Empreendedorismo Digital

Apresentação

Matheus Lima
CEO da HairU
Estudante

Apresentação

Revolução Digital

Mercado

Porque o Digital?

Baixo Custo Inicial

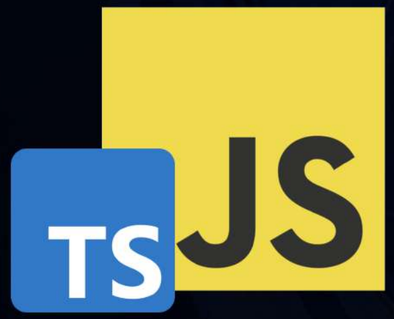
Escalabilidade

Acesso Global

Inovação Constante

Frameworks

Linguagens



Vou Contar uma História

CASTELO

WIDGET

ARGAMASSA

DART

COMO?

MANUAL

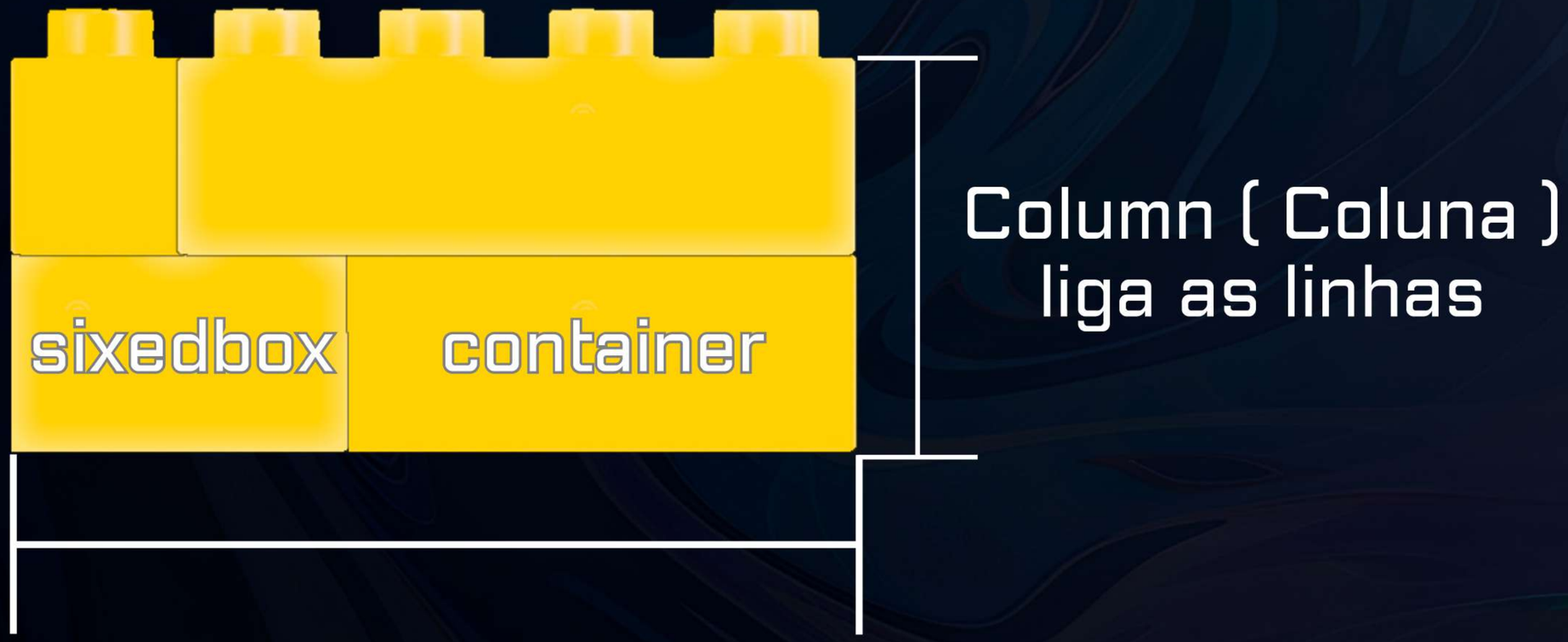
ARVORE DE WIDGETS

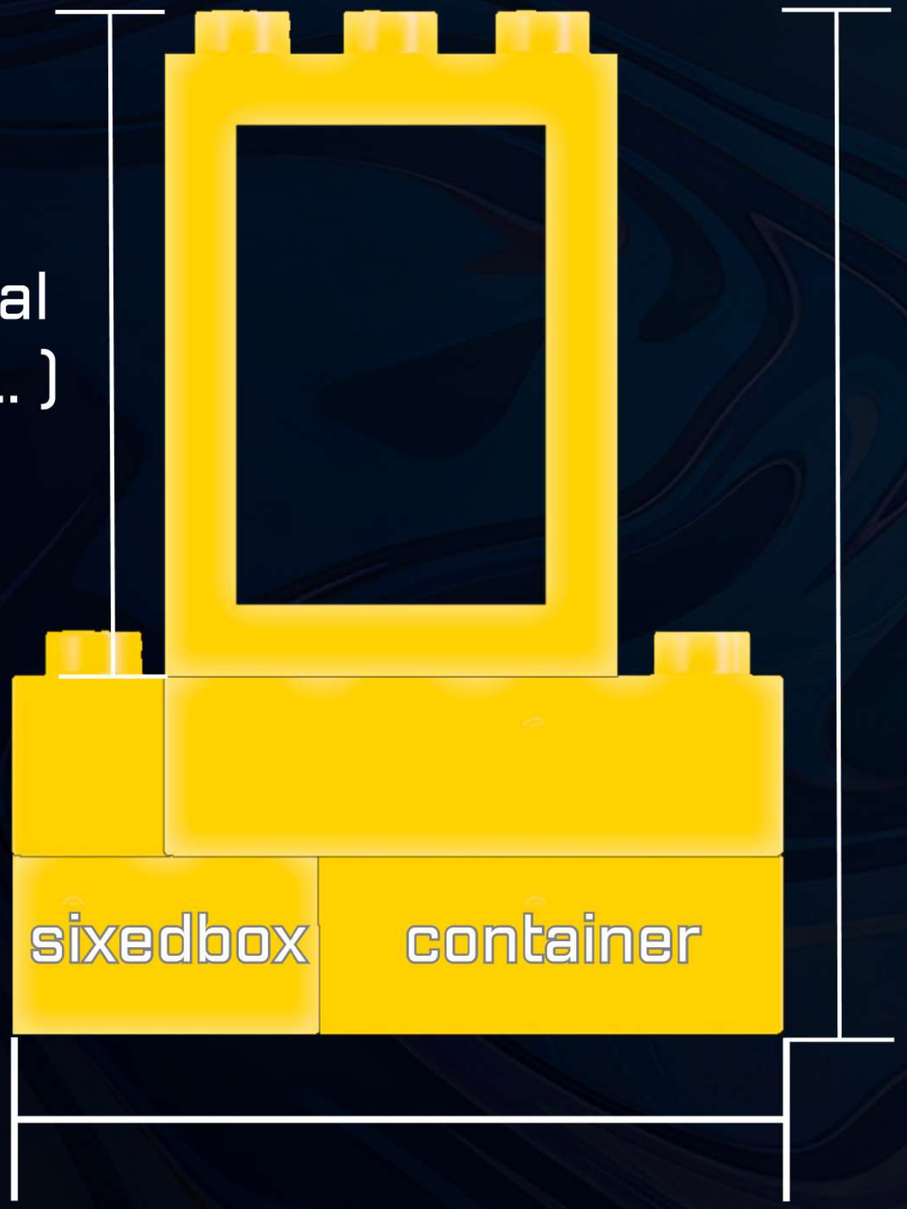
Depois de Muita Investigação

Criaram o LEGO

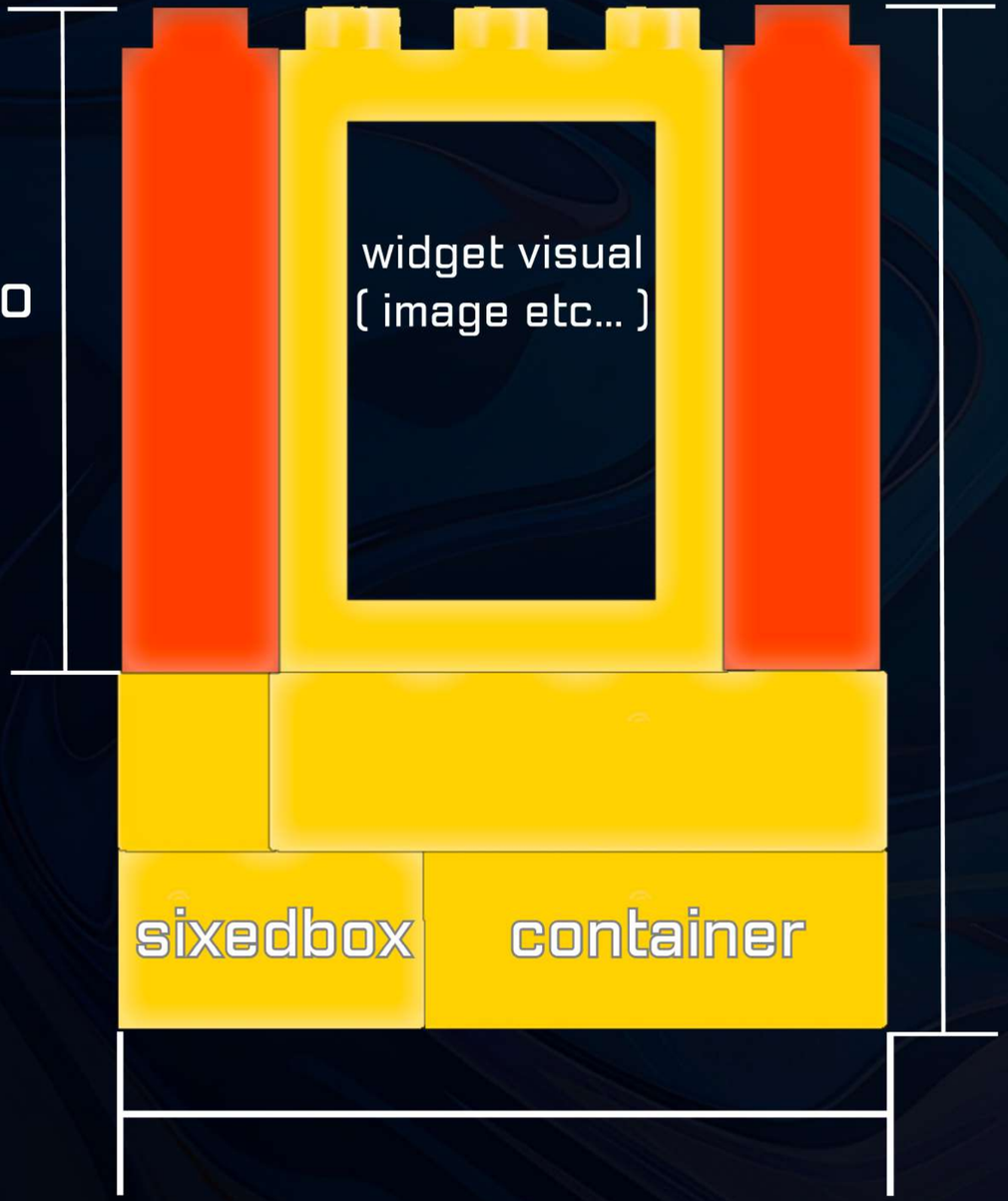
Vamos construir o Castelo?

