

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Таныгин Максим Олегович
Должность: и.о. декана факультета фундаментальной и прикладной информатики
Дата подписания: 21.09.2023 13:19:53
Уникальный программный ключ:
65ab2aa0d384efe8480e6a4c688eddbc475e411a

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Юго-Западный государственный университет»
(ЮЗГУ)

Кафедра программной инженерии

УТВЕРЖДАЮ
Проректор по учебной работе
О.И. Полюбова
« 24 » 10 / 2017 г.



ШАБЛОН ПРОЕКТИРОВАНИЯ «СОСТОЯНИЕ»

Методические указания по выполнению лабораторной работы по дисциплине "Проектирование и архитектура программных систем" для студентов направления подготовки 09.03.04 "Программная инженерия"

Курск 2017

УДК 004.65

Составители: В.Г. Белов, Т.М. Белова

Рецензент

Кандидат технических наук, доцент кафедры программной инженерии ЮЗГУ И.Н. Ефремова

Шаблон проектирования «Состояние»: методические указания по выполнению лабораторной работы по дисциплине "Проектирование и архитектура программных систем" для студентов направления подготовки 09.03.04 "Программная инженерия"/ Юго-Зап. гос. ун-т; сост.: В.Г. Белов, Т.М. Белова, – Курск, 2017. – 16 с.: ил. 15.

Изложена последовательность действий по разработке и применению шаблона проектирования «Состояние» при работе в интегрированной среде разработки Eclipse.

Материал предназначен для студентов направления подготовки бакалавров 09.03.04 «Программная инженерия», а также будет полезен студентам всех направлений подготовки, изучающим технологии разработки баз данных.

Текст печатается в авторской редакции.

Подписано в печать 24.10.17. Формат 60x84 1/16.

Усл. печ. л. 0,4. Уч.-изд. л. 0,6. Тираж 100 экз. Заказ 4966. Бесплатно.

Юго-Западный государственный университет
305040, Курск, ул.50 лет Октября, 94.

Содержание

1 Цель лабораторной работы	4
2 Порядок выполнения лабораторной работы	5
3 Содержание отчета по лабораторной работе	13
4 Вопросы к защите лабораторной работы	14
5 Индивидуальные задания	15

1 Цель лабораторной работы

Целью лабораторной работы является приобретение знаний умений и навыков для использования возможностей шаблона проектирования «Состояние» в проектировании программно-информационных систем.

Данный поведенческий шаблон предназначен, прежде всего, для использования в таких случаях, когда в решаемой задаче в каком-то определенном месте необходимо использовать различные алгоритмы в зависимости от конкретного состояния системы и/или ее окружения в конкретный момент времени.

2 Порядок выполнения лабораторной работы

1. Открыть Eclipse и создать новый JavaProject с именем State (рисунок 1).

