CALENDARIO

El proyecto se conforma de una script **main** donde se crea el menú principal y se guardan los datos, una page para el **formulario** donde se introducen los datos y se envían al main, y un par de modelos, un **modelo** para las asignaturas (los cuadrados del grid) y otro modelo para el color picker.

MAIN

En el main tenemos 3 array normales y 1 multidimensional.
Dos de los arrays son de Strings con los valores de las horas y los días. El otro array es de objetos del tipo modelo que he creado y que se usará para formar el gird del horario.
Con el array multidimensional créo el grid con una primera fila con los elementos de tipo del modelo y como texto los datos del array de días, la primera columna igual pero con los datos del array de horas y el resto con el widget clicable basado en el objeto modelo.

ARRAYS

Horas

```
var horas=['HORAS', '8.00\n8.55','8.55\n9.50','9.50\n10.45','10.45\n11.40','11
.40\n12.05','12.05\n13.00','13.00\n13.55','13.55\n14.50'];
```

Días

```
var dias =  ['HORAS', 'LUNES','MARTES','MIERC.','JUEVES','VIERNES'];
```

Cuadrados

```
var cuadrados=[
  Cuadrado("0,0","","",Color(0xFFFFFC107)),Cuadrado("0,1","","",Color(0xFFFFFC
107)),…]
```

Grid

```
for(int row = 0 ; row < twoDList.length ; row++)
            for(int col = 0 ; col < twoDList[row].length ; col++)
              if(row==0 && col==0)
                 twoDList[0][0]=CuadradoFijo(c: Cuadrado("$row$col",dias[row],
"",Colors.amber))
              else if(row==0 && col!=0)
                twoDList[0][col]=CuadradoFijo(c:Cuadrado("$row$col",dias[col],
"",Colors.amber))
              else if (col==0)
                twoDList[row][0]=CuadradoFijo(c:Cuadrado("$row$col",horas[row]
,"",Colors.amber))
              else
                twoDList[row][col]=_generarCuadrado(cuadrados[int.parse("$row$
col")]),
```

MODELO

El modelo es un objeto llamado Cuadrado que tiene como atributos una ID, dos textos (asignatura y aula) y un color de clase Color.

Este objeto será el que conforme el grid, es cada hueco del horario.

```
import 'dart:ui';

class Cuadrado{
  String id;
  String texto;
  String aula="";
  Color color;

  Cuadrado(this.id, this.texto, this.aula, this.color);
}
```

WIDGET

El único widget es el que se usa para los cuadrados de las horas y días. Se forma a partir del modelo y se le envían los datos desde el main.

```

import 'package:calendario3/model/cuadrado.dart';
import 'package:flutter/cupertino.dart';
import 'package:flutter/material.dart';

class CuadradoFijo extends StatefulWidget {
  CuadradoFijo({Key key, Cuadrado this.c}) : super(key: key);
  Cuadrado c;
  @override
  _CuadradoFijoState createState() => _CuadradoFijoState();
}


class _CuadradoFijoState extends State<CuadradoFijo> {

  @override
  Widget build(BuildContext context) {
    return Container(
      height: 200,
      width: 200,
      decoration:
          BoxDecoration(color: Colors.amber, borderRadius: BorderRadius.circul
ar(5)),
      child: Column(
          children: [
            SizedBox(height: 10),
```

```
            Text(widget.c.texto),
            Expanded(
              flex: 1,
              child: Container(
                width: 0,
                height: 0,
              ),
            ),
            SizedBox(
              height: 10,
            ),
          ],
        ),
      );
  }

}
```