Row (Linha)


sixedbox
container
Column (Coluna)
liga as linhas
Row (Linha)


widget visual
(image etc...)
Column (Coluna)
liga as linhas
sixedbox
container
Row (Linha)

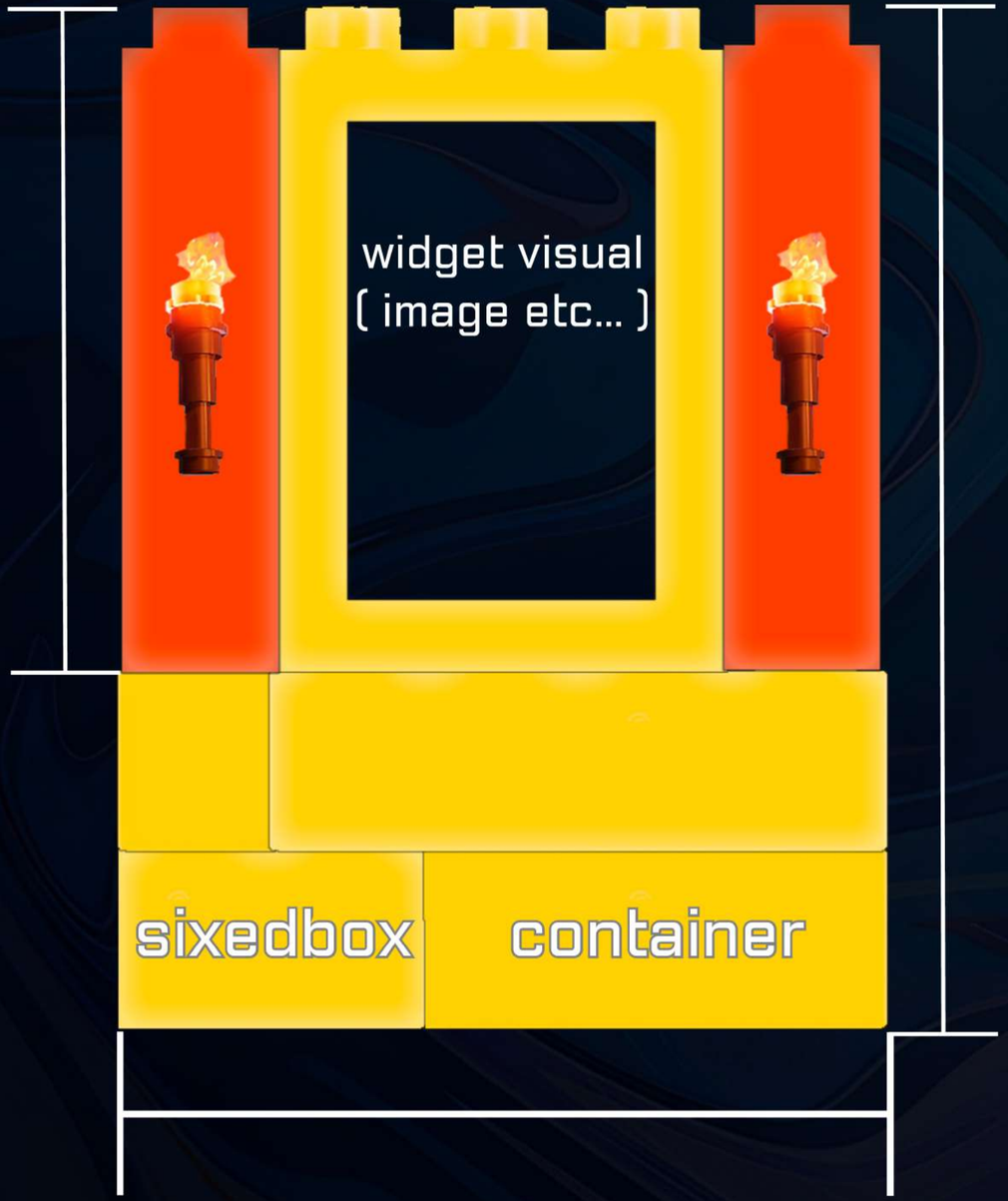
Widget modificado
(color: red)



