

TAGARELLA BRAILLE – APP PARA AUXÍLIO NO APRENDIZADO AO BRAILLE

Aluno(a): Leonardo Pereira Vieira

Orientador: Dalton Solano dos Reis

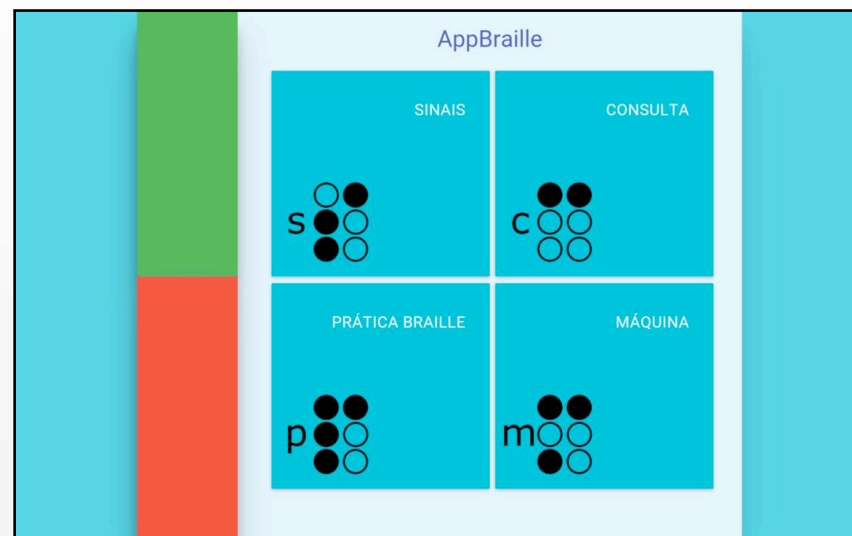
Roteiro

- Introdução
- Objetivos
- Fundamentação Teórica
- Trabalhos Correlatos
- Requisitos
- Especificação
- Implementação
- Análise dos Resultados
- Conclusões e Sugestões

Introdução

Projeto atual

- O Tagarela Braille, criado em 2016 por Lucas Cazagrande.
- A ideia inicial era criar um aplicativo para auxiliar as pessoas sem deficiência visual a aprender braille



Introdução

Importância do aplicativo

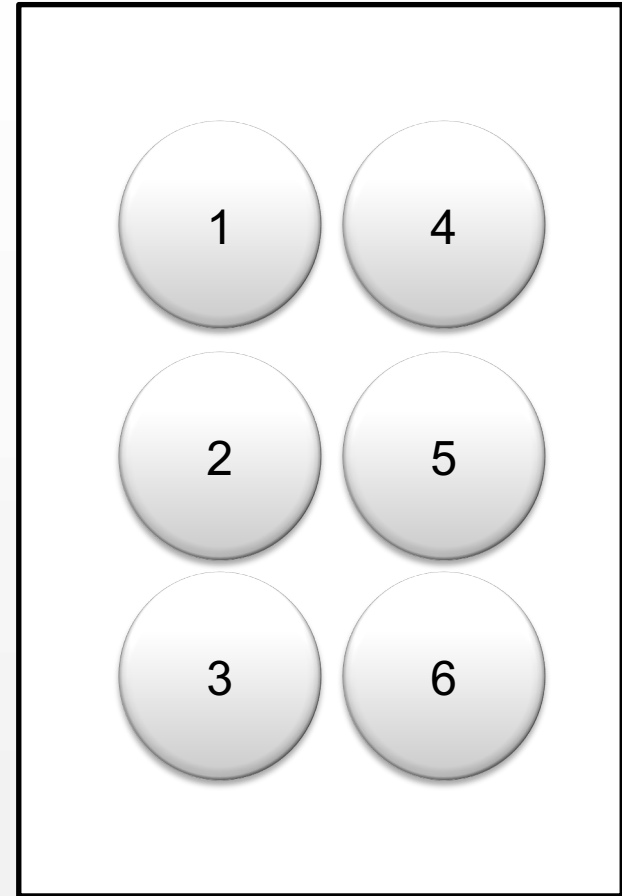
- Este trabalho é importante porque no Brasil cerca de 24% da população possui algum tipo de deficiência (IBGE 2017).
- Precisamos de mais ferramentas de inclusão.
- Quanto mais ferramentas pra estimular o aprendizado melhor.

Objetivos

- Criar um aplicativo para auxiliar o aprendizado do braille
 - ✓ Migrar aplicativo de IONIC para Flutter.
- Tornar o aplicativo acessível
 - ✓ Através de áudio.
 - ✓ Interação por gestos na tela.
- Utilizar componentização
 - ✓ Criar estrutura reaproveitável.
 - ✓ Componentes que possam ser usados em outros projetos ou Módulos.

