

Nombre: Keury Ramirez

Matricula: 2023-1101

Carrera: Desarrollo de software

Materia: Diplomado Desarrollo Web

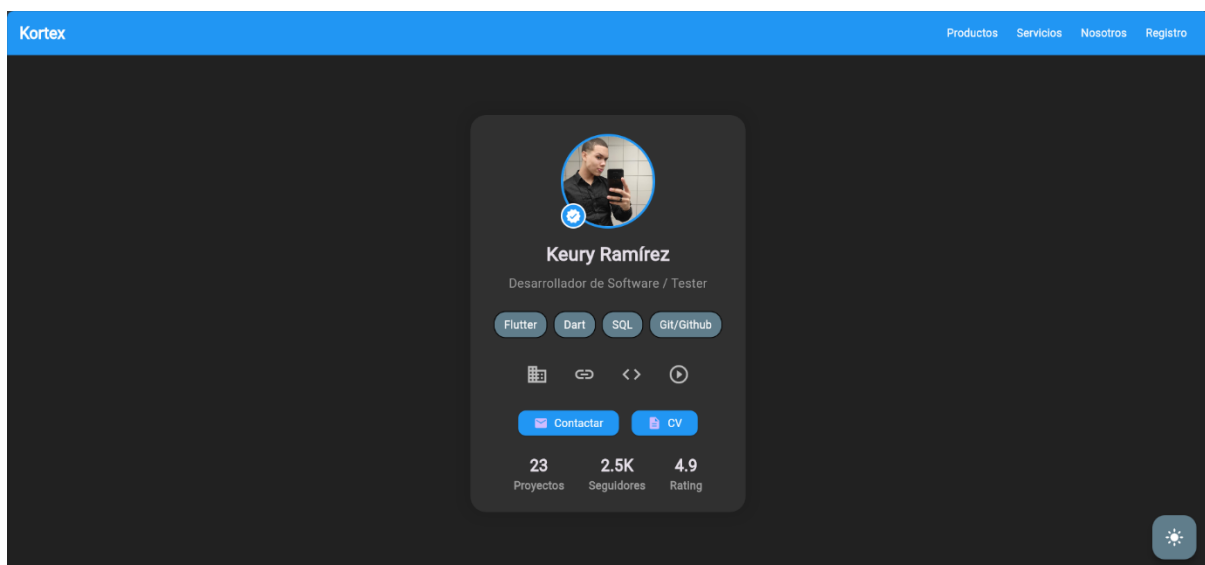
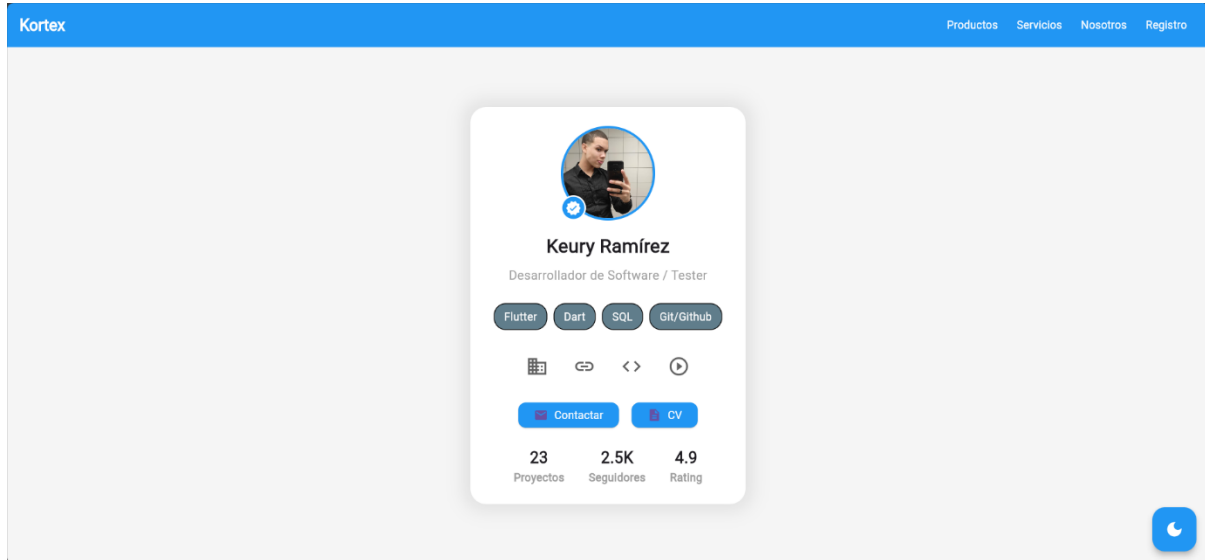
Docente: Adrián Francisco Fondeur
Fernandez

