

# Testify Test Plan

## **A. Identification and classification:**

Test Case 1

System: Multiple Choice Questions

Create question

Severity: 1

## **B. Instructions:**

1. Execute the instructor JAR
2. Click create test
3. Create a new Test
3. Create a multiple choice question
4. Fill in all fields

## **C. Expected result:**

Choice of language should have been disabled

When entering choices each should appear only once

## **A. Identification and classification:**

Test Case 2

System: Question deletion

Delete question

Severity: 2

## **B. Instructions:**

1. Execute the instructor JAR
2. Click create test
3. Create a new test
4. Create a question of any type
5. Create a second one so that there are two
6. Delete the second one

## **C. Expected result:**

The only question visible should be question 1

## **A. Identification and classification:**

Test Case 3

System: Multiple Choice Questions

Delete a choice of answer

Severity: 2

## **B. Instructions:**

1. Execute the instructor JAR
2. Click create test
3. Create a new Test
3. Create a multiple choice question
4. Fill in all fields
5. Delete one of the answer choices

**C. Expected result:**

The choice that you selected should be deleted

**A. Identification and classification:**

Test Case 4

System: Question Creation

Create question

Severity: 1

**B. Instructions:**

1. Execute the instructor JAR
2. Click create test
3. Create a new Test
3. Create a multiple choice question
4. Fill in all fields
5. Create a new question and fill in all fields, make sure you don't make them the same
6. Cycle back and forth between questions

**C. Expected result:**

The information should be consistent with what you had previously entered.

**A. Identification and classification:**

Test Case 5

System: Short answer question

Create question

Severity: 1

**B. Instructions:**

1. Execute the instructor JAR
2. Click create test
3. Create a new Test
3. Create a short answer question
4. Fill in all fields

**C. Expected result:**

A new short answer question should be created.

**A. Identification and classification:**

Test Case 6

System: Programming question

Create question

Severity: 1

**B. Instructions:**

1. Execute the instructor JAR
2. Click create test
3. Create a new Test
3. Create a programming question
4. Fill in all fields

**C. Expected result:**

Choices should be disabled, language and file should be enabled. A new programming question should be created.

**A. Identification and classification:**

Test Case 7

System: Create Test

Cycling between questions

Severity: 2

**B. Instructions:**

1. Execute the instructor JAR
2. Click create test
3. Create a new Test
3. Create several questions
4. Fill in all fields
5. Cycle back and forth between them at random using previous and next

**C. Expected result:**

It should progress in a logical fashion, ie the next of the final question is not the first question, 3 is not the prev of 2, etc

**A. Identification and classification:**

Test Case 8

System: Create Test

Create new test without saving

Severity: 2

**B. Instructions:**

1. Execute the instructor JAR
2. Click create test
3. Create a new Test
3. Create a few questions
4. Create a new test without saving the previous

**C. Expected result:**

New blank test should appear, old one should not be accessible

**A. Identification and classification:**

Test Case 9

System: Create test

Loading and saving tests

Severity: 1

**B. Instructions:**

1. Execute the instructor JAR
2. Click create test
3. Create a new Test
3. Create a few questions
4. Save the test
5. Close window
6. Load the test you just created.

**C. Expected result:**

The test should display with questions filled in.

**A. Identification and classification:**

Test Case 10

System: Create Test

Point assignment

Severity: 2

**B. Instructions:**

1. Execute the instructor JAR
2. Click create test
3. Assign a point value to a question.

**C. Expected result:**

The question should show the specified point value and does not affect the other questions' point values.

**A. Identification and classification:**

Test Case 11

System: Create Test

Invalid point assignment

Severity: 2

**B. Instructions:**

1. Execute the instructor JAR
2. Click create test
3. Assign a negative point value to a question.

**C. Expected result:**

The question should inform the user it is not a valid point assignment.

**A. Identification and classification:**

Test Case 12

System: Create Test

Invalid point assignment

Severity: 2

**B. Instructions:**

1. Execute the instructor JAR
2. Click create test
3. Assign a point value of 0 to a question.

**C. Expected result:**

The question should inform the user it is not a valid point assignment.

**A. Identification and classification:**

Test Case 13

System: Create Test

Invalid point assignment

Severity: 2

**B. Instructions:**

1. Execute the instructor JAR
2. Click create test
3. Type in non numerical value into point value assignment

**C. Expected result:**

The question should inform the user it is not a valid point assignment.