PÁGINAS

La página principal es el main donde se crea el array y grid así como donde se crean los objetos clicables y se cambia su setState.

```
class _MyHomePageState extends State<MyHomePage> {

 var cuadrados=[
  Cuadrado("0,0","","",Color(0xFFFFFC107)),Cuadrado("0,1","","",Color(0xFFFFFC
107)),Cuadrado("0,3","","",Color(0xFFFFFC107)),Cuadrado("0,4","","",Color(0xFF
FFFC107)),Cuadrado("0,5","","",Color(0xFFFFFC107)),Cuadrado("0,6","","",Color(
0xFFFFFC107)),Cuadrado("0,7","","",Color(0xFFFFFC107)),Cuadrado("0,8","","",Co
lor(0xFFFFFC107)),Cuadrado("0,9","","",Color(0xFFFFFC107)),Cuadrado("0,10","",
"",Color(0xFFFFFC107)),
  Cuadrado("1,0","","",Color(0xFFFFFC107)),Cuadrado("1,1","","",Color(0xFF9E9E
9E9E)),Cuadrado("1,3","","",Color(0xFF9E9E9E9E)),Cuadrado("1,4","","",Color(0x
FF9E9E9E9E)),Cuadrado("1,5","","",Color(0xFF9E9E9E9E)),Cuadrado("1,6","","",Co
lor(0xFF9E9E9E9E)),Cuadrado("1,7","","",Color(0xFF9E9E9E9E)),Cuadrado("1,8","
","",Color(0xFF9E9E9E9E)),Cuadrado("1,9","","",Color(0xFF9E9E9E9E)),Cuadrado("1
,10","","",Color(0xFF9E9E9E9E)),
  Cuadrado("2,0","","",Color(0xFFFFFC107)),Cuadrado("2,1","","",Color(0xFF9E9E
9E9E)),Cuadrado("2,3","","",Color(0xFF9E9E9E9E)),Cuadrado("2,4","","",Color(0x
FF9E9E9E9E)),Cuadrado("2,5","","",Color(0xFF9E9E9E9E)),Cuadrado("2,6","","",Co
lor(0xFF9E9E9E9E)),Cuadrado("2,7","","",Color(0xFF9E9E9E9E)),Cuadrado("2,8","
","",Color(0xFF9E9E9E9E)),Cuadrado("2,9","","",Color(0xFF9E9E9E9E)),Cuadrado("2
,10","","",Color(0xFF9E9E9E9E)),
  Cuadrado("3,0","","",Color(0xFFFFFC107)),Cuadrado("3,1","","",Color(0xFF9E9E
9E9E)),Cuadrado("3,3","","",Color(0xFF9E9E9E9E)),Cuadrado("3,4","","",Color(0x
FF9E9E9E9E)),Cuadrado("3,5","","",Color(0xFF9E9E9E9E)),Cuadrado("3,6","","",Co
```

```
lor(0xFF9E9E9E9E)),Cuadrado("3,7","","",Color(0xFF9E9E9E9E)),Cuadrado("3,8","
","",Color(0xFF9E9E9E9E)),Cuadrado("3,9","","",Color(0xFF9E9E9E9E)),Cuadrado("3
,10","","",Color(0xFF9E9E9E9E)),
  Cuadrado("4,0","","",Color(0xFFFFFC107)),Cuadrado("4,1","","",Color(0xFF9E9E
9E9E)),Cuadrado("4,3","","",Color(0xFF9E9E9E9E)),Cuadrado("4,4","","",Color(0x
FF9E9E9E9E)),Cuadrado("4,5","","",Color(0xFF9E9E9E9E)),Cuadrado("4,6","","",Co
lor(0xFF9E9E9E9E)),Cuadrado("4,7","","",Color(0xFF9E9E9E9E)),Cuadrado("4,8","
","",Color(0xFF9E9E9E9E)),Cuadrado("4,9","","",Color(0xFF9E9E9E9E)),Cuadrado("4
,10","","",Color(0xFF9E9E9E9E)),
  Cuadrado("5,0","","",Color(0xFFFFFC107)),Cuadrado("5,1","","",Color(0xFF9E9E
9E9E)),Cuadrado("5,3","","",Color(0xFF9E9E9E9E)),Cuadrado("5,4","","",Color(0x
FF9E9E9E9E)),Cuadrado("5,5","","",Color(0xFF9E9E9E9E)),Cuadrado("5,6","","",Co
lor(0xFF9E9E9E9E)),Cuadrado("5,7","","",Color(0xFF9E9E9E9E)),Cuadrado("5,8","
","",Color(0xFF9E9E9E9E)),Cuadrado("5,9","","",Color(0xFF9E9E9E9E)),Cuadrado("5
,10","","",Color(0xFF9E9E9E9E)),
  Cuadrado("6,0","","",Color(0xFFFFFC107)),Cuadrado("6,1","","",Color(0xFF9E9E
9E9E)),Cuadrado("6,3","","",Color(0xFF9E9E9E9E)),Cuadrado("6,4","","",Color(0x
FF9E9E9E9E)),Cuadrado("6,5","","",Color(0xFF9E9E9E9E)),Cuadrado("6,6","","",Co
lor(0xFF9E9E9E9E)),Cuadrado("6,7","","",Color(0xFF9E9E9E9E)),Cuadrado("6,8","
","",Color(0xFF9E9E9E9E)),Cuadrado("6,9","","",Color(0xFF9E9E9E9E)),Cuadrado("6
,10","","",Color(0xFF9E9E9E9E)),
  Cuadrado("7,0","","",Color(0xFFFFFC107)),Cuadrado("7,1","","",Color(0xFF9E9E
