aprender.

PÚBLICO-ALVO

A partir dos dados coletados é possível concluir que atualmente o público-alvo do nosso jogo educacional é na grande maioria crianças. Porém é caracterizada como uma área versátil e abrangente, assim pode atingir os mais variados públicos. Apresentamos um jogo educacional que engloba mais de uma forma de ensinar música de uma maneira divertida e não monótona. Fazendo com que seu jogador tenha acesso ao jogo físico mas ao mesmo tempo tenha elementos de jogos digitais.

O PROJETO

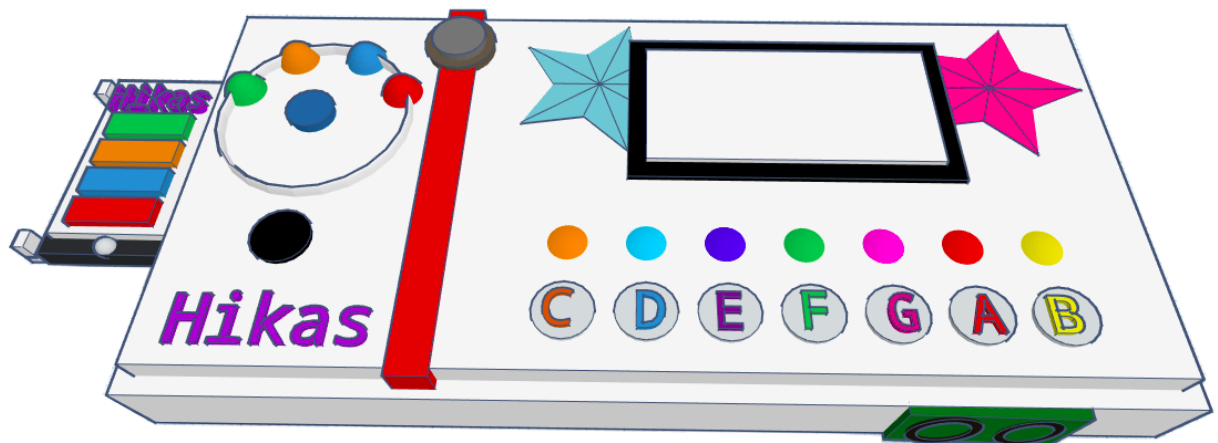
O Hikas foi desenvolvido inteiramente online, através plataforma Tinkercad. E como tal plataforma possui algumas limitações, o Hikas ganhou mais uma versão, sendo esta desenvolvida no Fritzing de forma mais avançada. Essa versão mais completa será descrita posteriormente. Abaixo o circuito do Hikas, desenvolvido no Tinkercad (Clique na imagem para acessar o projeto).



Utilizamos 7 botões que representam as notas: Dó, Ré, Mi, Fá, Sol, Lá e Si. O intuito foi de simular um mini teclado capaz de tocar as notas básicas e de modificar suas escalas a partir da manipulação do potenciômetro mais a esquerda.

Modelo 3D

Com a finalidade de simular o jogo na vida real, foi desenvolvido, também no Tinkercad, o modelo 3D abaixo (Acesse o modelo clicando na imagem):



O modelo foi feito de maneira a ficar mais compacto e de fácil utilização, com cores vibrantes e letras para identificar cada nota disposta em sua base. A partir do modelo, é possível identificar onde os materiais e componentes utilizados ficariam dispostos.

Materiais Utilizados e Preço Aproximado

Quantidade	Nome	Preço (Aprox.)
1	Arduino Uno R3	59,90
1	LCD 16x2	20,00
8	Botões	1,92
2	Potenciômetros	3,00
11	LEDs	2,75
12	Resistores 180 Ω	1,80
1	Resistor 200 Ω	0,08
1	Alto-falante	8,00
1	Bateria 9v	7,50
1	Conversor de 5v	4,20
2	Shift Register (74HC595)	2,60
1	Sensor Ultrassônico	11,00
1	EEPROM	10,00
1	Capacitor	0,50
Total		132,80

COMO FUNCIONA?

Hikas busca ser dinâmico e divertido e para isso possui quatro modos base, são eles:

1 - **Modo Livre:** Nesse modo, pode-se tocar livremente as notas alterando suas escalas entre grave e agudo o quando quiser.

