QA Team “Apple”

Telerik Academy 2014

**Requirements for Evaluation & Teamwork Modules**

Version 3.1

November 16, 2014

# Table of Contents

[Table of Contents 2](#_Toc403925529)

[1. INTRODUCTION 3](#_Toc403925530)

[2. FUNCTIONAL REQUIREMENTS 3](#_Toc403925531)

[2.1. Administrator functionality 3](#_Toc403925532)

[2.1.1. Evaluation modules 3](#_Toc403925533)

[2.1.2. Teamwork modules 5](#_Toc403925534)

[2.2. Front-end user functionality 7](#_Toc403925535)

[3. TECHNICAL REQUIREMENTS 8](#_Toc403925536)

[4. ENVIRONMENT REQUIREMENTS 8](#_Toc403925537)

[5. HARDWARE REQUIREMENTS 8](#_Toc403925538)

[6. SOFTWARE QUALITIES 9](#_Toc403925539)

[7. GLOSSARY 10](#_Toc403925540)

# INTRODUCTION

This document contains the detailed requirements for testing all “Evaluation” and “Teamwork” modules in [Telerik Academy Student System (TASS)](http://test.telerikacademy.com/).

# FUNCTIONAL REQUIREMENTS

## Administrator functionality

### Evaluation modules

|  |  |  |
| --- | --- | --- |
| *Priority* | *Requirement* | *Module* |
| *1* | *The Admin UI should allow the administrator to create new practical exam information for some course, to retrieve and edit existing exam information and also to delete some exam.* | [*Practical Exams*](http://test.telerikacademy.com/Administration_Courses/CoursePracticalExams) |
| *1* | *The Admin UI should allow a student to be assigned to practical exam, or some already assigned student to be deleted from the event.* |
| *1* | *The Admin UI should allow calculation of the results for each one practical exam.* |
| *1* | *The Admin UI should allow practical exam’s points to be transferred from* [*BGCoder*](http://bgcoder.com/) *to TASS.* |
| *3* | *The Admin UI should allow all information about the practical exams to be exported to an Excel file.* |
| *2* | *The Admin UI should allow the administrator to upload student practical exam file, to retrieve and edit some existing file upload and also to delete some uploaded file.* | [*Practical Exam Files*](http://test.telerikacademy.com/Administration_Courses/CoursePracticalExamFiles) |
| *3* | *The Admin UI should allow all information about uploaded files to be exported to an Excel file.* |
| *3* | *The Admin UI should allow all uploaded exam files to be exported to an archive file.* |
| *1* | *The Admin UI should allow the administrator to evaluate student practical exam, to retrieve and edit evaluation and also to delete some evaluation.* | [*Practical Exam Evaluations*](http://test.telerikacademy.com/Administration_Courses/CoursePracticalExamFilesEvaluations) |
| *3* | *The Admin UI should allow all exam evaluations to be exported to an Excel file.* |
| *1* | *The Admin UI should allow the administrator to create new test exam information for some course, to retrieve and edit existing exam information and also to delete some exam with all information about assigned students.* | [*Test Exams*](http://test.telerikacademy.com/Administration_Courses/CourseTestExams) |
| *1* | *The Admin UI should allow a student to be assigned to test exam or some already assigned student to be deleted from the event.* |
| *1* | *The Admin UI should allow calculation of the results for each one test exam.* |
| *2* | *The Admin UI should allow the administrator to upload Excel file with students’ records.* | [*Students’ Records*](http://test.telerikacademy.com/Administration_Courses/StudentsRecords) |
| *3* | *The Admin UI should allow the administrator to download an already uploaded Excel file.* |
| *3* | *The Admin UI should allow all information about the uploaded files to be exported in an Excel file.* |
| *3* | *The Admin UI should allow the administrator to export in an Excel file the whole or part of information for student’s results from some course or group of courses.* | [*Export Results*](http://test.telerikacademy.com/Administration_Courses/ExportResults) |
| *2* | *The Admin UI should allow the administrator to add bonus points to some students or group of students (from an Excel file), to retrieve and edit existing bonus information and also to delete some bonus.* | [*Bonuses*](http://test.telerikacademy.com/Administration_Courses/UsersInCoursesBonuses) |
| *3* | *The Admin UI should allow all information about the given bonuses to be exported to an Excel file.* |
| *2* | *The Admin UI should allow the administrator to retrieve and edit existing homework evaluation information and also to delete some homework evaluation.* | [*Homework Evaluation*](http://test.telerikacademy.com/Administration_Courses/HomeworkEvaluation) |
| *3* | *The Admin UI should allow all information about the evaluation to be exported to an Excel file.* |
| *3* | *The Admin UI should allow the administrator to add student helper for some course, to retrieve and edit existing helper information and also to delete some helper.* | [*Helpers*](http://test.telerikacademy.com/Administration_Courses/UserInCourseHelpers) |
| *3* | *The Admin UI should allow all information about the helpers to be exported to an Excel file.* |
| *3* | *The Admin UI should allow the administrator to generate pie chart statistics for some course, lecture and student.* | [*Homework Statistics*](http://test.telerikacademy.com/Administration_Courses/HomeworkStatistics) |
| *2* | *The Admin UI should allow the administrator to retrieve the existing information about the students’ homework evaluation reactions.* | [*Homework Evaluation Reactions*](http://test.telerikacademy.com/Administration_Courses/HomeworkEvaluationReactions) |
| *3* | *The Admin UI should allow all information about the homework evaluation reactions to be exported to an Excel file.* |
| *1* | *The Admin UI should allow the administrator to add group of students for practical exam by uploading an Excel file with information about the students.* | [*Add Students for Practical Exam*](http://test.telerikacademy.com/Administration_Courses/CoursePracticalExams/AddStudents) |
| *1* | *The Admin UI should allow the administrator to add group of students for test exam by uploading an Excel file with information about the students.* | [*Add Students for Test Exam*](http://test.telerikacademy.com/Administration_Courses/CourseTestExams/AddStudents) |