9E9E)),Cuadrado("7,3","","",Color(0xFF9E9E9E9E)),Cuadrado("7,4","","",Color(0x
FF9E9E9E9E)),Cuadrado("7,5","","",Color(0xFF9E9E9E9E)),Cuadrado("7,6","","",Co
lor(0xFF9E9E9E9E)),Cuadrado("7,7","","",Color(0xFF9E9E9E9E)),Cuadrado("7,8","
","",Color(0xFF9E9E9E9E)),Cuadrado("7,9","","",Color(0xFF9E9E9E9E)),Cuadrado("7
,10","","",Color(0xFF9E9E9E9E))
  ];


  @override
  Widget build(BuildContext context) {
    var dias =  ['HORAS', 'LUNES','MARTES','MIERC.','JUEVES','VIERNES'];
    var horas=['HORAS', '8.00\n8.55','8.55\n9.50','9.50\n10.45','10.45\n11.40'
,'11.40\n12.05','12.05\n13.00','13.00\n13.55','13.55\n14.50'];
    var twoDList = List.generate(8, (i) => List(6), growable: false);//el prim
er valor es el número de líneas

    return Scaffold(
      appBar: AppBar(
        title: Text(widget.title),
      ),
      body: GridView.count(
        padding: const EdgeInsets.all(5),//padding general alrededor
        crossAxisSpacing: 4.0,
        mainAxisSpacing: 8.0,
        crossAxisCount:6,
         children:[
```

```
          for(int row = 0 ; row < twoDList.length ; row++)
            for(int col = 0 ; col < twoDList[row].length ; col++)
              if(row==0 && col==0)
                 twoDList[0][0]=CuadradoFijo(c: Cuadrado("$row$col",dias[row],
"",Colors.amber))
              else if(row==0 && col!=0)
                twoDList[0][col]=CuadradoFijo(c:Cuadrado("$row$col",dias[col],
"",Colors.amber))
              else if (col==0)
                twoDList[row][0]=CuadradoFijo(c:Cuadrado("$row$col",horas[row]
,"",Colors.amber))
              else
                twoDList[row][col]=_generarCuadrado(cuadrados[int.parse("$row$
col")]),
        ]

      ),

    );
  }


  Widget _generarCuadrado(Cuadrado c){
    return Container(
      height: 200,
      width: 200,
      decoration:
          BoxDecoration(color: c.color, borderRadius: BorderRadius.circular(5)
),
      child: InkWell(
        onTap: ()async{
          var asignatura = Cuadrado(c.id,c.texto,c.aula,c.color);//creamos una
 variable de tipo Cuadrado
          Cuadrado response = await Navigator.push(context,//recogemos un obje
to de tipo Cuadrado desde la página FormPage pasando la asignatura
            MaterialPageRoute(
              builder: (context) => FormPage(asignatura: asignatura),
            ),
          );

          if (response != null) {//si devuelve algo cambiamos los datos con lo
s datos recibidos
            setState(() {
              c.texto = response.texto;
              c.aula=response.aula;
              c.color=response.color;
            });
            //Guardamos los datos en el array
```

```
            cuadrados[int.parse(c.id)].color=response.color;
            cuadrados[int.parse(c.id)].texto=response.texto;
            cuadrados[int.parse(c.id)].aula=response.aula;
          }

        },
        child: Column(
          children: [
            SizedBox(height: 10),
            Text(c.texto),
            Text(c.aula),
            Expanded(
              flex: 1,
              child: Container(
                width: 0,
                height: 0,
              ),
            ),
            SizedBox(
              height: 10,
            ),
          ],
        ),

      ));
  }
```

