МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ   
РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное образовательное учреждение высшего образования «Самарский национальный исследовательский университет имени академика С.П. Королева»  
(Самарский университет)

Институт информатики и кибернетики\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Кафедра программных систем\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**ОТЧЕТ**  
  
по лабораторной работе №2

«Документы тестирования»

по дисциплине «Тестирование и отладка ПО»

Обучающаяся в группе 6301-020302D Соколова Алёна

Преподаватель Лобанков Антон Алексеевич

Самара 2024

**Оглавление**

[Чек-лист для проверки приложения 3](#_Toc181109522)

[Общие требования: 3](#_Toc181109523)

[Тест-кейс 1: Успешный запуск приложения с полным набором параметров 3](#_Toc181109524)

[Тест-кейс 2: Обработка поврежденного аудиофайла 4](#_Toc181109525)

[Чек-лист по функциональности программы для варианта 26 5](#_Toc181109526)

[Тест-кейс 1: Проверка успешной генерации случайных чисел 5](#_Toc181109527)

[Тест-кейс 2: Проверка реакции программы на неверный диапазон (min больше max) 6](#_Toc181109528)

[Тест-кейс 3: Проверка обработки нечисловых значений 6](#_Toc181109529)

[Тест-кейс 4: Проверка ввода нечислового значения для количества чисел 6](#_Toc181109530)

[Тест-кейс 5: Проверка генерации слишком большого количества чисел 6](#_Toc181109531)

[Тест-кейс 6: Проверка ввода отрицательных значений для диапазона 7](#_Toc181109532)

[Тест-кейс 7: Проверка генерации чисел с плавающей точкой 7](#_Toc181109533)

[Тест-кейс 8: Проверка генерации случайных чисел с одинаковыми границами диапазона 7](#_Toc181109534)

[Часть 3: Заключение 7](#_Toc181109535)

**Вариант:** 26 – Генератор случайных чисел.

**Цель работы**

Целью выполнения данной лабораторной работы является ознакомление с процессом создания и анализа документации, применяемой в процессе тестирования программного обеспечения. В ходе работы необходимо изучить типы и содержание документации, используемой для тестирования, а также разработать собственные документы для программы, соответствующей индивидуальному заданию.

**Задание**

1. Используя требования из приложения 1 лабораторной работы 1 создать чек-лист покрывающий все требования и 2 тест-кейса на выбор;
2. Создать чек-лист и написать тест-кейсы функциональности программы в соответствии с индивидуальным вариантом;
3. Составить отчет о проделанной работе.

# Чек-лист для проверки приложения

## Общие требования:

1. Приложение является консольным;

2. Поддерживается минимальная версия JRE — 8.0.60;

3. Приложение работает без дополнительных библиотек;

4. Приложение не должно завершаться аварийно в случае исключений;

5. Поддерживаемые форматы аудио: mp3, flac, wav, ogg, wma;

6. Запуск приложения консольной командой с обязательными параметрами;

7. Вывод в формате HTML и CSV;

8. Проверка, что при отсутствии обязательного параметра выводится сообщение об ошибке;

9. Корректный вывод сообщений об ошибках.

## Тест-кейс 1: Успешный запуск приложения с полным набором параметров

Предусловие:

1. Убедитесь, что у вас есть доступ к каталогу, содержащему аудиофайлы по указанным путям: /path/to/audio1 и /path/to/audio2;

2. Убедитесь, что файл AudioCataloger.jar находится в доступном каталоге;

3. Убедитесь, что у вас есть права на запись в текущий каталог для создания файлов HtmlOutputFile.html и CsvOutputFile.csv.

Шаги:

1. Запустите приложение командой:

*java -jar AudioCataloger.jar HtmlOutputFile.html CsvOutputFile.csv /path/to/audio1 /path/to/audio2;*

2. Дождитесь завершения работы приложения.

Ожидаемый результат:

● Приложение успешно завершает сканирование каталогов;

● В указанные файлы выводится информация об аудиофайлах в форматах HTML и CSV;

● В консоли выводится информация о ходе работы приложения.

## Тест-кейс 2: Обработка поврежденного аудиофайла

Предусловие:

1. Создайте каталог, который будет содержать поврежденный аудиофайл (например, файл с некорректным форматом или неполный);

2. Убедитесь, что файл AudioCataloger.jar находится в доступном каталоге;

3. Убедитесь, что у вас есть права на запись в текущий каталог для создания файлов HtmlOutputFile.html и CsvOutputFile.csv.

Шаги:

1. Запустите приложение командой:

*java -jar AudioCataloger.jar HtmlOutputFile.html CsvOutputFile.csv /path/to/audio;*

2. Дождитесь завершения работы приложения.