Column (Coluna)
liga as linhas

Row (Linha)

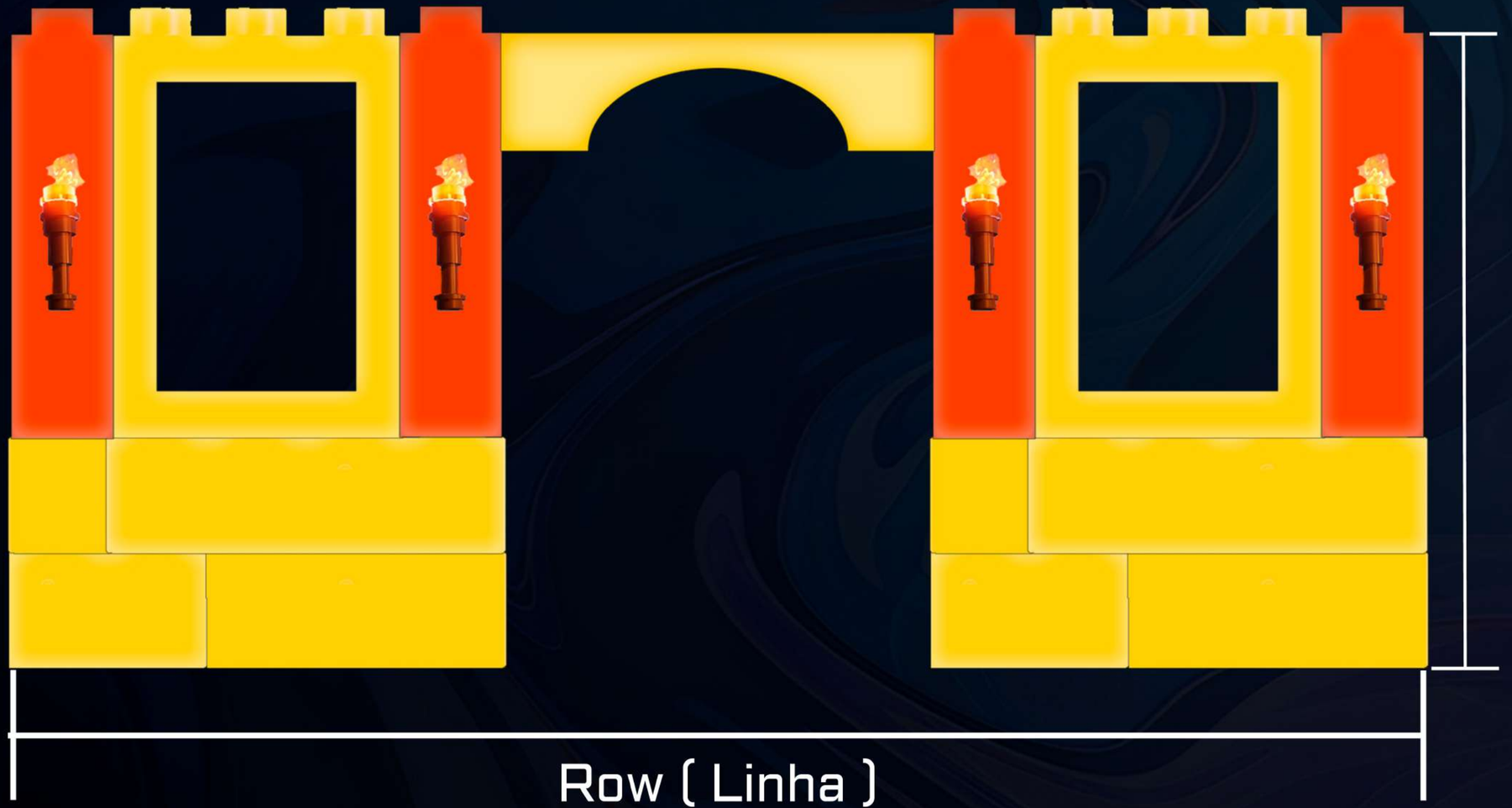
Stack
(Empilhar)



Column (Coluna)
liga as linhas

Row (Linha)