No he creado un widget para los cuadrados clicables ya que me daba problemas el traspaso de datos al main para guardar los cambios en el array de cuadrados. Si no cambia el array cada vez que hacemos un hot reload la página principal vuelve a su estado inicial.

La segunda página es un formulario con dos textField y un color picker que usaremos para modificar los datos del widget del horario (los clicables).

Al formulario accedemos clicando en el widget que queremos editar pasando el objeto Cuadrado que hemos clicado y lo volvemos a devolver con los cambios.

Paso el objeto completo por un lado para conseguir la id y así poder modificar el array en esa posición concreta y por otro lado, para que en caso de que quieras reeditar el objeto no tengas que volver a meter los datos si no que puedas ver que había ya de antes.

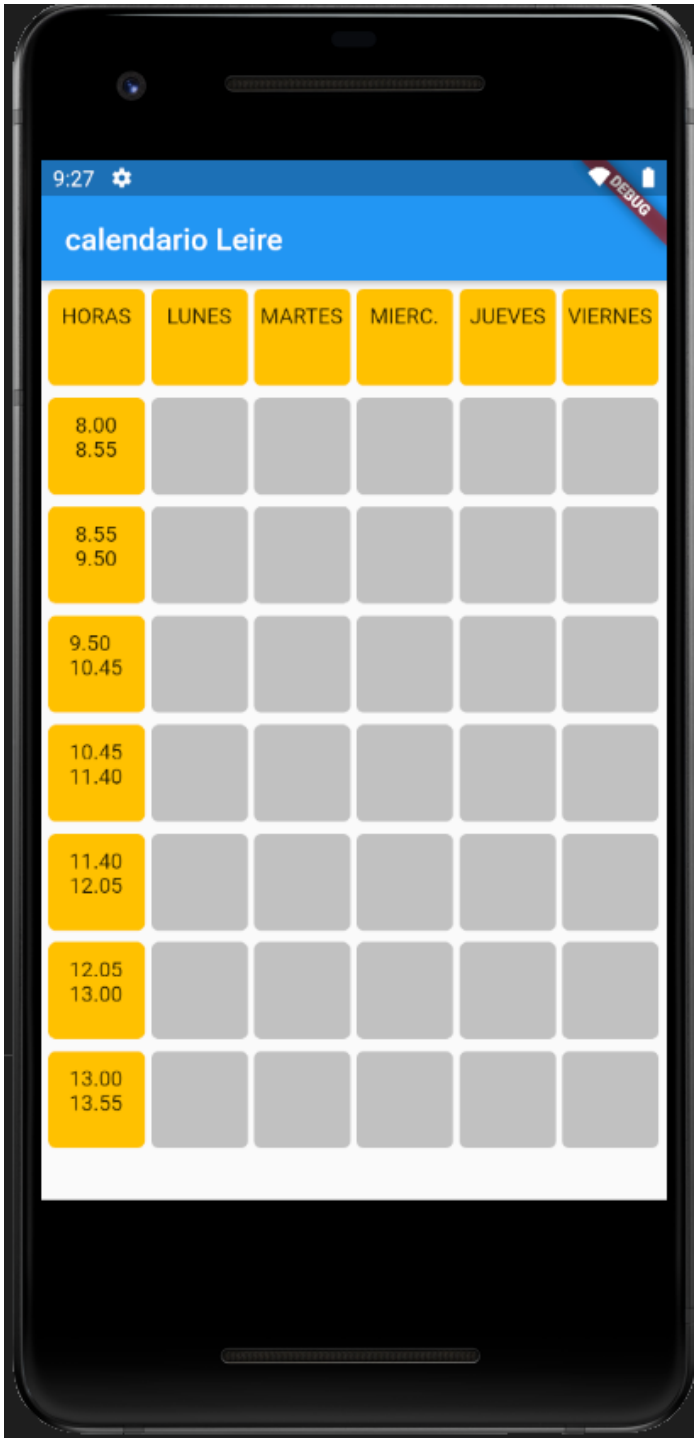
También hay la opción de resetear (borrar) la asignatura.

```
import 'package:calendario3/model/CircleColorPicker.dart';
import 'package:calendario3/model/cuadrado.dart';
import 'package:flutter/material.dart';
```