Fundamentação Teórica

- Braille
- O braille é um sistema de escrita e leitura tátil para as pessoas cegas e foi criado por Louis Braille (INSTITUTO BENJAMIN CONSTANT, 2018) .
- O sistema braille baseados em símbolos de alto-relevo resultantes da combinação de até seis pontos (OTSUKA, 2010).



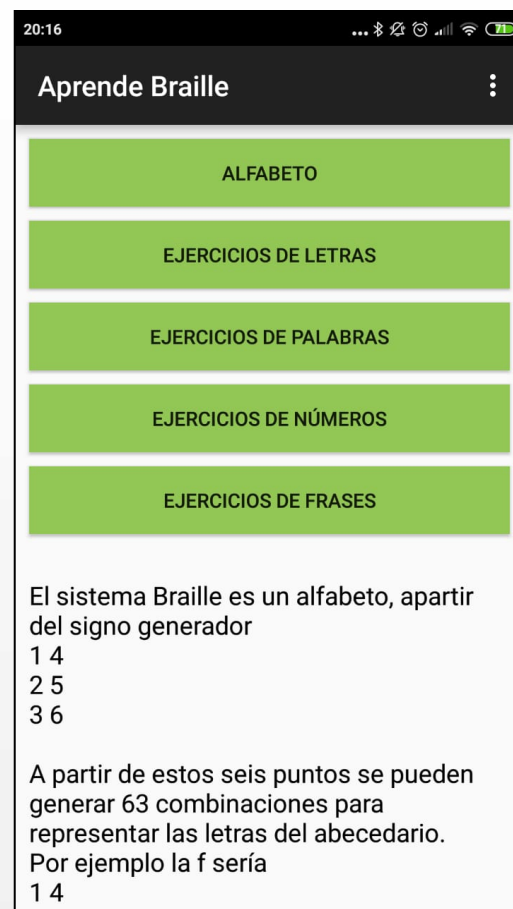
Fundamentação Teórica

- Componentização
- Produtividade: Pode-se economizar tempo de desenvolvimento, dependendo do portfólio de componentes já prontos;
- Robustez: Maior qualidade no produto final que utiliza componentes, pois os mesmos já foram largamente testados em um projeto dedicado à construção dos mesmos;
- Padrão de desenvolvimento: Equipe orientada a desenvolvimento nos moldes da componentização (OLIVEIRA, 2012, p. 1).

Trabalhos Correlatos

- **Aprende Braille**

- Ugedo (2016).
- alfabeto – lista com o alfabeto e seus respectivos sinais em braille.
- Exercícios – exercícios para o usuário digitar a letra, palavra ou frase que é representada nas células.



Trabalhos Correlatos

- **AbcNumBraille**
- Wermeson R. Aquino,
Zildomar C. Félix, Isledna
Rodrigues de Almeida, Ítalo
Cesar de Souza Belo;
- Feedback sonoro;
- Feedback tátil;
- Exercícios.



Trabalhos Correlatos

- **LêBraille**

- Agebson Rocha Façanha, Lívio Siqueira Lima, Maria da Conceição Carneiro Araújo, Windson Viana de Carvalho, Mauro Cavalcante Pequeno (2012).
- o aplicativo dispõe de uma tela com seis teclas que representam uma cela braille;
- interação por gestos na tela;
- feedback da interação por áudio.