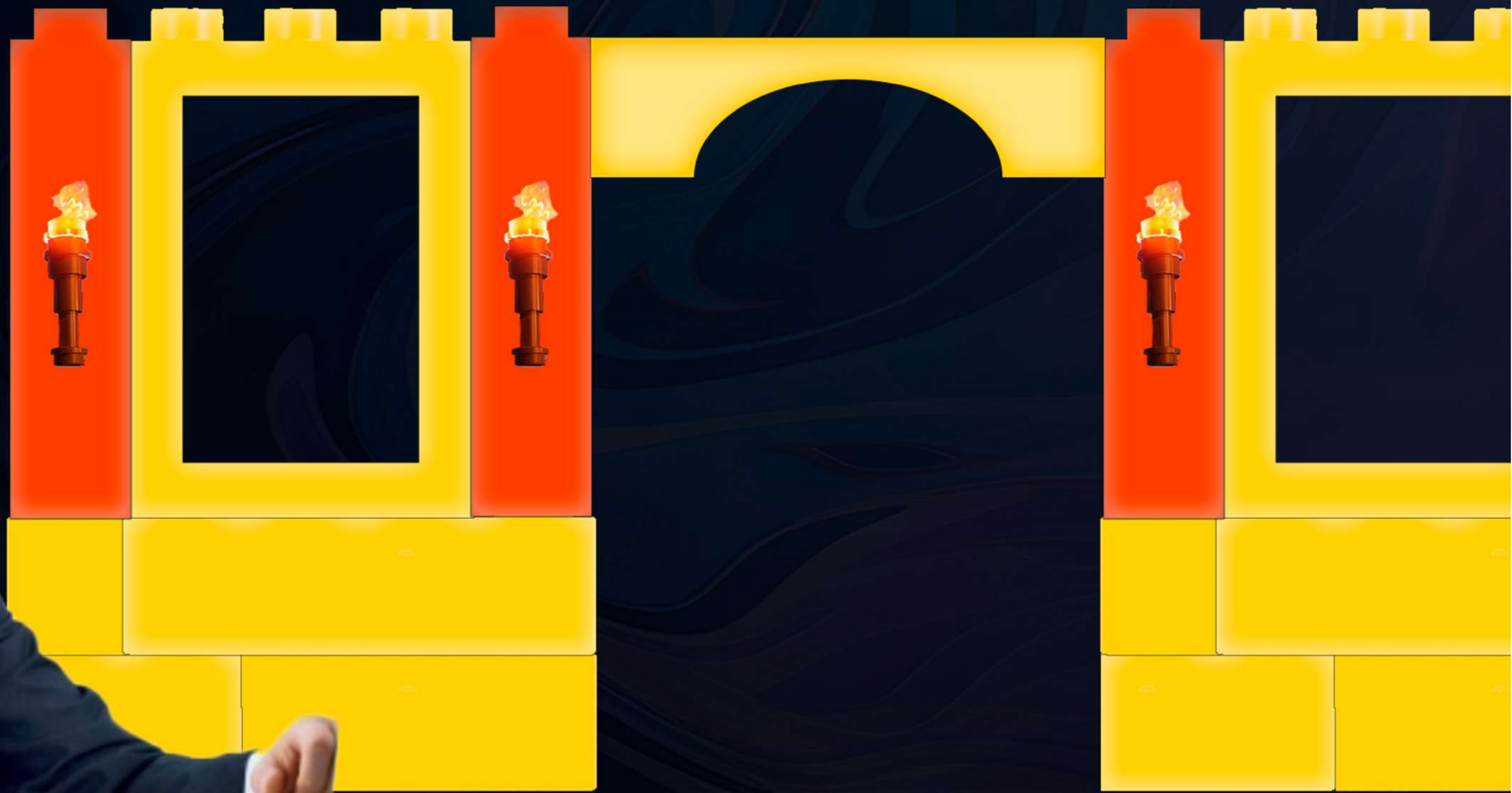
Crtl C + Crtl V



Column (Coluna)
liga as linhas

Row (Linha)

Aprovado !



Detalhes

Janela (filtro: vitral)

package portas
outro reino

Container
(color: #ffd100)

