



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ _____ «Информатика и системы управления»

КАФЕДРА _____ «Теоретическая информатика и компьютерные технологии»

Рубежный контроль № 1
по курсу «Разработка мобильных приложений»
«Проверка усвоенного материала по теме форматирования
контента виджетов и CustomPaint»

Студентка группы ИУ9-72Б Самохвалова П. С.

Преподаватель Посевин Д. П.

Москва 2023

1 Цель работы

Реализовать мобильное приложение выполняющее построение простейших геометрических фигур.

2 Задание

Приложение должно представлять из себя форму ввода входных данных и виджет вывода результата. Приложение должно запускаться на мобильном устройстве.

Индивидуальный вариант:

Входные данные: координаты вершин треугольника в пространстве. Результат: вид проекции на плоскость xOy .

3 Практическая реализация

Исходный код программы представлен в листинге 1.

Листинг 1: Мобильное приложение выполняющее построение простейших геометрических фигур

```
1 import 'package:flutter/material.dart';
2 import 'dart:ui' as ui;
3
4 void main() {
5   runApp(MyApp());
6 }
7
8 class Vertex {
9   double x;
10  double y;
11  double z;
12
13  Vertex(this.x, this.y, this.z);
14 }
15
16 class TriangleProjection extends StatefulWidget {
17   @override
18   _TriangleProjectionState createState() => _TriangleProjectionState();
19 }
20
```

```

21 class _TriangleProjectionState extends State<TriangleProjection> {
22   final vertex1Controller = TextEditingController();
23   final vertex2Controller = TextEditingController();
24   final vertex3Controller = TextEditingController();
25   List<Vertex> vertices = [];
26
27   @override
28   void initState() {
29     super.initState();
30     vertex1Controller.text = "0,0,0";
31     vertex2Controller.text = "100,0,0";
32     vertex3Controller.text = "0,100,0";
33     _updateVertices();
34   }
35
36   void _updateVertices() {
37     vertices.clear();
38     vertices.add(_parseVertex(vertex1Controller.text));
39     vertices.add(_parseVertex(vertex2Controller.text));
40     vertices.add(_parseVertex(vertex3Controller.text));
41     setState(() {});
42   }
43
44   Vertex _parseVertex(String vertexString) {
45     List<String> values = vertexString.split(",");
46     double x = double.parse(values[0]);
47     double y = double.parse(values[1]);
48     double z = double.parse(values[2]);
49     return Vertex(x, y, z);
50   }
51
52   @override
53   Widget build(BuildContext context) {
54     return MaterialApp(
55       home: Scaffold(
56         appBar: AppBar(
57           title: Text('Triangle Projection'),
58         ),
59         body: Padding(
60           padding: EdgeInsets.all(16.0),
61           child: Column(
62             crossAxisAlignment: CrossAxisAlignment.start,
63             children: <Widget>[
64               TextField(
65                 decoration: InputDecoration(labelText: 'Vertex 1 (x,y,z)

```

```

66         controller: vertex1Controller ,
67         onChanged: (_) => _updateVertices() ,
68     ),
69     TextField(
70         decoration: InputDecoration(labelText: 'Vertex 2 (x,y,z)
71     '),
72         controller: vertex2Controller ,
73         onChanged: (_) => _updateVertices() ,
74     ),
75     TextField(
76         decoration: InputDecoration(labelText: 'Vertex 3 (x,y,z)
77     '),
78         controller: vertex3Controller ,
79         onChanged: (_) => _updateVertices() ,
80     ),
81     SizedBox(height: 16.0) ,
82     Text(
83         'Projected Triangle:',
84         style: TextStyle(fontSize: 18.0, fontWeight: FontWeight.
85         bold) ,
86     ),
87     ),
88     SizedBox(height: 8.0) ,
89     Container(
90         width: 300,
91         height: 300,
92         child: CustomPaint(
93             painter: TrianglePainter(vertices) ,
94         ),
95     ),
96 ],
97 ),
98 );
99 }
100 }
101
102 class TrianglePainter extends CustomPainter {
103     List<Vertex> vertices;
104
105     TrianglePainter(this.vertices);
106
107     @override
108     void paint(Canvas canvas, Size size) {
109         Paint paint = Paint()
110             ..color = Colors.blue

```

```

109         ..style = PaintingStyle.fill;
110
111     Path path = Path();
112     path.moveTo(vertices[0].x, vertices[0].y);
113     path.lineTo(vertices[1].x, vertices[1].y);
114     path.lineTo(vertices[2].x, vertices[2].y);
115     path.close();
116
117     canvas.drawPath(path, paint);
118 }
119
120 @override
121 bool shouldRepaint(TrianglePainter oldDelegate) => true;
122 }
123
124 class MyApp extends StatelessWidget {
125     @override
126     Widget build(BuildContext context) {
127         return MaterialApp(
128             home: TriangleProjection(),
129         );
130     }
131 }

```

4 Результаты

Результаты работы программы представлены на рисунках 1 – 3.