Requisitos

Requisitos funcionais

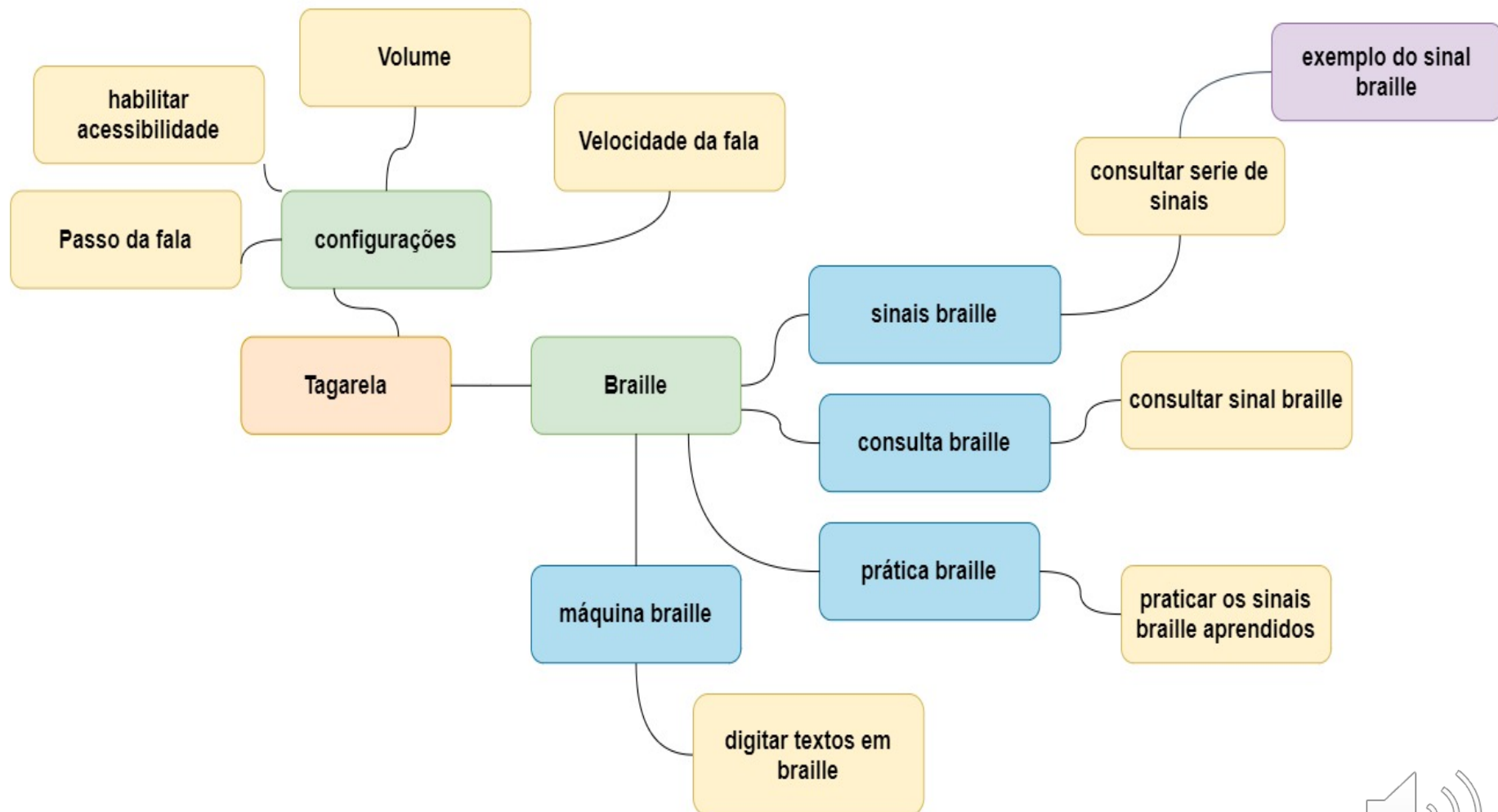
- Possuir *feedback* com áudio (Requisito Funcional - RF);
- Interação por gestos na tela (RF);
- Implementar módulo Braille (RF);
- Implementar série de sinais (RF);
- Implementar consulta de sinais (RF);
- Implementar prática (RF);
- Implementar máquina Braille (RF);

