



# TEST CASES CREATION Part 1

**TAT TRAINING**

JANUARY 1, 2018

# AGENDA

**Test Case Definition**

**Test Cases Goals**

**Tools for Test Cases**

**Test Case Structure**

**Test Cases Creation Process**

# **TEST CASE DEFINITION**

# TEST CASE DEFINITION

---

A **test case** is a set of test inputs, execution conditions, and expected results developed for a particular objective, such as to exercise a particular program path or to verify compliance with a specific requirement.

# TEST CASES GOALS

# TEST CASES GOALS

## Structured Approach

- **Plan**, only then run -> Structured approach, more bugs found

## Find Problems in Requirements

- Test the Requirements documentation before application is available

## Accelerate Regression Testing

- The right set of test cases helps to conduct qualitative regression testing

## Pass Information to New Testers

- Passing test cases is a good start point for a new tester

## Store Information

- Test cases store information about functionality, configuration, credentials, execution status and other relative data

## Track Testing Progress

- X% of tests executed, Y% requirements covered

## Satisfy Customer Requirements

- Satisfy **Customer**\EPAM process requirements

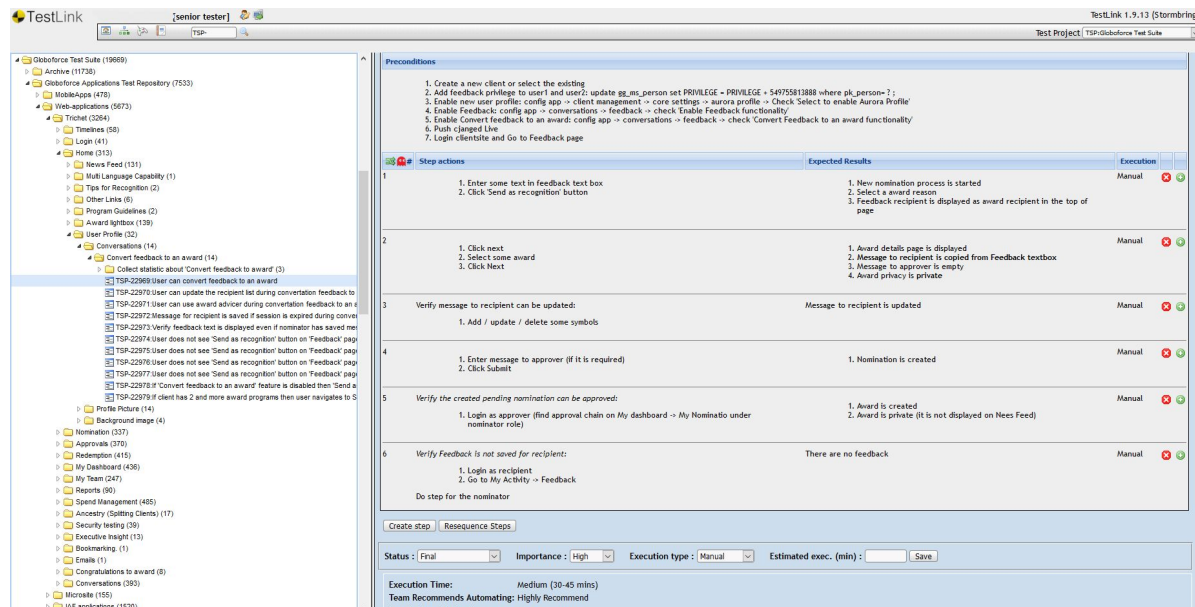


**TOOLS**

**FOR TEST CASES**

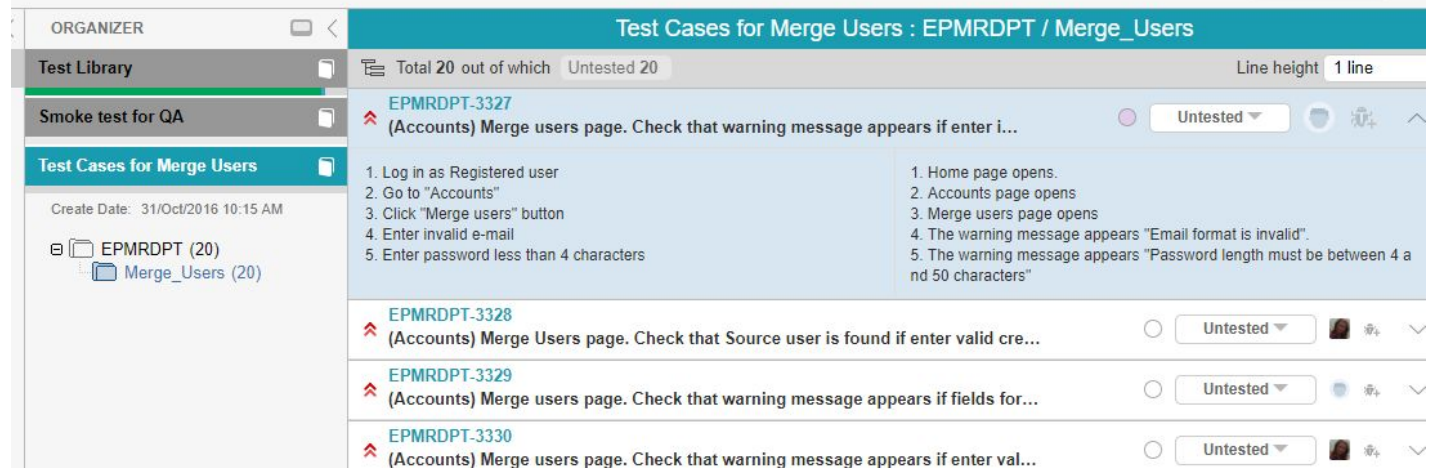


# TOOLS FOR TEST CASES



TestLink

Jira





# NO TOOLS?

## Use Excel

File Home Insert Page Layout Formulas Data Review View Tell me what you want to do...							
Clipboard		Font		Alignment		Number	
Conditional Formatting		Format as Table		Styles		Insert Delete	
2							
A	B	C	D	E	F	G	H
ID	Title	Preconditions	Steps	Expected result	Priority	Result	Comments
<i>My Approvals page</i>							
<u>AP1</u>	Verify user with approval role can disapprove a nomination.	1. New Nomination Process is enabled. 2. There are nominations waiting for approval.	1. Log in to Client site as a user with approver role and go to 'My Dashboard' -> 'My approvals' page. 2. Click 'Disapprove' button for one of the nominations. 3. Click 'No' button. 4. Click 'Disapprove' button for one of the nominations and then 'Yes' button on the confirmation page. 5. Disapprove the rest of nominations one by one.	1. 