Focused Build for SAP Solution Manager 7.2 (SP6)
Test Management

Customer Experience & Solutions, SAP SE
June 2020
Disclaimer

This presentation outlines our general product direction and should not be relied on in making a purchase decision. This presentation is not subject to your license agreement or any other agreement with SAP. SAP has no obligation to pursue any course of business outlined in this presentation or to develop or release any functionality mentioned in this presentation. This presentation and SAP's strategy and possible future developments are subject to change and may be changed by SAP at any time for any reason without notice. This document is provided without a warranty of any kind, either express or implied, including but not limited to, the implied warranties of merchantability, fitness for a particular purpose, or non-infringement. SAP assumes no responsibility for errors or omissions in this document, except if such damages were caused by SAP intentionally or grossly negligent.
Agenda

Introduction:
- Capabilities, Test Strategy, Test Types, Branches & Test Systems in Focused Build

How-to:
- Unit Test
- Single Functional Test & Acceptance Test
- Functional Integration Test
- Regression Test

Summary

What’s New with Focused Build SP6

Excursus:
- Test Steps, Technical Preparation
Introduction:
Test Strategy, Test Types, Branches & Test Systems in Focused Build
SAP Solution Manager - Test Suite
Capabilities of SAP Solution Manager Test Suite and Focused Build

On Top Functionality of Focused Build for SAP Solution Manager

Prepare | Explore | Realize | Deploy | Run

- Requirements / Work Package based Test Plan Generation
- Test Steps Designer
- Test Execution: My Test Executions
- Test Suite Dashboard

SAP Solution Manager – Test Suite

Solution Documentation | Change Impact Analysis | Test Planning | Test System Setup | Test Execution and Analytics

- Manual Test Cases
- Automated Test Cases
- Test Automation Framework with 3rd party test tool integration
- Test Composition Environment to manage SAP and partner tools
- Test Data Management
- Change Impact Analysis
- Test Plan and Test Package Management
- Test Execution of manual and automated tests
- Gap and Completeness Analytics
- Test Execution Analytics
Focused Build for Application / Software

PMO
- Create project plan
  - Track project readiness, issues and risks
  - Manage scope change
- Manage q-gate deliverables, sign-off
- Hand-over release to customer

Explore Team (Business)
- Fit-Gap WS for Requirements

Build Team (Plan)
- Create Work Package
- Assign Work Items & build teams
- Manage show & tell

Build Team (Development)
- Develop, configure, unit test
- Document Work Item & progress
- Release Work Item
- Defect Correction
- Plan & execute Wave tests (SFT & AT)
- Manage Defects
- Execute Release Tests (FIT & RT)

Test Management

Release Management
- Build, test and validate release
- Manage hyper-care

© 2020 SAP SE or an SAP affiliate company. All rights reserved. | PUBLIC
Example of Project Structure
Build Project for Waterfall Approach

**Project structure sample with 1 wave and 1 sprint**

Wave 1
- Functional Specification Released
- Single Func. Test Completed
- Unit Test Completed
- (Work Item) Build Started
- (Work Item) Build Completed

Sprint 1
- Technical Design Released
- Unit Test Completed
- (Work Item) Build Started
- (Work Item) Build Completed

Handover to Release
- Functional Integration Test (Final)
- Acceptance Test
- Regression Test
- Note. Point of NO RETURN for the Transports in these WP
Example of Project Structure
Build Project for Agile development of a single release

Project structure sample with 1 waves, of 3 sprints

Wave 1
- W1 Scope defined
- Functional Spec released/available
- Single Functional Test (SFT) completed
- Work Package (WP) Build completed
- Acceptance Test
- Functional Integration Test (Final)
- Regression Test
- (Note. Point of NO RETURN for the Transports in these WP)

Sprints
- Sprint 1
  - Tech. Design Released
  - Build Started
  - Build Completed
  - Unit Test Completed
- SFT
- Unit Test
- Sprint 2
  - Tech. Design Released
  - Build Started
  - Build Completed
  - Unit Test Completed
- SFT
- Unit Test
- Sprint 3
  - Tech. Design Released
  - Build Started
  - Build Completed
  - Unit Test Completed
- SFT
- Unit Test
Example of Project Structure
Build Project for Agile development of a single release

