

# Requirements Based Test Cases for TAPS

**Revision 1**  
**March 24, 2019**

**Team 17**  
**Mai Nguyen, Tasha Bornemann, Xinyi Yan, Guanpin Zhong**

## Document Revision History

Rev	Date	Author	Change Description
1	3/24/19	Team 17	Initial Document

## Table of Contents

---

<b>1. TEST REQUIREMENTS</b>	<b>2</b>
1.1 OBJECTIVE	2
1.2 DEFINITIONS AND ACRONYMS	2
1.3 TRACEABILITY MATRIX	2
<b>2. TEST CASES</b>	<b>6</b>
2.1 DESCRIPTION	6
2.2 TEST CASES	6

## 1. Test Requirements

### 1.1 Objective

The purpose of the Test Requirements section is to list all software test requirements for TAPS. The requirements referred to are contained in the Software Requirements Specification.

### 1.2 Definitions and Acronyms

Here are some technical terms or acronyms used in the document, along with their meanings:

SRS	Software Requirements Specification
TM	Traceability Matrix
TAPS	Teaching Assistant Processing System

### 1.3 Traceability Matrix

The Traceability Matrix below shows which test cases verify which requirements.

	Req 1	Req 2	Req 3	Req 4	Req 5	Req 6	Req 7	Req 8	Req 9	Req 10
Test Case ID 1	X									
Test Case ID 2	X									
Test Case ID 3		X								

---

Test Case ID 4			X							
Test Case ID 5				X						
Test Case ID 6					X					
Test Case ID 7					X					
Test Case ID 8					X					
Test Case ID 9					X					
Test Case ID 10					X					
Test Case ID 11						X				
Test Case ID 12						X				
Test Case ID 13						X				
Test Case ID 14						X				
Test Case ID 15						X				
Test Case ID 16							X			
Test Case ID 17							X			
Test Case ID 18								X		
Test Case ID 19								X		
Test Case ID 20									X	

## Requirements Based Test Cases for TAPS

Revision 1

3/24/2019

---

Test Case ID 21									X	
Test Case ID 22									X	
Test Case ID 23										X
Test Case ID 24										X
Test Case ID 25										X
Test Case ID 26										X
Test Case ID 27										X
Test Case ID 28										X
Test Case ID 29										X
Test Case ID 30	X	X	X	X	X	X	X	X	X	X

## 2. Test Cases

### 2.1 Description

For convenience, test cases may be logically grouped together in what are called Test Modules, based on areas of testing. The use of Test Modules helps make the Traceability Matrices and archiving Test Cases more manageable.

### 2.2 Test Cases

#### **Test Case ID: 1**      **Log in a valid user**

**Description:** Checks that the credentials are valid.

**Test Inputs:** Account username and password

**Expected Results:** User is logged in

**Dependencies:** None

**Initialization:** The username is already set in the system.

**Test Steps:**

1. Enter an existing username and valid password.
2. Verify that the user is logged in.

#### **Test Case ID: 2**      **Log in an invalid user**

**Description:** Checks that the credentials are invalid such that the username does not exist

**Test Inputs:** Account username and password

**Expected Results:** User is unable to log in

**Dependencies:** None

**Initialization:** None

**Test Steps:**

1. Enter a username that does not exist.
2. Verify that the user is not logged in.

#### **Test Case ID: 3**      **Complete TA Application**

**Description:** Checks that the required fields exist in the TA application and that a prospective TA can fill out and submit an application.

**Test Inputs:** None

**Expected Results:** All required fields exist, which include type of appointment requested, prior appointment details, course preferences, personal and

academic details, and language and technical scores. Application can be filled out and submitted.

**Dependencies:** Test Case ID 1

**Initialization:** The user logged in is a prospective TA.

**Test Steps:**

1. Open/start application.
2. Verify that all of the required fields listed in Expected Results are available on the application.
3. Verify that information can be entered in these fields.
4. Fill out all of the required fields listed in Expected Results.
5. Submit application.

#### **Test Case ID: 4      Complete Faculty Recommendation**

**Description:** Checks that a faculty member can fill out and submit a recommendation..

**Test Inputs:** None

**Expected Results:** The faculty member is able to fill out a recommendation, including the TA's name, and submit it.

**Dependencies:** Test Case ID 1

**Initialization:** The user logged in is a faculty member.

**Test Steps:**

1. Verify that a recommendation form is available.
2. Verify that the faculty member is able to fill out the recommendation form with information.
3. Submit recommendation.