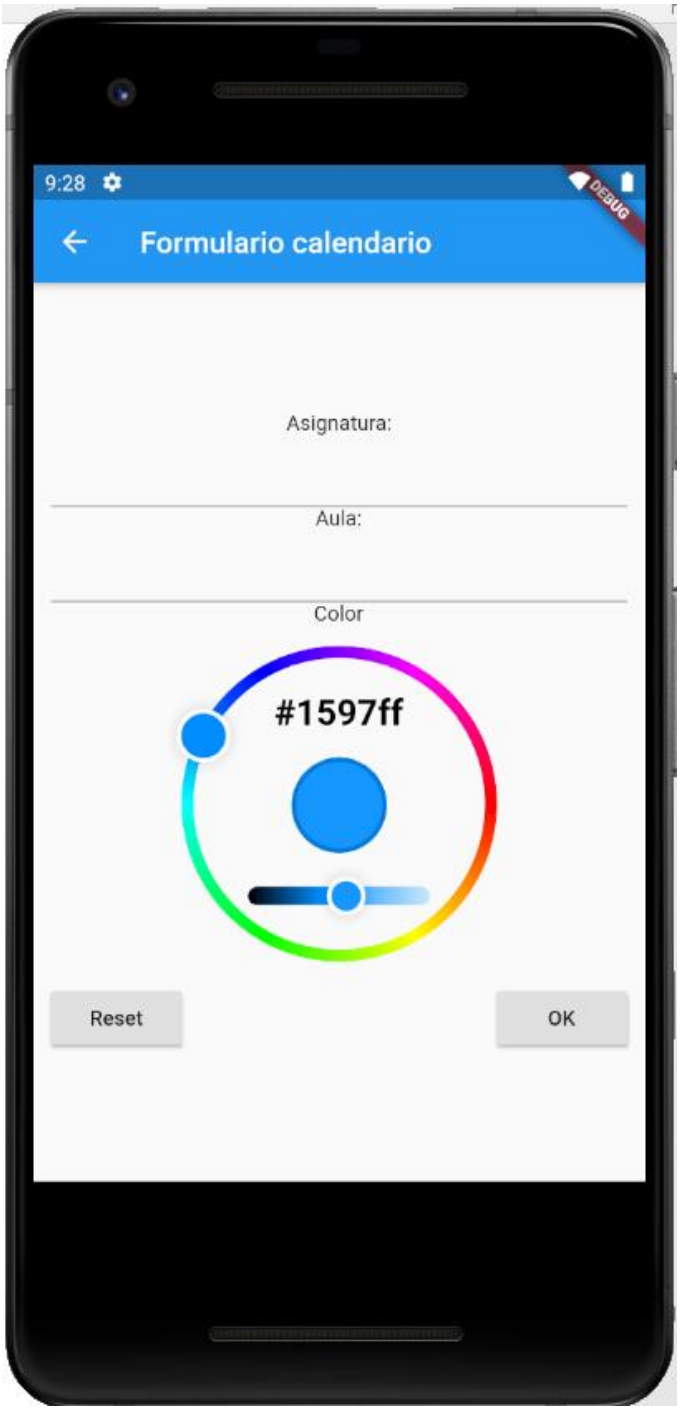
```
class FormPage extends StatelessWidget  {
  final Cuadrado asignatura;
  FormPage({this.asignatura});

  @override
  Widget build(BuildContext context) {
    asignatura.color=Colors.blue;//por si no cambia el color que se le ponga e
l que esta definido en el picker
    return Scaffold(
      appBar: AppBar(
        title: Text('Formulario calendario'),
      ),
      body: Container(//creamos un contenedor
        padding: EdgeInsets.all(12.0),
        alignment: Alignment.center,//alineamos en el centro los elementos
        child: ListView.builder(//creamos una lista para que cuando el usuario
 escriba no muestre un error de tamaño de pantalla
              shrinkWrap: true,
              itemCount: 1,
              itemBuilder: (BuildContext context, int index) {
              return Column(
                children: <Widget>[//los hijos serán el textfield, el color pi
cker y los botones para enviar los datos o resetearlos
                  Container(
                    child:Column(
                      children:<Widget> [
                        Text("Asignatura:"),
                        TextField(
                          controller: TextEditingController()..text = asignatu
ra.texto,
                          onChanged: (valor){//cuando cambie el valor lo asign
amos a la variable asignatura
                            asignatura.texto=valor;
                          },
                        ),
                        Text("Aula:"),
                        TextField(
                          controller: TextEditingController()..text = asignatu
ra.aula,
                          onChanged: (valor){//cuando cambie el valor lo asign
amos a la variable asignatura
                            asignatura.aula=valor;
                          },
                        ),
                        Text('Color'),
                        Center(
                          child: CircleColorPicker(//colorpickeer
```

```
                        initialColor: Colors.blue,
                        onChanged: (color) => asignatura.color= color,
                        size: const Size(240, 240),
                        strokeWidth: 4,
                        thumbSize: 36,
                      )),
                    ],
                  ),
                ),
                Row(//botones
                  mainAxisAlignment: MainAxisAlignment.spaceBetween,
                  children:<Widget> [
                  RaisedButton(
                    child: Text("Reset"),
                    onPressed: () {//dejamos los valores de asignatura como
al inicio de la aplicacion
                      asignatura.texto="";
                      asignatura.aula="";
                      asignatura.color=Color(0xFF9E9E9E9E);
                      Navigator.of(context).pop(asignatura);//pasamos la asi
gnatura al main.
                    }
                  ),
                  RaisedButton(
                    child: Text("OK"),
                    onPressed: () {//enviamos la asignatura con los datos ac
tuales
                      Navigator.of(context).pop(asignatura);
                    }
                  ),
                ],),
              ],
            );
          }
        ),
      )
    );
  }
}
```