**A. Identification and classification:**

#### Test Case 14

System: Create Test

Invalid point assignment

Severity: 2

##### **B. Instructions:**

1. Execute the instructor JAR
2. Click create test
3. Leave point value assignment blank.

##### **C. Expected result:**

The question should inform the user it is not a valid point assignment.

##### **A. Identification and classification:**

#### Test Case 15

System: Properties

Create question

Severity: 2

##### **B. Instructions:**

1. Execute the JAR
2. Click create test
3. Create a new Test
4. Check the properties
5. Change a property, end time of test

##### **C. Expected result:**

Properties of test should now be changed.

##### **A. Identification and classification:**

#### Test Case 16

System: Properties

Create question - invalid property

Severity: 2

##### **B. Instructions:**

1. Execute the JAR
2. Click create test
3. Create a new Test
4. Check the properties
5. Change a property, end time of test to a date in the past.

##### **C. Expected result:**

System should inform user they cannot have a test be due in the past

##### **A. Identification and classification:**

#### Test Case 17

System: Properties

Create question - invalid property

Severity: 2

##### **B. Instructions:**

1. Execute the JAR
2. Click create test
3. Create a new Test

4. Check the properties
5. Change a property, end time of test to an invalid date

**C. Expected result:**

System should inform user they have entered an invalid date

**A. Identification and classification:**

Test Case 18

System: Edit Test

Valid test

Severity: 1

**B. Instructions:**

1. Start the instructor client
2. Press the Edit Test button
3. Select a valid test you created earlier

**C. Expected result:**

1. The test should open up in a screen that looks similar to Create test.

**A. Identification and classification:**

Test Case 19

System: Edit Test

Invalid test

Severity: 2

**B. Instructions:**

1. Start the instructor client
2. Press the Edit Test button
3. Select an invalid test

**C. Expected result:**

1. The program should tell you the test cannot be opened

**A. Identification and classification:**

Test Case 20

System: Edit Test

Null test

Severity: 2

**B. Instructions:**

1. Start the instructor client
2. Press the Edit Test button
3. Don't select a test

**C. Expected result:**

1. The program should tell you no test is available

**A. Identification and classification:**

Test Case 21

System: Edit Test

Add multiple choice question

Severity: 1

**B. Instructions:**

1. Start the instructor client

2. Press the Edit Test button
3. Select a valid test
4. add a multiple choice question
5. save the test

**C. Expected result:**

1. The program should save the test and the question should save

**A. Identification and classification:**

Test Case 22

System: Edit Test

Add free response question

Severity: 1

**B. Instructions:**

1. Start the instructor client
2. Press the Edit Test button
3. Select a valid test
4. add a free response question
5. save the test

**C. Expected result:**

1. The program should save the test and the question should save

**A. Identification and classification:**

Test Case 23

System: Edit Test

Add Programming question

Severity: 1

**B. Instructions:**

1. Start the instructor client
2. Press the Edit Test button
3. Select a valid test
4. add a programming question
5. save the test

**C. Expected result:**

1. The program should save the test and the question should save

**A. Identification and classification:**

Test Case 24

System: Edit Test

Add free response question, multiple choice question, and a programming question

Severity: 1

**B. Instructions:**

1. Start the instructor client
2. Press the Edit Test button
3. Select a valid test
4. add a free response question, multiple choice question, and a programming question
5. save the test

**C. Expected result:**

1. The program should save the test and the questions should save

**A. Identification and classification:**

Test Case 25

System: Review Test

Valid student test

Severity: 1

**B. Instructions:**

1. Start the instructor client
2. Press the open test button
3. Select a valid student test

**C. Expected result:**

1. The program should open the students test
2. Their answers should be visible.
3. There should be some way to compare with the key