Tema: Practica 2

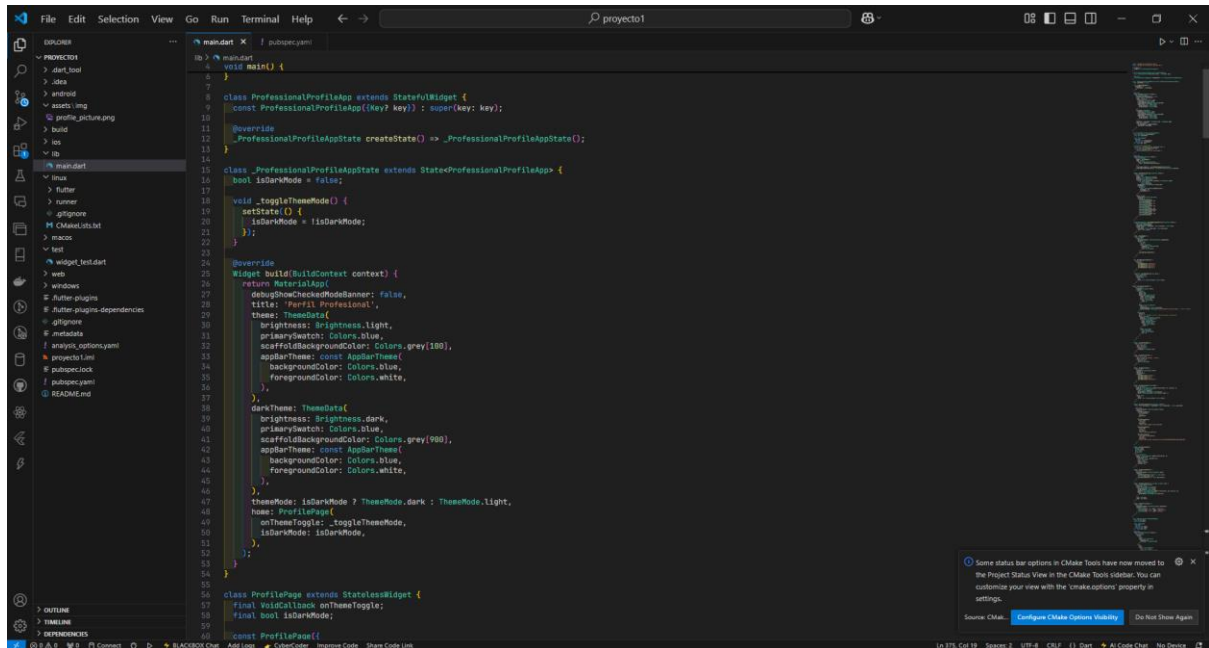
Fecha:
2/2/2025

Práctica 2

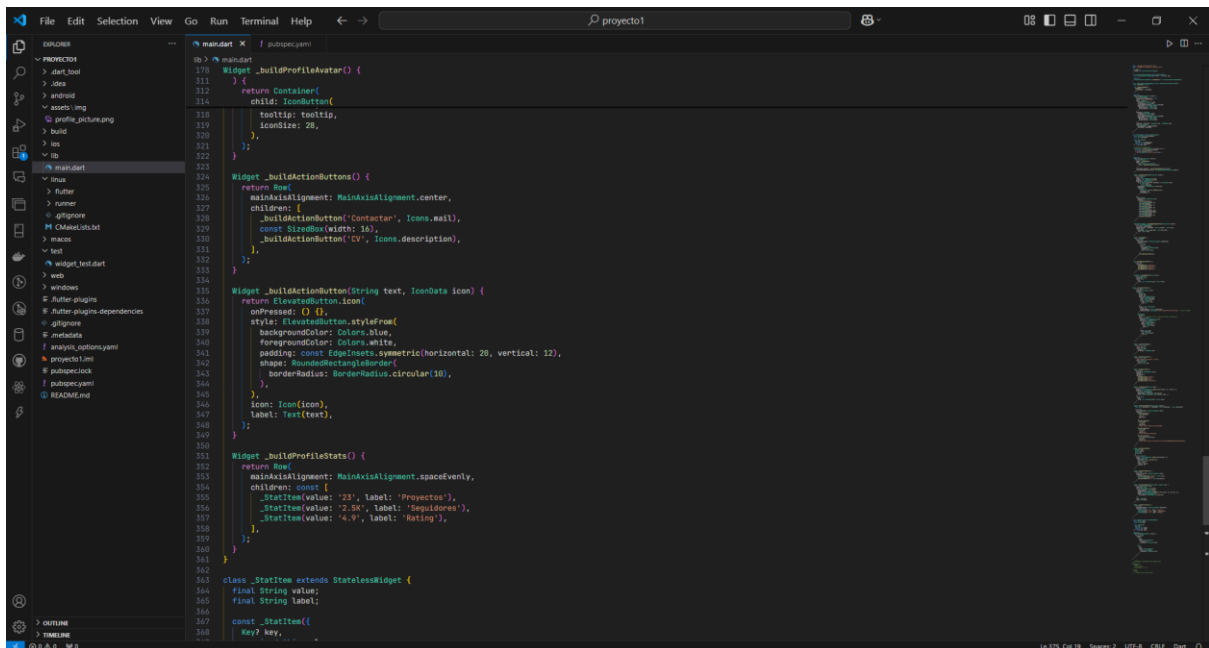
Intenta replicar la siguiente pantalla con sus funcionalidades:



