## «Audio Cataloger» Project

Test Plan

Background Estimations, schedule, strategy, and metrics are needed to organize the testing process efficiently.

Purpose To organize the testing process effective and efficient during the whole project period.

Scope Testing process description, metrics, schedule, resources.

Audience Management staff, QA team, project team.

File AudioTestPlan.docx

### Contents

1. Project scope and main goals......................................................................3

2. Requirements to be tested...........................................................................3

3. Requirements NOT to be tested .................................................................3

4. Test strategy and approach .........................................................................3

4.1. General approach......................................................................................3

4.2. Functional testing levels ..........................................................................3

5. Criteria.........................................................................................................4

6. Resources ....................................................................................................4

7. Schedule ......................................................................................................4

8. Roles and responsibilities ............................................................................4

9. Risk evaluation.............................................................................................4

10. Documentation ..........................................................................................5

11. Metrics .......................................................................................................5

### 1. Project scope and main goals

Development of a tool to:

∙ Catalog audio files.

∙ Find duplicates of audio files.

∙ Find corrupted audio files.

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).

### 2. Requirements to be tested:

See referenced sections in “Audio\_Cataloger\_Requirements.pdf”:

∙ UR-1.\*: smoke test.

∙ UR-2.\*: smoke test, critical path test.

∙ UR-3.\*: critical path test.

∙ BR-1.\*: smoke test, critical path test.

∙ BR-2.\*: smoke test, critical path test.

∙ BR-3.\*: critical path test.

∙ BR-4.\*: critical path test.

∙ QA-1.\*: smoke test, critical path test.

∙ QA-2.\*: smoke test, critical path test.

∙ QA-3.\*: critical path test.

∙ DS-2: critical path test.

∙ DS-3: critical path test.

∙ DS-4: critical path test.

∙ DS-5: critical path test.

### 3. Requirements NOT to be tested See referenced sections in “Audio\_Cataloger\_Requirements.pdf”:

∙ SC-1: the application is a console one by design.

∙ 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: no implementation required.

∙ L-4: no implementation required.

∙ DS-1: The application should work with just standard JRE, i.e. without any additional specific libraries and/or tools.

### 4. Test strategy and approach

4.1. General approach

The tool should not fail (for any reason) during its working process (unlike many competing tools).

4.2. Functional testing levels

∙ Smoke test: automated with batch files under Windows and Linux.

∙ Critical path test: executed manually.

Deep testing for negative and critical situations should be implemented to ensure the application reliability.

### 5. Criteria

∙ Acceptance criteria: 100% success of test cases on smoke test level and 90%

success of test cases on critical path test level (see “Test cases success

percentage” metric) if 100% of critical and major bugs are fixed (see “Overall defects fixed percentage” metric). Final requirements coverage by tests (see “Requirements coverage by tests” metric) should be at least 80%.

∙ Testing start criteria: new build.

∙ Testing pause criteria: critical path test must begin only after 100% success of testcases on the smoke test (see “Test cases success percentage”); test process may

be paused is with at least 25% test-cases executed there is at least 50% failure rate

(see “Stop-factor” metric).

∙ Testing resumption criteria: more than 50% of bugs found during the previous

iteration are fixed (see “Ongoing defects fixed percentage” metric).

∙ Testing finish criteria: more than 80% planned for the current iteration test cases are executed (see “Test-cases execution percentage”).

### 6. Resources

∙ Software: four virtual machines (two with Windows 10 Ent x64, two with Linux Ubuntu 18 LTS x64), Java version (see DS-1.1).

∙ Hardware: two standard workstations (8GB RAM, i7 3GHz).

∙ Personnel: Two senior developers with testing experience (100% workload during all project time). Roles: team lead, senior developer. Two testers with Java knowledge (100% workload during all project time). Role: testers.

∙ Time: 14 working days (112 hours).

∙ Finances: according to the approved budget