Teste e Apresentação

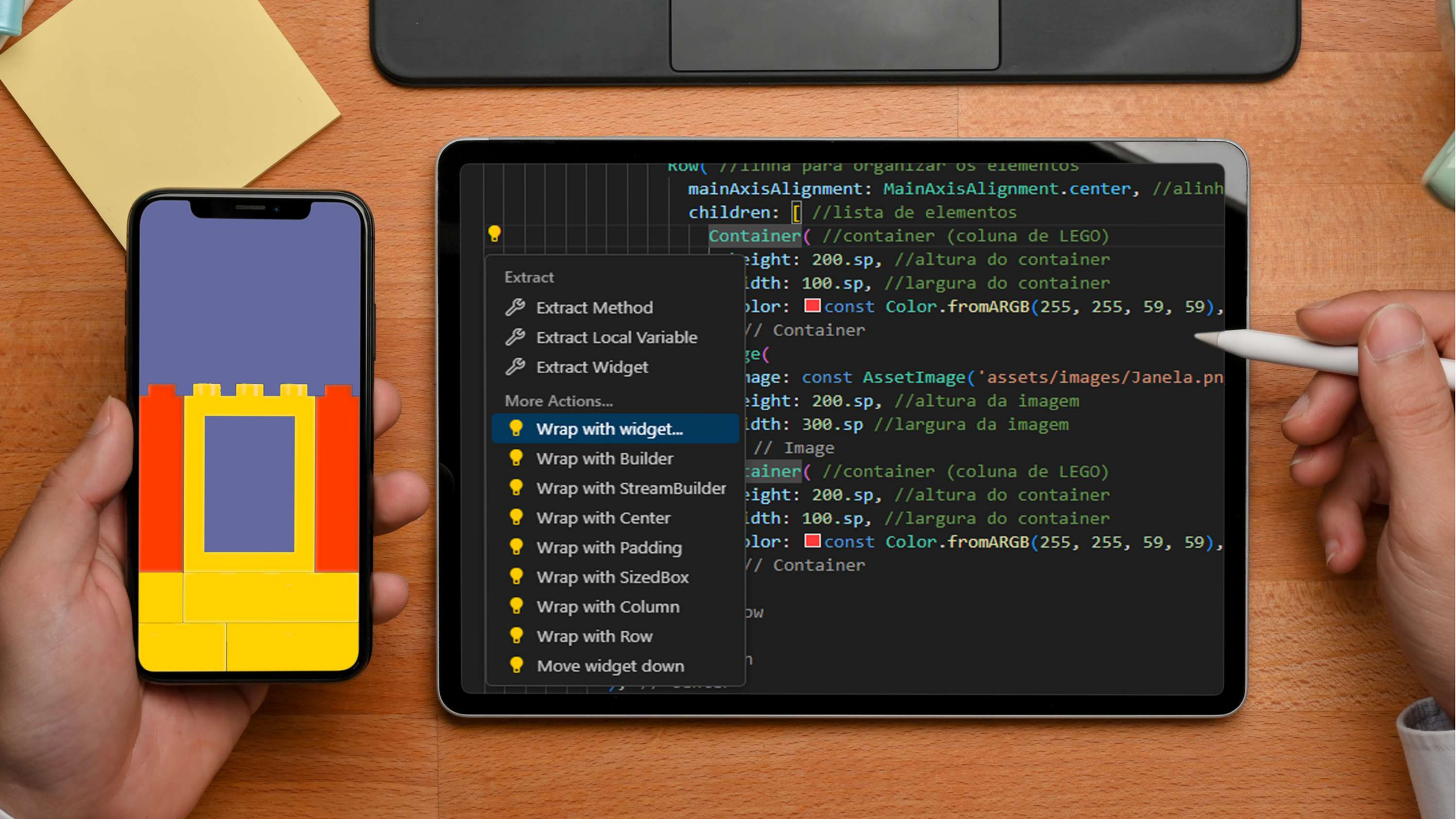
Chega de História !!

```dart
import 'package:flutter/material.dart';
import 'package:flutter_screenutil/flutter_screenutil.dart';

class Castelo extends StatefulWidget {
  const Castelo({super.key});

  @override
  State<Castelo> createState() => CasteloState();
}

class CasteloState extends State<Castelo> {
  @override
  Widget build(BuildContext context) {
    return Scaffold( //estrutura básica de layout visual
        backgroundColor: const Color.fromARGB(255, 28, 41, 50),
        body: SafeArea( //área segura para o conteúdo
          child: Container( //container para o conteúdo
            width: 500.sp,
            decoration: const BoxDecoration(
                gradient: LinearGradient(
                    begin: Alignment.topCenter,
                    end: Alignment.bottomCenter,
                    colors: [
                  Color.fromARGB(255, 12, 20, 27),
                  Color.fromARGB(255, 20, 28, 36)
                ])), // LinearGradient // BoxDecoration
```


begin: Alignment.topCenter,
end: Alignment.bottomCenter,
colors: [
Color.fromARGB(255, 12, 20, 27),
Color.fromARGB(255, 20, 28, 36)
])), // LinearGradient // BoxDecoration
child: Center(
child: Column(//coluna para organizar os elementos
mainAxisAlignment: MainAxisAlignment.end, //alinhamento principal no final
children: [
Row(//linha para organizar os elementos
mainAxisAlignment: MainAxisAlignment.center, //alinhamento principal no centro
children: [//lista de elementos
Container(//container (Bloco de LEGO)
height: 200.sp, //altura do container
width: 200.sp, //largura do container
color: Colors.yellow,
), // Container
Container(//container (Bloco de LEGO)
height: 200.sp, //altura do container
width: 300.sp, //largura do container
color: Colors.yellow,
), // Container
],
), // Row
],

```
children: [
  Row( //linha para organizar os elementos
    mainAxisAlignment: MainAxisAlignment.center, //alinhamento principal no centro
    children: [ //lista de elementos
      Container( //container (Bloco de LEGO)
        height: 200.sp, //altura do container
        width: 200.sp, //largura do container
        color: Colors.yellow,
      ), // Container
      Container( //container (Bloco de LEGO)
        height: 200.sp, //altura do container
        width: 300.sp, //largura do container
        color: Colors.yellow,
      ), // Container
    ],
  ), // Row
  Row( //linha para organizar os elementos
    mainAxisAlignment: MainAxisAlignment.center, //alinhamento principal no centro
    children: [ //lista de elementos
      Container( //container (Bloco de LEGO)
        height: 200.sp, //altura do container
        width: 100.sp, //largura do container
        color: Colors.yellow,
      ), // Container
      Container( //container (Bloco de LEGO)
        height: 200.sp, //altura do container
        width: 400.sp, //largura do container
        color: Colors.yellow,
```

