

## Проект " Аудиокаталогизатор "

### План испытаний

### Проектная документация

#### **Справочная информация**

Оценки, график, стратегия и метрики необходимы для эффективной организации процесса тестирования.

#### **Назначение**

Организовать процесс тестирования эффективным и результативным в течение всего периода проекта.

#### **Область применения**

Описание процесса тестирования, метрики, график, ресурсы.

#### **Аудитория**

Руководящий персонал, группа контроля качества, проектная группа.

#### **Файл**

План тестирования.pdf

## Содержание

1. Объем проекта и основные цели .....	3
2. Требования, подлежащие проверке .....	3
3. Требования, НЕ подлежащие проверке .....	3
4. Стратегия и подход к тестированию.....	4
4.1. Общий подход.....	4
4.2. Уровни функционального тестирования.....	4
5. Критерии.....	4
6. Ресурсы.....	4
7. Расписание .....	5
8. Роли и обязанности.....	5
9. Оценка рисков .....	5
10. Документация.....	5
11. Метрика .....	6

## **1. Объем проекта и основные цели**

### Разработка инструмента для:

- Каталогизация аудиофайлов.
- Поиск дубликатов аудиофайлов.
- Найдите поврежденные аудиофайлы.

### Основные цели:

Предоставьте клиенту быстрый и простой инструмент для создания списка всех аудиофайлов, находящихся в его распоряжении, вместе с перекрестными ссылками на дубликаты; Полученный список должен быть доступен для просмотра через веб-браузер для быстрого просмотра и редактирования в программе электронных таблиц для тщательной проверки и обработки;

Инструмент не должен выходить из строя (по любой причине) в процессе работы (в отличие от многих конкурирующих инструментов).

## **2. Требования, подлежащие проверке**

См. соответствующие разделы в документе "Требования к аудиокаталогизатору.docx":

UR-1\*: Начало и конец приложения.

UR-1.1\*: Запуск приложения должен быть выполнен следующей консольной командой: "java -jar AudioCataloger.jar [DuplicatesOnly] HtmlOutputFileName CsvOutputFileName StartingDirectory1 [... StartingDirectoryN]" (описание параметров см. в DS-2.1, сообщения об ошибках в случае неправильной конфигурации см. в DS-2.2, DS-2.3 и DS-2.4).

UR-1.2\*: Остановка (завершение работы) приложения должна выполняться путем применения Ctrl+C к консольному окну, в котором находится запущенное приложение.

UR-2\*: Конфигурация приложения.

UR-2.1\*: Единственная доступная конфигурация - через параметры командной строки (см. DS-2).

UR-2.2\*: Целевая кодировка для выходных текстовых сообщений - UTF8.

UR-3\*: Журнал регистрации заявлений.

UR-3.1\*: Приложение должно выводить свой журнал на консоль (см. DS-4).

UR-3.2\*: Содержание и формат журнала описаны в DS-4.2 и DS-4.3.

## **3. Требования, НЕ подлежащие проверке**

См. соответствующие разделы в документе "Требования к аудиокаталогизатору.docx":

SC-1\*: Приложение должно быть консольным.

SC-2\*: Приложение должно быть разработано с использованием Java и распространяться в виде JAR-контейнера.

SC-3\*: Приложение должно быть мультиплатформенным (с учетом L-4).

## **4. Стратегия и подход к тестированию**

### **4.1. Общий подход**

Приложение должно быть настроено один раз опытным специалистом и в дальнейшем использоваться конечными пользователями, для которых доступна только одна операция - создание списка всех аудиофайлов, находящихся в распоряжении пользователя, вместе с перекрестными ссылками на дубликаты. Поэтому вопросы удобства использования, безопасности и т.д. не рассматривались в ходе тестирования.

### **4.2. Уровни функционального тестирования**

- Дымовое тестирование: автоматизировано с помощью пакетных файлов под Windows и Linux.
- Тест критического пути: выполняется вручную.
- Расширенный тест: не выполняется, так как вероятность обнаружения дефектов на этом уровне пренебрежимо мала.

## **5. Критерии**

Критерии приемлемости: 100% успешность тестовых случаев на уровне дымовых тестов и 90% успешность тестовых случаев на уровне тестов критического пути (см. метрику "Процент успешности тестовых случаев"), если исправлено 100% критических и основных ошибок (см. метрику "Общий процент исправленных дефектов"). Окончательное покрытие требований тестами (см. метрику "Покрытие требований тестами") должно быть не менее 80%.

Критерии начала испытаний: новое строительство.

Критерии приостановки тестирования: тест критического пути должен начинаться только после 100% успеха тест-кейсов на дымовом тесте (см. "Процент успеха тест-кейсов"); процесс тестирования может быть приостановлен, если при выполнении не менее 25% тест-кейсов наблюдается не менее 50% отказов (см. метрику "Стоп-фактор").

Критерий возобновления тестирования: исправлено более 50% ошибок, найденных во время предыдущей итерации (см. метрику "Процент исправленных текущих дефектов").

Критерии завершения тестирования: более 80% запланированных на текущую итерацию тест-кейсов выполнены (см. "Процент выполнения тест-кейсов").

## **6. Ресурсы**

- Программное обеспечение: четыре виртуальные машины (две с Windows 10 Ent x64, две с Linux Ubuntu 18 LTS x64), две лицензии PHP Storm (доступна последняя версия).
- Оборудование: две стандартные рабочие станции (8 ГБ ОЗУ, i7 3 ГГц).
- Персонал:
  - Один старший разработчик с опытом тестирования (100% загрузка в течение всего времени проекта). Роли: руководитель группы, старший разработчик.
  - Один разработчик с опытом тестирования (100% загрузка в течение всего времени проекта). Роли: разработчик.