Project structure sample with 2 waves, of 2 sprints

Wave 1
- SFT (Single Functional Test)
  - Sprint 1: WP Build Finished
  - Sprint 2: SFT Finished
- AT1 (Final)
  - SFT completed

Wave 2
- SFT (Single Functional Test)
  - Sprint 1: WP Build Finished
  - Sprint 2: SFT Finished
- AT1 (Optional)
  - FIT1 completed
- Functional Integration Test (Final)
- Regression Test

Phases of Release Cycle
- Prepare
- Explore
- Realize
- Deploy
- Run

Milestones
- FS Released
- Tech. Design Released
- Build Started
- Build Completed
- Unit Test Completed

(Note. Point of NO RETURN for the Transports in these WP)
# Testing Types

## Definition

<table>
<thead>
<tr>
<th>Test Types</th>
<th>Test Level</th>
<th>Test Requirement</th>
<th>Definition</th>
<th>Test System</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unit Test</td>
<td>Sprint</td>
<td>Work Item (Tech. Design)</td>
<td>Test to ensure that the work (e.g. WRICEF, Config) was implemented correctly</td>
<td>QAS</td>
</tr>
<tr>
<td>Single Functional Test</td>
<td>Wave</td>
<td>Single Work Package (Func. Spec)</td>
<td>Business function test to ensure that the feature was implemented correctly. SFT of a Work Package (WP) can be executed when all Work Items (WI) of a WP have been successfully unit tested. (Depending on the project approach, the Single Functional Test might be without Test Plan creation)</td>
<td>QAS</td>
</tr>
<tr>
<td>(Business) Acceptance Test</td>
<td>Wave</td>
<td>Work Packages (Requirement)</td>
<td>Validation of requested functionality by the requestor. After validation, corresponding Work Packages (WP) and Defect Corrections (DC) would be ready for Handover to Release. (This is the most important test, as after setting the Work Package status ‘Handover to Release’, all assigned transports will end in PRD system.)</td>
<td>QAS</td>
</tr>
<tr>
<td>Functional Integration Test (Optional)</td>
<td>Wave(s)</td>
<td>Related Work Packages (Process with Requirements)</td>
<td>Early validation of modular processes. (Can include tests from earlier Wave(s))</td>
<td>QAS</td>
</tr>
<tr>
<td>Functional Integration Test (Final)</td>
<td>Release</td>
<td>E2E Processes</td>
<td>E2E tests of the new or extended functionality, validating the end to end business process. (Remember all Transports in WP/DC that were handed over to release will go live)</td>
<td>PRE</td>
</tr>
<tr>
<td>Regression Test</td>
<td>Release</td>
<td>Productive Processes</td>
<td>Test of productive processes or functions</td>
<td>PRE</td>
</tr>
</tbody>
</table>

* Excluding: Data Migration Tests; End User AT DR; Performance Tests; Technical System Tests
Requirements based Test Approach
All Test Types

- SBX: Definition and Specification
- DEV: Implementation and Configuration
- QAS: Functional Tests
- PRE: Process Tests
- PRD: PRD use

Requirements

Work Packages (Functional Specification)

Work Items (Technical Design)

Unit Test

Software Implementation and related Configuration

(Business) Acceptance Test

Regression Tests (Optional)

Functional Integration Test (Final)

Functional Integration Test (Optional)

Regression Test (Final)

Handover to Release

(Note: Point of NO RETURN for the Transports in these WP)

* Excluding: Data Migration Tests; End User AT DR; Performance Tests; Technical System Tests

© 2020 SAP SE or an SAP affiliate company. All rights reserved. | PUBLIC
Focused Build methodology
Standard Workflow

- Business Analyst/Process Expert
- Solution Architect
- Developer

- Requirements
- Work Package
- Work Item

- Approved
- In Realization
- Created
- Scoping
- Scope Finalized
- To Be Developed
- In Development
- In Repair
- To Be Tested
- Successfully Tested
- Hand over to Release
- Productive/Completed
- Realized

- Test Coordinator
- Technical Tester (Developer)
- Unit Test

© 2020 SAP SE or an SAP affiliate company. All rights reserved. | PUBLIC
### Focused Build methodology – Branches

#### Standard Workflow

**Design Branch**
- **Approved**
- **In Realization**

**Development Branch**
- **Created**
- **Scoping**
- **Scope Finalized**
- **To Be Developed**
- **In Development**
- **In Repair**
- **To Be Tested**
- **Successfully Tested**
- **Handed over to Release**

