# «Audio Cataloger» Project

# Requirements

**Project Documentation**

| | |
|---|---|
| **Background** | Full set of requirements specification. |
| **Purpose** | To organize both development and testing process. |
| **Scope** | Business requirements, user requirements, detailed specification, limitations. |
| **Audience** | Management staff, project team. |
| **File** | Audio Cataloger Requirements.docx |

# Contents

## 1. Project scope

Development of a tool to:

- Catalog audio files.
- Find duplicates of audio files.
- Find corrupted audio files.

## 2. Main goals

- Provide the Customer with a quick and simple tool to create a list of all audio files in his possession along with duplications cross-reference.
- The resulting list should be viewable via web-browser for quick review, and editable in spreadsheet software for thorough review and processing.
- The tool should not fail (for any reason) during its working process (unlike many competing tools).

## 3. Criteria for main goals achievement

- The tool should be a console one (for simplicity), support smart comparison algorithms (for good performance) and the following audio formats: see BR-1.
- HTML (for browser) and CSV (for spreadsheet editor) output support.
- Deep testing for negative and critical situations should be implemented to ensure the application reliability.

## 4. Risks

- Complexity of accurate parsing of some audio formats.
- Complexity (or impossibility) to detect encoding for non-English tags in files.

## 5. System characteristics

- SC-1: The application should be a console one.
- SC-2: The application should be developed using Java and distributed as JAR-container.
- SC-3: The application should be a multi-platform one (taking into account L-4).

## 6. User requirements

- UR-1: Start and stop of the application.
  - UR-1.1: The application start should be performed by the following console command: "java -jar AudioCataloger.jar [DuplicatesOnly] HtmlOutputFileName CsvOutputFileName StartingDirectory1 [... StartingDirectoryN]" (see DS-2.1 for parameters description, see DS-2.2, DS-2.3, and DS-2.4 for error messages on any misconfiguration situation).
  - UR-1.2: The application stop (shutdown) should be performed by applying Ctrl+C to the console window, which holds the running application.

- UR-2: Configuration of the application.
  - UR-2.1: The only configuration available is through command line parameters (see DS-2).
  - UR-2.2: Target encoding for output text messages is UTF8.
- UR-3: Application log.
  - UR-3.1: The application should output its log to the console (see DS-4).
  - UR-3.2: Log contents and format are described in DS-4.2 and DS-4.3.

## 7. Business rules

- BR-1: Supported formats are: mp3, flac, wav, ogg, wma.
- BR-2: Output formats are both HTML and CSV.
- BR-3: In DuplicatesOnly mode the background color for duplicates should be white in HTML output. In normal mode the background for duplicates should be red.
- BR-4: Any directory or file name in console output should be fully qualified normalized one.

## 8. Quality attributes

- QA-1: Resilience to input data
  - QA-1.1: See BR-1 for the requirements to input file formats.
  - QA-1.2: See DS-5.2 for the requirements to input file size.
  - QA-1.3: See DS-5.3 for the details on application reaction on incorrect input file format.
- QA-2: Exception handling: under no circumstances the application should crash with unhandled exception. No matter how broken an audio file is, the application should either extract necessary data or replace the data with predefined stubs in the output.
- QA-3: If several starting directories are specified, the application should analyze the set for nesting and/or duplication in order to scan each real directory only once.

## 9. Limitations

- L-1: The application should be developed using Java as the most convenient cross-platform environment.
- L-2: See DS-1 for JRE version and configuration details.
- L-3: JRE setup and configuration process are out of this project scope and therefore are NOT described in any product/project documentation.
- L-4: Multi-platform capabilities of the application are the next: it should work with Windows and Linux assuming that proper Java version (see DS-1.1) works there.

## 10. Detailed specifications

**DS-1: Java**
DS-1.1: Minimal JRE version – 8.0.60.
DS-1.2: The application should work with just standard JRE, i.e. without any additional specific libraries and/or tools.

**DS-2: Command line parameters**

DS-2.1: The application receives the following command line parameters during the start process:

- [DuplicatesOnly] – optional parameter indicating that only duplicate audio files should appear in the output;
- HtmlOutputFileName – mandatory parameter, points to the file for HTML output;
- CsvOutputFileName – mandatory parameter, points to the file for CSV output;
- StartingDirectory1 – mandatory parameter, points to the directory to scan;
- [... StartingDirectoryN] – optional parameters, each points to another directory to scan (see also QA-3).

DS-2.2: If some mandatory command line parameter is omitted, the application should shut down displaying standard usage-message (see DS-3.1).

DS-2.3: Any number of command line parameters after StartingDirectory1 should be interpreted as the set of directories to scan (see also QA-3).

DS-2.4: If the value of any command line parameter is incorrect, the application should shut down displaying standard usage-message (see DS-3.1) and incorrect parameter name, value, and proper error message (see DS-3.2).

**DS-3: Messages**

DS-3.1: Usage message: "Usage: java -jar AudioCataloger.jar [DuplicatesOnly] HtmlOutputFileName CsvOutputFileName StartingDirectory1 [... StartingDirectoryN]".

DS-3.2: Error messages:
- "The following directory is not found or is inaccessible: {full path}";
- "The following file is not writable: {full path}".
- "No audio header or audio tag data in: {full path}".

**DS-4: log**

DS-4.1: The application should display its current activity in the console. No log files needed.

DS-4.2: The console log format is up to developers.

DS-4.3: [Optional] The application should list and describe the list of given command line parameters in the log.

**DS-5: File format and size**

DS-5.1: The application should process files in the following formats: see BR-1.

DS-5.2: The application should process files up to 2 GB (inclusive).

DS-5.3: If a broken file or a file with unsupported inner structure detected, the application should display a log message "No audio header or audio tag data in: {full path}".