

Лабораторная работа №9

АУДИО ПЛЕЕР

Цель работы: создать приложение с возможностью выбора аудио из библиотеки и управления его воспроизведением.

```
Navigation.kt  Lab_9_Screen.kt x
34  @Composable
35  fun Lab_9_Screen(navController: NavController) {
36      var selectedAudioUri by remember { mutableStateOf<Uri?>(value: null) }
37
38      Column(
39          modifier = Modifier
40              .fillMaxSize()
41              .padding(16.dp),
42          verticalArrangement = Arrangement.Center
43      ) { this: ColumnScope
44          AudioPicker(onAudioSelected = { uri ->
45              selectedAudioUri = uri
46          })
47
48          Spacer(modifier = Modifier.height(16.dp))
49
50          AudioPlayer(audioUri = selectedAudioUri)
51      }
52  }
53
54  @OptIn(UnstableApi::class)
55  @Composable
56  fun AudioPlayer(audioUri: Uri?) {
57      val context = LocalContext.current
58      val player = remember { ExoPlayer.Builder(context).build() }
59
60      LaunchedEffect(audioUri) { this: CoroutineScope
61          audioUri?.let { uri ->
62              player.setMediaItem(MediaItem.fromUri(uri))
63              player.prepare()
64          }
65      }
66
67      DisposableEffect(Unit) { this: DisposableEffectScope
68          onDispose {
69              player.release()
70          }
71      }
72
73      Column(
74          modifier = Modifier
75              .fillMaxSize()
76              .padding(16.dp),
77          horizontalAlignment = Alignment.CenterHorizontally,
```

Рис 9.1. Код лабораторной работы

```
Navigation.kt  Lab_9_Screen.kt x
56 fun AudioPlayer(audioUri: Uri?) {
72
73     Column(
74         modifier = Modifier
75             .fillMaxSize()
76             .padding(16.dp),
77         horizontalAlignment = Alignment.CenterHorizontally,
78         verticalArrangement = Arrangement.Center
79     ) { this: ColumnScope
80         if (audioUri != null) {
81             AndroidView(
82                 factory = { ctx ->
83                     PlayerControlView(ctx).apply { this: PlayerControlView
84                         setPlayer(player)
85                     }
86                 },
87                 modifier = Modifier.fillMaxWidth()
88             )
89         } else {
90             Text(text: "Аудиофайл не выбран")
91         }
92     }
93 }
94
95 @Composable
96 fun AudioPicker(onAudioSelected: (Uri?) -> Unit) {
97     val audioPickerLauncher = rememberLauncherForActivityResult(
98         contract = ActivityResultContracts.GetContent()
99     ) { uri: Uri? ->
100         onAudioSelected(uri)
101     }
102
103     Button(
104         onClick = {
105             audioPickerLauncher.launch(input: "audio/*")
106         }
107     ) { this: RowScope
108         Text(text: "Выбрать аудиофайл")
109     }
```

Рис 9.2. Код лабораторной работы

Результаты работы программы:



Рис 9.3. Запуск

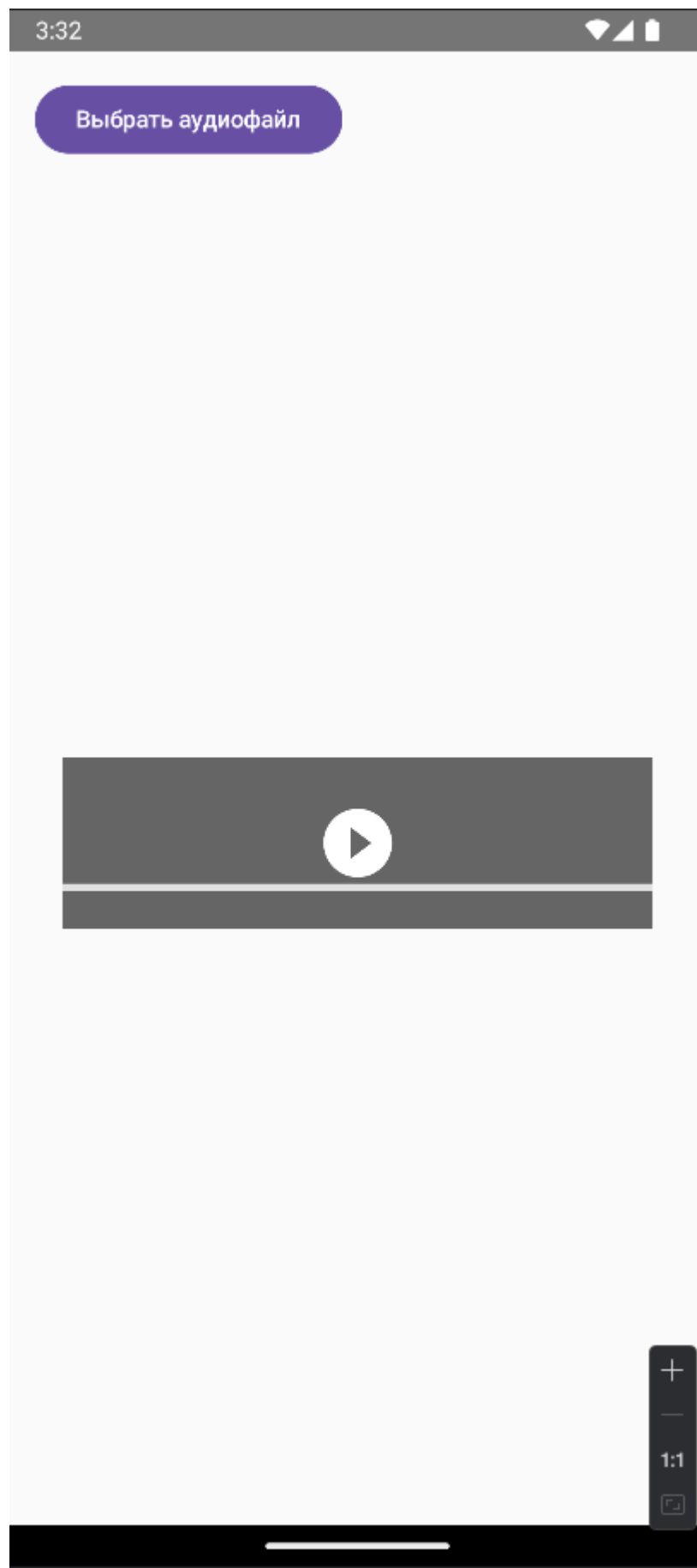


Рис 9.4. Отображение аудио плеера

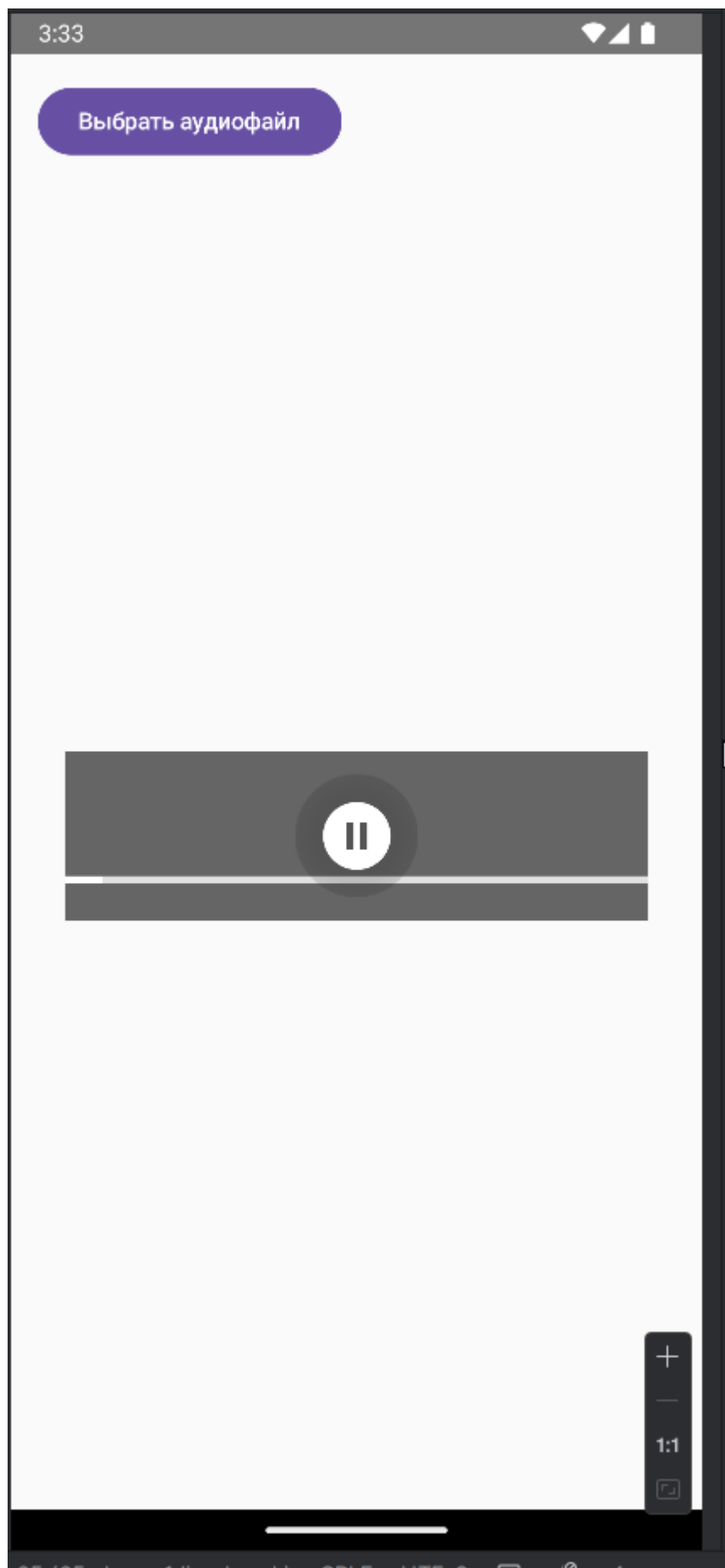


Рис 9.5. Отображение проигрывания звука

Выполнил	Баранько Д.А. 090303-ПИА-о22
Проверил	Елкин Н.С.