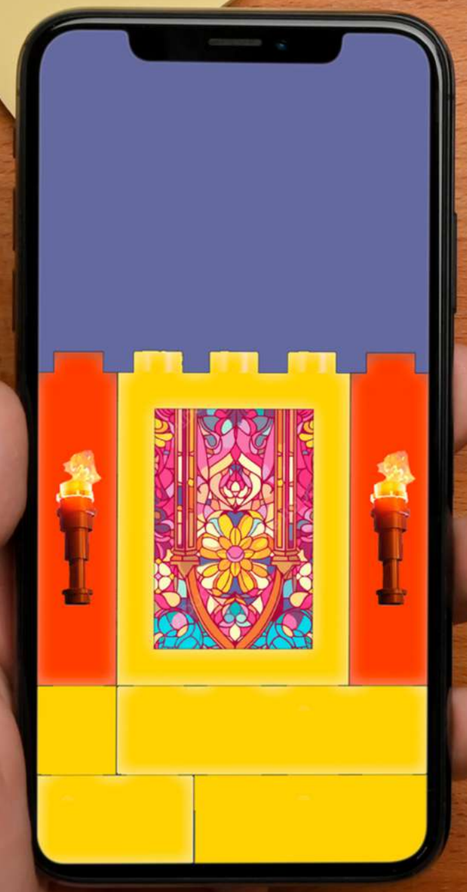
```
      Container( //container (Bloco de LEGO)
        height: 200.sp, //altura do container
        width: 400.sp, //largura do container
        color: Colors.yellow,
      ), // Container
    ],
  ), // Row
  Row( //linha para organizar os elementos
    mainAxisAlignment: MainAxisAlignment.center, //alinhamento principal no centro
    children: [ //lista de elementos
      Container( //container (coluna de LEGO)
        height: 200.sp, //altura do container
        width: 100.sp, //largura do container
        color: const Color.fromARGB(255, 255, 59, 59),
      ), // Container
      Image(
        image: const AssetImage('assets/images/Janela.png'), //imagem
        height: 200.sp, //altura da imagem
        width: 300.sp //largura da imagem
      ),  // Image
      Container( //container (coluna de LEGO)
        height: 200.sp, //altura do container
        width: 100.sp, //largura do container
        color: const Color.fromARGB(255, 255, 59, 59),
      ), // Container
    ],
```