Ожидаемый результат:

● Приложение успешно завершает работу, даже при наличии поврежденного аудиофайла;

● В логах выводится сообщение: «Нет аудиозаголовка или данных аудиотега в: {полный путь}»;

● В выходных данных вместо информации о поврежденном файле указаны заглушки.

# Чек-лист по функциональности программы для варианта 26

На основании предложенного шаблона требований, составлены следующие требования к приложению:

**Запуск программы:**

1. программа запускается через командную строку или отдельное приложение;
2. программа запрашивает ввод диапазона для генерации случайных чисел;
3. программа корректно обрабатывает ввод данных.

**Генерация случайных чисел:**

1. программа запрашивает ввод диапазона значений (min, max) для генерации чисел;
2. программа запрашивает количество чисел, которые необходимо сгенерировать;
3. программа корректно выводит список случайных чисел, соответствующий заданным параметрам.

**Обработка ошибок:**

1. программа выводит сообщение об ошибке, если введенные данные не соответствуют ожидаемому формату (например, если диапазон или количество чисел не являются числами).
2. программа не завершается аварийно при неверных входных данных, а выводит сообщение об ошибке.

## Тест-кейс 1: Проверка успешной генерации случайных чисел

Необходимо проверить, что программа корректно генерирует случайные числа в заданном диапазоне.

* + - 1. запустить программу;
      2. ввести диапазон от 1 до 10;
      3. ввести количество чисел, допустим 5

Ожидаемый результат: программа выводит пять случайных чисел в диапазоне от 1 до 10.

## Тест-кейс 2: Проверка реакции программы на неверный диапазон (min больше max)

Необходимо убедиться, что программа корректно выводит сообщение об ошибке, если введен некорректный диапазон.

* + - 1. запустить программу;
      2. ввести диапазон от 10 до 1.

Ожидаемый результат: программа выводит сообщение об ошибке "Ошибка: минимальное значение не может быть больше максимального".

## Тест-кейс 3: Проверка обработки нечисловых значений

Необходимо убедиться, что программа корректно обрабатывает нечисловой ввод.

* + - 1. запустить программу;
      2. ввести диапазон от “a” до 1.

Ожидаемый результат: программа выводит сообщение об ошибке "Ошибка: введите числовые значения для диапазона".

## Тест-кейс 4: Проверка ввода нечислового значения для количества чисел

Необходимо проверить, что программа корректно обрабатывает нечисловой ввод для количества чисел и выводит сообщение об ошибке.

* + - 1. запустить программу;
      2. ввести диапазон от 1 до 10;
      3. ввести количество чисел: “abc”.

Ожидаемый результат: программа выводит сообщение об ошибке "Ошибка: введите числовое значение для количества чисел".

## Тест-кейс 5: Проверка генерации слишком большого количества чисел

Необходимо убедиться, что программа корректно обрабатывает запрос на генерацию большого объема данных.

* + - 1. запустить программу;
      2. ввести диапазон от 1 до 10;
      3. ввести количество чисел: 1 000 001.

Ожидаемый результат: программа выводит сообщение об ошибке "Ошибка: превышено допустимое количество чисел (до 1 000 000)".

## Тест-кейс 6: Проверка ввода отрицательных значений для диапазона

Необходимо убедиться, что программа корректно генерирует случайные числа в диапазоне отрицательных значений.

* + - 1. запустить программу;
      2. ввести диапазон от -10 до -1.

Ожидаемый результат: программа выводит 5 случайных чисел в диапазоне от -10 до -1.

## Тест-кейс 7: Проверка генерации чисел с плавающей точкой

Необходимо убедиться, что программа корректно обрабатывает числа с плавающей точкой (если поддерживается).

* + - 1. запустить программу;
      2. ввести диапазон от 1.1 до 10.5.
      3. ввести количество чисел: 5.

Ожидаемый результат: программа выводит 5 случайных чисел с плавающей точкой.

## Тест-кейс 8: Проверка генерации случайных чисел с одинаковыми границами диапазона

Необходимо убедиться, что программа корректно обрабатывает диапазон, где min = max.

* + - 1. запустить программу;
      2. ввести диапазон от 5 до 5;
      3. ввести количество чисел: 3.

Ожидаемый результат: программа выводит три числа, каждое из которых равно 5.

# Часть 3: Заключение

В процессе выполнения работы была изучена документация, используемая для тестирования программного обеспечения. На основе анализа были составлены тестовые документы, включая тест-план и тест-кейсы для приложения, реализующего генерацию случайных чисел.