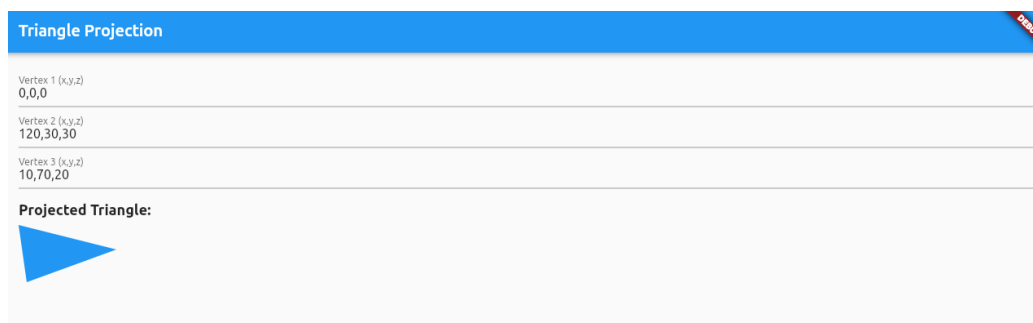


Рис. 1 — Результаты

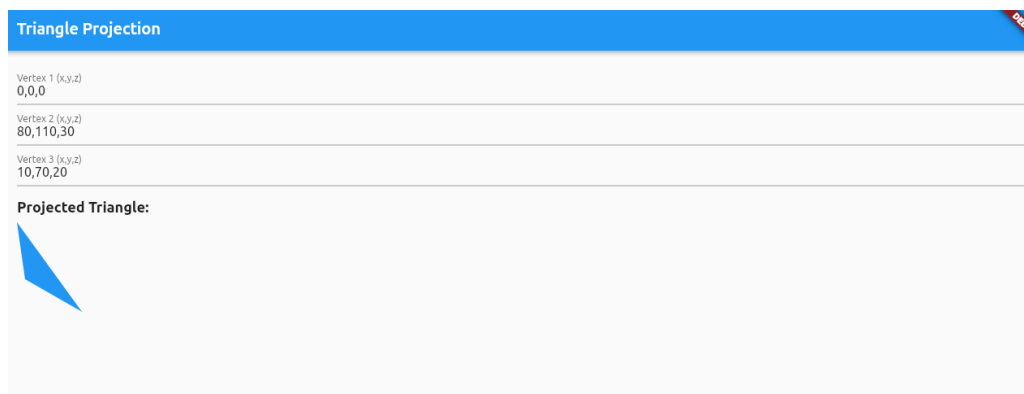


Рис. 2 — Результаты

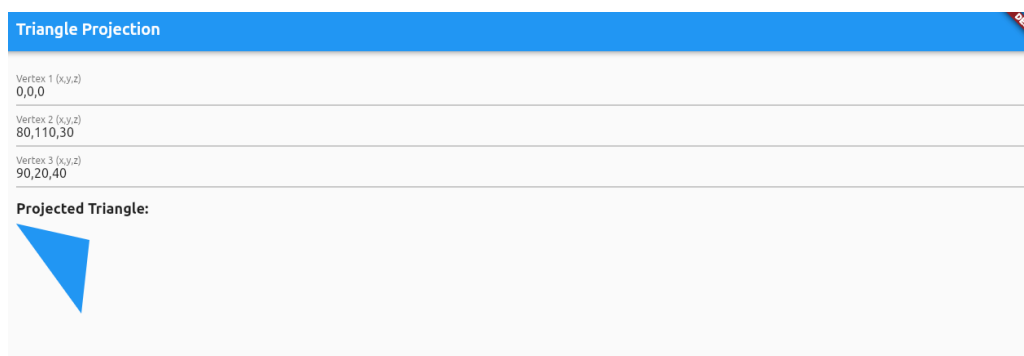


Рис. 3 — Результаты

5 Выводы

В результате выполнения рубежного контроля было реализовано мобильное приложение выполняющее построение простейших геометрических фигур.