### Teamwork modules

|  |  |  |
| --- | --- | --- |
| *Priority* | *Requirement* | *Module* |
| *2* | *The Admin UI should allow the administrator to create new teamwork for some course, to retrieve and edit existing teamwork information and also to delete some teamwork.* | [*Teamworks*](http://test.telerikacademy.com/Administration_Teamwork/Teamworks) |
| *2* | *The Admin UI should allow the administrator to create new team for some teamwork, to retrieve and edit existing team information and also to delete some team.* |
| *2* | *The Admin UI should allow the administrator to add some student in team and also to delete some student from the team.* |
| *2* | *The Admin UI should allow the administrator to import an Excel file with students’ teamwork evaluations.* |
| *3* | *The Admin UI should allow all information about one team to be exported to an Excel file.* |
| *2* | *The Admin UI should allow the administrator to send suggestion for teamwork to some student, to retrieve and edit existing teamwork suggestions and also to delete teamwork suggestion.* | [*Team Member Suggestions*](http://test.telerikacademy.com/Administration_Teamwork/TeamMemberSuggestions) |
| *3* | *The Admin UI should allow all information about teamwork suggestions to be exported to an Excel file.* |
| *2* | *The Admin UI should allow the administrator to import teams for some teamwork and to add students to them by uploading an Excel file with information about the students.* | [*Import Teams*](http://test.telerikacademy.com/Administration_Teamwork/ImportTeams) |
| *2* | *The Admin UI should allow the administrator to create new teamwork type, to retrieve and edit existing teamwork type information and also to delete some teamwork type.* | [*Team Name Suggestion*](http://test.telerikacademy.com/Administration_Teamwork/TeamNameSuggestions)*s* |
| *2* | *The Admin UI should allow the administrator to create new team name for some teamwork type, to retrieve and edit existing team name information and also to delete some team name.* |
| *3* | *The Admin UI should allow all team names for some teamwork type to be exported to an Excel file.* |
| *2* | *The Admin UI should allow the administrator to import a list with team names by uploading an Excel file.* | [*Import Team Names*](http://test.telerikacademy.com/Administration_Teamwork/TeamNameSuggestions/ImportTeamNameSuggestions) |
| *2* | *The Admin UI should allow the administrator to evaluate student teamwork, to retrieve and edit existing teamwork evaluation and also to delete some teamwork evaluation.* | [*Team Member Evaluations*](http://test.telerikacademy.com/Administration_Teamwork/TeamMemberEvaluations) |
| *3* | *The Admin UI should allow the administrator to evaluate group of students by uploading one or group of Excel files.* |
| *2* | *The Admin UI should allow the administrator to evaluate and comment some student teamwork, to retrieve and edit existing teamwork evaluation and also to delete some teamwork evaluation.* | [*Team Member All Evaluations*](http://test.telerikacademy.com/Administration_Teamwork/TeamMemberAllEvaluations) |
| *3* | *The Admin UI should allow the administrator to evaluate group of students by uploading one or group of Excel files.* |
| *3* | *The Admin UI should allow all teamwork evaluation has to be exported to an Excel file.* |