2 - **Modo Genius:** Inspirado no clássico jogo Genius dos anos 80. O modo genius tem diferenças importantes. Nele, ao invés de quatro, tem-se sete teclas que simulam um mini teclado. Assim, é possível selecionar a música e a velocidade desejada e o jogo repetirá a música estimulando a memória auditiva do jogador. Ao contrário do que ocorre no genius o qual as notas são dadas aleatoriamente pelo controlador.

3 - **Modo Scroll:** Como o próprio nome diz, o modo Scroll se baseia no famoso estilo de jogo musical, o qual as notas chegam, uma atrás da outra, como é o caso do jogo Guitar Hero. Para vencer esse modo é necessário precisão e velocidade.

3 - **Modo Programação:** Neste modo tem-se duas opções:

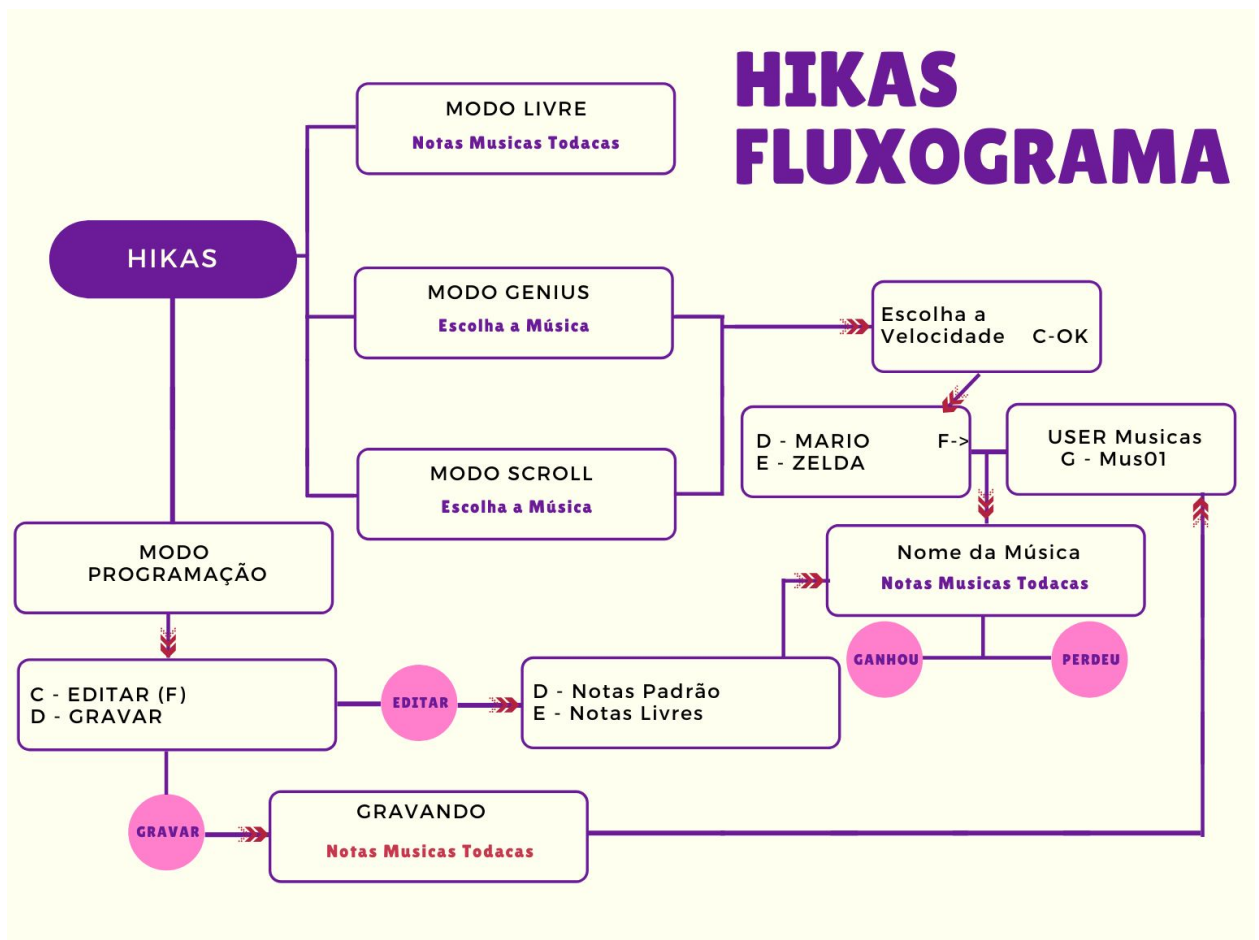
3.1 - **Edição:** Nele pode-se liberar as notas do jogo genius e Scroll para serem livres, ou seja, não mais se prenderem as notas padrões (com frequências predefinidas);

3.2 - **Gravar:** Aqui é possível gravar sua música na memória do Hikas para que posteriormente possa ser jogada no modo Genius ou Scroll. Na hora da gravação também é possível modificar as escalas musicais, dando assim mais opções e mais vida a música.