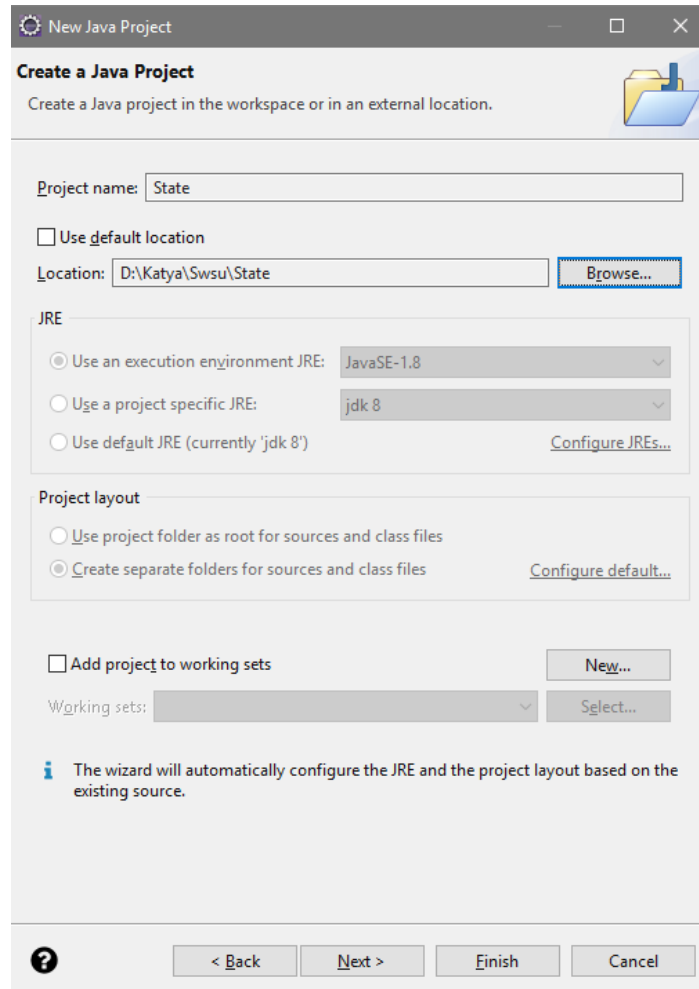


Рисунок 1

2. Схема задачи представлена ниже в виде UML-диаграммы, которая будет постепенно наполняться содержимым. Схема создается в этом проекте путем добавления PapyrusModel (рисунок 2).

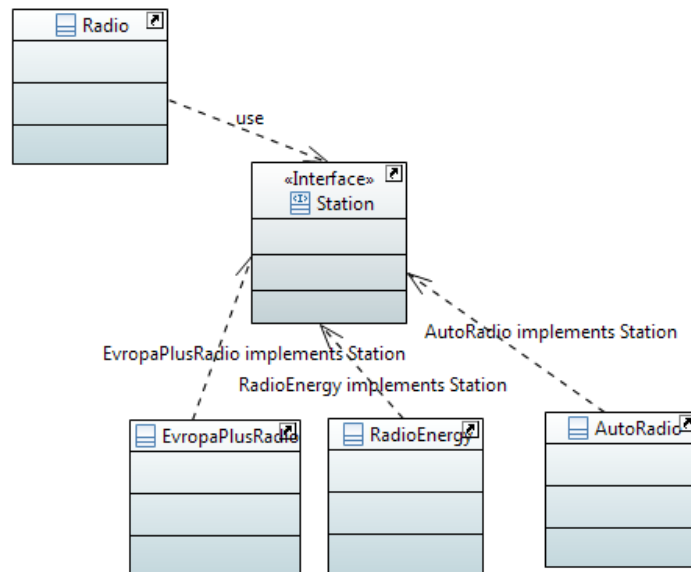


Рисунок 2

3. По этой схеме сгенерировать Java-код. Автоматически создаются заготовки для будущих классов и интерфейса.
4. Добавить в интерфейс Station метод play (рисунок 3).

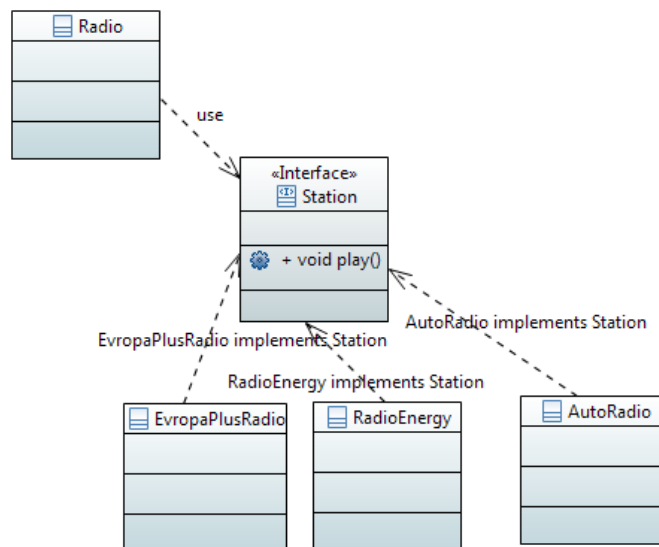


Рисунок 3

Интерфейс примет следующий вид (рисунок 4):

```

Station.java  *model.di
1 package States;
2
3 //State
4 public interface Station {
5
6     void play();
7 }
8

```

Рисунок 4

5. В классе AutoRadio реализовать метод play, который будет выводить результат проигрывания радиостанции (рисунок 5).

