

Ministerul Educației, Culturii și Cercetării al Republicii Moldova
Universitatea Tehnică a Moldovei
Departamentul Ingineria Software și Automatică

RAPORT

Lucrare de laborator nr.1, varianta II

Disciplina: PAM

Tema: Introducerea in programarea cross platform utilizind Flutter framework. Utilizarea de Componente UI simple (UI controls) si evenimente asupra lor.

A efectuat:

Russu Iulia TI-224

A verificat :

Buza D.

Chișinău 2025

Sarcina de lucru: Varianta 2. Conversie temperatură

Realizați o aplicație mobilă care va efectua conversia temperaturii între Celsius ↔ Fahrenheit ↔ Kelvin.

Input: temperatură numerică (TextField)

UI controls: RadioButton pentru selectarea unității sursă și destinație, ElevatedButton

Output: valoarea convertită afișată pe ecran.

Mersul lucrării:

Pentru Input a fost folosit TextField:

```
111  TextField(  
112    controller: controller,  
113    keyboardType: TextInputType.number,  
114    decoration: const InputDecoration(  
115      labelText: "Introduceți temperatura",  
116      border: OutlineInputBorder(),  
117    ), // InputDecoration  
118  ), // TextField  
119  const SizedBox(height: 16),
```

Figura 1 – TextField

Pentru Selectare a fost folosit RadioButton:

```
87  Text(label, style: const TextStyle(fontWeight: FontWeight.bold)),  
88  Row(  
89    children: [  
90      Radio<Unit>(value: Unit.celsius, groupValue: current, onChanged: onChanged),  
91      const Text("Celsius"),  
92      Radio<Unit>(value: Unit.fahrenheit, groupValue: current, onChanged: onChanged),  
93      const Text("Fahrenheit"),  
94      Radio<Unit>(value: Unit.kelvin, groupValue: current, onChanged: onChanged),  
95      const Text("Kelvin"),  
96    ],
```

Figura 2 – Radio

Pentru butonul de conversie a fost folosit ElevatedButton:

```
125  Center(  
126    child: ElevatedButton(  
127      onPressed: convert,  
128      child: const Text("Convertește"),  
129    ), // ElevatedButton  
130  ), // Center  
131
```

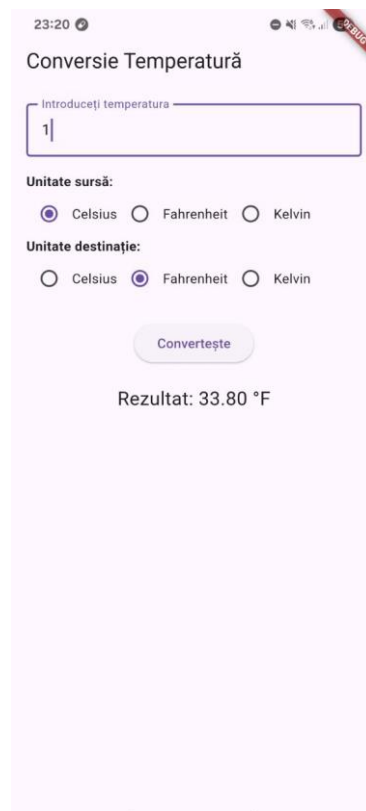
Figura 3 – Radio

Calcularea a fost efectuată cu formule matematice:

```
41 double celsius;  
42 switch (from) {  
43     case Unit.celsius:  
44         celsius = value;  
45         break;  
46     case Unit.fahrenheit:  
47         celsius = (value - 32) * 5 / 9;  
48         break;  
49     case Unit.kelvin:  
50         celsius = value - 273.15;  
51         break;  
52 }  
53  
54 double output;  
55 switch (to) {  
56     case Unit.celsius:  
57         output = celsius;  
58         break;  
59     case Unit.fahrenheit:  
60         output = celsius * 9 / 5 + 32;  
61         break;  
62     case Unit.kelvin:  
63         output = celsius + 273.15;  
64         break;  
65 }  
66
```

Figura 4 – Calcule

Rezultat:



The screenshot shows a mobile application interface for temperature conversion. At the top, the status bar displays the time 23:20 and battery level. The app title is "Conversie Temperatură". Below the title is a text input field labeled "Introduceți temperatura" with the value "1" entered. Underneath the input field are two sections for unit selection. The first section, "Unitate sursă:", has three radio buttons: "Celsius" (selected), "Fahrenheit", and "Kelvin". The second section, "Unitate destinație:", also has three radio buttons: "Celsius", "Fahrenheit" (selected), and "Kelvin". Below these sections is a button labeled "Convertește". At the bottom, the result is displayed as "Rezultat: 33.80 °F".

Figura 5 – Aplicația creată

De asemenea, în laborator am utilizat și alte componente precum **Scaffold**, pentru structurarea paginii principale a aplicației, **AppBar**, pentru afișarea titlului, **Padding** și **Column**, pentru organizarea și alinierea elementelor din interfață, precum și **Text**, pentru afișarea etichetelor și a rezultatului conversiei. Aceste componente suplimentare au contribuit la realizarea unei interfețe clare, ordonate și ușor de utilizat.

Concluzie

În concluzie, acest laborator a demonstrat modul practic de realizare a unei aplicații mobile simple în Flutter, capabilă să efectueze conversia temperaturii între diferite unități. Prin utilizarea widgeturilor de bază, a controlului stării și a elementelor de interfață precum **TextField**, **RadioButton** și **ElevatedButton**, am înțeles mai bine structura unei aplicații Flutter și modul în care logica poate fi integrată cu designul pentru a oferi o experiență interactivă și intuitivă utilizatorului.