

Tarea Combo

Form_register.dart

```
import 'package:flutter/material.dart';
import 'package:flutter/services.dart';
import '../models/contact.dart';
import 'list_contact.dart';

class FormRegister extends StatelessWidget {
  FormRegister({Key? key}) : super(key: key);
  final Contact _contact =
    Contact(id: 0, name: "", age: 0, email: "", materia: "");
  List<Contact> _contacts = [];
  final _formKey = GlobalKey<FormState>();
  final _scaffoldkey = GlobalKey<ScaffoldState>();
  final _name = TextEditingController(text: 'Alex Reyes');
  final _age = TextEditingController(text: '25');
  final _email = TextEditingController(text: 'areyes@est.ups.edu.ec');
  final String titulo = "Mi pagina de registro";
  void _okSmartAlert(BuildContext context) {
    showDialog(
      context: context,
      builder: (BuildContext context) => AlertDialog(
        title: const Text('Mensaje del Sistema'),
        content: const Text('Registro Almacenado con éxito'),
        actions: <Widget>[
          TextButton(
            onPressed: () => Navigator.pop(context, 'ok'),
            child: const Text('Listo'),
          ),
          /*TextButton(
            onPressed: () => Navigator.pop(context, 'OK'),
            child: const Text('OK'),
          ),*/
        ],
      ),
    );
  }

  _onSubmit(BuildContext context) async {
    var form = _formKey.currentState;
    if (form!.validate()) {
      form.save();
      // ignore: avoid_print
      print(''
        Nombre : ${_contact.name}
```

```

        Edad : ${_contact.age}
        Email : ${_contact.email}
        Materia : ${_contact.materia}
        \n
    ''');
    _contacts.add(Contact(
        id: -1,
        name: _contact.name,
        age: _contact.age,
        email: _contact.email,
        materia: _contact.materia));
    form.reset();
    _okSmartAlert(context);
    /*showDialog(
        context: context,
        builder: (BuildContext context) {
            return new AlertDialog(
                title: new Text("My Super title"),
                content: new Text("Hello World"),
            );
        });*/
}
}

@override
Widget build(BuildContext context) {
    return Scaffold(
        // Fija los botones en el pie de la ventana,
        // por ejemplo, si se abre el teclado los botones se suben
        // encima del teclado si esta propiedad está en true
        // resizeToAvoidBottomInset: false,
        key: _scaffoldkey,
        appBar: AppBar(
            title: Text(titulo),
            centerTitle: true,
        ),
        body: Form(
            key: _formKey,
            child: Column(
                children: [
                    _wname(),
                    _wage(),
                    const SizedBox(
                        height: 20,
                    ),
                    _wemail(),
                    const SizedBox(
                        height: 20,
                    ),
                ],
            ),
        ),
    );
}

```

```

        _wmateria(),
        // MateriaCombo(),
        const SizedBox(
          height: 20,
        ),
        ElevatedButton(
          /*onPressed: () {
            if (_formKey.currentState!.validate()) {
              var form = _formKey.currentState;
              form!.save();
              _name.clear();
              _age.clear();
              _email.clear();
              /* _scaffoldkey.currentState!.showSnackBar(
                const SnackBar(
                  duration: Duration(seconds: 2),
                  content: Text('Procesando'),
                ),
              );*/
            }
          }, */
          onPressed: () => _onSubmit(context),
          child: const Text("Enviar"),
        )
      ],
    ),
  ),
  bottomNavigationBar: BottomAppBar(
    shape: const CircularNotchedRectangle(),
    child: Container(height: 40.0),
  ),
  floatingActionButton: FloatingActionButton(
    //onPressed: () => {Navigator.pushNamed(context, "/list")},
    onPressed: () {
      print(_contacts.length);
      //Navigator.pushNamed(context, "/list", arguments: _contacts);
      Navigator.push(
        context,
        MaterialPageRoute(
          builder: (context) => ListContact(contacts: _contacts),
        ),
      );
    },
    tooltip: 'Increment Counter',
    child: const Icon(Icons.list),
  ),
  floatingActionButtonLocation:
FloatingActionButtonLocation.centerDocked,
);