- Один тестировщик со знанием PHP (100% загруженность в течение всего времени проекта). Роль: тестировщик.
- Один тестировщик ручной (100% загруженность в течение всего времени проекта). Роль: тестировщик.
- Время: 15 рабочих дней (120 рабочих часов).
- Финансы: в соответствии с утвержденным бюджетом.

## **7. Расписание**

10.10 - тестирование и доработка требований.

12.10-14.10 - тест-кейсы и скрипты для создания автоматизированного тестирования.

17.10-24.10 - основной этап тестирования (выполнение тест-кейсов).

25.10 - основной этап тестирования (создание отчетов о дефектах).

27.10-28.10 - завершение тестирования, отчетность.

## **8. Роли и обязанности**

Старший разработчик: участие в тестировании требований и проверке кода.

Разработчик: участие в тестировании требований и проверке кода.

Тестировщик (PHP): создание документации, выполнение тест-кейсов, участие в рецензировании кода.

Тестировщик (ручной): тестирование требований, выполнение тест-кейсов, создание отчетов о дефектах.

## **9. Оценка риска**

Персонал (низкая вероятность): если какой-либо член команды будет недоступен, мы можем распределить его обязанности между другими членами команды.

Время (высокая вероятность): заказчик указал срок до 29.10, поэтому время является критическим ресурсом. Рекомендуется приложить все усилия для завершения проекта к 27.10, чтобы два дня (28.10 и 29.10) остались свободными для решения любых непредвиденных вопросов.

Прочие риски: Сложность точного разбора некоторых аудиоформатов; Сложность (или невозможность) определения кодировки для неанглийских тегов в файлах.

## **10. Документация**

Требования. Ответственное лицо - тестировщик, срок - 12.10.

Тестовые случаи и отчеты о дефектах. Ответственный - тестировщик, период создания - 17.10-26.10.

Отчет о результатах тестирования. Ответственное лицо - испытатель, срок - 29.10.

## 11. Метрика

### Процент успешности тестовых случаев:

$$T_{SP} = (T_{Success} / T_{Total}) \cdot 100\%,$$

где  $T_{SP}$  - процент успешно пройденных тестовых случаев,

$T_{Success}$  - количество успешно пройденных тестовых случаев,

$T_{Total}$  - общее количество выполненных тестовых случаев.

#### Минимально допустимые границы:

Начальная фаза проекта - 10%.

Основная фаза проекта - 40%.

Заключительный этап проекта - 80%.

### Общий процент исправленных дефектов:

$$DFTP_{Level} = (D_{Closed Level} / D_{Found Level}) \cdot 100\%,$$

где  $DFTP_{Level}$  - общий процент устранения дефектов на  $Level$  за все время существования проекта,

$D_{Closed}$  - количество дефектов  $Level$ , исправленных за все время жизни проекта,

$D_{Found}$  - количество дефектов  $Level$ , обнаруженных за все время жизни проекта.

#### Минимально допустимые границы:

		Серьезность дефекта			
		Минор	Средний	Главная	Критический
Стадия проекта	Начало	10%	40%	50%	80%
	Главная	15%	50%	75%	90%
	Окончательный	20%	60%	100%	100%

### Процент устранения текущих дефектов:

$$DFCP_{Level} = (D_{Closed Level} / D_{Found Level}) \cdot 100\%,$$

где  $DFCP_{Level}$  - процент исправления дефектов на  $Level$  (дефекты, найденные в предыдущей сборке и исправленные в текущей),

$D_{Closed Level}$  - количество дефектов  $Level$ , исправленных в текущей сборке,

$D_{Found Level}$  - количество дефектов  $Level$ , найденных в предыдущей сборке.

#### Минимально допустимые границы:

		Серьезность дефекта			
		Минор	Средний	Главная	Критический
Стадия проекта	Начало	60%	60%	60%	60%
	Главная	65%	70%	85%	90%
	Окончательный	70%	80%	95%	100%

Стоп-фактор:

$$S = \{Yes, T^E \geq 25\% \ \&\& \ T^{SP} < 50\% , \\ No, T^E < 25\% \ || \ T^{SP} \geq 50\%$$

где  $S$  - решение приостановить процесс тестирования,

$T^E$  - текущее  $T^E$  значение,

$T^{SP}$  - текущее  $T^{SP}$  значение.

Процент выполнения тест-кейсов:

$$T^E = (T_{Executed} / T_{Planned}) \cdot 100\%,$$

$T^E$  - процент выполнения тест-кейсов,

$T_{Executed}$  - количество выполненных тест-кейсов,

$T_{Planned}$  - количество запланированных (к выполнению) тест-кейсов.

Уровни (границы):

Минимальный: 80%.

Желаемые: 95%-100%.

Покрытие требований тестами:

$$C^R = (R_{Covered} / R_{Total}) \cdot 100\%,$$

где  $C^R$  - покрытие требований тестами (в процентах),

$R_{Covered}$  - количество требований, покрытых тест-кейсами,

$R_{Total}$  - общее количество требований.

Минимально приемлемые границы:

Начальный этап проекта: 40%.

Основная фаза проекта: 60%.

Заключительный этап проекта: 80% (рекомендуется 90%+)