#### **Test Case ID: 5      Complete Faculty Request**

**Description:** Checks that a faculty member can fill out and submit a request..

**Test Inputs:** None

**Expected Results:** The faculty member is able to fill out a request, including the TA's name, and submit it.

**Dependencies:** Test Case ID 1

**Initialization:** The user logged in is a faculty member.

**Test Steps:**

1. Verify that a request form is available.
2. Verify that the faculty member is able to fill out the request form with information.
3. Submit request.

#### **Test Case ID: 6      Input Data**

**Description:** Checks that the system can allow data to be input manually. This could be budget information, course and position information, or statistics from previous semesters.

**Test Inputs:** None  
**Expected Results:** Manually input data is stored.  
**Dependencies:** Test Case ID 1  
**Initialization:** The user logged in is administrative staff.  
**Test Steps:**

1. Input data.
2. Submit data.

**Test Case ID: 7**      **View budget information**

**Description:** Checks that the administrative staff is able to view budget information.  
**Test Inputs:** None  
**Expected Results:** The system displays the budget information.  
**Dependencies:** Test Case ID 1, 6  
**Initialization:** The user logged in is an administrative staff. Budget information has been provided to system.  
**Test Steps:**

1. Verify that administrative staff can view the allocated budget.
2. Verify that administrative staff can view the amount of money that has been spent.

**Test Case ID: 8**      **View course and position information**

**Description:** Checks that the system can display the course and position information.  
**Test Inputs:** None  
**Expected Results:** The system displays the course and position information.  
**Dependencies:** Test Case ID 1, 6  
**Initialization:** The user logged in is an administrative staff. Course and position information has been provided to the system.  
**Test Steps:**

1. Verify that the administrative staff is able to see the list of offered courses.
2. Verify that the administrative staff is able to show the estimated number of TAs for each course.

**Test Case ID: 9**      **View summary information**

**Description:** Checks that the administrative staff is able to view the summary information.  
**Test Inputs:** None  
**Expected Results:** The system displays the summary information.  
**Results:**

**Dependencies:** Test Case ID 1, 6

**Initialization:** The user logged in is an administrative staff.

**Test Steps:**

1. View summary information.
2. Verify that the administrative staff is able to see information about the current semester.

**Test Case ID: 10      View statistics from past semesters**

**Description:** Checks that the administrative staff is able to view statistics from past semesters.

**Test Inputs:** None

**Expected Results:** The system displays the statistics from past semesters.

**Dependencies:** Test Case ID 1, 6

**Initialization:** The user logged in is an administrative staff. Statistics information is available to the system.

**Test Steps:**

1. View past statistics.
2. Verify that administrative staff can view information about past semesters, including the proportion of TAs who were in each program.

**Test Case ID: 11      View TA applications without an order**

**Description:** Checks that administrative staff is able to view the submitted applications.

**Test Inputs:** None

**Expected Results:** All submitted applications are displayed in an unsorted order.

**Dependencies:** Test Case ID 1, 3

**Initialization:** The user logged in is an administrative staff. TA applications have been submitted.

**Test Steps:**

1. Verify that all submitted applications are viewable by administrative staff.
2. Verify that the applications are not sorted in an order respective to priority score.

**Test Case ID: 12      View TA applications with an order**

**Description:** Checks that administrative staff is able to view the submitted applications in order of the priority score.

**Test Inputs:** None



**Expected Results:** All submitted applications are displayed in an order based on the priority score.

**Dependencies:** Test Case ID 1, 11

**Initialization:** The user logged in is an administrative staff. TA applications have been submitted.

**Test Steps:**

1. View submitted applications.
2. Specify to sort applications by priority score.
3. Verify that the applications are sorted in order of priority score.

**Test Case ID: 13      Appoint TA**

**Description:** Checks that administrative staff is able to appoint a TA to a position.

**Test Inputs:** None

**Expected Results:** TA is appointed.

**Dependencies:** Test Case ID 1, 12

**Initialization:** The user logged in is an administrative staff. TA applications have been submitted.

**Test Steps:**

1. View the list of TA applications.
2. Choose a TA and verify that they can be appointed.

**Test Case ID: 14      Assign course to chosen TA**

**Description:** Checks that administrative staff is able to assign an appointed TA to a course.

**Test Inputs:** None

**Expected Results:** TA is assigned to the course.

**Dependencies:** Test Case ID 1, 13

**Initialization:** The user logged in is an administrative staff. The TA has been appointed.

**Test Steps:**

1. Choose a course for assignment.
2. Assign chosen course to an appointed TA.
3. Verify that the appointed TA has the correct course assigned to them.