'My Approvals' page opens. Nominations waiting for approval are displayed. 2. Confirmation page appears with the following text: 'Are you sure you want disapprove the nomination?' 3. Confirmation page closes, this nomination is still displaying. 4. The toast message in green "Nomination disapproved successfully" appears and fades away. This disapproved nomination fades out. The rest of the award cards reshuffle. Approval count is decreased by 1. 5. No pending approval requests are displayed. allApprovedTickMark and the following message are displayed: 'Well done, you have nothing more to approve'. 'Approve All' button is not displayed.	P1		
<u>AP2</u>	Verify that user is redirected My Approvals page to the Login page if timeout happens	1. New Nomination Process is enabled. 2. There are nominations waiting for approval.	1. Log in to Client site as a user with approver role who has nominations to approve. 2. Go to 'My Dashboard' -> 'My approvals' page. 3. Wait for session expiration. 4. Click 'Approve' button for one of the nominations.	User is redirected to the login page.	P2		

# **TEST CASE STRUCTURE**

# STRUCTURE OF TEST CASES

## Required fields of a test case:

Title / Verification Point / Summary	•Summary of the test objective typically written in one line.
Preconditions	•Conditions that must be met before test case is executed
Steps	•The sequences of actions to be followed of or executed
Expected result	•Expected output from the system or application for the action performed on it

## Optional elements:

ID	•A unique number to identify the test case
Module / Sub-module	•Module / Sub-module name.
Description	•Detailed description of test case
Priority	•Helps to choose the order of test cases execution
Status	•Not run / Passed / Failed / Blocked
Comments	•Additional information or any note required while execution test cases

# WHAT IS A GOOD TEST CASE?

---

Specific or General?

Positive or Negative?

Simple or Complex?

Independent or Tied Together?

# SPECIFIC OR GENERAL?

Both tests cases test the same feature. What are the good and bad points of each test case?

