



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»

Лабораторная работа № 3
по курсу «Разработка мобильных приложений»
«Проверка усвоенного материала по теме использования
библиотек работы с 3D объектами »

Студент группы ИУ9-71Б Яровикова А. С.

Преподаватель Посевин Д. П.

Москва 2023

1 Цель

Реализовать мобильное приложение выводящее трехмерный объект по вариантам. Использование библиотеки на усмотрение программиста из рассмотренных на лекции.

2 Задание

Реализовать приложение моделирования движения глаз человека влево и вправо, модель головы человека можно взять из лекции. Движение глаз реализуется ползунком. Голова должна вращаться.

3 Реализация

Исходный код представлен в листинге 1.

Листинг 1: main.dart

```
1 import 'dart:ui' as ui;
2 import 'dart:math' as math;
3 import 'package:flutter/material.dart';
4 import 'package:flutter_cube/flutter_cube.dart';
5
6 void main() => runApp(MyApp());
7
8 class MyApp extends StatelessWidget {
9   @override
10  Widget build(BuildContext context) {
11    return MaterialApp(
12      debugShowCheckedModeBanner: false,
13      title: 'FLY4 - skull with eyeballs',
14      theme: ThemeData.dark(),
15      home: MyHomePage(title: 'Skull'),
16    );
17  }
18 }
19
20 class MyHomePage extends StatefulWidget {
21   MyHomePage({Key? key, this.title}) : super(key: key);
22
23   final String? title;
```

```

25  @override
26  _MyHomePageState createState() => _MyHomePageState();
27  }
28
29  class _MyHomePageState extends State<MyHomePage> with
    SingleTickerProviderStateMixin {
30    late Scene _scene;
31    Object? _skull_bottom;
32    Object? _skull_top;
33    Object? _eye1;
34    Object? _eye2;
35    late AnimationController _controller;
36    double _ambient = 0.1;
37    double _diffuse = 0.8;
38    double _specular = 0.5;
39    // double _shininess = 0.0;
40    double top_pos = 0.3;
41    double eyes_z = 0.0;
42    double eyes_x = 0.0;
43
44    void _onSceneCreated(Scene scene) {
45      _scene = scene;
46      scene.camera.position.y = 7;
47      scene.camera.position.z = 7;
48      scene.light.position.setFrom(Vector3(0, 0, 10));
49      scene.light.setColor(Colors.white, _ambient, _diffuse, _specular);
50      // eye 1
51      _eye1 = Object(
52        position: Vector3(-0.5, 1.4, 0.9),
53        scale: Vector3(0.6, 0.6, 0.6),
54        rotation: Vector3(0.0, 0.0, 0.0),
55        backfaceCulling: false,
56        lighting: true,
57        fileName: 'assets/Eyeball/eyeball.obj');
58      scene.world.add(_eye1!);
59
60      // eye 2
61      _eye2 = Object(
62        position: Vector3(0.5, 1.4, 0.9),
63        scale: Vector3(0.6, 0.6, 0.6),
64        rotation: Vector3(0.0, 0.0, 0.0),
65        // backfaceCulling: false,
66        lighting: true,
67        fileName: 'assets/Eyeball/eyeball.obj');
68      scene.world.add(_eye2!);
69

```

```

70 // skull
71 _skull_bottom = Object(
72     position: Vector3(0.0, 0.0, 0.0),
73     scale: Vector3(5, 5, 5),
74     rotation: Vector3(-90.0, 0.0, 0.0),
75     backfaceCulling: false,
76     lighting: true,
77     // fileName: 'assets/face.obj'
78     fileName: '../assets/12140_Skull_v3_L2.obj'
79 );
80
81 scene.world.add(_skull_bottom!);
82
83 _skull_top = Object(
84     position: Vector3(0.0, 0.3, 0),
85     scale: Vector3(5, 5, 5),
86     rotation: Vector3(-90.0, 0.0, 0.0),
87     backfaceCulling: false,
88     // lighting: true,
89     fileName: 'assets/skull/1.obj');
90 // scene.world.add(_skull_top!);
91 }
92
93
94 @override
95 void dispose() {
96     _controller.dispose();
97     super.dispose();
98 }
99
100 @override
101 Widget build(BuildContext context) {
102     return Scaffold(
103         appBar: AppBar(
104             title: Text(widget.title!),
105         ),
106         body: Stack(
107             children: <Widget>[
108                 Cube(onSceneCreated: _onSceneCreated),
109                 Column(
110                     mainAxisAlignment: MainAxisAlignment.end,
111                     children: <Widget>[
112
113                         Row(
114                             mainAxisAlignment: MainAxisAlignment.end,
115                             children: <Widget>[