Requisitos

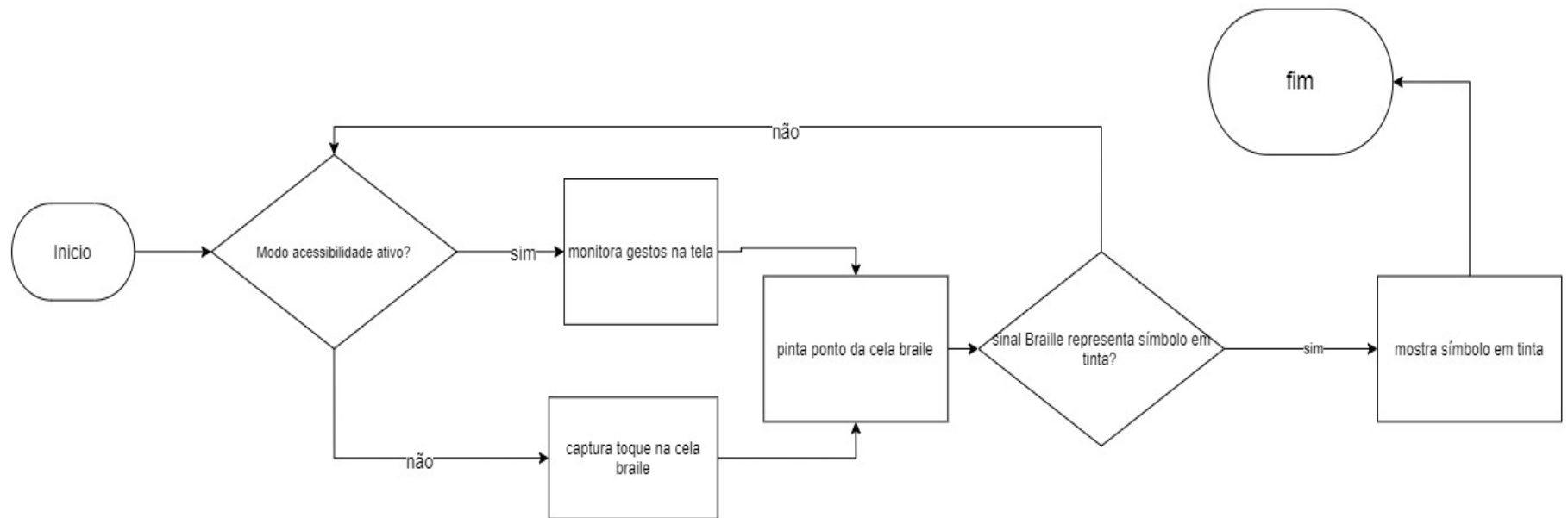
Requisitos não funcionais

- Utilizar flutter para criação do aplicativo (RNF);
- Permitir a criação de módulos em forma de componentização para facilitar novas implementações (RNF);
- Possuir o módulo Braille implementado para ser utilizado como modelo (RNF).

