



15° CLASS

LOGIN PAGE

ROTEIRO

1



**FINAL TEXT EDITING
CONTROLLER.**


2



TEXT FIELD.


ADORO FLUTTER.




 Common Text field Demo 

 Name

 Password 

 Email

 Message

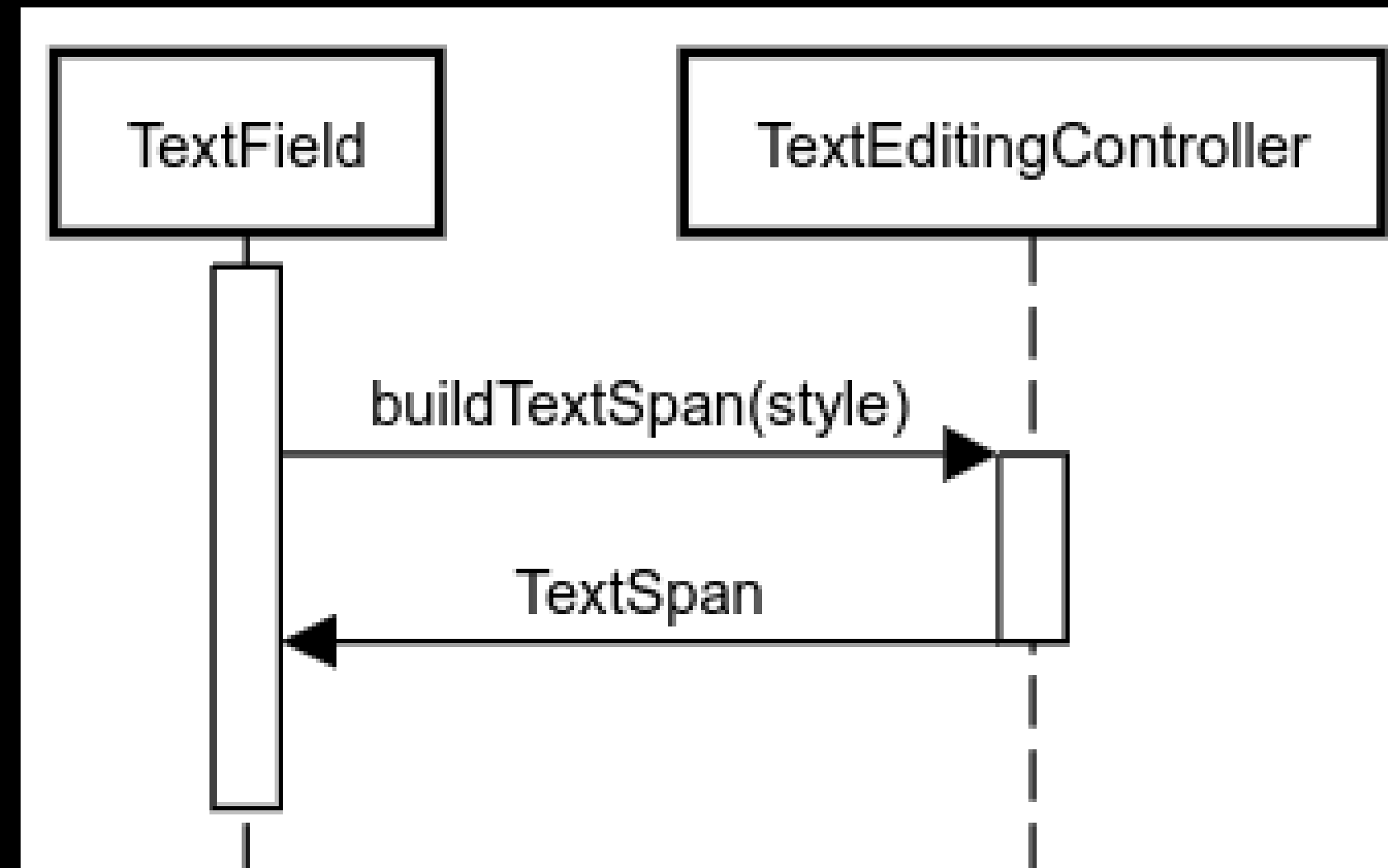
Submit

TEXT FIELD

É O CAMPO ONDE O USUÁRIO DIGITA.

TEXT EDITING CONTROLLER.

É A VARIÁVEL QUE "OBSERVA" O QUE O
USUÁRIO DIGITOU.