**A. Identification and classification:**

Test Case 26

System: Review Test

Invalid student test

Severity: 2

**B. Instructions:**

1. Start the instructor client
2. Press the open test button
3. Select an invalid student test

**C. Expected result:**

1. The program should tell you the test cannot be opened

**A. Identification and classification:**

Test Case 27

System: Review Test

Null student test

Severity: 2

**B. Instructions:**

1. Start the instructor client
2. Press the open test button
3. Do not select a students test

**C. Expected result:**

1. The program should tell you to select a test

**A. Identification and classification:**

Test Case 28

System: Open instructors test

Valid instructors test

Severity: 1

**B. Instructions:**

1. Start the Student client
2. Press the open test button
3. Select a valid instructor client



**C. Expected result:**

1. The program should open the test
2. the program should let you take the test

**A. Identification and classification:**

Test Case 29

System: Open instructors test

Invalid instructors test

Severity: 2

**B. Instructions:**

1. Start the Student client
2. Press the open test button
3. Select an invalid instructor client

**C. Expected result:**

1. The program should tell you it can't open the test

**A. Identification and classification:**

Test Case 30

System: Open instructors test

Null instructors test

Severity: 2

**B. Instructions:**

1. Start the Student client
2. Press the open test button
3. Do not select an instructor's test

**C. Expected result:**

1. The program should tell you to select a test

**A. Identification and classification:**

Test Case 31

System: Instructor Client

Navigating through the user interface

Severity: 2

**B. Instructions:**

1. Run the TestifyBuggedInstructor.jar
2. Click on one of the buttons available
3. Click a button or a menu that brings user back to the main menu

**C. Expected result:**

1. The application should launch properly
2. The button can correctly navigate user from and back to the main (administrative) page

**A. Identification and classification:**

Test Case 32

System: Instructor Client

Trying to remove a question when no question is in the list

Severity: 2

**B. Instructions:**

1. Run the TestifyBuggedInstructor.jar

2. Click the “Remove” question button when the list of questions is empty

**C. Expected result:**

1. The application should launch properly.
2. The application should warn user that the list of questions is empty.

**A. Identification and classification:**

Test Case 33

System: Instructor Client

Editing a question

Severity: 1

**B. Instructions:**

1. Run the TestifyBuggedInstructor.jar, assuming there is at least 1 question
2. Select that question and click the “Edit” question button

**C. Expected result:**

1. The application should launch properly.
2. User is able to change the properties of the questions
3. Make sure the change is permanent (the old values are erased)

**A. Identification and classification:**

Test Case 34

System: Instructor Client

Editing a question, change the question to an invalid string

Severity: 2

**B. Instructions:**

1. Run the TestifyBuggedInstructor.jar, assuming there is at least 1 question
2. Select that question and click the “Edit” question button
3. Erase the current question leaving it as null.

**C. Expected result:**

1. The application should launch properly.
2. User is able to change the properties of the questions.
3. The application should warn user (now or later when saving the test) that a particular test question has invalid string.

**A. Identification and classification:**

Test Case 35

System: Instructor Client

Trying to edit a question when no question is in the list

Severity: 2

**B. Instructions:**

1. Run the TestifyBuggedInstructor.jar, assuming there is at least 1 question
2. Click the “Edit” question button when the list of questions is empty

**C. Expected result:**

1. The application should launch properly.
2. The application should warn user that the list of questions is empty.

**A. Identification and classification:**

Test Case 36

System: Instructor Client

Saving a test without any questions

Severity: 2

**B. Instructions:**

1. Run the TestifyBuggedInstructor.jar
2. Remove all questions
2. Click the "Save" test

**C. Expected result:**

1. The application should launch properly.
2. The application should warn user that the list of questions is empty.

**A. Identification and classification:**

Test Case 37

System: Student

Student wants to take a test - Boundary test

Severity: 2

**B. Instructions:**

1. Select the first or last test file from the directory and click it to take the test

**C. Expected result:**

1. The test should open up and you should see all the different questions

**A. Identification and classification:**

Test Case 38

System: Student

Student wants to skip a question and come back to it.

Severity: 1

**B. Instructions:**

1. Select a test to take and fill out half of the questions. Click next and then previous

**C. Expected result:**

1. User should be able to go back to the previous unanswered question and answer it

**A. Identification and classification:**

Test Case 39

System: Student

Student wants to submit a complete test.