<i>1. In the field A type 2 2. In the field B type 3 3. Click Add button. 4. Check value of the field C</i>	<i>4. The value is 5.</i>
---	---------------------------

<i>1. Verify that the program sums A and B correctly</i>	<i>1. It is so.</i>
--	---------------------

- When all details are specified, the same action will be repeated each time. Much less chances to find bug.
- When test case is very general, it can very well stay untested
- Integration tests tend to be more general than others
- More details leads to more time to support and write

# SPECIFIC OR GENERAL?

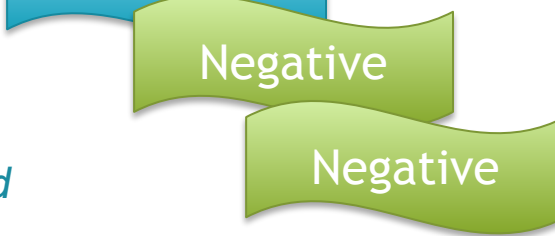
<ol style="list-style-type: none"><li>1. In the field A enter valid integer</li><li>2. In the field B enter valid integer</li><li>3. Click Add button.</li><li>4. Check value of the field C</li><li>5. Repeat steps 1-4 for 0, 9999 (max allowed value), -9999, 1.5, aaa (invalid values)// <i>this point can be moved into separate test case</i></li></ol>	<ol style="list-style-type: none"><li>4. The value of C field equals <math>A+B</math>.</li></ol>
---	--

- We are not tied to specific value
- We still know how to check if the result is correct
- We save support time by referencing steps 1-4

# POSITIVE OR NEGATIVE?

What test cases will you create to test login to an application?

Examples:

- *Test case 1: Login with valid user name and password*
  - *Test case 2: Login with invalid user name and password*
  - *Test case 3: Try to go to main application page bypassing login*
- 
- **Positive test:** attempts to show that application does what it is supposed to do
  - **Negative test:** attempts to show that application doesn't do something that it is not supposed to do (including checking error messages).



# POSITIVE AND NEGATIVE AND BOUNDARY

Both positive and negative test cases are valuable.

Positive tests should be started with

Negative tests usually are more likely to find bugs

There are much more negative tests than positive. We need to select most valuable (higher risk, higher probability)

Boundary tests (border between positive and negative) are extremely useful

# SIMPLE OR COMPLEX?

---

**Test scenario** is a set of test cases for some purpose.

Good test scenario flows along some logic-typical usage, convenience to test, by modules.

# INDEPENDENT OR TIED TOGETHER?

## What are advantages of stand alone test?

- Test can be executed quickly and easily
- Testing can continue when some tests fail
- Tests can be run in any order, any subset
- Variation is possible (after test case #1, created item was modified (new idea). This does not run tests #2-10)

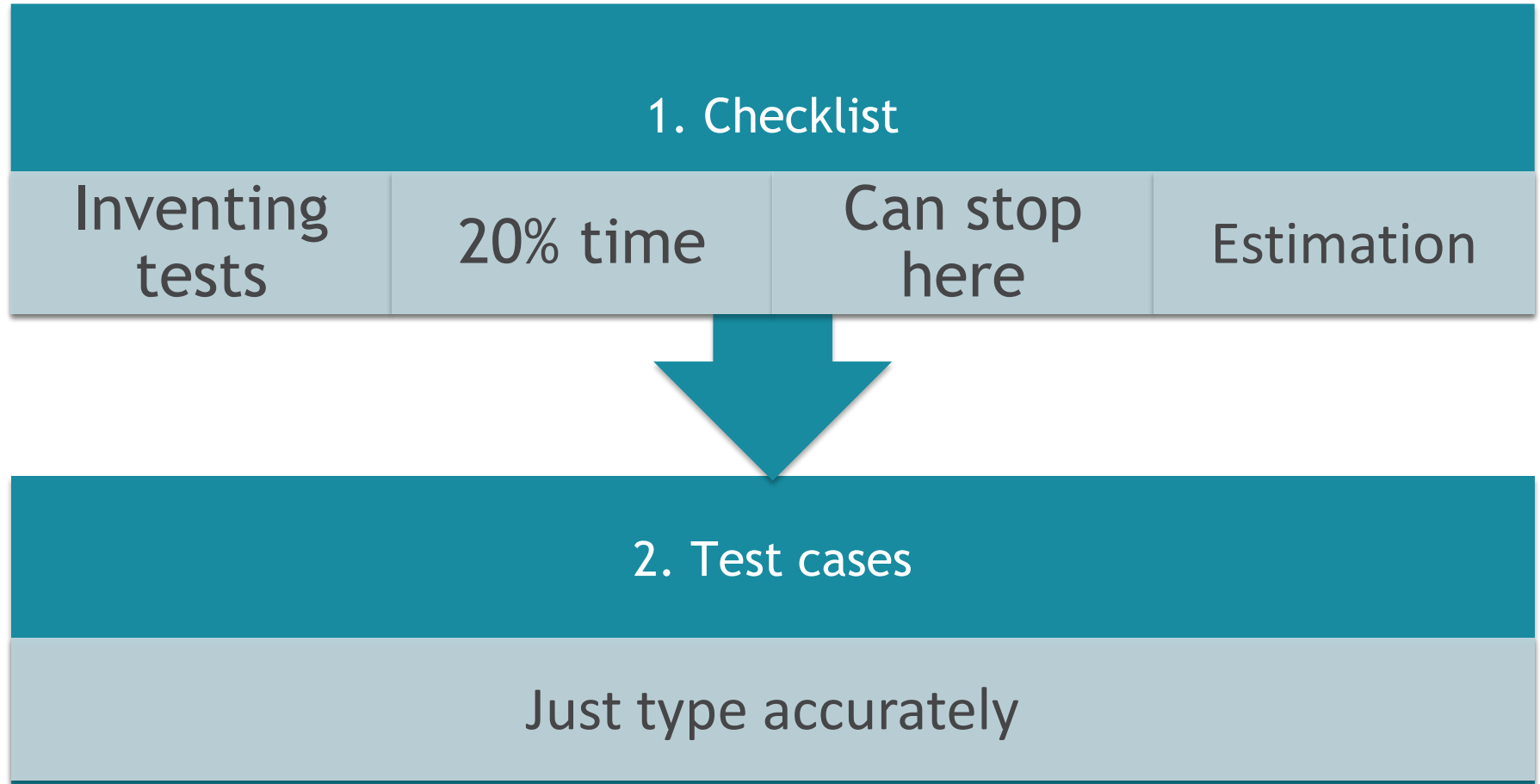
## What are advantages of set of test cases that flow one from another?

