



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»**

Студентка группы ИУ9-72Б Самохвалова П. С.

Преподаватель Посевин Д. П.

*Москва 2023*

# 1 Задание

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

Индивидуальный вариант

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

## 2 Практическая реализация

Исходный код программы представлен в листинге 1.

Листинг 1: Работа с трехмерными объектами

```
1 import 'package:flutter/material.dart';
2 import 'package:flutter_cube/flutter_cube.dart';
3
4 void main() => runApp(MyApp());
5
6 class MyApp extends StatelessWidget {
7   @override
8   Widget build(BuildContext context) {
9     return MaterialApp(
10       title: 'Flutter Cube',
11       theme: ThemeData.dark(),
12       home: MyHomePage(title: 'Flutter Cube Home Page'),
13     );
14   }
15 }
16
17 class MyHomePage extends StatefulWidget {
18   MyHomePage({Key? key, this.title}) : super(key: key);
19
20   final String? title;
21
22   @override
23   _MyHomePageState createState() => _MyHomePageState();
24 }
25
26 class _MyHomePageState extends State<MyHomePage> with
    SingleTickerProviderStateMixin {
```

```

27 late Scene _scene;
28 Object? _bunny;
29 Object? _eyes;
30 late AnimationController _controller;
31 double _ambient = 0.1;
32 double _diffuse = 0.8;
33 double _specular = 0.5;
34 double _shininess = 0.0;
35
36 double x1 = 0.0;
37 double y1 = 0.0;
38
39 final Object cube1 = Object(
40     scale: Vector3(0.5, 0.5, 0.5),
41     position: Vector3(0.2, 0.9, 0.5)..scale(3),
42     lighting: true,
43     fileName: 'assets/eyeball/eyeball.obj',
44 );
45
46 final Object cube2 = Object(
47     scale: Vector3(0.5, 0.5, 0.5),
48     position: Vector3(-0.2, 0.9, 0.5)..scale(3),
49     lighting: true,
50     fileName: 'assets/eyeball/eyeball.obj',
51 );
52
53 void _onSceneCreated(Scene scene) {
54     _scene = scene;
55     scene.camera.position.z = 10;
56     scene.light.position.setFrom(Vector3(0, 10, 10));
57     // scene.light.setColor(Colors.white, _ambient, _diffuse, _specular)
58     ;
59     _bunny = Object(position: Vector3(0, 0.0, 0), scale: Vector3(10.0,
60     10.0, 10.0), lighting: true, fileName: 'assets/face/face.obj');
61     _eyes = Object();
62     _eyes!.add(cube1);
63     _eyes!.add(cube2);
64
65     _bunny!.lighting = true;
66     _eyes!.lighting = false;
67
68     scene.world.add(_bunny!);
69     scene.world.add(_eyes!);
70 }
71
72 @override

```

```

71 void initState() {
72     super.initState();
73     _controller = AnimationController(duration: Duration(milliseconds:
74         30000), vsync: this)
75         ..addListener(() {
76             // if (_bunny != null) {
77                 // _bunny!.rotation.y = _controller.value * 360;
78                 // _bunny!.updateTransform();
79                 // _scene.update();
80             // }
81             // if (_eyes != null) {
82                 // _eyes!.rotation.y = _controller.value * 360;
83                 // _eyes!.updateTransform();
84                 // _scene.update();
85             // }
86         })
87         ..repeat();
88 }
89
90 @override
91 void dispose() {
92     _controller.dispose();
93     super.dispose();
94 }
95
96 @override
97 Widget build(BuildContext context) {
98     return Scaffold(
99         appBar: AppBar(
100             title: Text(widget.title!),
101         ),
102         body: Stack(
103             children: <Widget>[
104                 Cube(onSceneCreated: _onSceneCreated),
105                 Column(
106                     mainAxisAlignment: MainAxisAlignment.end,
107                     children: <Widget>[
108                         Row(
109                             mainAxisAlignment: MainAxisAlignment.end,
110                             children: <Widget>[
111                                 Flexible(flex: 2, child: Text('diffuse')),
112                                 Flexible(
113                                     flex: 8,
114                                     child: Slider(
115                                         value: _diffuse,

