|  |  |
| --- | --- |
|  | «File Converter» Project |
| Test Plan SAMPLE  **Project Documentation** |
| **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** | 02 03 - Test Plan Sample.docx |

Contents

1. [Project scope and main goals 3](#_bookmark0)
2. [Requirements to be tested 3](#_bookmark1)
3. [Requirements NOT to be tested 3](#_bookmark2)
4. [Test strategy and approach 3](#_bookmark3)
   1. [General approach 3](#_bookmark4)
   2. [Functional testing levels 3](#_bookmark5)
5. [Criteria 4](#_bookmark6)
6. [Resources 4](#_bookmark7)
7. [Schedule 4](#_bookmark8)
8. [Roles and responsibilities 4](#_bookmark9)
9. [Risk evaluation 4](#_bookmark10)
10. [Documentation 5](#_bookmark11)
11. [Metrics 5](#_bookmark12)

# Project scope and main goals

Correct automated conversion of text documents in different source encodings to one destination encoding with performance significantly higher than human performance during the same actions.

# Requirements to be tested

See referenced sections in “File Converter Requirements.docx”:

* UR-1.\*: smoke test.
* UR-2.\*: smoke test, critical path test.
* UR-3.\*: critical path test.
* BR-1.\*: smoke test, critical path test.
* QA-2.\*: smoke test, critical path test.
* L-4: smoke test.
* L-5: smoke test.
* DS-\*: smoke test, critical path test.

# Requirements NOT to be tested

See referenced sections in “File Converter Requirements.docx”:

* SC-1: the application is a console one by design.
* SC-2, L-1, L-2: the application is developed with proper PHP version.
* QA-1.1: this performance characteristic is at the bottom border of typical operations performance for such applications.
* L-3: no implementation required.
* L-6: no implementation required.

# Test strategy and approach

## General approach

The application is to be configured once by an experienced specialist and later used by end users, for whom only one operation is available – placing the file into the input directory. Therefore, issues of usability, security, etc. not explored during testing.

## Functional testing levels

* Smoke test: automated with batch files under Windows and Linux.
* Critical path test: executed manually.
* Extended test: not executed as the probability of defects detection on this level is negligibly small.

Due to the team cross-functionality, a significant contribution to quality improvement can be expected from the code review combined with manual testing using the white box method. Unit-testing will not be applied due to extreme time limitations.

# 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](#_bookmark13)” metric) if 100% of critical and major bugs are fixed (see “[Overall defects fixed percentage](#_bookmark14)” metric). Final requirements coverage by tests (see “[Requirements coverage by tests](#_bookmark18)” metric) should be at least 80%.
* Testing start criteria: new build.
* Testing pause criteria: critical path test must begin only after 100% success of test- cases on the smoke test (see “[Test cases success percentage](#_bookmark13)”); test process may be paused is with at least 25% test-cases executed there is at least 50% failure rate (see “[Stop-factor](#_bookmark16)” metric).
* Testing resumption criteria: more than 50% of bugs found during the previous iteration are fixed (see “[Ongoing defects fixed percentage](#_bookmark15)” metric).
* Testing finish criteria: more than 80% planned for the current iteration test cases are executed (see “[Test-cases execution percentage](#_bookmark17)”).

# Resources

* Software: four virtual machines (two with Windows 10 Ent x64, two with Linux Ubuntu 18 LTS x64), two PHP Storm licenses (latest version available).
* Hardware: two standard workstations (8GB RAM, i7 3GHz).
* Personnel:
  + One senior developer with testing experience (100% workload during all project time). Roles: team lead, senior developer.
  + One tester with PHP knowledge (100% workload during all project time). Role: tester.
* Time: one workweek (40 work hours).
* Finances: according to the approved budget.

# Schedule

* 25.05 – requirements testing and finalizing.
* 26.05 – test-cases and scripts for automated testing creation.
* 27.05-28.05 – main testing stage (test-cases execution, defect reports creation).
* 29.05 – testing finalization, reporting.

# Roles and responsibilities

* Senior developer: participation in requirements testing and code review.
* Tester: documentation creation, test-cases execution, participation in code-review.

# Risk evaluation

* Personnel (low probability): if any team member is inaccessible, we can contact the representatives of the “Cataloger” project to get a temporary replacement (the commitment from the “Cataloger” PM John Smith was received).
* Time (high probability): the customer has indicated a deadline of 01.06, therefore time is a critical resource. It is recommended to do our best to complete the project by 28.05 so that one day (29.05) remains available for any unexpected issues.
* Other risks: no other specific risks have been identified.

# Documentation

* Requirements. Responsible person – tester, deadline – 25.05.
* Test cases and defect reports. Responsible – tester, creation period – 26.05-28.05.
* Test result report. Responsible person – tester, deadline – 29.05.

