Telerik Academy Learning System

Release 5.7

Master Test Plan

Version 1.0

**Draft**

November 2015

Submitted by

Team Giant

***TABLE OF CONTENST***

|  |  |  |
| --- | --- | --- |
| 1. | Test Plan Identifier ………………………………………………………… | [3](#TEST_PLAN_IDENTIFIER) |
| 2. | References ………………………………………………………………….... | [3](#REFERENCES) |
| 3. | Introduction ………………………………………………………………….. | [3](#INTRODUCTION) |
|  | 3.1 Scope………………………..……………………………………………….. | [3](#Scope) |
|  | 3.2 Test Objectives……..…………………..………….……..………………. | [3](#Test_Objectives) |
|  | 3.3 Testing Goals…..……………………………………………………………. | [4](#TestingGoals) |
| 4. | Test Items ………………………………………………………………………. | [4](#TEST_ITEMS_FUNCTIONS) |
|  | 4.1 Admin panel ………………………………………………………...……… | [4](#Admin_panel) |
|  | 4.2 Front End.………………….………………………………………………… | [5](#FrontEnd) |
| 5. | Software Risk Issues ……………………………………………………….. | [5](#SOFTWARE_RISK_ISSUES) |
| 6. | Features to be Tested ……………………………………….…………… | [5](#FEATURES_TO_BE_TESTED6) |
| 7. | Features not to be Tested …………………………………………….. | [6](#FEATURES_NOT_TO_BE_TESTED7) |
| 8. | Approach ………………………………………………………………………. | [6](#APPROACH) |
|  | 8.1 Testing Strategy………………………………………………………………………. | [6](#a8) |
|  | 8.2 Testing Tools…………………………………………………………………………… | [6](#b8) |
| 9. | Item Pass/Fail Criteria ……………………………………………………. | [7](#ITEM_PASS_FAIL_CRITERIA) |
| 10. | Suspension Criteria and Resumption Requirements……….. | [7](#SUSPENSION_CRITERIA_AND_RESUMPTION_REQUI) |
| 11. | Test Deliverables ……………………………………………………………. | [7](#TEST_DELIVERABLES) |
| 12. | Remaining Test Tasks …………………………………………………….. | [8](#REMAINING_TEST_TASKS) |
| 13. | Environmental Needs …………………………………………………….. | [8](#ENVIRONMENTAL_NEEDS) |
|  | 13.1 Available Client-side Environments………………………………….. | [8](#AvailableClientideEnvironments) |
|  | 13.2 Available Server-side Environments…………………………………. | [8](#AvailableServerSideEnvironments) |
| 14. | Staffing and Training Needs …………………………………………… | [9](#STAFFING_AND_TRAINING_NEEDS) |
| 15. | Responsibilities ……………………………………………………………… | [9](#RESPONSIBILITIES) |
| 16. | Schedule ……………………………………………………………………….. | [9](#SCHEDULE) |
| 17. | Planning Risks and Contingencies ………………………………….. | [9](#PLANNING_RISKS_AND_CONTINGENCIES) |
| 18. | Approvals ………………………………………………………………………. | [10](#APPROVALS) |
| 19. | Glossary ………………………………………………………………………… | [10](#GLOSSARY) |
| 20. | Revision History …………………………………………………………….. | [10](#REVISION_HISTORY) |

***1***  ***TEST PLAN IDENTIFIER***

**TALS -MTP v5.7 (build 20151119.44a964b)**

Note, the structure of this document is primarily based on the IEEE 829-1998 Standard for Software Test Documentation. Additional reference standards include IEEE 1008 (Unit Testing), 1012 & 1059 (Validation & Verification) and 1074 (Software Life Cycle process).

***2*** ***REFERENCES***

Distribution of Teamwork.xlsx

<http://telerikacademy.com/Courses/Teamwork/List>

***3*** ***INTRODUCTION***

***3.1 Scope***

The purpose of this document is to describe the overall test plan and strategy for testing the Telerik Academy Learning System web API located at <http://stage.telerikacademy.com/>. The approach described in this document provides the framework for all testing related to this application. Individual test cases will be written for each version of the application that is released. This document will also be updated as required for each release.