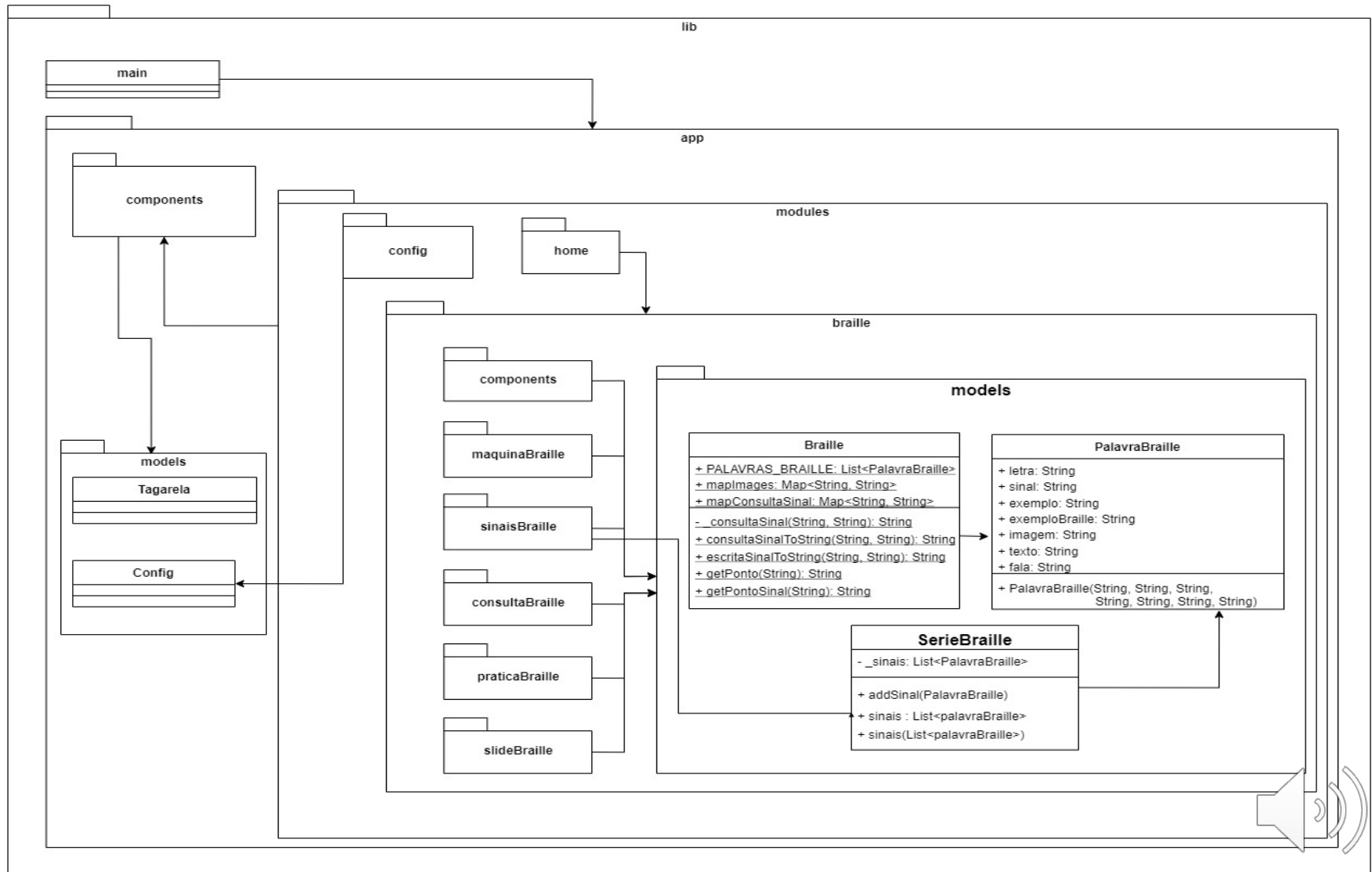
Especificação: Tagarela Braille



Especificação: Cella Braille



Especificação: Módulos



Implementação

- Sinais em binários

```
425
426     static Map<String, String> mapConsultaSinal = {
427         '001111' + '100000': '1',
428         '001111' + '110000': '2',
429         '001111' + '100100': '3',
430         '001111' + '100110': '4',
431         '001111' + '100010': '5',
432         '001111' + '110100': '6',
433         '001111' + '110110': '7',
434         '001111' + '110010': '8',
435         '001111' + '010100': '9',
436         '001111' + '010110': '0',
437         '100000' + '000000': 'a',
438         '110000' + '000000': 'b',
439         '100100' + '000000': 'c',
440         '100110' + '000000': 'd',
441         '100010' + '000000': 'e',
442         '110100' + '000000': 'f',
443         '110110' + '000000': 'g',
```


Implementação

- Mapeamento gestos na tela

```
76
77  ✓ PositionTap getPosition(double x, y) {
78      Size size = MediaQuery.of(context).size;
79  ✓      if (x < size.width * .5 && y < size.height * .33) {
80          return PositionTap.leftTop;
81  ✓      } else if (x > size.width * .5 && y < size.height * .33) {
82          return PositionTap.rightTop;
83  ✓      } else if (x < size.width * .5 && y < size.height * .66) {
84          return PositionTap.leftCenter;
85  ✓      } else if (x > size.width * .5 && y < size.height * .66) {
86          return PositionTap.rightCenter;
87  ✓      } else if (x < size.width * .5) {
88          return PositionTap.leftButton;
89          } else
90          return PositionTap.rightButton;
91      }
92  }
93
```


Implementação

```
1  GestureAccessibilityWidget(  
2      active: Tagarela.config.acessible,  
3      onTap: (position) {  
4  
5          controller.setLetra();  
6          String pontos1 = Braille.getPontoSinal(controller.sinal1);  
7          String pontos2 = Braille.getPontoSinal(controller.sinal2);  
8          return '${(pontos1 != '' ? 'Pontos primeira célula, ' + pontos1 : '')} ${  
          (pontos2 != '' ? 'Pontos segunda célula, ' + pontos2 : '')} ${  
          (controller.letra != '' ? '. Letra: ' + controller.letra : '')}';  
9      },  
10     options: [  
11         OptionGesture(  
12             action: () {  
13                 cela = 1;  
14             },  
15             speak: 'Célula Braille 1, click duas vezes para confirmar'),  
16         OptionGesture(  
17             action: () {  
18                 cela = 2;  
19             },  
20             speak: 'Célula Braille 2, click duas vezes para confirmar')  
21     ],  
22     primarySpeak:  
23         'Consulta sinais braille, \n click na tela para marcar um ponto, \n  
24         pontos 1 e 4 na parte superior, \n' +  
         ' pontos 2 e 5 no centro e pontos 3 e 6 na parte infeirior. \n  
         para navegar entre as células Braille arraste para o lado',
```



• Caracteres em braille

calligraphr

!	"	%	&	'	(	
)	+	,	-	.	/		
0	1	2	3	4	5	6	7
8	9	:	;	=	?	@	A
B	C	D	E	F	G	H	I
J	K	L	M	N	O	P	Q
R	S	T	U	V	W	X	Y
Z	a	b	c	d	e	f	g




← Include all those four markers untrimmed on your photo or scan.



)	+	,	-	.	/
0	1	2	3	4	5
8	9	:	;	=	?
B	C	D	E	F	G



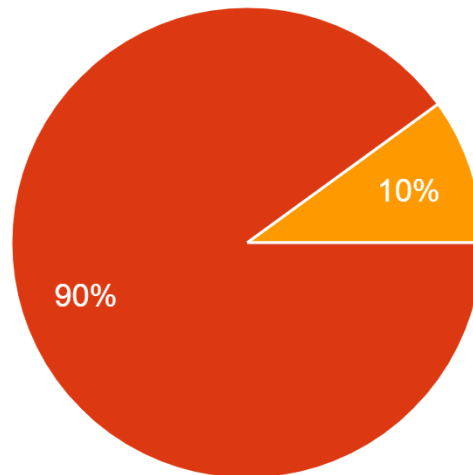
Análise dos Resultados

	Aprende Braille (UGEDO, 2016)	AbcNum (AQUINO, W. et al. 2015)	LêBraille (FAÇANHA et al. 2012)	Tagarela Braille
Plataforma	Android	Android	Android	Android, iOS
Feedback com áudio	Sim	Sim	Sim	Sim
Captura de gestos	Não	Não	Sim	Sim
Ensino de escrita	Não	Sim	Sim	Sim
Ensino de Leitura	Sim	Sim	Não	Sim
Consultar sinais braille	Sim	Não	Não	Sim 