Para ver os modos em funcionamento acesse nosso video no YouTube:



A seguir um fluxograma ilustrativo que descreve o comportamento desses modos no jogo:



O Manual do Jogo

Como todo jogo que demanda mais conhecimento prévio, Hikas possui um mini manual que acompanha o jogo, explicando de forma simples e resumida seus processos:

<div> <div>Hikas Manual</div> <div> <p>Hikas é um jogo musical que estimula de diversas formas o aprendizado musical e sonoro. Com seus modos diversificados e interativos, Hikas procura se manter divertido e não monótono juntando em um só lugar várias táticas de ensino e jogo musical.</p> </div> </div>				
	LIVRE	GENIUS	SCROLL	PROGRAMAÇÃO
ACESSO	Um click no botão principal	Dois clicks no botão principal	Três clicks no botão principal	Quatro clicks no botão principal
FUNCIONAMENTO	Toque sem restrições	Baseado no jogo Genius - repita a sequência musical sem errar	Acerte as notas a medida que elas chegam	Grave sua música para jogá-la nos outros modos
REGULAÇÃO	Mude o som entre grave e agudo o quanto quiser	Mude a velocidade do jogo e tenha maiores desafios	Aumente a velocidade das notas e se supere	Mude o som entre grave e agudo e grave sua música
APRENDIZADO	Aprenda as notas e seus sons	Aprenda a tocar uma música básica a partir da memorização	Aprenda a tocar com velocidade e precisão	Estimule sua criatividade criando sua própria música

<div> <div>Músicas Base</div> <div> <p>Teste no modo livre, modifique através de sua criatividade e se divirta gravando e jogando sua música nos outros modos.</p> </div> </div>	
O REI LEÃO HAKUNA MATATA	SOL LA SI LA SI DO SI LA SOL LA SI LA SOL SI LA RE SI LA SI RE DO SI LA SOL LA SI LA SOL SI LA SOL RE RE DO SI SI SOL SOL SOL
STAR WARS MARCHA IMPERIAL	SI SI SI SOL RE SI SOL RE SI FA FA FA SOL RE LA SOL RE SI SI SI SI LA LA SOL SOL SOL DO FA MI RE RE DO RE SOL LA SOL LA RE SI RE FA SI SI SI SI LA LA SOL SOL SOL DO FA MI RE RE DO RE SOL LA SOL RE SI SOL RE SI
PARABENS PARA VOCÊ	SOL SOL LA SOL DO SI SOL SOL LA SOL RE DO DO MI SOL MI DO SI LA FA FA MI DO RE DO DO SOL DO DO DO SOL RE SI DO SOL DO MI SOL MI DO SOL RE SI DO
POKÉMON (ABERTURA)	LA LA LA LA LA SOL FA RE RE LA LA LA SOL FA SOL SIB SIB SIB SIB DO LA SOL FA FA LA LA SOL FA LA LA LA LA LA SOL FA RE RE LA LA SOL FA SOL SIB SIB SIB SIB DO LA SOL FA LA LA SOL FA LA REFRÃO: LA DO RE LA LA DO RE RE RE RE MI FA MI RE DO LA DO RE RE DO LA RE DO SOL FA FA LA LA SOL FA LA LA DO RE LA LA DO RE RE RE RE MI FA MI RE DO FA FA FA FA DO FA FA FA FA FA MI LA LA DO RE RE LA LA DO RE RE

Tecnologias e Diferenciais

Economia

O sensor de distância ultrassônico tem um papel fundamental dentro do jogo. O de economizar bateria se não há ninguém jogando o jogo por um determinado tempo. Para isso, utilizamos o modo Sleep.