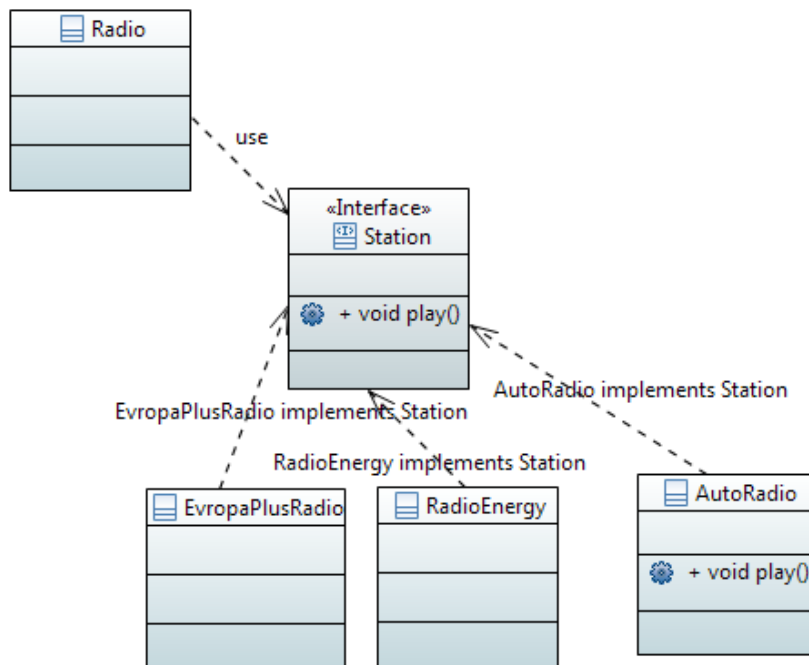


Рисунок 5

Класс примет следующий вид (рисунок 6):

```

Station.java  *model.di  AutoRadio.java
1 package States;
2
3 public class AutoRadio implements Station{
4
5     public void play() {System.out.println("AutoRadio plays...");}
6 }
7

```

Рисунок 6

6. В классе RadioEnergy реализовать метод play, который будет выводить результат проигрывания радиостанции (рисунок 7).