***3.2*** ***Test Objectives***

The quality objectives of testing the Telerik Academy Learning System web API are to ensure complete validation of the business and software requirements:

* Verify software requirements are complete and accurate
* Perform detailed test planning
* Identify testing standards and procedures that will be used on the project
* Prepare and document test scenarios and test cases
* Regression testing to validate that unchanged functionality has not been affected by changes
* Manage defect tracking process
* Provide test metrics/testing summary reports

***3.3*** ***Testing Goals***

The goals in testing this application include validating the quality, usability, reliability and performance of the application. Another goal is to make the tests repeatable for use in regression testing during the project lifecycle, and for future application upgrades. Quality software is reasonably bug-free, meets requirements and/or expectations, and is maintainable.

***4 TEST ITEMS (FUNCTIONS)***

The scope of this Testing activity will include:

Telerik Academy Learning System - v5.7 (build 20151119.44a964b) Web API - Courses and Lectures

***4.1 Admin panel***

* Courses and Lectures
* Course Categories
* Category Hierarchy
* Students in Courses
* Groups
* Homework
* Courses, Instances and Lectures
* Single Courses (Kids Academy)
* Licenses
* Students Statistics
* Students Statistics (by age)
* Course's Surveys
* Statistics for course's surveys
* Course’s [unsubscribe](https://www.google.bg/search?safe=off&q=unsubscribe&spell=1&sa=X&ved=0ahUKEwi9ieLwz6vJAhWHWiwKHYUrCO8QvwUIGCgA)d students

***4.2*** ***Front End***

* Navigation between different courses
* My Courses
* Archive

***5*** ***SOFTWARE RISK ISSUES***

There are several parts of the project that are not within the control of the Reassigned Sales application but have direct impacts on the process and must be checked as well.

* Team member get sick and need some days to be ready for work.
* The project schedule is too tight, it's hard to complete this project on time
* Test Manager has poor management skill
* A lack of cooperation negatively affects your employees' productivity
* Wrong budget estimate and cost overruns

***6*** ***FEATURES TO BE TESTED***

Testing will consist of several phase, each phase may or may not include testing of anyone or more of the following aspects of the Telerik Academy Learning System Web site (listed alphabetically):

* Accessibility
* Accuracy
* Availability
* Coding standards
* Compatibility
* Content
* Functional
* Legal
* Navigation
* Performance
* Reliability
* Scalability
* Security
* Suitability
* Usability

***7*** ***FEATURES NOT TO BE TESTED***

It is the intent that all of the individual test cases contained in each test plan will be performed. However, if time does not permit, some of the low priority test cases may be dropped.

***8*** ***APPROACH***

***8.1. Testing Strategy***

The testing philosophy is risk-based testing, i.e. each test case will be prioritized as High, Medium, or Low priority and then scheduled accordingly (Highest first). Exceptions to this general rule might include instances where:

* A large number of low priority test cases can be executed using a small amount of resources;
* A lower priority test is a pre-requisite of another higher priority test - e.g. an expensive and high priority usability test might necessitate many of the inexpensive low priority navigational tests to have passed;
* Due to the lack of comprehensive requirements, navigational and functional tests may be scheduled first, so as to allow the testers the opportunity to gain familiarity with the Web site (thereby also allowing them to developing pseudo requirements).
* Web site’s source code will be frozen while being tested. If critical fixes that are blocking the testing efforts arise, changes will not be applied while a unit of code is being tested.
* Regression testing will be done during any level of testing (Unit, Integration, System, or Acceptance). The regression testing will be based on severity of defects detected. If critical fix is done all tests until the fix obtain priority High and regression testing starts.

***8.2. Testing tools***

The testing will use a combination of manual and automated testing. Automated tools:

* Selenium
* Selenium Web Driver
* Telerik Test Studio
* Other tools (if required and if the training is applied in short period of time without breaking the time schedule plan)

Basic metrics will be kept for test effort (i.e. hours), test cases executed, and incidents.

***9*** ***ITEM PASS/FAIL CRITERIA***

***9.1. Entry/Exit Criteria***

The entry/exit criteria for each phase of testing must be met before the next phase can commence. Formal approval will be granted by the Product Owner.

***9.2. Defect Severity Classification***

The actual terminologies, and their meaning, can vary depending on people, projects, organizations, or defect tracking tools, but the following is a normally accepted classification.

* Critical: The defect affects critical functionality or critical data. It does not have a workaround. Example: Unsuccessful installation, complete failure of a feature.
* High: The defect affects major functionality or major data. It has a workaround but is not obvious and is difficult. Example: A feature is not functional from one module but the task is doable if 10 complicated indirect steps are followed in another module/s.
* Medium: The defect affects minor functionality or non-critical data. It has an easy workaround. Example: A minor feature that is not functional in one module but the same task is easily doable from another module.
* Low: The defect does not affect functionality or data. It does not even need a workaround. It does not impact productivity or efficiency. It is merely an inconvenience. Example: Petty layout discrepancies, spelling/grammatical errors.