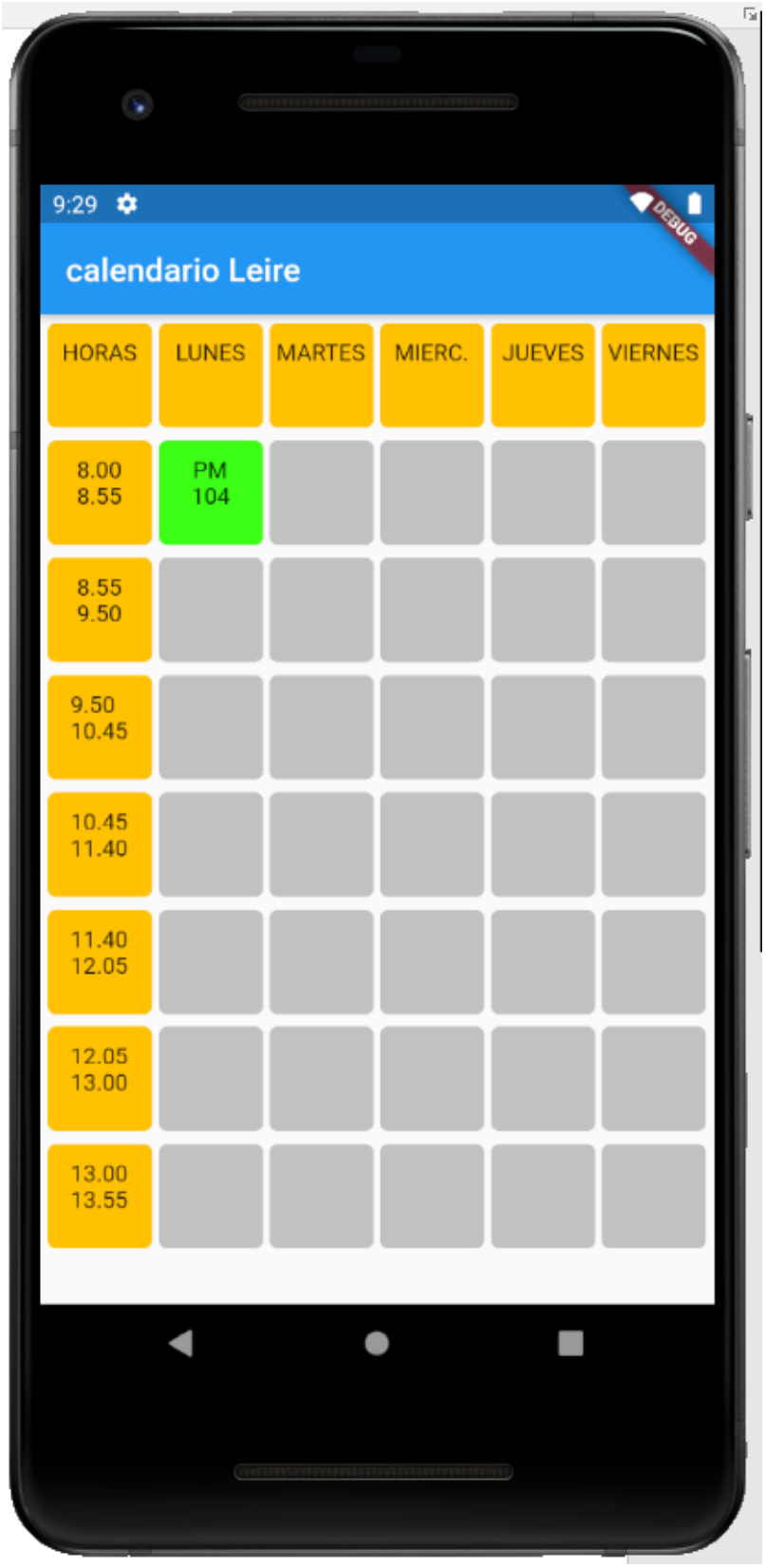
Menú inicial


9:27
DEBUG
calendario Leire
HORAS
LUNES
MARTES
MIERC.
JUEVES
VIERNES
8.00
8.55
8.55
9.50
9.50
10.45
10.45
11.40
11.40
12.05
12.05
13.00
13.00
13.55