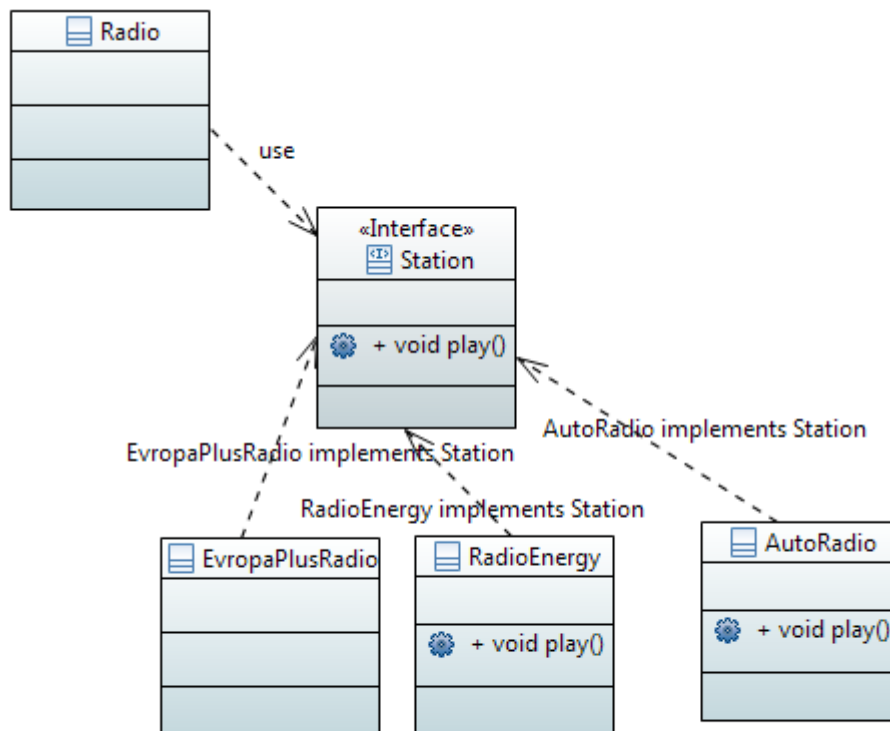


Рисунок 7

Класс примет следующий вид (рисунок 8):


```

Station.java  *model.di  AutoRadio.java  RadioEnergy.java
1 package States;
2
3 public class RadioEnergy implements Station{
4
5     public void play() {System.out.println("RadioEnergy plays...");}
6 }
7

```

Рисунок 8

7. В классе `EvropaPlusRadio` реализовать метод `play`, который будет выводить результат проигрывания радиостанции (рисунок 9).

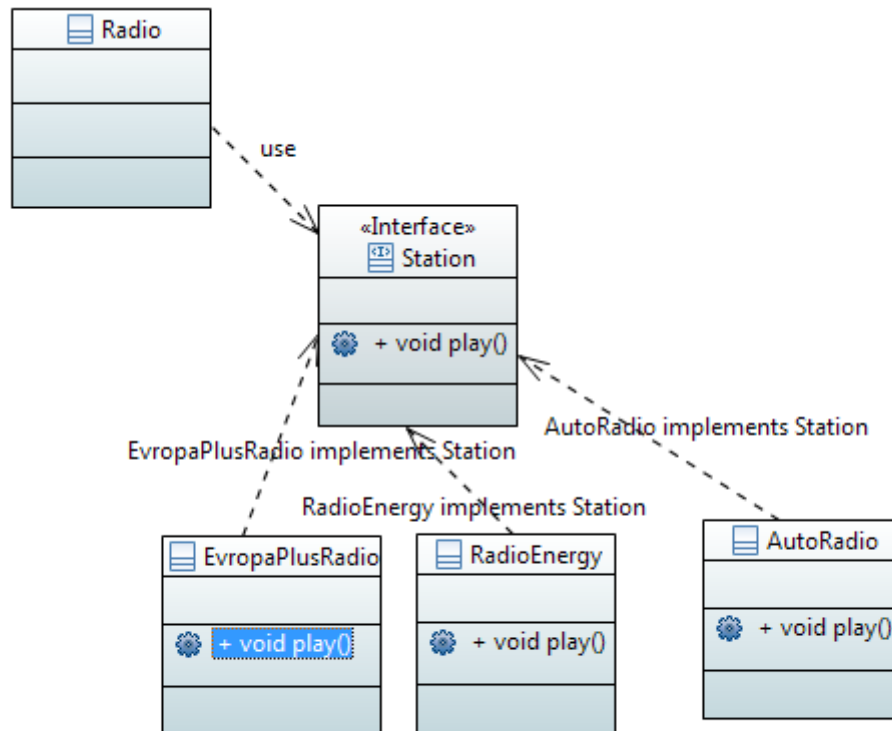


Рисунок 9

Класс примет следующий вид (рисунок 10):

```

Station.java *model.di AutoRadio.java RadioEnergy.java EvropaPlusRadio.java
1 package States;
2
3 public class EvropaPlusRadio implements Station{
4
5     public void play() {System.out.println("EvropaPlusRadio plays...");}
6 }
7

```

Рисунок 10

8. В класс Radio (рисунок 11), использующий интерфейс Station, добавить методы setStation (устанавливает конкретное состояние) и nextStation (реализует установленное состояние).