Análise dos Resultados

Você conhece Braille?

10 respostas

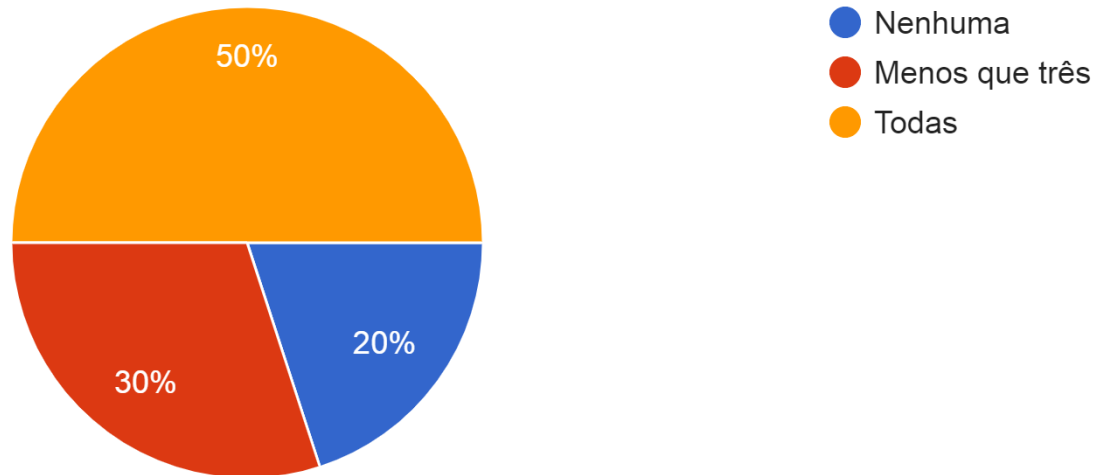


- Sim, sei ler Braille e tenha muito conhecimento
- Sim, mas só ouvi falar, não tenho muito conhecimento sobre
- Não, nunca ouvi falar

Análise dos Resultados

Quantas tarefas você concluiu sem NENHUM auxílio externo?

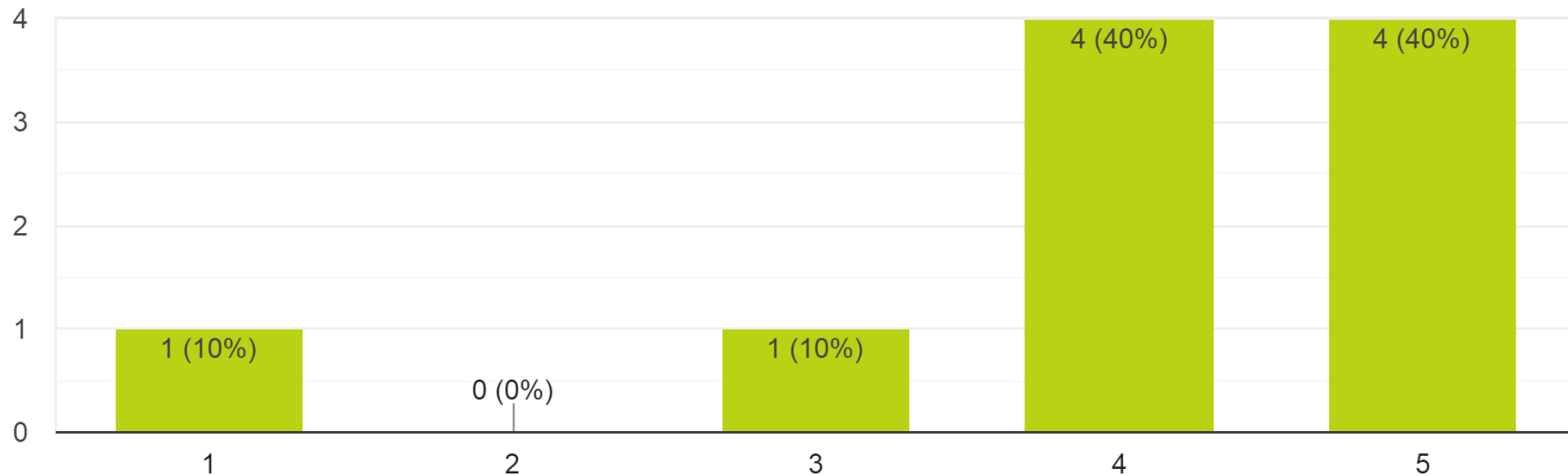
10 respostas



Análise dos Resultados

Como você classifica a usabilidade por pessoas com deficiência?

