



# Tips and Tricks for Forms in Flutter



Rona Kilmer  
[@rkunboxed](#)



# Agenda

- Validation
- Keyboard Considerations
- Button Styling
- Handling Custom Form Fields
- Focus Widget and FocusNode
- Text Input Formatters
- Raffle



# Validation

```
final GlobalKey<FormState> _formKey = GlobalKey<FormState>();
```

```
Form(  
  key: _formKey,  
  child: Column(  
    children: <Widget>[  
      TextFormField(  
        validator: (String? value) {...},  
        autovalidateMode: AutovalidateMode.onUserInteraction,  
      ),  
    ],  
  ),  
)  
_formKey.currentState.validate();  
// can also save and reset
```

# Keyboard Considerations

```
TextFormField(  
  // for mobile keyboard buttons  
  textInputAction: TextInputAction.next,  
  // for all keyboards  
  autofocus: true,  
)
```



# Button Styling

```
// Style Button States with MaterialStateProperty.resolveWith()
```

```
TextButton(  
  style: ButtonStyle(  
    backgroundColor: MaterialStateProperty.resolveWith(...),  
  ),  
  child: const Text('SUBMIT'),  
  onPressed: onSubmit,  
)
```



# Demo: Example 1

[https://github.com/rkunboxed/flutter-forms-demo/blob/main/lib/example\\_one.dart](https://github.com/rkunboxed/flutter-forms-demo/blob/main/lib/example_one.dart)

# Handling Custom Form Fields

```
// We can wrap our custom form widgets with a Focus widget to handle keyboard  
// events and styling changes when the widget is focused.
```

```
Focus(  
  onKeyEvent: (focusNode, event) {  
    // can handle keyboard events here when widget is focused  
  },  
  onFocusChange: (bool focused) {  
    // can make style changes here when focus changes  
  },  
  child: Container(...)  
)
```

```
// What about validation of the custom form widget? We can use the FormField  
// widget for that.
```

```
FormField<dynamic>(  
  validator: customValidator,  
  builder: (FormFieldState fieldState) {  
    if (fieldState.hasError) {  
      // can adjust style properties here when field doesn't validate  
    }  
    return Container(...);  
  }  
)
```

# Demo: Example 2

[https://github.com/rkunboxed/flutter-forms-demo/blob/main/lib/example\\_two.dart](https://github.com/rkunboxed/flutter-forms-demo/blob/main/lib/example_two.dart)

FocusNode

```
// create in initState and don't forget to dispose!  
final FocusNode _myFocusNode = FocusNode();  
  
TextField(  
  // assign the FocusNode  
  focusNode: _myFocusNode,  
)  
  
// move focus to the FormField with this FocusNode  
_myFocusNode.requestFocus();
```



# Text Input Formatters

```
// Input Formatters
```

```
import 'package:flutter/services.dart';
```

```
TextFormField(  
  inputFormatters: [  
    FilteringTextInputFormatter.digitsOnly, // out-of-the-box formatter  
    FilteringTextInputFormatter.deny(RegExp(...)), // blacklist  
    FilteringTextInputFormatter.allow(RegExp(...)), // whitelist  
    MyCustomFormatter(), // custom formatter  
  ],  
)
```

```
class MyCustomFormatter extends TextInputFormatter {

  @override
  TextEditingValue formatEditUpdate(TextEditingValue oldValue, TextEditingValue newValue) {
    // filtering logic here

    return TextEditingValue(
      // formatted/filtered string
      text: 'New String',
      // indicates position of the cursor
      selection: TextSelection.collapsed(offset: index),
    );
  }
}
```

# Demo: Example 3

[https://github.com/rkunboxed/flutter-forms-demo/blob/main/lib/example\\_three.dart](https://github.com/rkunboxed/flutter-forms-demo/blob/main/lib/example_three.dart)

# Resources

- Demo Repo: <https://github.com/rkunboxed/flutter-forms-demo>
- Cookbook: <https://docs.flutter.dev/cookbook/forms>
- Flutter Gallery Demo:  
[https://github.com/flutter/gallery/blob/master/lib/demos/material/text\\_field\\_demo.dart](https://github.com/flutter/gallery/blob/master/lib/demos/material/text_field_demo.dart)
- Details on FocusNode:  
<https://api.flutter.dev/flutter/widgets/FocusNode-class.html>