mainAxisAlignment: MainAxisAlignment.center, //alinh
children: [//lista de elementos
Container(//container (coluna de LEGO)
eight: 200.sp, //altura do container
idth: 100.sp, //largura do container
olor: const Color.fromARGB(255, 255, 59, 59),
// Container
ge(
mage: const AssetImage('assets/images/Janela.pn
eight: 200.sp, //altura da imagem
idth: 300.sp //largura da imagem
// Image
tainer(//container (coluna de LEGO)
eight: 200.sp, //altura do container
idth: 100.sp, //largura do container
olor: const Color.fromARGB(255, 255, 59, 59),
// Container
Extract
Extract Method
Extract Local Variable
Extract Widget
More Actions...
Wrap with widget...
Wrap with Builder
Wrap with StreamBuilder
Wrap with Center
Wrap with Padding
Wrap with SizedBox
Wrap with Column
Wrap with Row
Move widget down

```
Row( //linha para organizar os elementos
  mainAxisAlignment: MainAxisAlignment.center, //alinhamento principal no centro
  children: [ //lista de elementos
    Stack(
      children: [
        Container( //container (coluna de LEGO)
          height: 200.sp, //altura do container
          width: 100.sp, //largura do container
          color: const Color.fromARGB(255, 255, 59, 59),
        ), // Container
        Image(
          image: const AssetImage('assets/images/Toxa.png'), //imagem
          height: 100.sp, //altura da imagem
          width: 50.sp //largura da imagem
        ), // Image
      ],
    ), // Stack
    Image(
      image: const AssetImage('assets/images/Janela.png'), //imagem
      height: 200.sp, //altura da imagem
      width: 300.sp //largura da imagem
    ),  // Image
    Stack(
      children: [
        Container( //container (coluna de LEGO)
          height: 200.sp, //altura do container
```

```
        Color.fromARGB(255, 12, 20, 27),
        Color.fromARGB(255, 20, 28, 36)
    ])), // LinearGradient // BoxDecoration
child: Center(
  child: Row(
    children: [
      Column( //coluna para organizar os elementos
        mainAxisAlignment: MainAxisAlignment.end, //alinhamento principal no final
        verticalDirection: VerticalDirection.up, //direção vertical de baixo para cima
        children: [
          Row( //linha para organizar os elementos
            mainAxisAlignment: MainAxisAlignment.center, //alinhamento principal no centro
            children: [ //lista de elementos
              Container( //container (Bloco de LEGO)
                height: 200.sp, //altura do container
                width: 200.sp, //largura do container
                color: Colors.yellow,
              ), // Container
              Container( //container (Bloco de LEGO)
                height: 200.sp, //altura do container
                width: 300.sp, //largura do container
                color: Colors.yellow,
              ), // Container
            ],
          ), // Row
```

```
                ), // Container
                Image(
                  image: const AssetImage('assets/images/Toxa.png'), //imagem
                  height: 100.sp, //altura da imagem
                  width: 50.sp //largura da imagem
                ), // Image
              ],
            ), // Stack
          ],
        ), // Row
      ],
    ), // Column
    Container( //espaçador
      height: 500.sp,
      width: 500.sp,
      color: Colors.yellow,
    ), // Container
    Column( //coluna para organizar os elementos
      mainAxisAlignment: MainAxisAlignment.end, //alinhamento principal no final
      verticalDirection: VerticalDirection.up, //direção vertical de baixo para cima
      children: [
        Row( //linha para organizar os elementos
          mainAxisAlignment: MainAxisAlignment.center, //alinhamento principal no centro
          children: [ //lista de elementos
            Container( //container (Bloco de LEGO)
              height: 200.sp, //altura do container
              width: 200.sp, //largura do container
              color: Colors.yellow,
```



```
      ), // Container
      Image(
        image: const AssetImage('assets/images/Toxa.png'), //imagem
        height: 100.sp, //altura da imagem
        width: 50.sp //largura da imagem
      ), // Image
    ],
  ), // Stack
  Image(
    image: const AssetImage('assets/images/Janela.png'), //imagem
    height: 200.sp, //altura da imagem
    width: 300.sp, //largura da imagem
    fit: vitral //filtro de vitral
  ),  // Image
  Stack(
    children: [
      Container( //container (coluna de LEGO)
        height: 200.sp, //altura do container
        width: 100.sp, //largura do container
        color: const Color.fromARGB(255, 255, 59, 59),
      ), // Container
      Image(
        image: const AssetImage('assets/images/Toxa.png'), //imagem
        height: 100.sp, //altura da imagem
        width: 50.sp //largura da imagem
      ), // Image
```

pubspec.yaml

```
intl: any
permission_handler: ^11.3.1
custom_info_window: ^1.0.1
clippy_flutter: ^2.0.0-nullsafety.1
clipboard: ^0.1.3
qr_flutter: ^4.1.0
share_plus: ^8.0.3
flutter_keyboard_visibility: ^6.0.0
whatsapp_bot_flutter: ^2.0.4
syncfusion_flutter_charts: ^24.2.9
pix_flutter: ^2.2.0
awesome_notifications: ^0.8.2
flutter_background_service: ^5.0.5
page_transition: ^2.1.0
app_links: ^6.1.1
portas: ^1.0.0
```

```dart
import 'package:flutter/material.dart';
import 'package:flutter_screenutil/flutter_screenutil.dart';
import 'package:portas/portas.dart';

        ), // Stack
      ],
    ), // Row
  ],
), // Column
Container( //espaçador
  height: 500.sp,
  width: 500.sp,
  color: Colors.yellow,
  child: const Porta(), //chamada do widget Porta
), // Container
Column( //coluna para organizar os elementos
  mainAxisAlignment: MainAxisAlignment.end, //alinhamento principal no final
  verticalDirection: VerticalDirection.up, //direção vertical de baixo para cim
  children: [
    Row( //linha para organizar os elementos
      mainAxisAlignment: MainAxisAlignment.center, //alinhamento principal no c
      children: [ //lista de elementos
        Container( //container (Bloco de LEGO)
          height: 200.sp, //altura do container
          width: 200.sp, //largura do container
```

1. Empreendedorismo Digital
2. Frameworks e Linguagens
3. Widgets como Blocos de Construção
4. Dart como Argamassa
5. Árvore de Widgets
6. Layout Dinâmico
7. Personalização e Detalhes
8. Pacotes Externos (Packages)
9. Testar e Publicar o App

Links

Como instalar o Flutter

Introdução ao Dart/Flutter

Flutter do básico ao avançado (EN)

Flutter do básico ao avançado (PT-BR)

Obrigado pela atenção!

Matheus Lima - CEO

(11) 97057-9393

Matheus Lima Castro

@hairu.oficial

HairU

contato@hairu.com.br