# Metrics

* Test cases success percentage:

𝑇 𝑆𝑃 = 𝑇𝑆𝑢𝑐𝑐𝑒𝑠𝑠 ∙ 100%, where

𝑇𝑜𝑡𝑎𝑙

𝑇

𝑇 𝑆𝑃 – percentage of successfully passed test cases,

𝑇 𝑆𝑢𝑐𝑐𝑒𝑠𝑠 – quantity of successfully passed test cases,

𝑇𝑇𝑜𝑡𝑎𝑙 – total quantity of executed test cases. Minimally acceptable borders:

* Beginning project phase: 10%.
* Main project phase: 40%.
* Final project phase: 80%.
* Overall defects fixed percentage:

𝐷𝐹𝑇𝑃 = 𝐷 𝐶𝑙𝑜𝑠𝑒𝑑 ∙ 100%, where

𝐿𝑒𝑣𝑒𝑙

𝐿𝑒𝑣𝑒𝑙

𝐷

𝐹𝑜𝑢𝑛𝑑

𝐿𝑒𝑣𝑒𝑙

𝐷𝐹𝑇𝑃 – overall defects fixation percentage by 𝐿𝑒𝑣𝑒𝑙 during all project lifetime,

𝐿𝑒𝑣𝑒𝑙

𝐷𝐶𝑙𝑜𝑠𝑒𝑑 – quantity of defects of 𝐿𝑒𝑣𝑒𝑙 fixed during all project lifetime,

𝐿𝑒𝑣𝑒𝑙

𝐷𝐹𝑜𝑢𝑛𝑑 – quantity of defects of 𝐿𝑒𝑣𝑒𝑙 found during all project lifetime.

𝐿𝑒𝑣𝑒𝑙

Minimally acceptable borders:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | | Defect severity | | | |
| Minor | Medium | Major | Critical |
| Project phase | Beginning | 10% | 40% | 50% | 80% |
| Main | 15% | 50% | 75% | 90% |
| Final | 20% | 60% | 100% | 100% |

* Ongoing defects fixed percentage:

𝐷𝐹𝐶𝑃 = 𝐷 𝐶𝑙𝑜𝑠𝑒𝑑 ∙ 100%, where

𝐿𝑒𝑣𝑒𝑙

𝐿𝑒𝑣𝑒𝑙

𝐷

𝐹𝑜𝑢𝑛𝑑

𝐿𝑒𝑣𝑒𝑙

𝐷𝐹𝐶𝑃 – defects fixation percentage by 𝐿𝑒𝑣𝑒𝑙 (defects found in the previous build and fixed in the current build),

𝐿𝑒𝑣𝑒𝑙

𝐷𝐶𝑙𝑜𝑠𝑒𝑑 – quantity of defects of 𝐿𝑒𝑣𝑒𝑙 fixed in the current build,

𝐿𝑒𝑣𝑒𝑙

𝐷𝐹𝑜𝑢𝑛𝑑 – quantity of defects of 𝐿𝑒𝑣𝑒𝑙 found in the previous build.

𝐿𝑒𝑣𝑒𝑙

Minimally acceptable borders:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | | Defect severity | | | |
| Minor | Medium | Major | Critical |
| Project phase | Beginning | 60% | 60% | 60% | 60% |
| Main | 65% | 70% | 85% | 90% |
| Final | 70% | 80% | 95% | 100% |

* Stop-factor:

𝑆 = {𝑌𝑒𝑠, 𝑇𝐸 ≥ 25% && 𝑇𝑆𝑃 < 50% , where

𝑁𝑜, 𝑇𝐸 < 25% || 𝑇𝑆𝑃 ≥ 50%

𝑆 – decision to pause the testing process,

𝑇𝐸 – current 𝑇𝐸 value,

𝑇 𝑆𝑃 – current 𝑇 𝑆𝑃 value.

* Test-cases execution percentage:

𝐸 𝑇𝐸𝑥𝑒𝑐𝑢𝑡𝑒𝑑

𝑇 = ∙ 100%, where

𝑇𝑃𝑙𝑎𝑛𝑛𝑒𝑑

𝑇𝐸 – test-cases execution percentage,

𝑇𝐸𝑥𝑒𝑐𝑢𝑡𝑒𝑑 – quantity of executed test-cases,

𝑇𝑃𝑙𝑎𝑛𝑛𝑒𝑑 – quantity of planned (to execution) test-cases. Levels (borders):

o Minimal: 80%.

o Desired: 95%-100%.

* Requirements coverage by tests:

𝑅𝐶 = 𝑅𝐶𝑜𝑣𝑒𝑟𝑒𝑑 ∙ 100%, where

𝑇𝑜𝑡𝑎𝑙

𝑅

𝑅𝐶 – requirements coverage by tests (percentage),

𝑅𝐶𝑜𝑣𝑒𝑟𝑒𝑑 – quantity of requirements covered with test-cases,

𝑅𝑇𝑜𝑡𝑎𝑙 – overall quantity of requirements. Minimally acceptable borders:

* Beginning project phase: 40%.
* Main project phase: 60%.
* Final project phase: 80% (90%+ recommended).