## Шаблон

### 1. Введение

* Цели

По окончанию работы нужно получить ПО, с помощью которого можно будет массово редактировать информацию в тегах mp3-файлов, заменяя или полностью удаляя выбранные совпадения.

* Обзор

Иногда, при скачивании музыки через ботов или некоторые сайты, владельцы сайтов заменяют информацию некоторых IP3-тегов на названия своего сервиса/бота. С помощью данного ПО планируется удалять такие теги.

* Определения, сокращения, термины

Излишни.

* Ссылки

Нет

### 2. Общее описание

### ПО осуществляет поиск информации в метаданных mp3-файлов и производит замену тэгов, которые пользователь считает некорректными.

### Все изменения должны логироваться.

### 3. Требования

* Функциональность
* [Юзабилити](https://ru.wikipedia.org/wiki/%D0%AE%D0%B7%D0%B0%D0%B1%D0%B8%D0%BB%D0%B8%D1%82%D0%B8)
* Надежность
* [Производительность](https://ru.wikipedia.org/wiki/%D0%A2%D0%B5%D1%81%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5_%D0%BF%D1%80%D0%BE%D0%B8%D0%B7%D0%B2%D0%BE%D0%B4%D0%B8%D1%82%D0%B5%D0%BB%D1%8C%D0%BD%D0%BE%D1%81%D1%82%D0%B8)

Желательна реализация асинхронности.

* Поддерживаемость

Не предусмотрена.

* Проектные ограничения

Жизненные силы разработчика.

* Требования по документированности и поддержке пользователей

Нет

* Заимствованные компоненты

Нет

* Интерфейсы

1. Пользовательские интерфейсы

Одно окно с кнопками, полями ввода, флажками и текстом.

Флажки:

А) Искать в альбомах

Б) Искать в исполнителях

В) Искать в названиях песен

Г) Исправить ВСЁ

Д) Выбор с подпапками

Е) 3 флажка с текстом «Очистить»

Поля:

А) Что искать в альбомах?

Б) Что искать в исполнителях

В) Что искать в названиях песен

Г) 3 поля с текстом: «Заменить на»

Кнопки:

А) Выбрать папку

Б) Найти и вывести файлы с проблемами

В) Запустить исправитель

1. Аппаратные интерфейсы

Нет

1. Программные интерфейсы

Программа поделена на 4 модуля:

* Interface – пользовательский интерфейс: окно с кнопками, заполняемыми полями ввода, флажками и текстом.
* Get\_path – получает пути всех файлов формата mp3 в выбранной директории.
* Get\_tags – получает пути к файлам, проверяет метаданные и изменяет их.
* Make\_logs – логирует изменения файлов, производимые функцией get\_tags.

   - Коммуникационные интерфейсы

Нет  
- Лицензионные соглашения

Нет  
- Необходимые замечания по законодательству, авторским правам и прочие

Нет  
- Применяемые стандарты

Нет

### 4. Сопроводительная информация

### Поиск файлов с проблемами в названии:

### Проверяем тег «название» на наличие (в том числе) комбинации символов « - ».

### Если комбинация присутствует, то сравниваем её с названием файла, если есть различия, то помечаем как потенциально проблемный.

### Смотрим тег «исполнители», если в нём присутствует символ «/», то помечаем как потенциально проблемный. Если производится замена, то заменяется на «, ».

### Результаты проверки сохраняются в файл txt, который открывается по завершении процесса, дабы пользователь проверил, все ли выбранные файлы подходят под критерии проблемности.

### Лучше всего отдельно отредактировать помеченные файлы, не являющиеся проблемными.

### Должна быть реализована асинхронность.

### Interface получает путь и критерии отбора от пользователя.

### Get\_path получает путь от interface, формирует список файлов в папке и подпапках.

### Get\_tags получает список путей от get\_path и критерии отбора от interface, проверяет тэги на соответствие критериям. Передаёт список композиций и потенциально проблемных тегов.

### Файл открывается и показывает пользователю данные из check\_tags

### Пользователь устраняет те недочёты, что возможно, вручную. Затем запускает процесс заново, но в этот раз выбирает изменения.

### Correct\_tags получает список композиций и информацию по корректировке от interface. Вносит изменения. По завершении операций выводит информацию о количестве исправленных файлов и тегов.

### Logger записывает список изменений, произведённый correct\_tags