## Front-end user functionality

|  |  |  |
| --- | --- | --- |
| *Priority* | *Requirement* | *Area* |
| *1* | *The Front-end UI should allow the user to upload some homework (or teamwork) as archive file via course page.* | *Homework (Teamwork) Upload* |
| *1* | *The Front-end UI should allow the user to re-upload some homework (or teamwork) as archive file via course page.* |
| *2* | *The Front-end UI should allow the user to download the last uploaded homework (or teamwork) file via course page.* |
| *1* | *The Front-end UI should allow the user to evaluate and comment the homework (or teamwork) of some other student via course page.* | *Homework Evaluation* |
| *2* | *The Front-end UI should allow the user to access the comments for his uploaded homework (or teamwork) via course page.* | *Homework Comments & Grading* |
| *2* | *The Front-end UI should allow the user to evaluate the helpfulness of the comment to his uploaded homework (or teamwork) via course page.* |
| *1* | *The Front-end UI should allow the user to sign-up and choose date for practical and test exam.* | *Exam Sign-up* |
| *1* | *The Front-end UI should allow the user to upload some practical exam as archive file.* | *Exam Upload* |
| *1* | *The Front-end UI should allow the user to re-upload some practical exam as archive file.* |
| *2* | *The Front-end UI should allow the user to download the last uploaded practical exam file.* |
| *1* | *The Front-end UI should allow the user to evaluate and comment the practical exam of some other student.* | *Exam Evaluation* |
| *1* | *The Front-end UI should allow the user to check practical and test exam results from his personal information page.* | *Exam Results* |
| *1* | *The Front-end UI should allow the user to sign-up or not for teamwork.* | *Teamwork Sign-up* |
| *3* | *The Front-end UI should allow the user to change his choice for participation in teamwork.* |
| *2* | *The Front-end UI should allow the user to evaluate and comment some teammate from the list with teamwork.* | *Teammate Grading* |
| *2* | *The Front-end UI should allow the user to check the final teamwork scores from the list with teamwork.* |

# TECHNICAL REQUIREMENTS

* Anti-cheat security
* Score data alteration security
* Precise estimation of exam result and course total score

# ENVIRONMENT REQUIREMENTS

* Administrator access
* Server internet access

# HARDWARE REQUIREMENTS

* Server reliability

# SOFTWARE QUALITIES

* **User-friendliness** – since users of the TASS will be of all sorts and types and not all of them will be very experienced with computers, it is essential that the application be as user-friendly as possible.
* **Reliability** – reliability of the TASS is not expected to be essential, but nonetheless important. Accepted rate of failure is 1 crash per fortnight.
* **Performance** – it is highly essential that the TASS application runs efficiently. Any request should not take more than 5 seconds.
* **Reusability** – reusability will not be important in this application.
* **Extensibility** – over time the system will be enhanced with small features, therefore the application should be extensible.
* **Robustness** – it is expected that the TASS application does not crash if a user inputs wrong data, makes mistakes or if links are broken.
* **Maintainability** – “Telerik, Inc.” expects that the TASS will be used by different kinds of users over varying periods of time. Routine maintenance may be necessary at times; hence the system should be easily maintainable.
* **Reparability** – since it is possible that the application may not be fault-free, TASS application should be designed and implemented such that it can be easily repaired in the future.
* **Understandability** – to support evolvability, reparability, and maintainability, it is imperative that all aspects of the TASS (design, code and test cases) be easily understandable, even to future developers who are not currently involved in the creation of the TASS.
* **Interoperability** – since the TASS does not have to interoperate with any other software, interoperability is not a concern.
* **Productivity** – productivity is not a concern for the TASS.
* **Visibility** – visibility is not a concern in the development of the TASS.

# GLOSSARY

* **Telerik Academy Student System (TASS)** – a student system developed for the purpose of training at the Academy of Telerik.
* **User** – a student in Telerik Academy.
* **Admin UI** – it is used for all administration related functionality like giving user roles, creating and deleting of courses, lectures, exams and teamwork, assigning students to them, practical exam and test evaluations, bonuses, anti-cheat, certificates, etc.
* **Front-end UI** – the interface between the end-user of an application and the software behind it.