**Test Case ID: 15      Assign more than 2 courses to TA**

**Description:** Checks that administrative staff is unable to assign more than 2 courses to the appointed TA

**Test Inputs:** None

**Expected Results:** TA will not have more than 2 courses assigned to them.

**Dependencies:** Test Case ID 1, 13, 14

**Initialization:** The user logged in is an administrative staff. The TA has been appointed.

**Test Steps:**

1. Choose 3 courses for assignment.

2. Assign chosen courses to the appointed TA.
3. Verify that the system will not allow this action and the appointed TA can only have at most 2 courses assigned.

**Test Case ID: 16      View list of TA appointments**

**Description:** Checks that payroll staff shall be able to view the list of TA appointments made by administrative staff.

**Test Inputs:** None

**Expected Results:** The list of TA appointments is displayed.

**Dependencies:** Test Case ID 1, 13

**Initialization:** The user logged in is a payroll staff. Appointments have been made by administrative staff.

**Test Steps:**

1. Verify that the payroll staff member is able to see the list of appointments.
2. Verify that each appointment in the list includes the TA and the percentage.

**Test Case ID: 17      Correct fields in appointment offer**

**Description:** Checks that the correct fields such as name of the TA, appointment percentage, and the dates of the appointment are included in the appointment offer.

**Test Inputs:** None

**Expected Results:** The fields included in the appointment offer are correct.

**Dependencies:** Test Case ID 1

**Initialization:** The user logged in is a payroll staff.

**Test Steps:**

1. Verify that an appointment offer has a space for the TA's name, appointment percentage, and appointment dates.

**Test Case ID: 18      Send appointment offers to appointed TA**

**Description:** Checks that payroll staff is able to send an appointment offer to an appointed TA from the list of appointments.

**Test Inputs:** None

**Expected Results:** The appointment offer to an appointed TA is successfully sent.

**Dependencies:** Test Case ID 1, 16

**Initialization:** The user logged in is a payroll staff. TA appointments have been made.

**Test Steps:**

1. Choose an appointed TA from the appointment list.
2. Write the appointment offer for the appointed TA, including the TA's name, appointment percentage, and appointment dates.
3. Send appointment offer to appointed TA.
4. Verify that the appointment offer have been successfully sent to the appointed TA.

**Test Case ID: 19**      **Appointed TA notified of offer**

**Description:** Checks that an appointed TA is notified of an offer after it has been sent out by payroll staff.

**Test Inputs:** None

**Expected Results:** TA receives a notification of the offer

**Dependencies:** Test Case ID 1, 18

**Initialization:** The user logged in is an appointed TA. An offer has been sent by payroll staff.

**Test Steps:**

1. Verify that the appointed TA can see a notification of the offer sent by payroll staff.
2. For each user who is not the appointed TA, log in and verify that there is not a notification about the response.

**Test Case ID: 20**      **View appointment status of TA**

**Description:** Checks that administrative staff is able to view the appointment status of a TA.

**Test Inputs:** None

**Expected Results:** The system displays the appointment status of a TA specified by administrative staff.

**Dependencies:** Test Case ID 1

**Initialization:** The user logged in is an administrative staff.

**Test Steps:**

1. Specify a TA.
2. Verify that administrative staff can view the appointment status of the specified TA.

**Test Case ID: 21**      **Update appointment status of TA**

**Description:** Checks that administrative staff is able to update the appointment status of a TA.

**Test Inputs:** None

**Expected Results:** The appointment status of a TA is successfully updated in the system.

**Dependencies:** Test Case ID 1, 20

**Initialization:** The user logged in is an administrative staff

**Test Steps:**

1. Specify a TA.
2. View the appointment status of the specified TA.
3. Update the appointment status by making a change.
4. Verify the system successfully saves the updated appointment status of the TA.

**Test Case ID: 22      Notification of change in appointment status**

**Description:** Checks that a notification of the change is sent to the TA, the faculty member for the course, and payroll staff.

**Test Inputs:** None

**Expected Results:** The TA, faculty member for the course and payroll staff are notified of change in appointment status.

**Dependencies:** Test Case ID 1, 21

**Initialization:** The user logged in is a faculty member for the course, the TA, or payroll staff. There is a change in the appointment status.

**Test Steps:**

1. Log in as a user specified in initialization.
2. Verify that there is a notification of the updated change of the appointment status sent by administrative staff.
3. Repeat steps 1-2 for each user specified in initialization.
4. For each user who is not included in initialization (other TAs and other faculty members), log in and verify that there is not a notification about the status change.

**Test Case ID: 23      Offer Received**

**Description:** Checks that the TA receives the offer sent by payroll, and is given the option of accepting or declining the offer.