```

```

116         Flexible(flex: 2, child: Text('Eyes left/right')),
117     Flexible(
118         flex: 8,
119         child: Slider(
120             value: eyes_z,
121             min: -45.0,
122             max: 45.0,
123             divisions: 100,
124             onChanged: (value) {
125                 setState(() {
126                     eyes_z = value;
127                     _eye1!.rotation.y = eyes_z;
128                     _eye2!.rotation.y = eyes_z;
129                     _eye1!.updateTransform();
130                     _eye2!.updateTransform();
131                     _scene.update();
132                 });
133             },
134         ),
135     ),
136 ],
137 ),
138 Row(
139     mainAxisAlignment: MainAxisAlignment.end,
140     children: <Widget>[
141         Flexible(flex: 2, child: Text('Eyes up/down')),
142         Flexible(
143             flex: 8,
144             child: Slider(
145                 value: eyes_x,
146                 min: -45,
147                 max: 45,
148                 divisions: 100,
149                 onChanged: (value) {
150                     setState(() {
151                         eyes_x = value;
152                         _eye1!.rotation.x = eyes_x;
153                         _eye2!.rotation.x = eyes_x;
154                         _eye1!.updateTransform();
155                         _eye2!.updateTransform();
156                         _scene.update();
157                     });
158                 },
159             ),
160         ),
161     ],

```

```

162         ),
163         Row(
164             mainAxisAlignment: MainAxisAlignment.end,
165             children: <Widget>[
166                 Flexible(flex: 2, child: Text('diffuse')),
167                 Flexible(
168                     flex: 8,
169                     child: Slider(
170                         value: _diffuse,
171                         min: 0.0,
172                         max: 1.0,
173                         divisions: 100,
174                         onChanged: (value) {
175                             setState(() {
176                                 _diffuse = value;
177                                 _scene.light.setColor(Colors.white, _ambient,
178                                     _diffuse, _specular);
179                             });
180                         },
181                     ),
182                 ],
183             ),
184         ],
185     ),
186 ],
187 ),
188 );
189 }
190 }

```

4 Результаты

Результат представлен на рисунках 1 - 5.