**Production Branch**
- **Realized**
- **Completed**

### Work Package

<table>
<thead>
<tr>
<th>Work Item</th>
<th>Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Created</td>
<td><strong>Approved</strong></td>
</tr>
<tr>
<td>Scoping</td>
<td><strong>In Realization</strong></td>
</tr>
<tr>
<td>Scope Finalized</td>
<td><strong>Created</strong></td>
</tr>
<tr>
<td>To Be Developed</td>
<td><strong>Scoping</strong></td>
</tr>
<tr>
<td>In Development</td>
<td>** Scope Finalized**</td>
</tr>
<tr>
<td>In Repair</td>
<td><strong>To Be Developed</strong></td>
</tr>
<tr>
<td>To Be Tested</td>
<td><strong>In Development</strong></td>
</tr>
<tr>
<td>Successfully Tested</td>
<td><strong>In Repair</strong></td>
</tr>
<tr>
<td>Handled over to Release</td>
<td><strong>To Be Tested</strong></td>
</tr>
</tbody>
</table>

### Design Branch

- **Created**
- **Scoping**
- **Scope Finalized**

### Development Branch

- **Created**
- **Scoping**
- **Scope Finalized**
- **To Be Developed**
- **In Development**
- **In Repair**
- **To Be Tested**
- **Successfully Tested**
- **Handed over to Release**

### Production Branch

- **Realized**
- **Completed**

---

© 2020 SAP SE or an SAP affiliate company. All rights reserved. ǀ PUBLIC
Focused Build methodology – Systems
Standard Workflow

SBX
- Approved
- In Realization
- Create Work Package
  - Created
  - Scoping
  - Scope Finalized
  - To Be Developed
  - In Development
  - Create Work Item

QAS
- Single Functional Test & AT
  - In Repair
  - To Be Tested
  - Successfully Tested
  - Hand over to Release

PRE
- Realized
- FIT & RT
- Hand over to Release

PRD
- Completed
- Productive/Completed

DEV
- Created
- In Development
- Create Transports
  - To Be Tested
  - Successfully Tested
  - Hand over to Release
  - Productive/Completed

© 2020 SAP SE or an SAP affiliate company. All rights reserved. | PUBLIC
Focused Build methodology – Release Phase
Standard Workflow
## Unit Test

<table>
<thead>
<tr>
<th>Test Types</th>
<th>Test Level</th>
<th>Test Requirement</th>
<th>Definition</th>
<th>Test System</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Unit Test</strong></td>
<td>Sprint</td>
<td>Work Item (Tech. Design)</td>
<td>Test to ensure that the work (e.g. WRICEF, Config) was implemented correctly</td>
<td>QAS</td>
</tr>
<tr>
<td><strong>Single Functional Test</strong></td>
<td>Wave</td>
<td>Single Work Package (Func. Spec)</td>
<td>Business function test to ensure that the feature was implemented correctly</td>
<td>QAS</td>
</tr>
<tr>
<td><strong>(Business) Acceptance Test</strong></td>
<td>Wave</td>
<td>Work Packages (Requirement)</td>
<td>Validation of requested functionality by the requestor.</td>
<td>QAS</td>
</tr>
<tr>
<td><strong>Functional Integration Test</strong></td>
<td>Wave(s)</td>
<td>Related Work Packages (Process with Requirements)</td>
<td>Early validation of modular processes</td>
<td>QAS</td>
</tr>
<tr>
<td><strong>(Optional)</strong></td>
<td>Release</td>
<td>E2E Processes</td>
<td>E2E tests of the new or extended functionality, validating the end to end business process</td>
<td>PRE</td>
</tr>
<tr>
<td><strong>Functional Integration Test</strong></td>
<td>Release</td>
<td>Productive Processes</td>
<td>Test of productive processes or functions</td>
<td>PRE</td>
</tr>
<tr>
<td><strong>(Final)</strong></td>
<td>Release</td>
<td>E2E Processes</td>
<td>E2E tests of the new or extended functionality, validating the end to end business process</td>
<td>PRE</td>
</tr>
<tr>
<td><strong>Regression Test</strong></td>
<td>Release</td>
<td>Productive Processes</td>
<td>Test of productive processes or functions</td>
<td>PRE</td>
</tr>
</tbody>
</table>
Perform Unit Test
Start Unit Test by importing Transport of Copies into QAS