```

```

116         min: 0.0,
117         max: 1.0,
118         divisions: 100,
119         onChanged: (value) {
120             setState(() {
121                 _diffuse = value;
122                 _scene.light.setColor(Colors.white, _ambient,
123                 _diffuse, _specular);
124             });
125         },
126     ),
127 ],
128 ),
129 Row(
130     mainAxisAlignment: MainAxisAlignment.end,
131     children: <Widget>[
132         Flexible(flex: 2, child: Text('Eye rotation up and
133         down'))),
134         Flexible(
135             flex: 8,
136             child: Slider(
137                 value: x1,
138                 min: -45.0,
139                 max: 45.0,
140                 // divisions: 32,
141                 onChanged: (value) {
142                     setState(() {
143                         x1 = value;
144                         cube1.rotation.x = x1;
145                         cube1.updateTransform();
146                         cube2.rotation.x = x1;
147                         cube2.updateTransform();
148                     });
149                 },
150             // child: Slider(
151             //     value: _specular,
152             //     min: 0.0,
153             //     max: 1.0,
154             //     divisions: 100,
155             //     onChanged: (value) {
156             //         setState(() {
157             //             _specular = value;
158             //             _scene.light.setColor(Colors.white,
159             _ambient, _diffuse, _specular);

```

```

159         //     });
160         //   },
161         // ),
162     ),
163 ],
164 ),
165 Row(
166     mainAxisAlignment: MainAxisAlignment.end,
167     children: <Widget>[
168         Flexible(flex: 2, child: Text('Eye rotation left and
right ')),
169         Flexible(
170             flex: 8,
171             child: Slider(
172                 value: y1,
173                 min: -45.0,
174                 max: 45.0,
175                 // divisions: 32,
176                 onChanged: (value) {
177                     setState(() {
178                         y1 = value;
179                         cubel.rotation.y = y1;
180                         cubel.updateTransform();
181                         cube2.rotation.y = y1;
182                         cube2.updateTransform();
183                     });
184                 },
185             ),
186             // child: Slider(
187             //     value: _shininess,
188             //     min: 0.0,
189             //     max: 32.0,
190             //     divisions: 32,
191             //     onChanged: (value) {
192             //         setState(() {
193             //             _shininess = value;
194             //             _bunny!.mesh.material.shininess =
_shininess;
195             //             _eyes!.mesh.material.shininess = _shininess
;
196             //         });
197             //     },
198             // ),
199         ),
200     ],
201 ),

```

```

202         ],
203     ),
204     ],
205     ),
206 );
207 }
208 }

```

### 3 Результаты

Результаты работы программы представлены на рисунках 1 – 4.