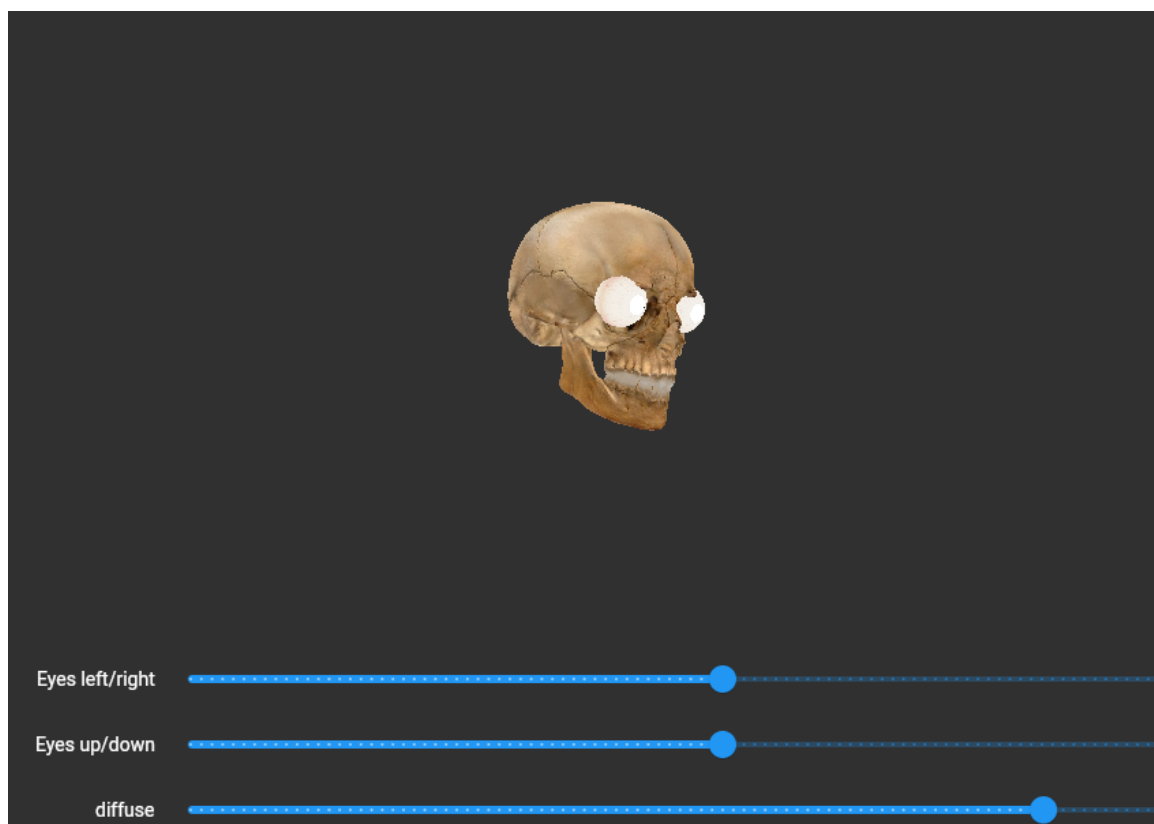


Рис. 1 — Поворот головы

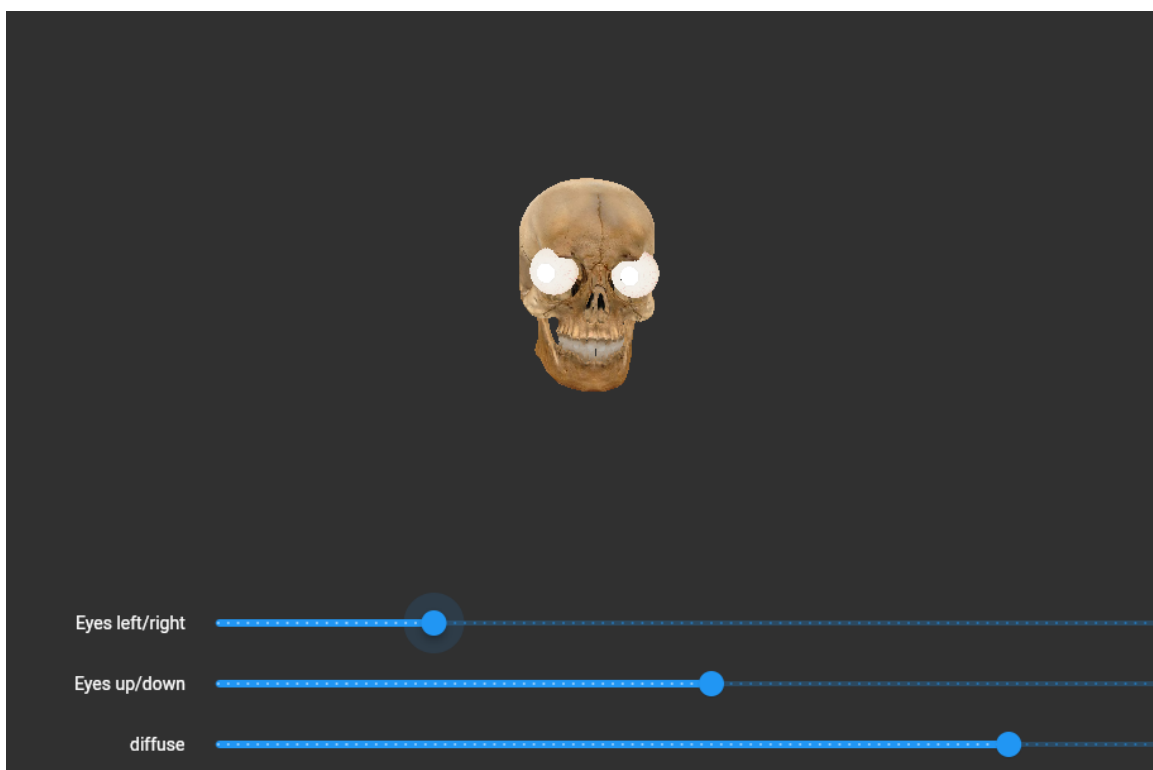


Рис. 2 — Глаза влево

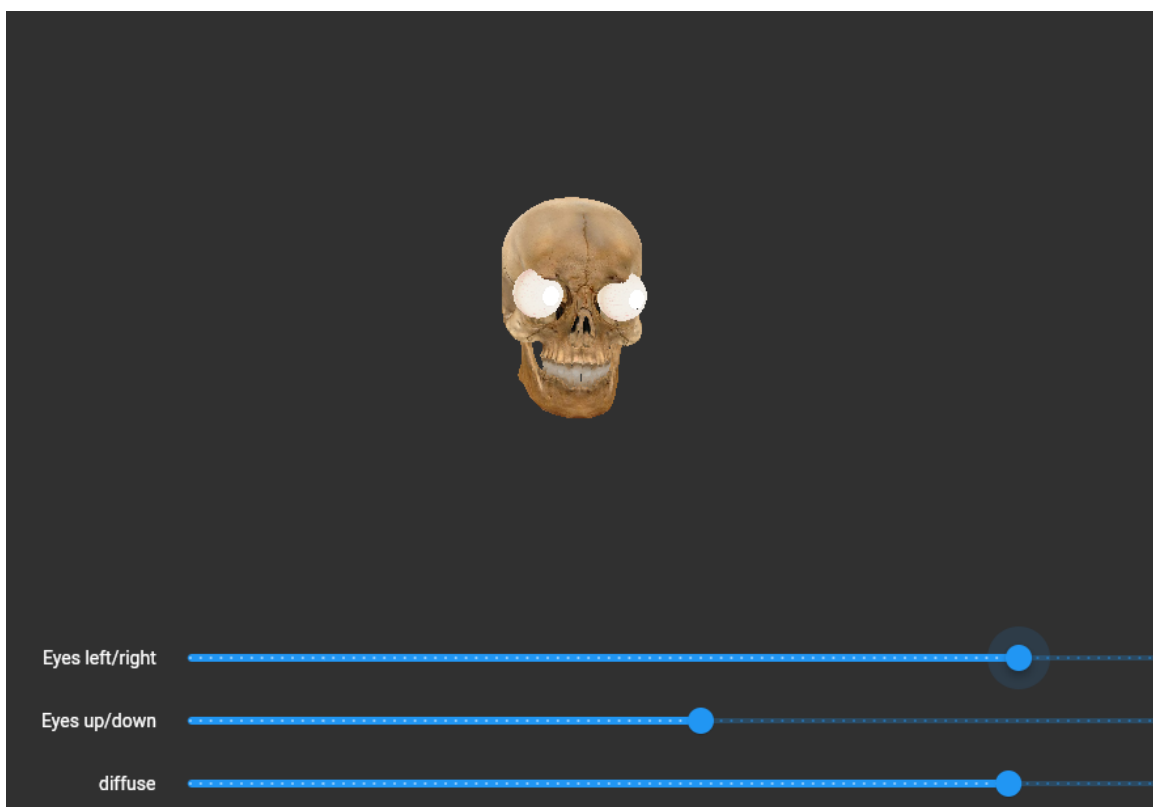


Рис. 3 — Глаза вправо