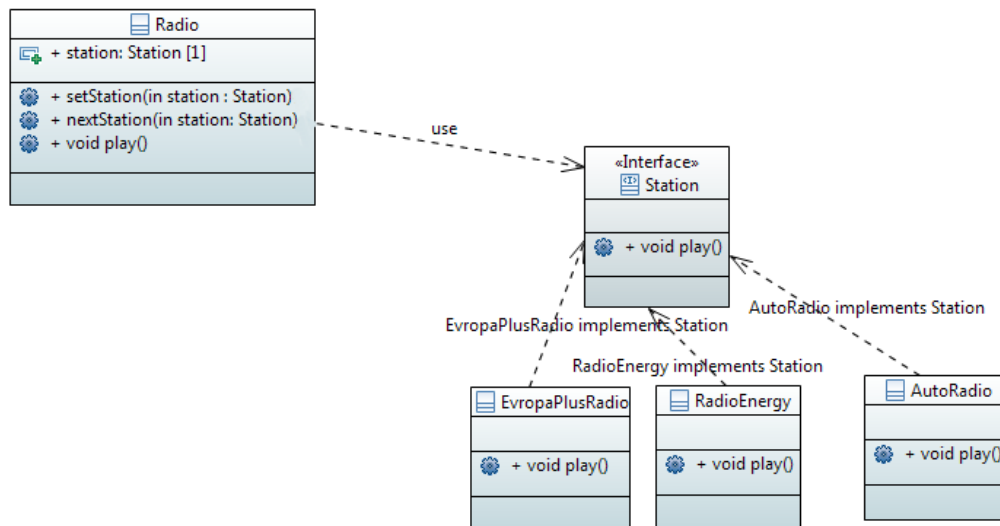


Рисунок 11

Класс примет следующий вид (рисунок 12):

```

Station.java *model.di AutoRadio.java RadioEnergy.java EvropaPlusRadio.java Radio.java
1 package States;
2
3 //Context
4 public class Radio {
5     Station station;
6     void setStation(Station st) {station = st;}
7     void nextStation() {
8         if (station instanceof RadioEnergy) {
9             setStation (new EvropaPlusRadio ());
10        }
11        else if (station instanceof EvropaPlusRadio) {
12            setStation (new AutoRadio ());
13        }
14        else if (station instanceof AutoRadio) {
15            setStation (new RadioEnergy ());
16        }
17    }
18 }
19 void play() {
20     station.play();
21 }
22 }
23

```

Рисунок 12

9. Добавьте класс ClassMain и добавить в него метод main, в котором следует написать тестовое приложение для шаблона «Состояние»(рисунок13).