Steps
1. Developer selects Work Item
2. Developer performs action 'Pass To Test':
   - Transport of Copies (ToC) will be imported to QAS system automatically
   - Unit test can be performed in QAS

Effects
- In Solution Readiness Dashboard, development will be shown as completed for this Work Item

Recommendations
- Please make sure the unit tester is not the same user as the developer
Perform Unit Test
Perform & confirm successful Unit Test

Steps
1. Tester executes Unit Test in QAS
2. Tester selects Work Item
3. (Optional): Tester documents test results
4. Tester performs action ‘Confirm Successful Test’

Effects
- In Solution Readiness Dashboard the Unit Test will be calculated as completed for this Work Item
- The Work Package will be changed to ‘To be Tested’ automatically when all its Work Items are successfully tested
- Transport(s) (TR(s)) will be released automatically and could be imported to QAS via scheduled job defined by Release Manager

Recommendations
- Test results can be maintained as plain text in the Text tab using text type ‘Test Report’
Check Q-Gate for Realize Phase Completed
Check Unit-Test milestone in Solution Readiness Dashboard

Steps

1. All Requirements should be completed
2. All Functional Spec should be completed
3. All Technical Design should be completed
4. All Development should be completed
5. All Unit Tests should be completed
# Single Functional Test & Acceptance Test

<table>
<thead>
<tr>
<th>Test Types</th>
<th>Test Level</th>
<th>Test Requirement</th>
<th>Definition</th>
<th>Test System</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unit Test</td>
<td>Sprint</td>
<td>Work Item (Tech. Design)</td>
<td>Test to ensure that the work (e.g. WRICEF, Config) was implemented correctly</td>
<td>QAS</td>
</tr>
<tr>
<td>Single Functional Test</td>
<td>Wave</td>
<td>Single Work Package (Func. Spec)</td>
<td>Business function test to ensure that the feature was implemented correctly.</td>
<td>QAS</td>
</tr>
<tr>
<td>(Business) Acceptance Test</td>
<td>Wave</td>
<td>Work Packages (Requirement)</td>
<td>Validation of requested functionality by the requestor.</td>
<td>QAS</td>
</tr>
<tr>
<td>Functional Integration Test</td>
<td>Wave(s)</td>
<td>Related Work Packages (Process with Requirements)</td>
<td>Early validation of modular processes.</td>
<td>QAS</td>
</tr>
<tr>
<td>(Final)</td>
<td>Release</td>
<td>E2E Processes</td>
<td>E2E tests of the new or extended functionality, validating the end to end business process.</td>
<td>PRE</td>
</tr>
<tr>
<td>Regression Test</td>
<td>Release</td>
<td>Productive Processes</td>
<td>Test of productive processes or functions</td>
<td>PRE</td>
</tr>
</tbody>
</table>
Single Functional Test (SFT) and Acceptance Test (AT)
Two Variants supported by Focused Build - Overview

Variant A:
Without Test Plan.
(Using option to create Defect Correction during Single Functional Test starting directly from the Work Package)

Variant B:
With Test Plan(s).

You can choose the variant fitting best to your individual test Requirements.
Single Functional Test (SFT) and Acceptance Test (AT)
Variants A supported by Focused Build

Variant A - With formal documentation, without test plan

- Use 'Show and Tell' sessions for testing
- Make sure that 'Show and Tell' are guided by test cases added to the WPs
- After the sessions, add 'Test Notes' documenting results of sessions to each WP
- In case of errors, create defect correction directly from the Work Package in tab Scope
- Work Package will switch into status "to be tested" for retesting when Defect Correction is confirmed
- Change status of WP from 'To be Tested' to 'Successfully Tested' manually WP by WP
Single Functional Test (SFT) and Acceptance Test (AT)
Variant B supported by Focused Build

**Variant B - With formal documentation, with test plan(s)**
- Use test plan as Agenda for 'Show and Tell' sessions
- Add test cases to the WPs (Even in phase scoping a test case or an empty template can be uploaded)
- Create at least SFT Test Plan and/or a AT test plan via Assignment Analysis, which makes sure that the test plan is linked to the project. It is recommended to use the option '1 Test Package per WP' during creation of the test plan, as this option makes it easy to identify the WP that can be set to 'Successfully Tested' when the TP is finished. This is also required for automatic status change of WP, depending on Defect Correction status (WP 'in repair') and enables the linking of Defect Corrections automatically to WP, which makes sure that DC must be released / imported together with the WP
- **Remark:** In case the DCs are not finished (user status Confirmed) please refer to L2 presentation Defect and Defect Correction for further details
  
  *Only one Test Plan:*
  a) Use Test Plan only for AT. For SFT, follow Variant A.