- Simulate typical user behavior
- Are convenient for integration testing
- Help when many steps (>10 ) are needed for #1 test
- Reuse the same test data (Item A created in test case #1 can be reused in test cases #2-10)

- **Industry standard is stand-alone tests.**
- **Still it is ok to have some tied scenarios, but they shouldn't be very long**

# **TEST CASE CREATION PROCESS**

# TWO STAGES



# CHECKLIST

Break applications into functions/modules to be tested



Write checklist for each function/area, add questions



Add any questions, as you go



Reorder checklist, add \ remove some tests

# STEP 1

## Break Application into Functions / Modules to be Tested

‘PMC’ application

1. *Login*
2. *Bugs*
  - 2.1 *Bug details form*
    - 2.1.1 *View details*
    - 2.1.2 *Edit details*
      - 2.1.2.1 *Add attachment*
  - 2.2 *Delete*
  - ...
  - 2.4 *Workflow*
  - 2.5 *Data Export*
- ....
3. *Documents*



# STEP 2

## Write Checklist for Each Function / Area, Add Any Questions

Start with “simple” tests

1. *Login, valid*

Don't forget about negative cases

2. *Login, invalid password*

3. *Login, inactive user*

4. *Login, blank password*

And any not evident situations you find interesting

5. *Login, using enter*

6. *Login, from favorites*

7. *Go to bugs page directly, bypassing login*

8. *Login from outside EPAM network???*

# STEP 3

## Fill in Details, Resolve Questions

### *Login, incorrect password 3 times*

#### *Steps:*

- 1. Go to Application A login page*
- 2. Enter valid user name and invalid password combination*
- 3. Click login button*
- 4. Repeat step 3 two more times*

#### *Expected Results:*

- 1. Application A login screen appears*
- 2. The values appear in the respective fields*
- 3. Message that password incorrect appears*
- 4. Is there limit for incorrect attempts???*

# STEP 4

## Add Cosmetics

- Add consistent numbering
- Correct any mistypes, spelling grammar
- Add Consistent formatting
- Edit badly worded, complicated sentences
- Add Excel grouping

# STEP 5

## Get Review

**Get review from other tester, developer, customer**

- Are some interesting tests missed?
- Are some tests redundant?
- Are test cases easy to understand by other person?  
Novice tester?
- Is it what customer expects?

# TEST CASES LANGUAGE

---

- Use active case, do this, do that
- Use “System displays this, does that”
- Use simple, conversational language
- Use exact, consistent names of fields, not generic
- Don't explain Windows basics

**THANK YOU**

**FOR YOUR ATTENTION**