O Modo Sleep é basicamente uma ferramenta que não está presente de forma nativa no arduino, todavia que pode ser implementada. Sua função é de desativar alguns componentes que estão sendo pouco usados, dessa forma economizando energia, geralmente desativa-se alguns dos clocks presentes no microcontrolador, tal recurso é de fundamental importância em componentes eletrônicos em geral para não só a redução do consumo de energia mais também para a maior duração da vida útil dos componentes caso necessário.

Essa funcionalidade pode reduzir em até 50% o consumo de energia de todo e qualquer aparelho que use um microcontrolador, como o caso do projeto apresentado do jogo HIKAS, onde o microcontrolador(arduino) reduz a quantidade de operações internas a cada intervalo de tempo, consumindo assim menos potência, o modo sleep é comumente chamado de Stand By, vejamos a tabela abaixo para melhor exemplificar como funciona tal tarefa:

Sleep Mode	Active Clock Domains					Oscillators		Wake-up Sources							Software BOD Disable
	clkCPU	clkFLASH	clkIO	clkADC	clkASY	Main Clock Source Enabled	Timer Oscillator Enabled	INT and PCINT	TWI Address Match	Timer2	SPM/EEPROM Ready	ADC	WDT	Other I/O	
Idle			Yes	Yes	Yes	Yes	Yes ⁽²⁾	Yes	Yes	Yes	Yes	Yes	Yes	Yes	
ADC Noise Reduction				Yes	Yes	Yes	Yes ⁽²⁾	Yes ⁽³⁾	Yes	Yes ⁽²⁾	Yes	Yes	Yes		
Power-down								Yes ⁽³⁾	Yes				Yes		Yes
Power-save					Yes		Yes ⁽²⁾	Yes ⁽³⁾	Yes	Yes			Yes		Yes
Standby ⁽¹⁾						Yes		Yes ⁽³⁾	Yes				Yes		Yes
Extended Standby					Yes ⁽²⁾	Yes	Yes ⁽²⁾	Yes ⁽³⁾	Yes	Yes			Yes		Yes

Como vemos o microcontrolador faz a desabilitação dos clocks, porém, pode surgir uma dúvida em como o arduino faz para ativá-los, para isso existe o modo wake-up sources.

Economia de energia: A economia de energia está totalmente ligada com o algoritmo e a forma como o circuito foi montado, além disso a utilização da plaquinha do Arduino também influencia, tendo em vista que os elementos dela geram um determinado gasto de energia. Por exemplo o regulador de tensão, que pode consumir cerca de 10 mA. Como utilizado no projeto

faremos a verificação e medição do gasto de corrente do Arduino UNO no modo normal e com o modo sleep ativo.

Observe que neste exemplo a efetividade do sleep mode no arduino é muito perceptível, uma vez que o consumo de energia cai drasticamente quando utilizado em modo Stand By. Podemos também fazer um paralelo no que diz respeito a jogos, a arquitetura dos componentes de um computador pode ser perfeitamente comparado com o nosso

Modo	Arduino UNO	Microcontrolador à parte
Normal	85,7mA	10,4mA
Sleep	58,8mA	0,32mA

exemplo, uma vez que, na programação de um game por exemplo, onde o foco do uso está mais centrado na GPU(placa de vídeo/memória de vídeo) por exemplo, o próprio jogo faz com que o uso do processador caia pela metade, conseqüentemente diminuindo a temperatura e as atribuições correlatas ao microcontrolador. Vejamos com um algoritmo simples como implementar o *modo sleep* e o *wake up sources*, lembrando que o arduino não possui essa biblioteca, tornando-se necessária a importação com o seguinte comando:

```
1  #define interrupcao 2
2  #define led 13
3  void setup()
4  {
5      // Define o pino 2 como entrada e ativa o resistor de pull-up
6      pinMode(interruptao, INPUT_PULLUP);
7      // Define o LED como saída
8      pinMode(led, OUTPUT);
9  }
```

Função Sleep(dormir):

```
1 void dormir(){
2     set_sleep_mode(SLEEP_MODE_PWR_DOWN);
3     sleep_enable();
4     attachInterrupt(digitalPinToInterrupt(interruptacao), acordar, LOW);
5     sleep_cpu();
6 }
```

Função Wake Up Sources:

```
1 void acordar(){
2     // Desabilita o sleep
3     sleep_disable();
4     // Desabilita a interrupção
5     detachInterrupt(0);
6 }
```

Memória e expansão

E EEPROM é uma forma de ampliar a memória do Arduino, onde para projetos um pouco maiores e uma boa escolha para utilizar. A memória disponível vai de acordo com a placa utilizada, no nosso caso um Arduino Uno. O EEPROM disponibiliza 1KB de memória, possui 1024 posições de memória para os dados de um byte (8 bits), ou seja, o maior valor que tem a capacidade de gravar e de 255, para o nosso projeto seria utilizado por causa das suas diversas funções, podendo armazenar dados de determinadas funções, valores de medição e determinados resultados de cálculos, a biblioteca vem com a função, EEPROM.write, onde permite que envie determinado dado a uma determinada posição, ou endereço.