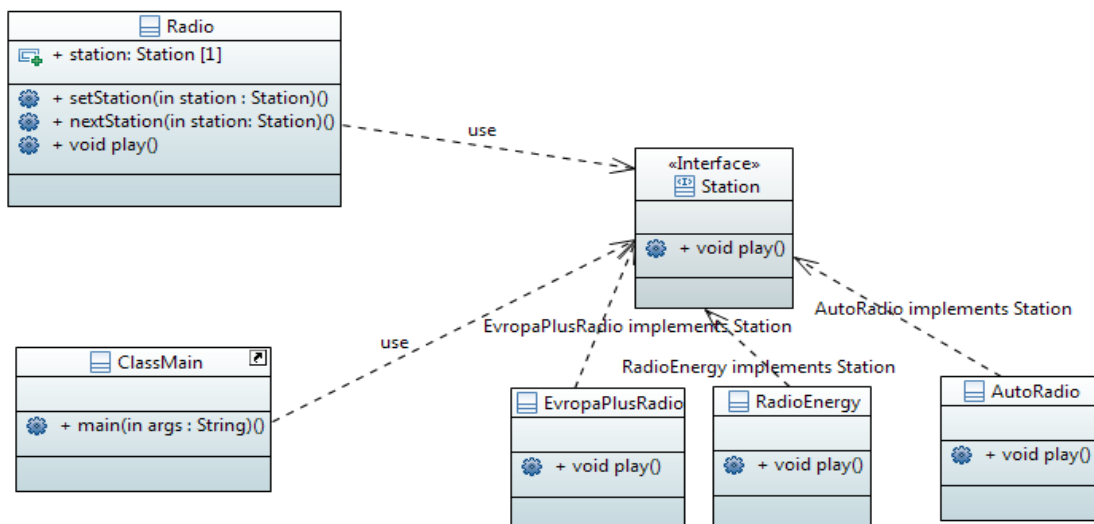
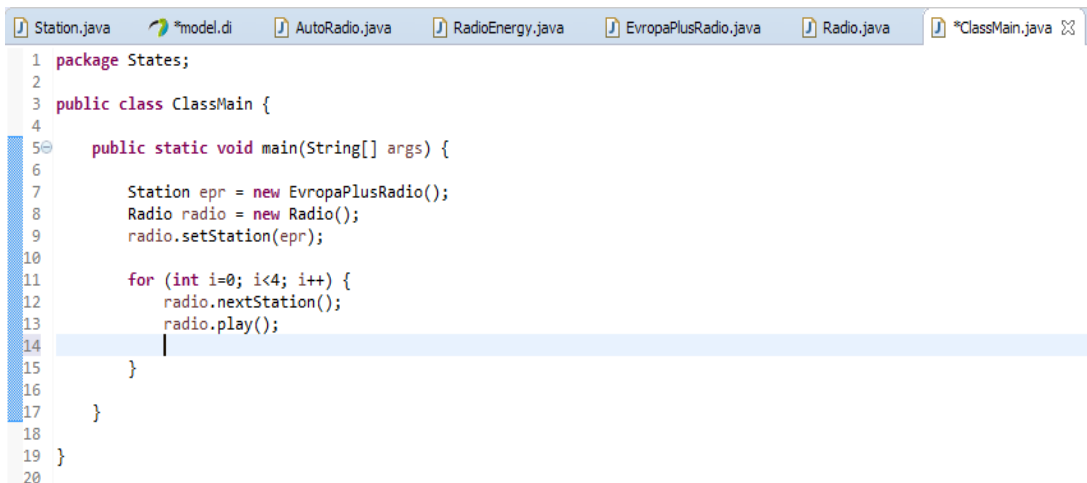


Рисунок 13

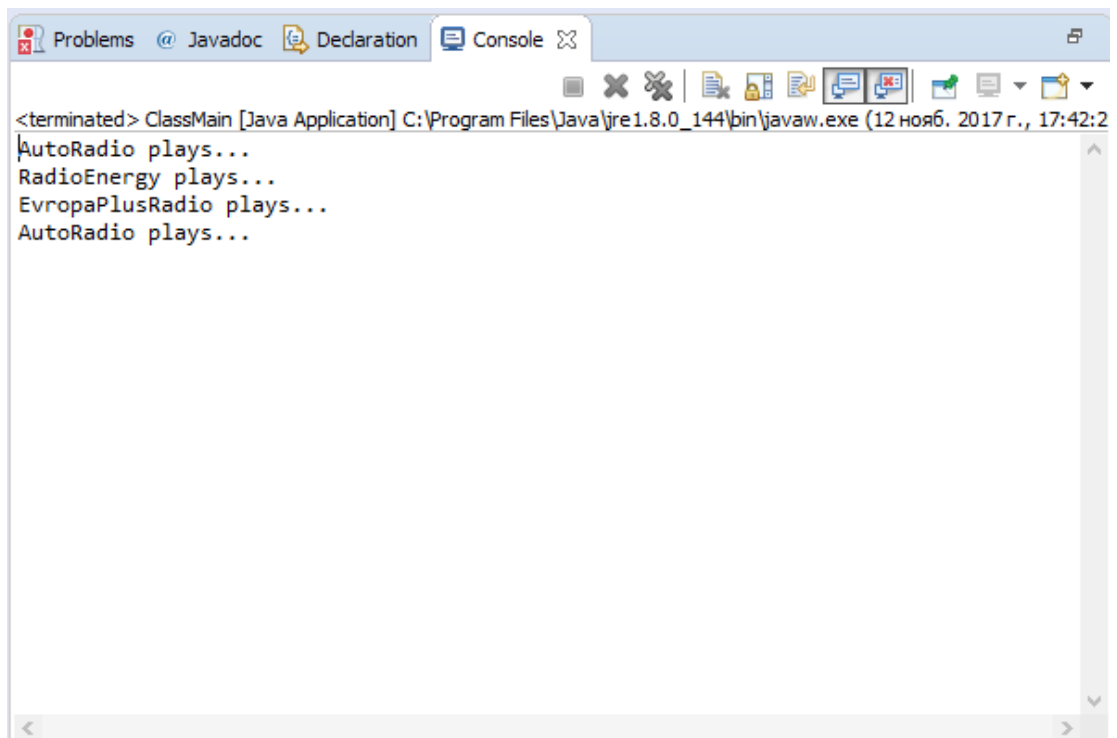
Класс тестового приложения примет следующий вид (рисунок 14):



```
1 package States;
2
3 public class ClassMain {
4
5     public static void main(String[] args) {
6
7         Station epr = new EvropaPlusRadio();
8         Radio radio = new Radio();
9         radio.setStation(epr);
10
11         for (int i=0; i<4; i++) {
12             radio.nextStation();
13             radio.play();
14         }
15     }
16 }
17
18 }
19 }
20 }
```