Código del proyecto



```
1 void main() {
2   runApp(MyApp());
3 }
4
5 class ProfessionalProfileApp extends StatefulWidget {
6   const ProfessionalProfileApp({Key? key}) : super(key: key);
7 }
8
9 @override
10 _ProfessionalProfileAppState createState() => _ProfessionalProfileAppState();
11
12
13 class _ProfessionalProfileAppState extends State<ProfessionalProfileApp> {
14   bool isDarkMode = false;
15
16   void toggleThemeMode() {
17     setState(() {
18       isDarkMode = !isDarkMode;
19     });
20   }
21
22   @override
23   Widget build(BuildContext context) {
24     return MaterialApp(
25       debugShowCheckedModeBanner: false,
26       title: 'Perfil Profesional',
27       theme: ThemeData(
28         brightness: Brightness.light,
29         primarySwatch: Colors.blue,
30         scaffoldBackgroundColor: Colors.grey(100),
31         appBarTheme: const AppBarTheme(
32           backgroundColor: Colors.blue,
33           foregroundColor: Colors.white,
34         ),
35       ),
36       darkTheme: ThemeData(
37         brightness: Brightness.dark,
38         primarySwatch: Colors.blue,
39         scaffoldBackgroundColor: Colors.grey(900),
40         appBarTheme: const AppBarTheme(
41           backgroundColor: Colors.blue,
42           foregroundColor: Colors.white,
43         ),
44       ),
45       themeMode: isDarkMode ? ThemeMode.dark : ThemeMode.light,
46     );
47   }
48
49   home: ProfilePage(),
50   onThemeToggle: toggleThemeMode,
51   isDarkMode: isDarkMode,
52 );
53
54 class ProfilePage extends StatelessWidget {
55   final VoidCallback onThemeToggle;
56   final bool isDarkMode;
57
58   const ProfilePage({
59     required VoidCallback onThemeToggle,
60     required bool isDarkMode,
61   }) : onThemeToggle = onThemeToggle, isDarkMode = isDarkMode;
62
63   @override
64   Widget build(BuildContext context) {
65     return Scaffold(
66       appBar: AppBar(
67         title: 'Perfil Profesional',
68         backgroundColor: isDarkMode ? Colors.black : Colors.white,
69         foregroundColor: isDarkMode ? Colors.white : Colors.black,
70       ),
71       body: Center(
72         child: Column(
73           mainAxisAlignment: MainAxisAlignment.center,
74           children: [
75             ProfileAvatar(),
76             Row(
77               mainAxisAlignment: MainAxisAlignment.spaceEvenly,
78               children: [
79                 buildActionButton('Contactar', Icons.email),
80                 buildActionButton('CV', Icons.description),
81               ],
82             ),
83             Stateful(
84               value: '2.5K',
85               label: 'Seguidores',
86             ),
87             Stateful(
88               value: '4.9',
89               label: 'Rating',
90             ),
91           ],
92         ),
93       ),
94     );
95   }
96 }
97
98 class ProfileAvatar extends StatelessWidget {
99   final String value;
100   final String label;
101
102   const ProfileAvatar({
103     required String value,
104     required String label,
105   }) : value = value, label = label;
106
107   @override
108   Widget build(BuildContext context) {
109     return Container(
110       width: 100,
111       height: 100,
112       margin: 10,
113       child: Image.network(
114         'https://i.pravatar.cc/100?img=1',
115       ),
116     );
117   }
118 }
119