Exemplo:

```
// Inclui a biblioteca dos comandos de EEPROM
#include <EEPROM.h>

void setup() {
  // Inicia a comunicação serial
  Serial.begin(9600);

  // Cria um delay para dar tempo da comunicação serial iniciar
  delay(1000);

  // Cria uma iteração de 0 até o valor limite de memória da EEPROM (no caso do UNO até 1024)
  for (int i = 0 ; i < EEPROM.length() ; i++) {
    // A cada iteração, o valor de i serve para definir o endereço de escrita, ou seja, os 1024 endereços são acessados

    // Antes de limpar o valor do byte, escreve o valor previamente armazenado
    Serial.println(EEPROM.read(i));

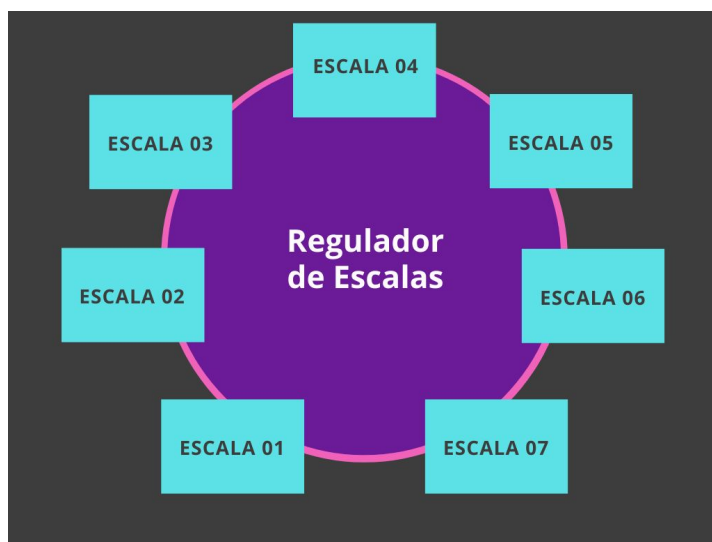
    // Para cada endereço 'i', coloca o valor do byte em 0
    EEPROM.write(i, 0);
  }
}

void loop() {
}
```

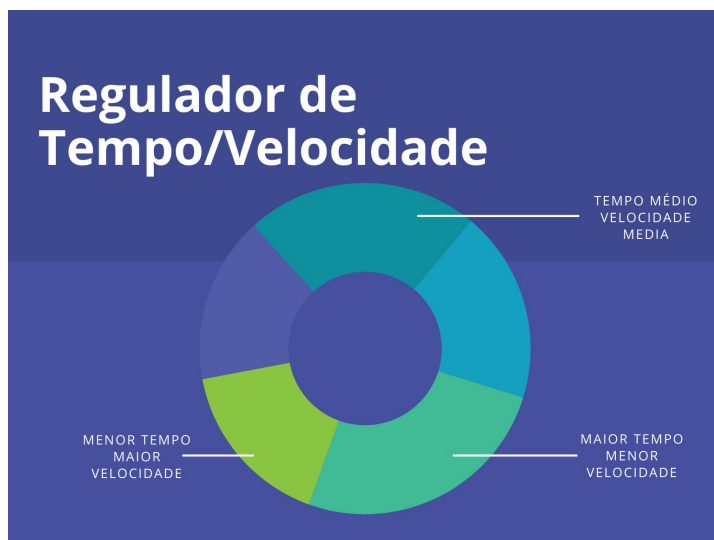
Regulação

O potenciômetro tem uma função de grande importância dentro do jogo, ele possibilita a modelagem do jogo de acordo com seu jogador.

Nos Modos Livre e Programação (Gravar): ele pode mudar a escala musical com um simples girar. Nele estão dispostas 7 escalas musicais (não atendendo as notas bemol e sustenido, por questões de espaço).