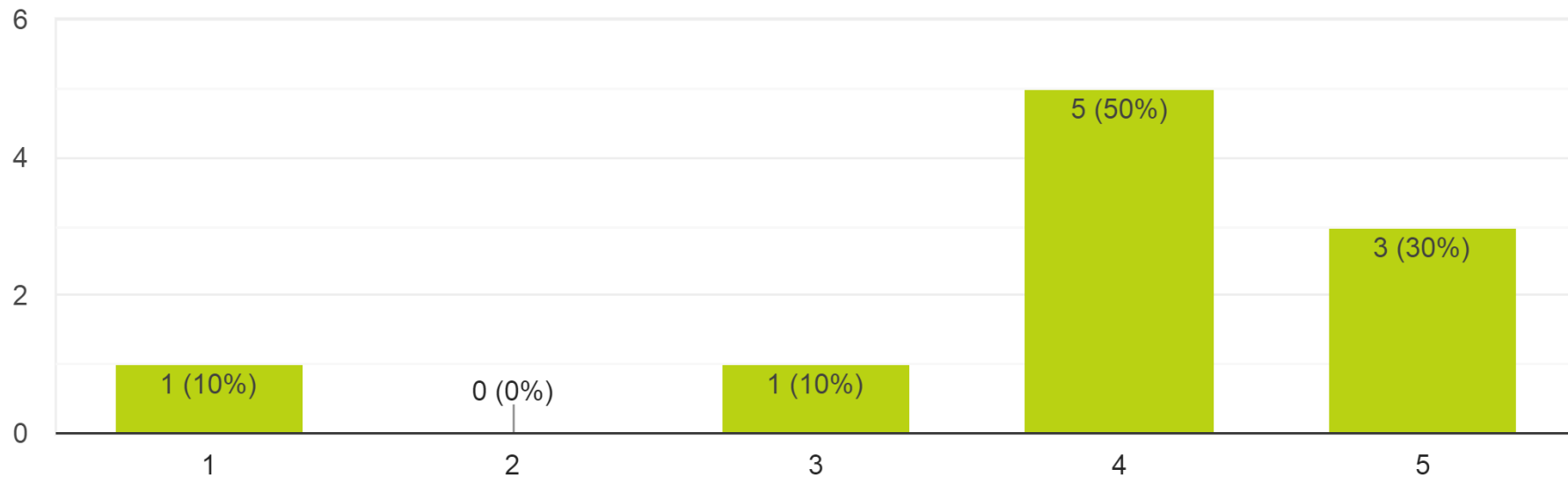
10 respostas



Análise dos Resultados

Como você classifica a usabilidade do tagarella Braille em Geral?