```

```

} // build

_wname() => TextFormField(
  keyboardType: TextInputType.text,
  decoration: const InputDecoration(
    labelText: "Ingreso nombre",
    icon: Icon(Icons.person),
    //suffixText: etiquetaFin,

    //border: InputBorder.none,
    //border: OutlineInputBorder(),
    hintText: "Nombre",
  ),
  controller: _name,
  maxLength: 30,
  // Evento que se dispara cuando se pasa la validación
  onSave: (val) => _contact.name = val!,
  //onSaved: (val) => setState(() => _contact.name = val!),
  validator: (val) => (val!.isEmpty ? 'Por favor ingresa nombre' :
null),
  /*validator: (value) {
    if (value!.isEmpty) {
      return "Por favor ingresa nombre";
    } else {
      return null;
    }
  },*/
);
_wage() => TextFormField(
  keyboardType: TextInputType.number,
  inputFormatters: <TextInputFormatter>[
    FilteringTextInputFormatter.allow(RegExp(r'[0-9]'))
  ],
  decoration: const InputDecoration(
    labelText: "Ingreso edad",
    icon: Icon(Icons.man),
    //border: InputBorder.none,
    hintText: "Edad"),
  controller: _age,
  validator: (value) {
    if (value!.isEmpty) {
      return "Por favor ingresa la edad";
    } else if (value.length > 2) {
      return "Edad no valida";
    } else {
      return null;
    }
  },
  // Evento que se dispara cuando se pasa la validación

```

```

        onSave: (val) => _contact.age = int.parse(val!.trim()),
    );
    _wemail() => TextFormField(
        keyboardType: TextInputType.emailAddress,
        inputFormatters: <TextInputFormatter>[
            FilteringTextInputFormatter.allow(RegExp(r"^[a-zA-Z0-9.a-zA-Z0-9@]+"))
        ],
        decoration: const InputDecoration(
            labelText: "Ingresa correo electrónico",
            icon: Icon(Icons.email),
            //border: InputBorder.none,
            hintText: "Email"),
        controller: _email,
        // Evento que se dispara cuando se pasa la validación
        onSave: (val) => _contact.email = val!,
        validator: (value) {
            if (value!.isEmpty) {
                return "Por favor ingresa el correo electrónico";
            } else if (!RegExp(
                r"^[a-zA-Z0-9.a-zA-Z0-9.!#$%&'*-+./=?^_`{|}~]+@[a-zA-Z0-9]+\.[a-zA-Z]+")
                .hasMatch(value)) {
                return "correo electrónico NO VALIDO";
            } else {
                return null;
            }
        },
    );

    _wmateria() => DropdownButtonFormField<String>(
        value: "Computación",
        icon: const Icon(Icons.arrow_downward),
        elevation: 16,
        style: const TextStyle(color: Colors.deepPurple),
        decoration: const InputDecoration(
            labelText: "Ingresa materia",
            icon: Icon(Icons.person),
            //suffixText: etiquetaFin,

            //border: InputBorder.none,
            //border: OutlineInputBorder(),
            hintText: "Materia",
        ),
        // Esta propiedad es solo de 'DropdownButton'
        // underline: Container(
        //   height: 2,
        //   color: Colors.deepPurpleAccent,
        // ),
    );

```

```

    onSave: (val) => _contact.materia = val!,
    onChanged: (String? newValue) {
      _contact.materia = newValue!;
    },
    items: <String>[
      'Computación',
      'Mecatrónica',
      'Contabilidad',
      'Arquitectura'
    ].map<DropdownMenuItem<String>>((String value) {
      return DropdownMenuItem<String>(
        value: value,
        child: Text(value),
      );
    }).toList(),
  );
}