***9.3. Defect Priority Classification***

* Critical: Must be fixed in the next build.
* High: Must be fixed in any of the upcoming builds but should be included in the release.
* Medium: May be fixed after the release / in the next release.
* Low: May or may not be fixed at all.

***10*** ***SUSPENSION CRITERIA AND RESUMPTION REQUIREMENTS***

In general, testing will only stop if the Web site Under Test (WUT) becomes unavailable. If testing is suspended due to the Web site becoming unavailable, testing will be resumed once access to the Web site is reestablished.

Certain individual test cases may be suspended, skipped or reduced if prerequisite tests have previously failed e.g. usability testing may be skipped if a significant number of Web page navigational tests fail.

***11*** ***TEST DELIVERABLES***

The following documents will be generated as a result of these testing activities:

* Master test plan (MTP - this document)
* Individual test plans for each phase of the testing cycle (as an Appendix to the MTP)
* Combination incident/test summary reports for each phase
* Test log for each phase
* Automated test scripts and supporting test data

With the exception of the automated test scripts, all documents will be delivered as Microsoft Office 2007 documents.

***12*** ***REMAINING TEST TASKS***

|  |  |  |
| --- | --- | --- |
| TASK | Assigned To | Status |
| Create Acceptance Test Plan | TM, PM, Client |  |
| Create System/Integration Test Plan | TM, PM, Dev. |  |
| Define Unit Test rules and Procedures | Dev. |  |
| Define Turnover procedures for each level | TM, Dev |  |

***13*** ***ENVIRONMENTAL NEEDS***

There are essentially two parts to the Telerik Academy Learning System WebApi in production:

***13.1.*** ***Available Client-side Environments***

* ***OS***: Windows 7, 8.1, 10
* ***Browser***: Internet Explorer 8, 9, 10, Apple Safari 5.1, Mozilla Firefox 27.0, Google Chrome 32.0, Opera 27 and later should be available to the tester.

***13.2.*** ***Available Server-side Environments***

* ***OS***: Windows Server 2012 R2
* ***Web Server***: MS IIS 8.5
* ***SQL Server***: MS SQL Server 2012
* ***CPU:*** Intel Xeon CPU @ 3.10 GHz (Quad Core)
* ***RAM***: 8GB
* ***Storage***: 100GB

***14*** ***STAFFING AND TRAINING NEEDS***

Continuously training Team Giant at each level of testing.

***15*** ***RESPONSIBILITIES***

Biser Hristov

Blagoy Shokov

Ventsislav Ivanov

Dushka Dragoeva

Plamen Kostadinov

***16*** ***SCHEDULE***

The estimated time line for this project is 3 months. Time has been allocated within the project plan for the following testing activities. The specific dates and times for each activity are defined in the project plan time line. The persons required for each process are detailed in the project time line:

* Review of Requirements document by test team personnel
* Development of Master test plan by test manager and test with time allocated for at least two reviews of the plan.
* Development of System/Integration and Acceptance test plans by test manager and team with time allocated for at least two reviews of the plans.
* Time allocated for both System/Integration and Acceptance test processes.

***17*** ***PLANNING RISKS AND CONTINGENCIES***

The following seeks to identify some of the more likely project risks and propose possible contingencies:

* Web site becomes unavailable – Testing will be delayed until this situation is rectified
* Web testing software is not available/does not work (e.g. Web site uses cookies and tool cannot handle cookies) - This will delay the introduction of automated testing and result in more manual testing
* A large number of defects/incidents makes it functionally impossible to run all of the test cases – As many test cases as possible will be executed, The PM will ultimately make the decision as to whether the number of defects/incidents warrants delaying the implementation of the production version.
* Not enough time to complete all test cases. If time cannot be extended, individual test cases will be skipped, starting with the lowest priority.

***18*** ***APPROVALS***

Telerik QA trainers

Product Owner must approve this plan

***19*** ***GLOSSARY***

TALS – Telerik Academy Learning System

***20*** ***REVISION HISTORY***

|  |  |  |  |
| --- | --- | --- | --- |
| Revision # | Revision Date | **Description of Change** | **Author** |
| 1.0 | 11/27/2015 | First Draft | Team Giant |
|  |  |  |  |