Рисунок 14

10. Результат выполнения тестового приложения (рисунок 15).



```
<terminated> ClassMain [Java Application] C:\Program Files\Java\jre1.8.0_144\bin\javaw.exe (12 нояб. 2017 г., 17:42:2
AutoRadio plays...
RadioEnergy plays...
EvropaPlusRadio plays...
AutoRadio plays...
```

Рисунок 15

3 Содержание отчета по лабораторной работе

В сводный отчет по лабораторным работам в качестве одного из разделов или подразделов включаются скриншоты, показывающие реализацию шаблона проектирования «Состояние» для индивидуального задания.

4 Вопросы к защите лабораторной работы

1. Что такое шаблон проектирования?
2. Для чего предназначен шаблон проектирования «Состояние»?
3. К какому типу шаблонов проектирования относится «Состояние»?
4. Какие классы/интерфейсы являются участниками «Состояние»?
5. Для чего необходим метод `setStation` в классе `Radio`?
6. Назовите родственные для «Состояния» шаблоны проектирования.
7. Выделите достоинства и недостатки этого шаблона проектирования.

5 Индивидуальные задания

1. Реализовать программный продукт, который является примером конечного автомата с двумя состояниями и двумя событиями. При разработке использовать поведенческий шаблон проектирования «Состояние».

2. Реализовать программный продукт на тему «Аудиоплеер». Основной класс плеера меняет своё поведение в зависимости от того, в каком состоянии находится проигрывание. При разработке использовать поведенческий шаблон проектирования «Состояние».

3. Реализовать программный продукт на тему «Светофор». При разработке использовать поведенческий шаблон проектирования «Состояние».

4. Реализовать программный продукт на тему «Вода». Предусмотреть случай нахождения воды в трех состояниях. При разработке использовать поведенческий шаблон проектирования «Состояние».

5. Реализовать программный продукт на тему «Телевизор». Предусмотреть случай нахождения телевизора в трех состояниях: «Выключен», «Включен», «Переключение канала». При разработке использовать поведенческий шаблон проектирования «Состояние».

6. Реализовать программный продукт на тему «Движение автомобиля». Предусмотреть случай нахождения автомобиля в трех состояниях. При разработке использовать поведенческий шаблон проектирования «Состояние».

7. Реализовать программный продукт на тему «Калькулятор». При

разработке использовать поведенческий шаблон проектирования
«Состояние».