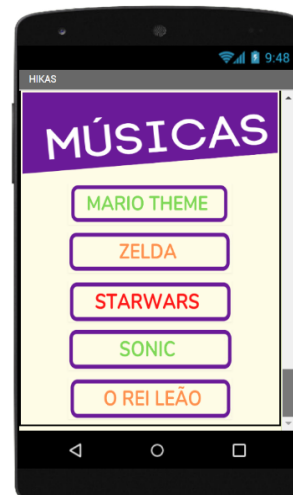
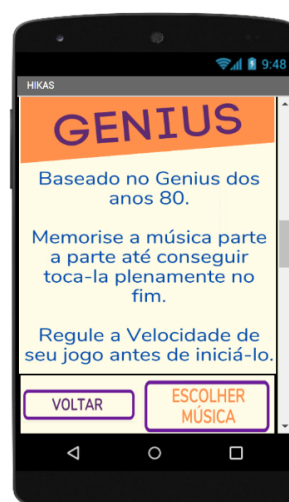
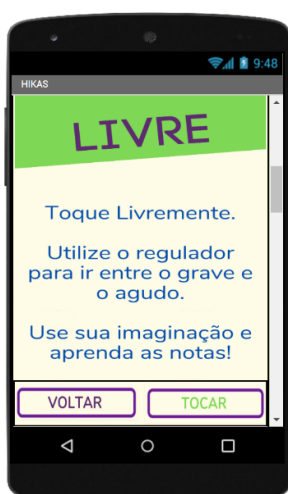
Nos modos Genius e Scroll: o potenciômetro regula o tempo entre as notas, alterando assim a velocidade do jogo.



Aplicativo Hikas

Os smartphones estão por toda parte hoje em dia, e pensando nisso e nas diversas possibilidades que o mesmo oferece, o aplicativo Hikas busca integrar o celular com o jogo físico através de uma conexão bluetooth. Assim, com o celular controlando as opções do jogo, o grau de acessibilidade aumenta consideravelmente e a expansão do jogo em si também.

O aplicativo foi desenvolvido através do site App Inventor 2, criado pelo MIT, e toda sua programação feita pelo mesmo site. Obviamente testes prévios não puderam ser feitos para comprovação total de seu funcionamento. Todo o código pode ser encontrado juntamente com todo material do projeto hospedado no GitHub. A seguir as telas:



Protótipo da versão completa

Como falado anteriormente, o Tinkercad oferece muitas limitações, porém a simulação e comprovação do funcionamento pode ser feita por ele. Pensando na expansão do Hikas, um protótipo de sua versão completa foi feito pelo Fritzing.

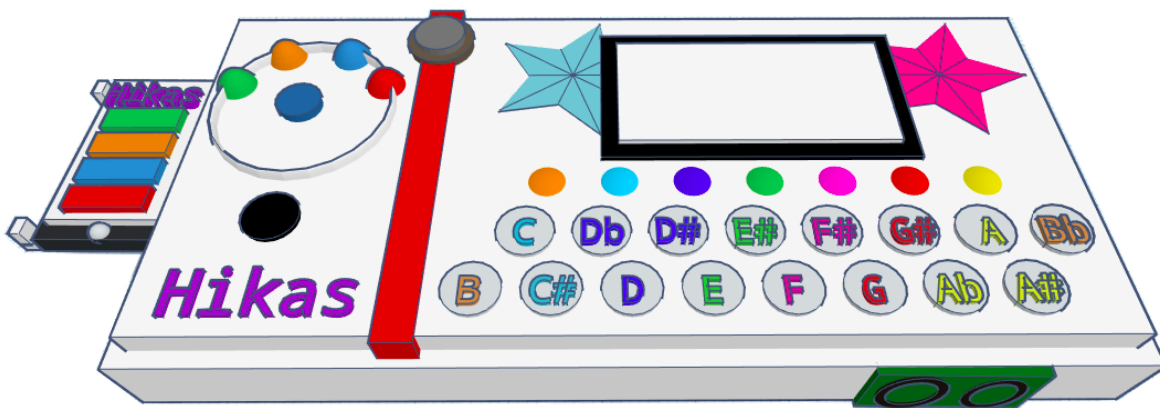
Essa versão oferece suporte ao App, com o bluetooth e uma gama maior de teclas podendo, agora sim, simular verdadeiramente um teclado musical.