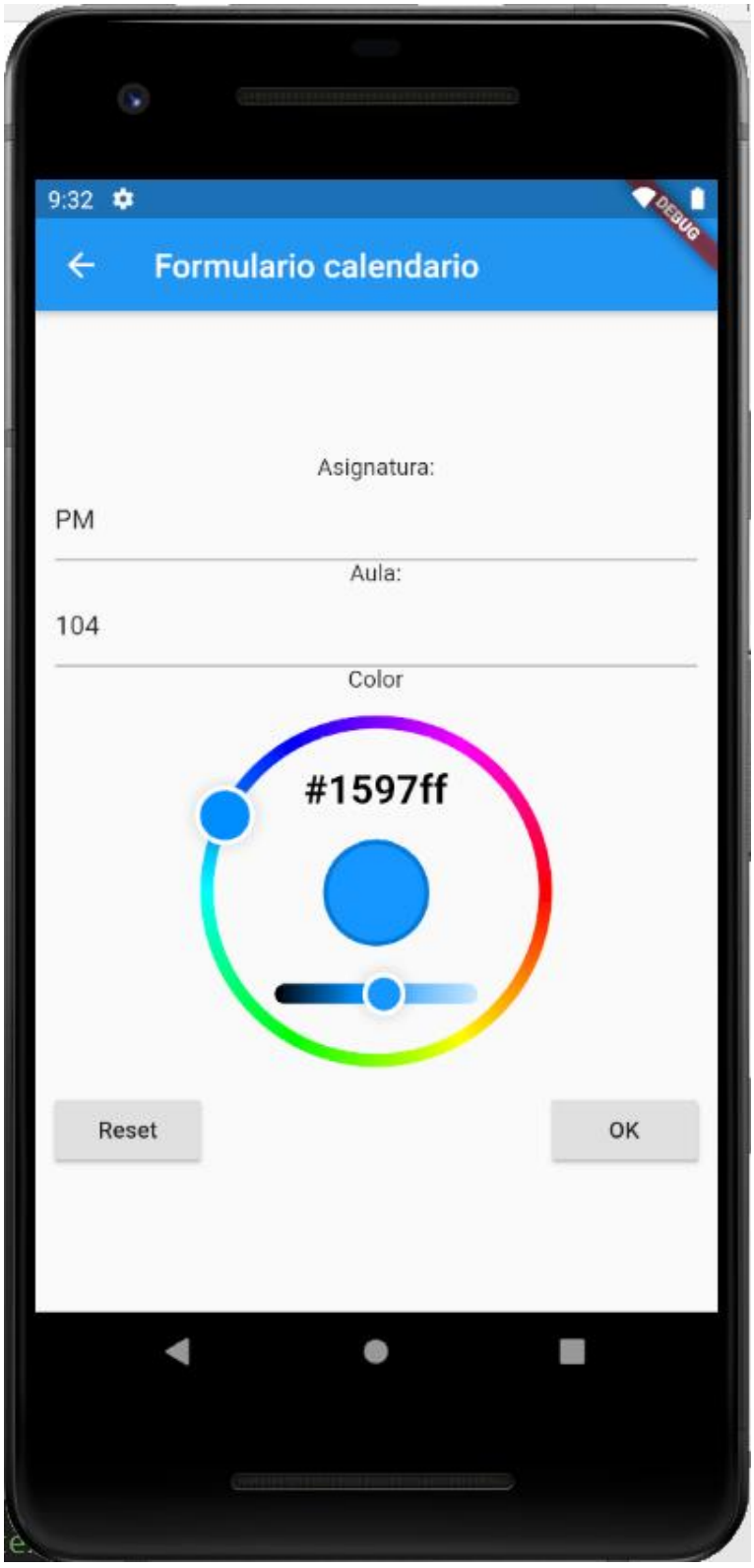
Formulario


9:28
DEBUG
Formulario calendario
Asignatura:
Aula:
Color
#1597ff
Reset
OK

Con datos



Formulario con datos anteriores


9:32
DEBUG
Formulario calendario
Asignatura:
PM
Aula:
104
Color
#1597ff
Reset
OK