|  |  |  |
| --- | --- | --- |
|  |  |  |
| МИНОБРНАУКИ РОССИИ | | |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** | | |

Институт Информационных технологий

Кафедра Математического обеспечения и стандартизации информационных технологий

**Отчет по практической работе №3**

по дисциплине «Тестирование и верификация программного обеспечения»

|  |  |
| --- | --- |
| **Выполнил:** | Студент группы ИКБО-01-19  Кузин Д. О. |
| **Проверил:** | ассистент Трушин С. М. |

2021 г.

**СОДЕРЖАНИЕ**

[1 Ход работы 3](#_Toc85743445)

[1.1 Исходный код 3](#_Toc85743446)

[1.2 Демонстрация работоспособности 5](#_Toc85743447)

[1.3 История коммитов 6](#_Toc85743448)

# Ход работы

## Исходный код

Ссылка на исходный код программы: <https://github.com/Shambonik/5th-semester-/tree/testing_3/Тестирование/melody>

Методология разработки ПО

Была разработана игра «Угадай мелодию» по методологии TDD. Сначала были написаны тесты (Листинги 1 - 2) к программе.

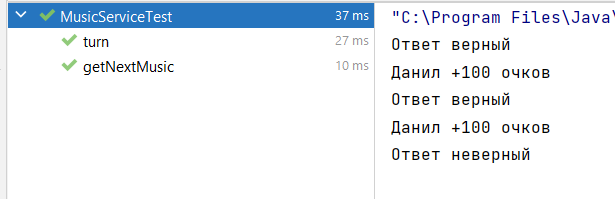
Листинг 1 – код тестов класса MusicService

public class MusicServiceTest {  
  
 private final String musicFileName = "files/music\_list.txt";  
  
 @Test  
 public void getNextMusic() throws IOException{  
 File musicFile = new File(musicFileName);  
 FileWriter fileWriter = new FileWriter(musicFile);  
 String fileContent = "";  
 fileWriter.write(fileContent);  
 fileWriter.close();  
 MusicService musicService = new MusicService();  
 assert musicService.getNextMusic() == null;  
 fileWriter = new FileWriter(musicFile);  
 fileContent = "кукла\_колдуна.wav -> Кукла колдуна; Кукла; Колдун";  
 fileWriter.write(fileContent);  
 fileWriter.close();  
 musicService = new MusicService();  
 assert musicService.getNextMusic().getFile().getName().equals("кукла\_колдуна.wav");  
 assert musicService.getNextMusic() == null;  
 fileWriter = new FileWriter(musicFile);  
 fileContent = "девятка.wav -> Девятка; А мой мальчик едет на девятке\n" +  
 "Я\_в\_моменте.wav -> Я в моменте\n" +  
 "кукла\_колдуна.wav -> Кукла колдуна; Кукла; Колдун\n" +  
 "знаешь\_ли\_ты.wav -> Знаешь ли ты; Вдоль ночных дорог\n" +  
 "сансара.wav -> Сансара\n";  
 fileWriter.write(fileContent);  
 fileWriter.close();  
 }  
  
 @Test  
 public void turn() {  
 Player player = new Player(1, "Данил", 0);  
 List<String> music\_names = new ArrayList<>();  
 music\_names.add("Название1");  
 music\_names.add("Название2");  
 Music music = new Music(music\_names, new File("files/music/сансара.wav"));  
 MusicService musicService = new MusicService();  
 System.*setIn*(new ByteArrayInputStream("Название1\n".getBytes()));  
 musicService.turn(player, music);  
 assert player.getScore() == 100;  
 System.*setIn*(new ByteArrayInputStream("Название2\n".getBytes()));  
 musicService.turn(player, music);  
 assert player.getScore() == 200;  
 System.*setIn*(new ByteArrayInputStream("Название3\n".getBytes()));  
 musicService.turn(player, music);  
 assert player.getScore() == 200;  
 }  
}