120 class Stateful extends StatelessWidget {
121   final String value;
122   final String label;
123
124   const Stateful({
125     required String value,
126     required String label,
127   }) : value = value, label = label;
128
129   @override
130   Widget build(BuildContext context) {
131     return Container(
132       width: 100,
133       height: 100,
134       margin: 10,
135       child: Row(
136         mainAxisAlignment: MainAxisAlignment.spaceEvenly,
137         children: [
138           Text(value),
139           Text(label),
140         ],
141       ),
142     );
143   }
144 }
145
146 class buildActionButton extends StatelessWidget {
147   final String text;
148   final IconData icon;
149
150   const buildActionButton({
151     required String text,
152     required IconData icon,
153   }) : text = text, icon = icon;
154
155   @override
156   Widget build(BuildContext context) {
157     return ElevatedButton(
158       onPressed: () {},
159       style: ElevatedButton.styleFrom(
160         backgroundColor: Colors.blue,
161         foregroundColor: Colors.white,
162         padding: const EdgeInsets.symmetric(horizontal: 20, vertical: 12),
163         shape: RoundedRectangleBorder(
164           borderRadius: BorderRadius.circular(10),
165         ),
166       ),
167       child: Row(
168         mainAxisAlignment: MainAxisAlignment.spaceEvenly,
169         children: [
170           Icon(icon),
171           Text(text),
172         ],
173       ),
174     );
175   }
176 }
```



```
177 Widget buildProfileAvatar() {
178   return Container(
179     width: 100,
180     height: 100,
181     margin: 10,
182     child: Image.network(
183       'https://i.pravatar.cc/100?img=1',
184     ),
185   );
186 }
187
188 Widget buildActionButton(String text, IconData icon) {
189   return ElevatedButton(
190     onPressed: () {},
191     style: ElevatedButton.styleFrom(
192       backgroundColor: Colors.blue,
193       foregroundColor: Colors.white,
194       padding: const EdgeInsets.symmetric(horizontal: 20, vertical: 12),
195       shape: RoundedRectangleBorder(
196         borderRadius: BorderRadius.circular(10),
197       ),
198     ),
199     child: Row(
200       mainAxisAlignment: MainAxisAlignment.spaceEvenly,
201       children: [
202         Icon(icon),
203         Text(text),
204       ],
205     ),
206   );
207 }
208
209 Widget buildProfileState() {
210   return Row(
211     mainAxisAlignment: MainAxisAlignment.spaceEvenly,
212     children: [
213       Stateful(
214         value: '2.5K',
215         label: 'Seguidores',
216       ),
217       Stateful(
218         value: '4.9',
219         label: 'Rating',
220       ),
221     ],
222   );
223 }
224
225 class Stateful extends StatelessWidget {
226   final String value;
227   final String label;
228
229   const Stateful({
230     required String value,
231     required String label,
232   }) : value = value, label = label;
233
234   @override
235   Widget build(BuildContext context) {
236     return Container(
237       width: 100,
238       height: 100,
239       margin: 10,
240       child: Row(
241         mainAxisAlignment: MainAxisAlignment.spaceEvenly,
242         children: [
243           Text(value),
244           Text(label),
245         ],
246       ),
247     );
248   }
249 }
```

Repositorio Github

<https://github.com/K3ury99/Proyecto-2-Flutter>