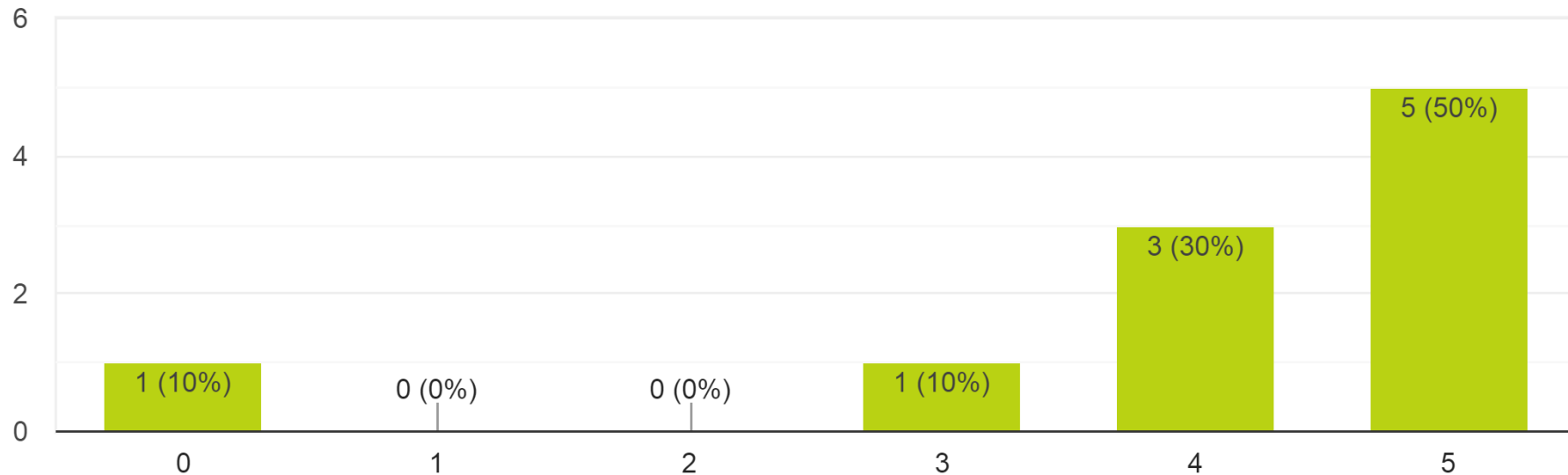
10 respostas



Análise dos Resultados

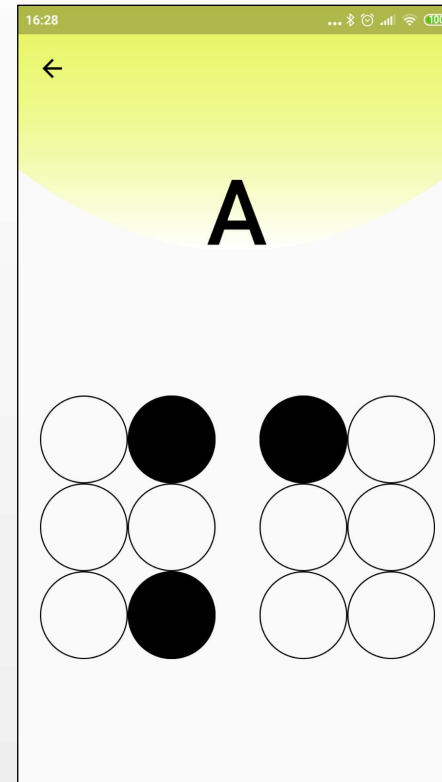
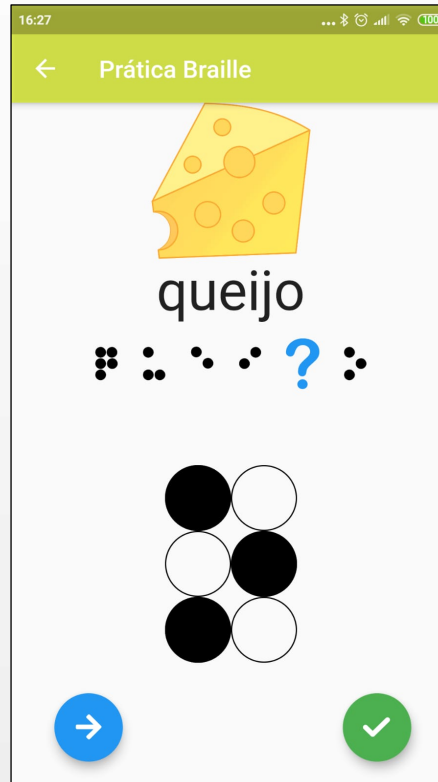
Você acha que o tagarella- Braille cumpriu o objetivo de ensinar e estimular o aprendizado por pessoas com e sem deficiência visual?

10 respostas



Análise dos Resultados

- Componentes reaproveitáveis por módulo



Análise dos Resultados

- Componentes reaproveitáveis por aplicativo

```
✓ lib
  ✓ app
    ✓ components
      > BoxText
      > ButtonDoubleFooter
      > ButtonErase
      > cardimageTitle
      > GestureAccessibility
      > GestureTagarela
      > headerBubble
      > headerCurve
      > headerWave
      > menuTagarela
      > tagarelaCard
    > icons
    > modules
    > shared
    > splash
    app_controller.dart
    app_controller.g.dart
    app_module.dart
    app_widget.dart
    main.dart
```



Conclusões e Sugestões

- ✓ Auxilia no aprendizado do braille
- ✓ Não apenas para normovisuais
- ✓ Interação por gestos
- ✓ feedback por áudio
- ✓ Maior acessibilidade



Conclusões e Sugestões

- Sugestões para trabalhos futuros:
 - a) criar *login* para o aplicativo salvar o progresso dos usuários;
 - b) incluir novos módulos para o aplicativo Tagarela;
 - c) criar funcionalidade de conversão de texto do alfabeto escrito para o braille;
 - d) migração para versão *web*;
 - e) criação de novos exercícios para a prática Braille;
 - f) tornar as configurações do aplicativo acessíveis.

Obrigado!