**Test Inputs:** None

**Expected Results:** The TA can view the offer and respond by accepting or declining.

**Dependencies:** Test Case ID 1, 19

**Initialization:** The user logged in is a TA. An offer has been sent by payroll staff, and a notification received.

**Test Steps:**

1. View the offer.
2. Verify that there are 2 responses available: accept or decline.

**Test Case ID: 24      Accept Offer**

**Description:** Checks that the response 'Accept' by a TA prompts a notification sent to administrative staff.

**Test Inputs:** None

**Expected Results:** Only administrative staff receives a notification that the TA accepted the offer.

**Dependencies:** Test Case ID 1, 23

**Initialization:** An offer has been sent by payroll staff.

**Test Steps:**

1. Logged in as the TA, accept the offer.
2. Logged in as administrative staff, verify that there is a notification of this response from the correct TA.
3. For each user who is not administrative staff, log in and verify that there is not a notification about the response.

**Test Case ID: 25      Decline Offer**

**Description:** Checks that the response 'Decline' by a TA prompts a notification sent to administrative staff.

**Test Inputs:** None

**Expected Results:** Only administrative staff receives a notification that the TA declined the offer.

**Dependencies:** Test Case ID 1, 23

**Initialization:** An offer has been sent by payroll staff.

**Test Steps:**

1. Logged in as the TA, decline the offer.
2. Logged in as administrative staff, verify that there is a notification of this response from the correct TA.
3. For each user who is not administrative staff, log in and verify that there is not a notification about the response.

**Test Case ID: 26      Ability to Resign**

**Description:** Checks that an appointed TA has the option of resigning from their accepted appointment.

**Test Inputs:** None

**Expected Results:** The TA is able to resign from their appointment.

**Dependencies:** Test Case ID 1, 24

**Initialization:** The user logged in is a TA that has previously accepted an offer.

**Test Steps:**

1. Verify that there is an option for resigning from the appointment.
2. Resign from appointment.

**Test Case ID: 27      Notification of Resignation**

**Description:** Checks that administrative staff receives a notification when a TA resigns.

**Test Inputs:** None

**Expected Results:** Only administrative staff receives a notification that the TA has resigned from the appointment.

**Dependencies:** Test Case ID 26

**Initialization:** The TA has previously accepted an offer and resigned.

**Test Steps:**

1. Logged in as administrative staff, verify that there is a notification of this resignation from the correct TA.
2. For each user who is not administrative staff, log in and verify that they do not have a notification about the resignation.

**Test Case ID: 28      Create Announcement**

**Description:** Checks that administrative staff is able to make an announcement.

**Test Inputs:** None

**Expected Results:** Administrative staff can write the announcement, choose the recipients, and send the announcement.

**Dependencies:** Test Case ID 1

**Initialization:** The user logged in is administrative staff.

**Test Steps:**

1. Create a new announcement.
2. Verify that administrative staff is able to write a message.
3. Verify that administrative staff is able to choose recipients of the announcement.
4. Send the announcement.

**Test Case ID: 29      Receive Announcement**

**Description:** Checks that recipients of an announcement receive a notification and are able to view it.

**Test Inputs:** None

**Expected Results:** All recipients of the announcement receive a notification and are able to view the message. Only the recipients of the announcement receive the notification.

**Dependencies:** Test Case ID 1, 28

**Initialization:** An announcement has been sent by administrative staff.

**Test Steps:**

1. Logged in as a recipient of the announcement, verify that there is a notification of the announcement.
2. Verify that the recipient sees the same message as the administrative staff wrote.
3. Repeat steps 1-2 for each recipient of the announcement.
4. Logged in as a user who is not a recipient of the announcement, verify that the user does not have a notification of the announcement.
5. Repeat step 4 for all users who are not recipients of the announcement.

**Test Case ID: 30      Ensure correct permissions**

- Description:** Checks that each type of user logged in has the correct access to their respective actions and data.
- Test Inputs:** None
- Expected Results:** Each type of user is not allowed to see data or perform actions not permitted to them.
- Dependencies:** Test Case ID 1-29
- Initialization:** The user logged in is a prospective TA, appointed TA, faculty staff, payroll staff, or administrative staff.
- Test Steps:**
1. Verify that only users who are administrative staff are able to view administrative data such as budget, summary and statistics, and priority scores.
  2. Verify that only users who are administrative staff are able to appoint TAs and assign TAs to courses.
  3. Verify that only users who are prospective TAs are able to complete a TA application.
  4. Verify that only users who are appointed TAs are able to receive and respond to an offer.
  5. Verify that only users who are faculty members are able to complete recommendations and requests.
  6. Verify that only users who are payroll staff are able to send appointment offers.