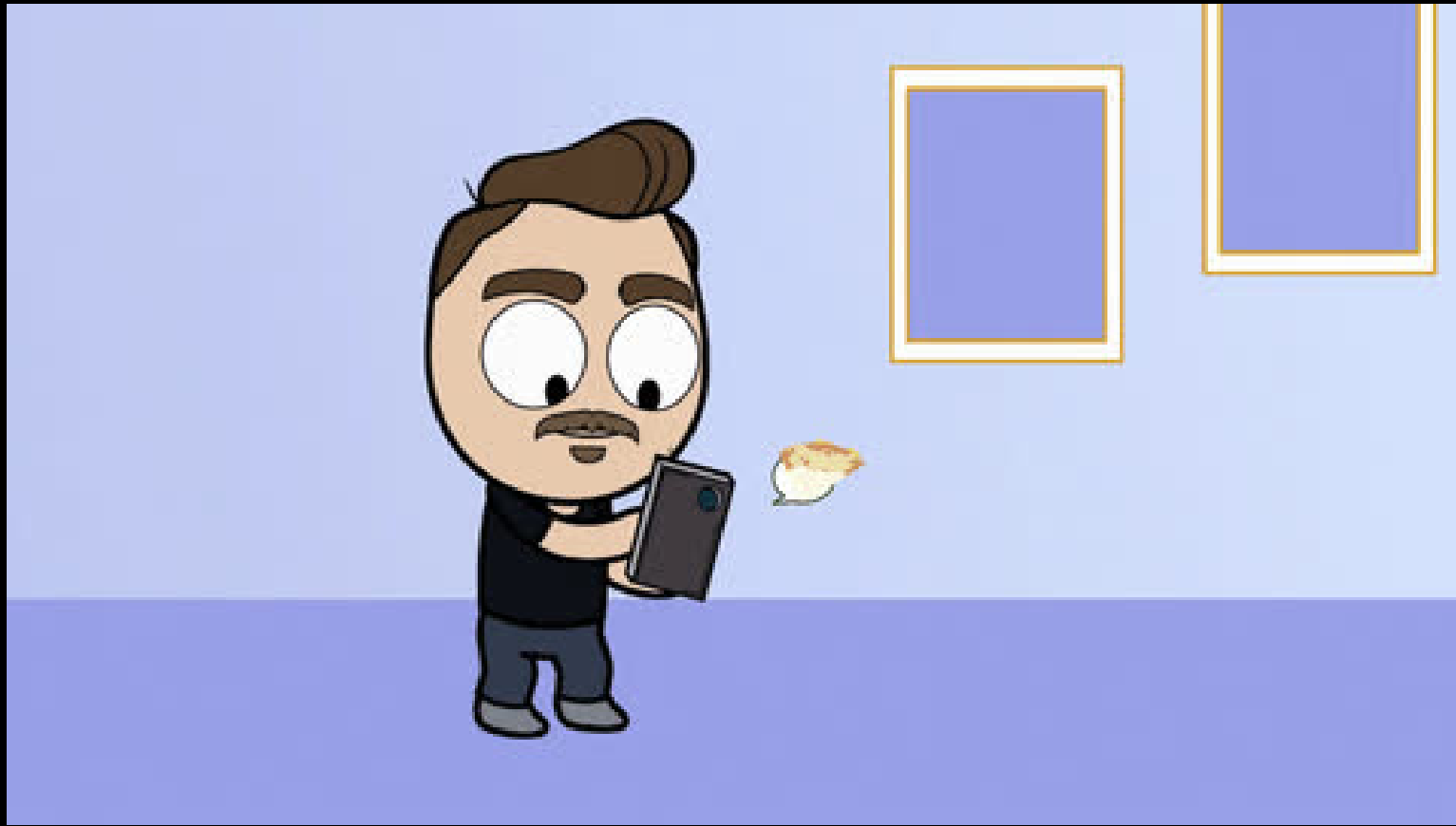


**TUDO QUE VOCÊ DIGITAR NO
TEXT FIELD, SERÁ O VALOR DA
VARIÁVEL
TEXTEDITINGCONTROLLER.**

**DESTA FORMA VOCÊ PODE
VERIFICAR AS INFORMAÇÕES
DIGITADAS.**



VAMOS



FAZER ?