Severity: 2

**B. Instructions:**

1. Select a test to take and open it. Click submit after completing all questions.

**C. Expected result:**

1. The test should be successfully saved under the specified directory.

**A. Identification and classification:**

Test Case 40

System: Student Client

Reopen test after submitting it

Severity: 2

**B. Instructions:**

1. Select a test to take and open it. Click submit after completing all questions and open it in the saved directory.

**C. Expected result:**

1. System should not let user to be able to edit a submitted test.

**A. Identification and classification:**

Test Case 41

System: Student

Multiple choice question

Severity: 1

**B. Instructions:**

1. Select a test to take and open it. Click around the different options in a multiple choice question and save it.

**C. Expected result:**

1. It should only let you select one option at a time and be set as the active option when you open the saved test.

**A. Identification and classification:**

Test Case 42

System: Student Client

Free response

Severity: 3

**B. Instructions:**

1. Select a test to take and open it. Click and type in a free response box.

**C. Expected result:**

1. It should provide you with enough characters to answer the question and have some way to inform you of character max.

**A. Identification and classification:**

Test Case 43

System: Administer

Severity: 1

**B. Instructions:**

1. Select test data to view.

**C. Expected result:**

A list of students should appear.

**A. Identification and classification:**

Test Case 44

System: Administer

Severity: 1

**B. Instructions:**

1. Select test data to view. Then select a student from the list.

**C. Expected result:**

1. Once a user is selected, all of their test answers appear on screen.

**A. Identification and classification:**

Test Case 45

System: Administer

Invalid test data

Severity: 2

**B. Instructions:**

1. Select a file that doesn't contain test data.

**C. Expected result:**

1. It should inform the user that the selected file is invalid.

**A. Identification and classification:**

Test Case 46

System: Administer

Switching between students

Severity: 2

**B. Instructions:**

1. Select test data to view. Select a student from the list of students, then select another student from the list.

**C. Expected result:**

1. It should provide the answers that each student put in for each question in the test.

**A. Identification and classification:**

Test Case 47

System: Administer

Different number of questions on tests

Severity: 2

**B. Instructions:**

1. Select test form to view. Select a student from the list. Select a different test form with less questions to view. Select a student from the list.

**C. Expected result:**

1. It should only show data on questions related to the newest test form and not have any leftover data from the first test data opened.

**A. Identification and classification:**

Test Case 48

System: Administer

Switch between questions

Severity: 2

**B. Instructions:**

1. Select test data to view. Select a student. Go to the next question and then after go back to the previous question.

**C. Expected result:**

1. It should switch correctly between the question and answers that the student had for each question.

**A. Identification and classification:**

Test Case 49

System: Student client

Student sign in

Severity: 1

**B. Instructions:**

1. Start the Student client

2. Enter info into first and last name fields

3. Click the next button.

**C. Expected result:**

1. The program should advance to the next screen prompting the user to open a test file.

**A. Identification and classification:**

Test Case 50

System: Student client

Exit Button

Severity: 1

**B. Instructions:**

1. Start the Student client

2. Enter info into first and last name fields

3. Click the exit button.

**C. Expected result:**

1. The program should exit.

**A. Identification and classification:**

Test Case 51

System: Student client

First and Last Name Text Fields

Severity: 1

**B. Instructions:**

1. Start the Student client

2. Enter invalid info into first and last name fields

**C. Expected result:**

1. The program should tell the user if the user name is invalid.

**A. Identification and classification:**

Test Case 52

System: Student client

Editing Short Answer Prompt

Severity: 2

**B. Instructions:**

1. Click on the Text box for the question prompt and type

**C. Expected result:**

1. The program should not allow the user to edit the question.

**A. Identification and classification:**

Test Case 53

System: Student client

Short Answer Next Button

Severity: 1

**B. Instructions:**

1. Click on Next Button

**C. Expected result:**

1. Student will be able to move to next question.

**A. Identification and classification:**

Test Case 54

System: Student client

Short Answer Previous Button

Severity: 1

**B. Instructions:**

1. Click on Previous Button

**C. Expected result:**

1. Student will be able to move to previous question.

**A. Identification and classification:**

Test Case 55

System: Student client

Verify Test Save File Name

Severity: 2

**B. Instructions:**

1. Submit test

**C. Expected result:**

1. The file will be saved with the format firstname\_lastname

**A. Identification and classification:**

Test Case 56

System: Student client

Previous button - boundary test

Severity: 2

**B. Instructions:**

1. On the first question, click the previous button.

**C. Expected result:**

1. Student should not be able to move to a previous question.

**A. Identification and classification:**

Test Case 57

System: Student client

Submit test

Severity: 1

**B. Instructions:**

1. Open a test and fill out all the questions
2. Submit Test

**C. Expected result:**

1. User should be directed to an exit screen.