Com essa expansão o Hikas consegue atingir 96 notas de um teclado musical com apenas 16 teclas e um regulador.

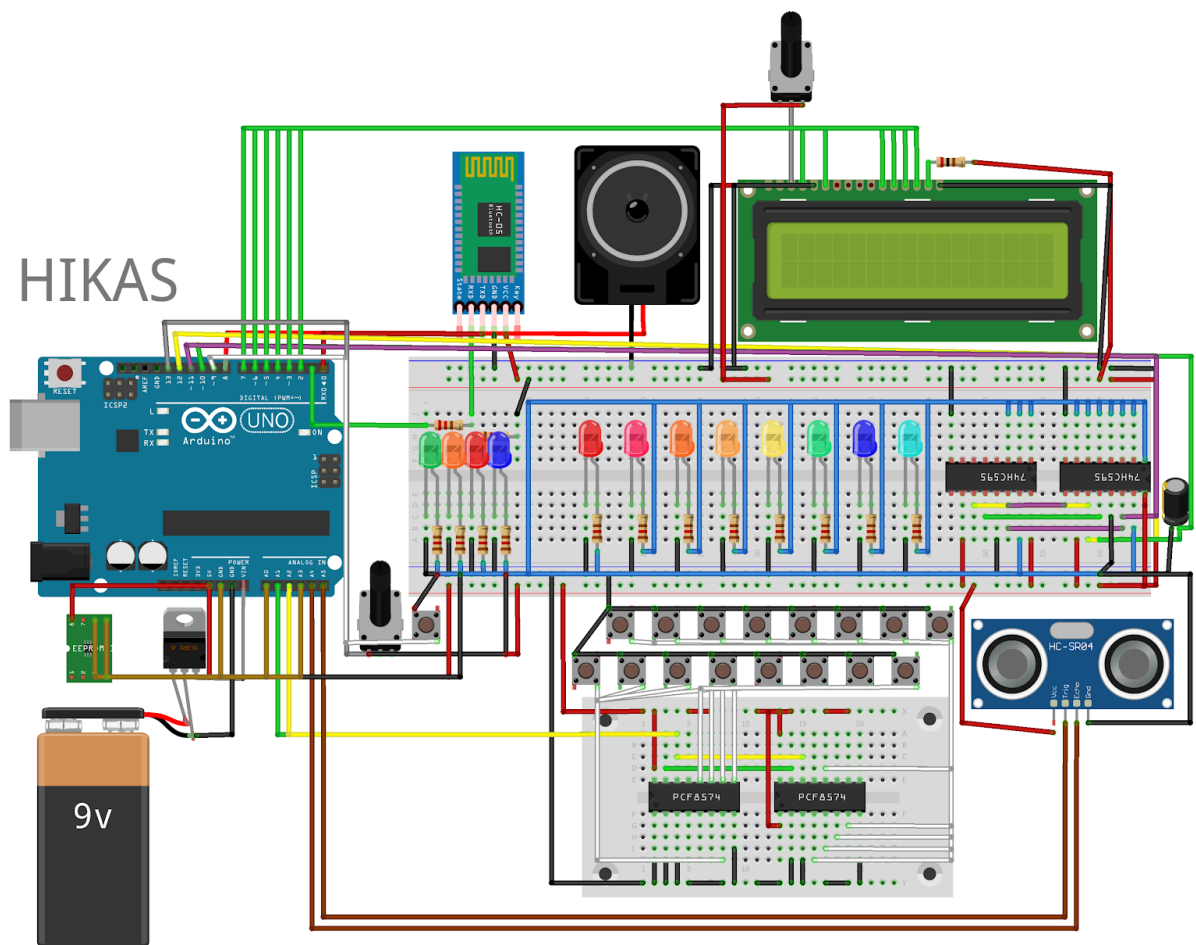
Teclado e frequências utilizados como base



Modelo 3D



Circuito (Fritzing)



fritzing

Com o acréscimo de dois PCF8574, módulo bluetooth e 9 botões, o preço aproximado do Hikás versão completa é de R\$150,36. Considerando o preço unitário de cada peça.

Como acessar o projeto completo?

Criamos um projeto no GitHub e lá se tem acesso a todos os documentos, desde o código até as imagens do projeto completo. Acesse clicando na imagem a seguir.