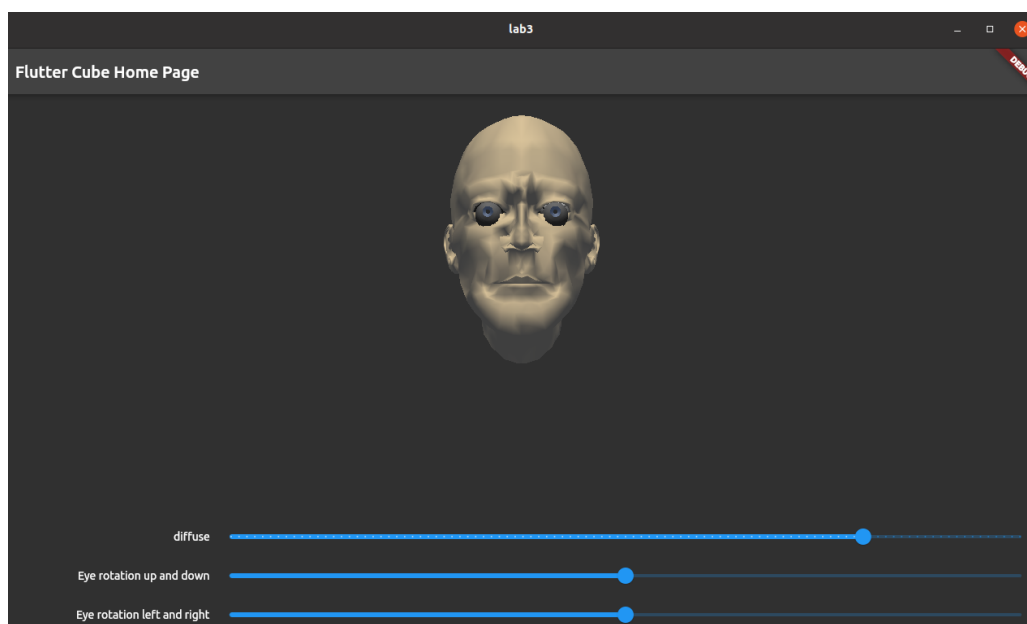


Рис. 1 — Приложение

### 4 Выводы

В результате выполнения лабораторной работы были получены навыки работы с трехмерными объектами, было реализовано приложение моделирования движения глаз человека влево и вправо.