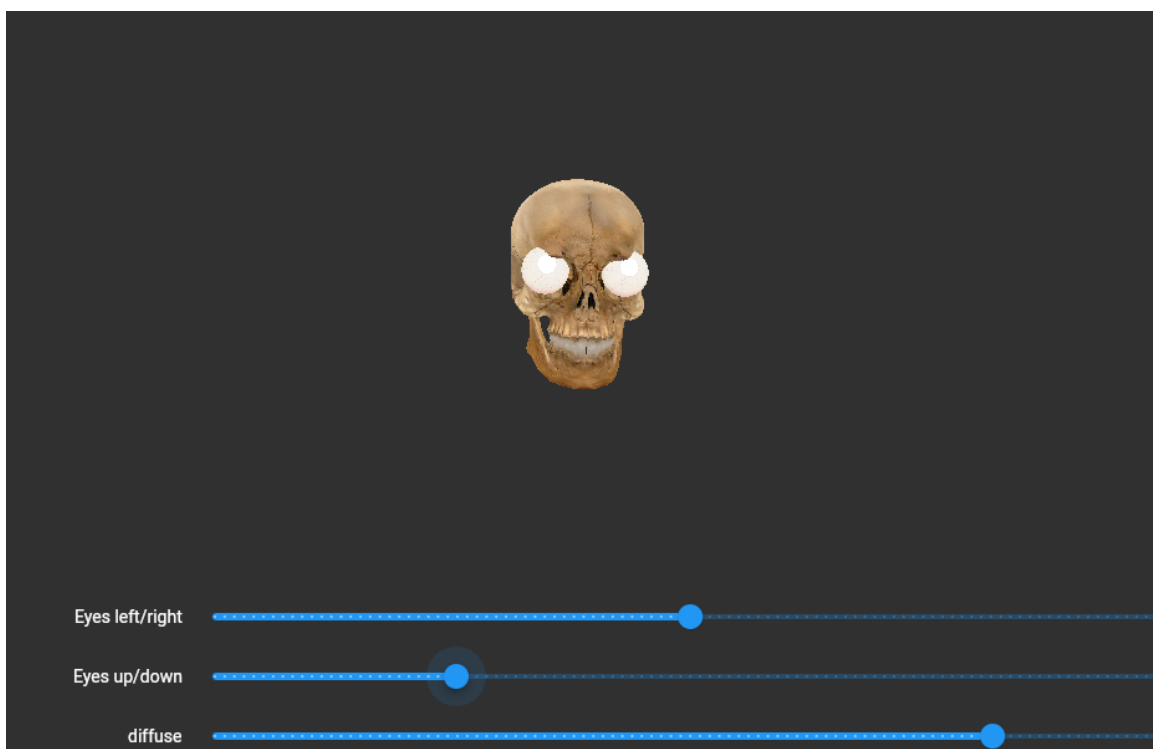


Рис. 4 — Глаза вверх

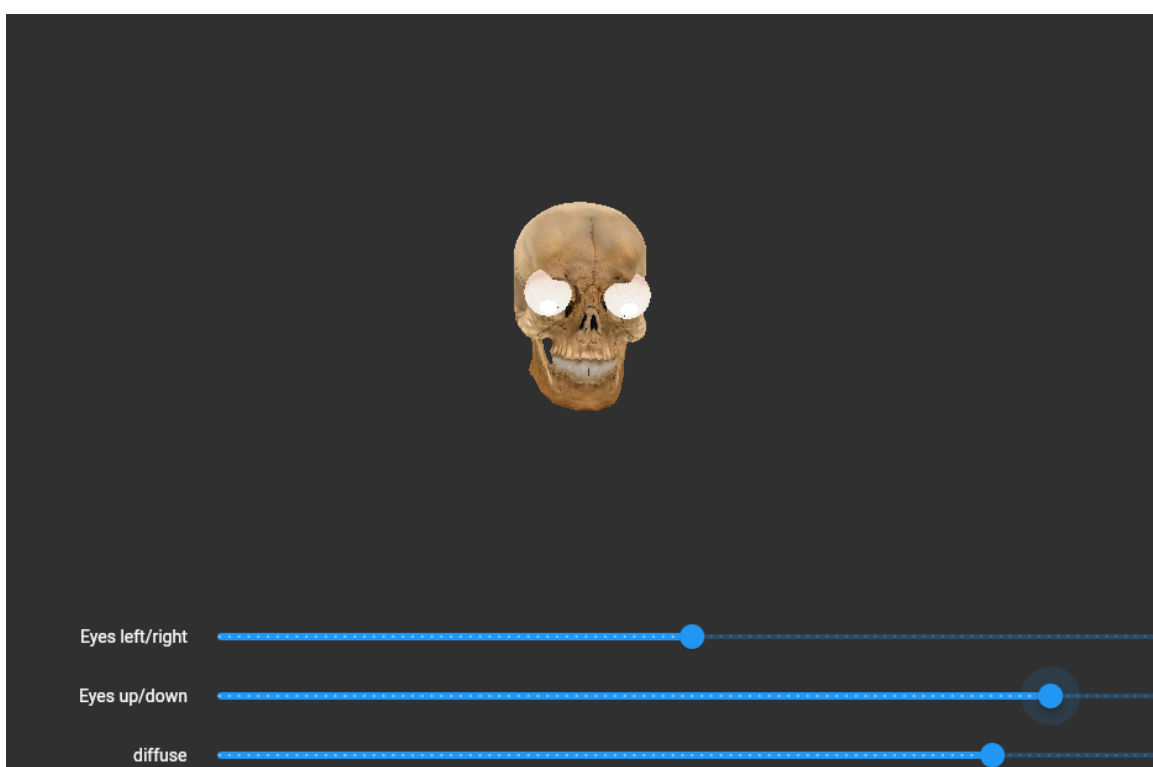


Рис. 5 — Глаза вниз

5 Выводы

В результате лабораторной работы было создано приложение с графическим пользовательским интерфейсом на языке программирования Dart. Приложение позволяет отображать трехмерный объект в виде головы (черепа) с возможностью вращать голову и двигать глаза человека влево, вправо, вверх и вниз.