EXEMPLO TELA DE LOGIN NO GIT

PASSO 1: CRIE SUA CLASSE STATEFUL E DEPOIS,,

CRIE AS VARIÁVEIS QUE OBSERVAM
OS DADOS DIGITADOS.

```
class _LoginPageState extends State<LoginPage> {  
  //Variavel que observa o que o usuario digita :) 😎  
  TextEditingController user = TextEditingController();  
  TextEditingController password = TextEditingController();  
}
```

PASSO 2:

CRIE AS VARIÁVEIS COM O VALOR
CORRETO

```
//Variavel com as informações corretas ✓🤖🐱🚲  
String correctUser = "vinicin";  
String correctPassword = "123";
```

PASSO 3:

CRIE VARIÁVEL PARA MOSTRAR O
ERRO

```
//Variavel para mostrar o erro 🗯️  
String erro = "";
```


PASSO 4:

CRIE A FUNÇÃO PARA VALIDAR AS
INFORMAÇÕES:

```
//Funcao para validar as informações
void login() {
  if(user.text == correctUser && password.text == correctPassword){
    Navigator.push(context, MaterialPageRoute(builder: (context)=> ApiPage()));
  }else {
    setState(() {
      erro = "Existem credenciais erradas";
    });
  }
}
```

PASSO 5:

PARA CADA CONTROLADOR CRIE UM
TEXT FIELD.

```
@override
Widget build(BuildContext context) {
  return MaterialApp(
    home: Scaffold(
      body: Center(
        child: Column(
          children: [
            Icon(Icons.person, size: 180, color: Colors.blue,),

            TextField(
              controller: user, //Aqui voce coloca a variavel que observa oq o usuario digita
              maxLength: 150, //Maximo de caracteres digitados
              decoration: InputDecoration(
                hintText: "Insira o seu nome", //Hint = Dica/o famoso "placeholder"
                border: OutlineInputBorder(//cria borda
                  borderRadius: BorderRadius.circular(20), //Faz a circunferencia da borda
                  borderSide: BorderSide(
                    color: Colors.blue
                  ) // BorderSide
                ) // OutlineInputBorder
              ), // InputDecoration
            // TextField
```

PASSO 6:

POR FIM , CRIE UM BOTÃO QUE
CHAMA A FUNÇÃO.

```
TextField(  
  obscureText: true, //Deixa a senha privada *****  
  controller: password, //Aqui voce coloca a variavel que observa oq o usuario digita  
  maxLength: 150, //Maximo de caracteres digitados  
  decoration: InputDecoration(  
    hintText: "Insira sua senha", //Hint = Dica/o famoso "placeholder"  
    border: OutlineInputBorder(//cria borda  
      borderRadius: BorderRadius.circular(20), //Faz a circunferencia da borda  
      borderSide: BorderSide(  
        color: Colors.blue  
      ) // BorderSide  
    ) // OutlineInputBorder  
  ), // InputDecoration  
), // TextField  
  
ElevatedButton(onPressed: login, child: Text("Login")),  
Text("$erro")
```



API PAGE

PASSO 1:

CRIE UMA TELA STATEFUL

```
class ApiPage extends StatefulWidget {  
  const ApiPage({super.key});  
  
  @override  
  State<ApiPage> createState() => _ApiPageState();  
}  
  
class _ApiPageState extends State<ApiPage> {
```

PASSO 2:

ANTES DO 2º OVERRIDE CRIE UMA
VARIÁVEL QUE PODE SER NULA.

```
class _ApiPageState extends State<ApiPage> {  
  String? value;
```

PASSO 3:

AGORA CRIE OUTRO OVERRIDE COM
A FUNÇÃO QUE RESETA A BUSCA
HTTP.

```
@override
void initState() { //funcao que recarrega o estado da pagina ,
//toda vez ao iniciar a pagina
  super.initState();
  getvalue(); //funcao que busca o valor , sera resetada sempre ao iniciar o app
}
```

PASSO 4:

CRIE A FUNÇÃO QUE FAZ A BUSCA
HTTP.

```
void getvalue() async { //funcao que busca o valor
  final response = await http.get(Uri.parse("https://dummyjson.com/products"));

  if(response.statusCode == 200){//se o status da requisicao for OK
    //json decode transforma as propriedades do json em tipos de dados
    final data = jsonDecode(response.body);

    setState(() {
      value = data[0]["title"];
    });
  }
}
```


PASSO 5:

CRIE OS COMPONENTES COM OPERADOR TERNÁRIO., CADA SITUAÇÃO PODE TER CONJUNTOS DE COMPONENTES.

```
@override
Widget build(BuildContext context) {
  return MaterialApp(
    home: Scaffold(
      body: value == null ? CircularProgressIndicator() : Text("$value")
    ) // Scaffold
  ); // MaterialApp
}
```

OBRIGADO!

FLUTTER WITH VINI;