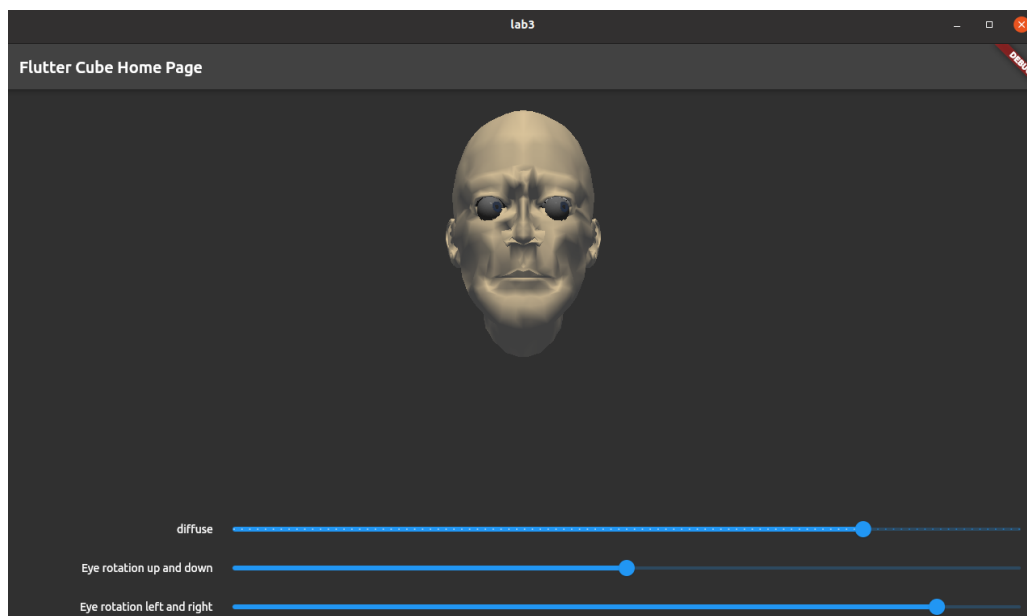


Рис. 2 — Поворот глаз вправо

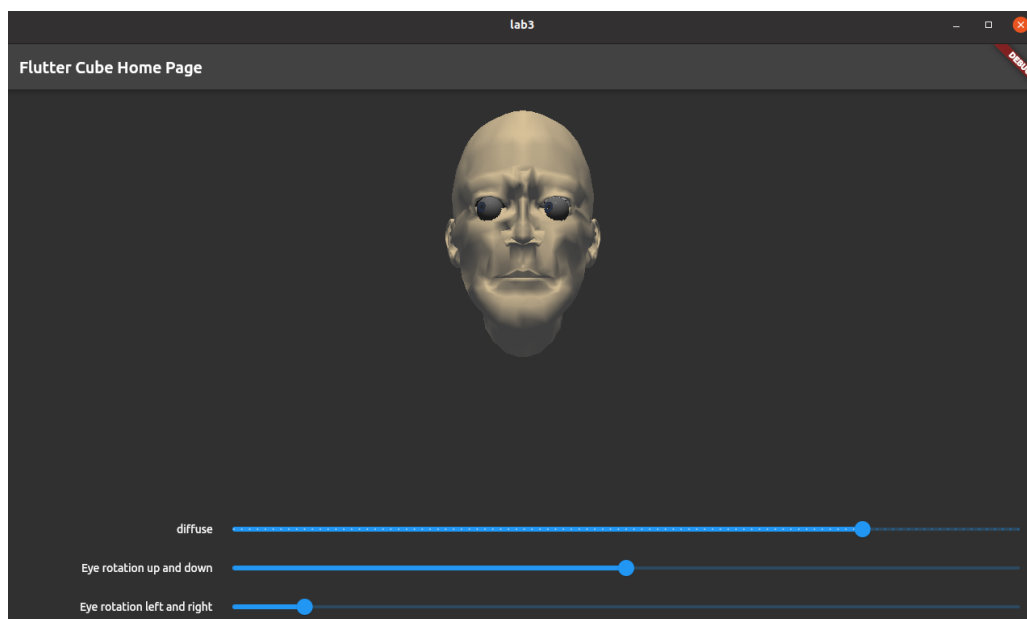


Рис. 3 — Поворот глаз влево



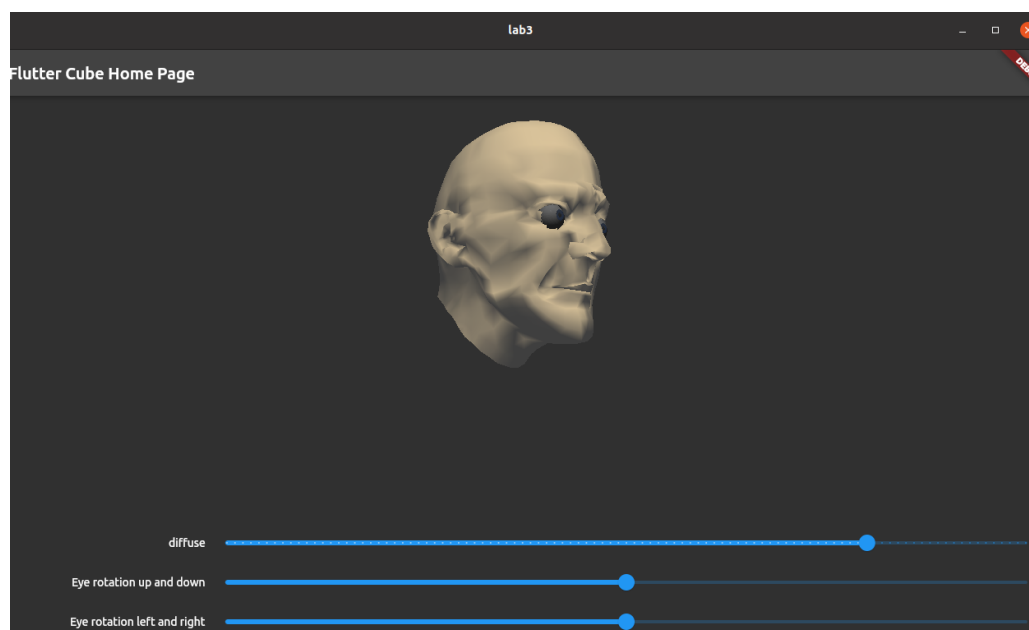


Рис. 4 — Вращение головы