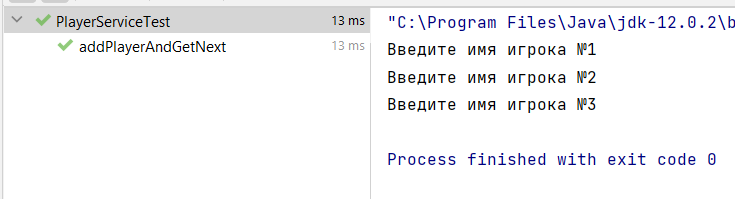
Листинг 2 – код тестов класса UserService

public class PlayerServiceTest {  
 private final PlayerService playerService = new PlayerService();  
  
 @Test  
 public void addPlayerAndGetNext() {  
 assert playerService.getNextPlayer() == null;  
 System.*setIn*(new ByteArrayInputStream("Игрок1\nИгрок2\nИгрок3\n".getBytes()));  
 playerService.addPlayers();  
 assert playerService.getNextPlayer().getName().equals("Игрок1");  
 assert playerService.getNextPlayer().getName().equals("Игрок2");  
 assert playerService.getNextPlayer().getName().equals("Игрок3");  
 assert playerService.getNextPlayer().getName().equals("Игрок1");  
 }  
}

Потом были написаны реализации функций и проведено тестирование (Рисунки 1 - 2).

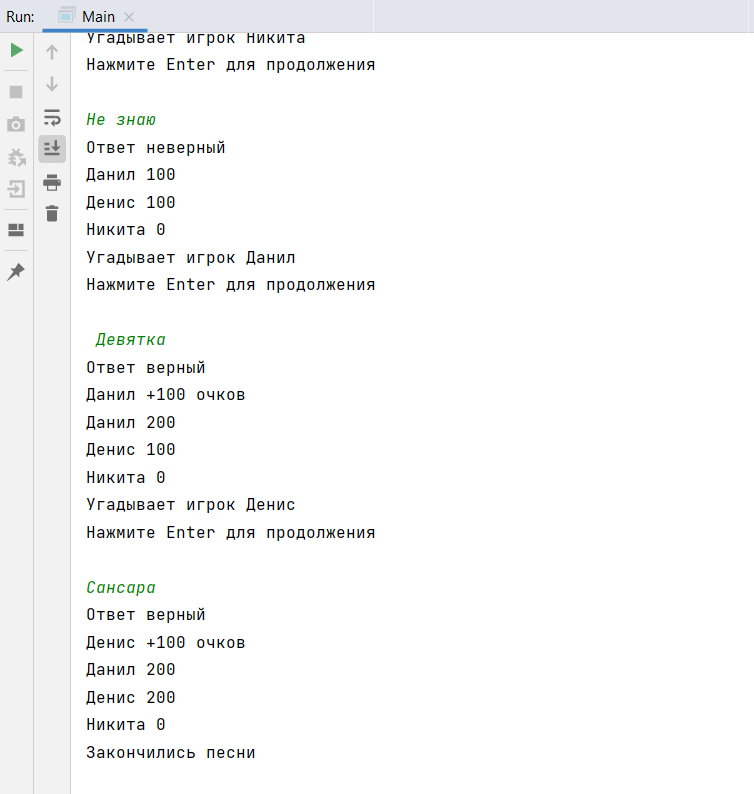


***Рисунок 1 – демонстрация успешного прохождения тестов класса MusicService***



***Рисунок 2 – демонстрация успешного прохождения тестов класса PlayerService***

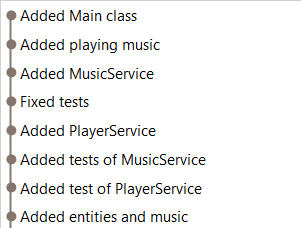
## Демонстрация работоспособности



***Рисунок 3 – демонстрация работоспособности игры***

## История коммитов

История коммитов представлена на Рисунке 3.



***Рисунок 4 – история коммитов***