```

Materia.dart

```

import 'package:flutter/material.dart';
import 'package:flutter/src/foundation/key.dart';
import 'package:flutter/src/widgets/framework.dart';

class MateriaCombo extends StatefulWidget {
  const MateriaCombo({Key? key}) : super(key: key);

  @override
  State<MateriaCombo> createState() => _MateriaComboState();
}

class _MateriaComboState extends State<MateriaCombo> {
  String dropdownValue = 'Computación';

  @override
  Widget build(BuildContext context) {
    return DropdownButtonFormField<String>(
      value: dropdownValue,
      icon: const Icon(Icons.arrow_downward),
      elevation: 16,
      style: const TextStyle(color: Colors.deepPurple),
      decoration: const InputDecoration(
        labelText: "Ingrese materia",
        icon: Icon(Icons.person),
        //suffixText: etiquetaFin,

        //border: InputBorder.none,
        //border: OutlineInputBorder(),

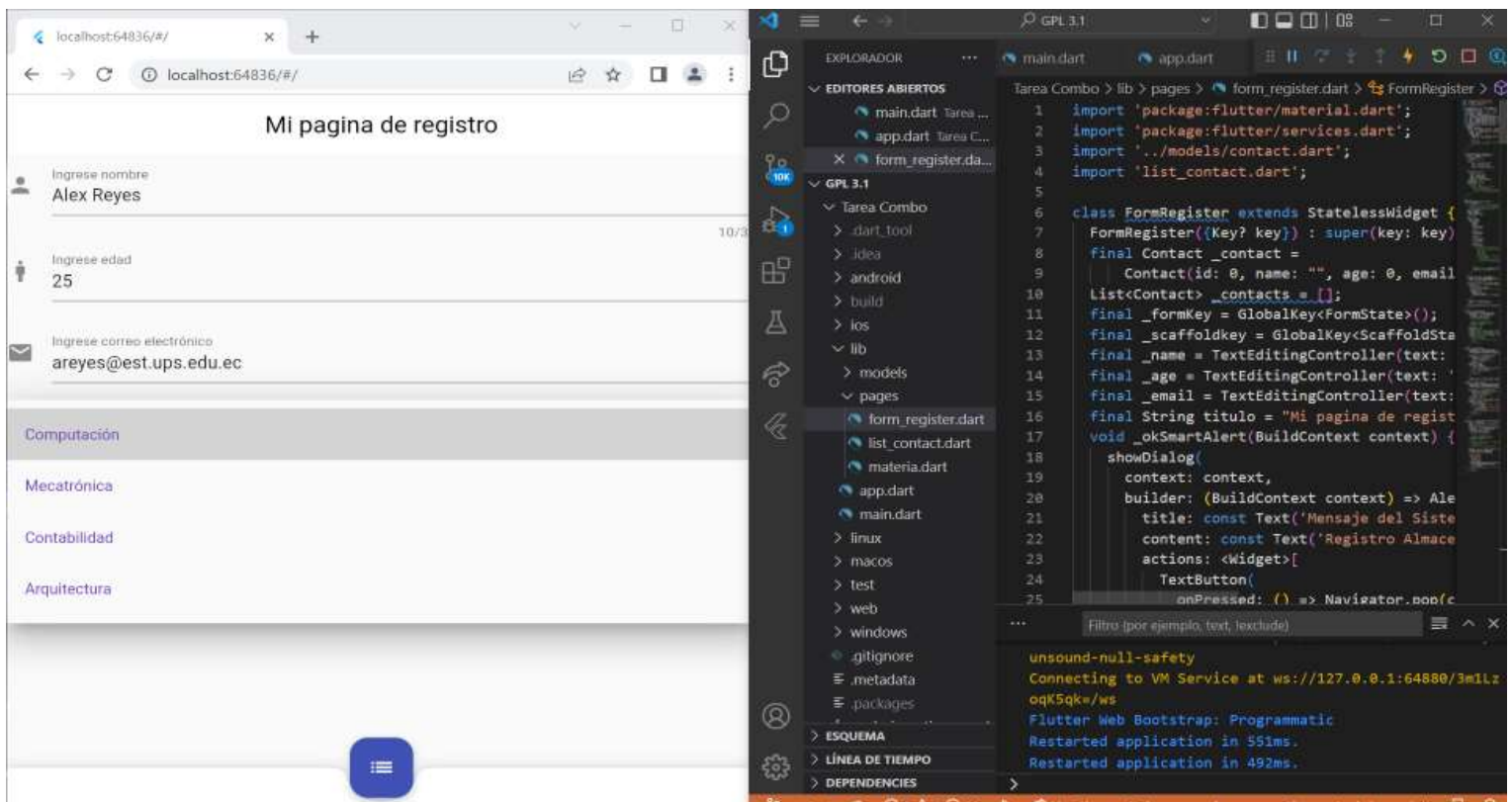
```

```

        hintText: "Materia",
      ),
      // Esta propiedad es solo de 'DropDownButton'
      // underline: Container(
      //   height: 2,
      //   color: Colors.deepPurpleAccent,
      // ),
      onChanged: (String? newValue) {
        setState(() {
          dropdownValue = newValue!;
        });
      },
      items: <String>[
        'Computación',
        'Mecatrónica',
        'Contabilidad',
        'Arquitectura'
      ].map<DropDownMenuItem<String>>((String value) {
        return DropDownMenuItem<String>(
          value: value,
          child: Text(value),
        );
      }).toList(),
    );
  }
}

```

Resultado.



Nombre: Alex Reyes – Felipe Miranda.