## 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

- Ссылки  
- Обзор

### 2. Общее описание

### Символы: ‘ - ‘, ‘/’, ‘…’, ‘!’

### 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" \o "Юзабилити)  
- 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)  
- Поддерживаемость  
- Проектные ограничения  
- Требования по документированности и поддержке пользователей

Нет  
- Заимствованные компоненты

Нет  
- Интерфейсы  
   - Пользовательские интерфейсы

Одно окно с кнопками, полями ввода и флажками.

Флажки:

Искать в альбомах

Искать в исполнителях

Искать в названиях

Только поиск

Поля:

Выбор директории

Названия альбомов

Исполнители

Названия песен

3 поля: «заменить на»

Старт

Выход  
   - Аппаратные интерфейсы  
   - Программные интерфейсы  
   - Коммуникационные интерфейсы  
- 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.

### Модули:

### Interface – tkinter

### Список плохой информации в метадате:

### «Official»

### «Video»

### ('/', '\\', ':', '\*', '?', '<', '>', '|', '...', ' ...', ' - ')

### «(» без «)»

### «VEVO»

### «mrSuicideSheep»

### «music»

### «alona chemerys»

### «aviencloud»

### 

### Если название разделить по ' - ' и правая с левой частью не будут совпадать с соответствующими тегами, то название повреждено.

### Check\_tags – классы для работы с ID3

### 

### Get\_path

### Проход по вложенным папкам и поиск файлов

### Correct\_tags

### Logger – работа с логами

### Информация, которая должна быть в логах:

### Время

### Тип операции: изменение или поиск

### Тип тэга

### Полный путь

### 

### 

### Buffer – работа с информацией в оперативной памяти

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

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

### Check\_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.

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