  *Separate Test Plans for SFT and AT:*
  b) AT test plan contains a subset of test packages / test cases of SFT test plan but is executed by different testers (key user, person responsible for Requirement)
  c) AT test plan is a 1:1 copy of SFT test plan but is executed by different testers (key user, person responsible for Requirement)
  d) AT test plan contains different test cases than SFT test plan

- Ensure that status is set and 'Test Results' are documented per test case
- In case of errors create test defects, defect correction and retest
- Optional: Create final Test Report for Test Plan from Test Plan Analytics and add it to respective WPs
- Change status of WP from 'To be Tested' to 'Successfully Tested' only if exit criteria for test phase(s) are met. Mass change functionality for WPs could be used for this if no final Test Report is used
Variant A - Single Functional Test and Acceptance Test

& finalize Test Cases, Execute Tests

Recommendation
- A Business Analyst should review the test cases available in Solution Documentation (SolDoc). If an update of existing test cases or the upload of additional test cases is required:
  - Test cases can be added directly in SolDoc
  - Test cases can be assigned to the respective Work Item (WI)
- Note: If Work Package (WP) is in Status ‘To be Tested’, a WI could be created directly in order to upload test cases. There’s no need to go back in the status ‘Scope extension’ to create the WI there. Even in phase scoping a test case or an empty template can be uploaded to the WP

Steps to upload Test Case to WI:
1. Select Work Item
2. Navigate to Documentation tab
3. Drag and drop the test cases to the Documents area from user’s local computer
4. Select New Version when using the same file name (Only for update)
5. Change test case status to ‘Released’ when finalized
6. Set Work Package status to ‘Confirm Successful Test’ when single test execution is finalized
All variants - Single Functional Test and Acceptance Test

Test Preparation

Tab 'Test Preparation':

- In this tile you can view the amount of Work Packages for a given Project and Wave
- It will help you to easily identify the Work Packages to which no test cases (of any types) are assigned
- It will also highlight the Work Packages which do not have any Test Case assigned according to their statuses
- In this example, you can see that a Work Package in the status 'To Be Tested' which does not have a test case assignment is ranked as 'Red'
- In the table you will find the Work Package details (ID, Name, Priority, Test Case Assignment, …)
- You can jump directly to the Work Package Application to update any Work Package
- You can also navigate to the 'Assignment Analysis' application to add any missing Test Cases
Variant B - Single Functional Test (SFT) and Acceptance Test (AT)
Schematic view on test plan creation

In Focused Build projects, the creation of test plans is done with the **Assignment Analysis and Test Plan Generation** application. With this application, you can check the test plan coverage of the Work Packages of a project and create a test plan based on the test cases assigned to the Work Packages.
Variant B - Single Functional Test (SFT) and Acceptance Test (AT)
Sample Workflow

**Solution Documentation**
- Setup of Solution Documentation
- Setup of Project(s) and Release
- Create Requirements and Work Packages

**Test Planning**
- Check Assignment of Test Case(s) to Work Package (WP)
- Create Test Plan based on WP status
- Create / Adjust Test Packages
- Assign Tester(s)

**Test Execution and Analytics**
- Access My Test Executions (MTE)
- Execute Test Case(s) (manual / automatic)
- Document test results/ set status/ create defects
- Defect Correction (Developer provides fix)
- Retest and set test case status

Test Suite Dashboard

Test Suite Analytics
Variant B - Single Functional Test (SFT) and Acceptance Test (AT)
Update & finalize Test Cases

Recommendation

- A Business Analyst should review the test cases available in Solution Documentation (SolDoc). If an update of existing test cases or the upload of additional test cases is required:
  - Test cases can be added directly in SolDoc
  - Functionality ‘Test Steps’ can be used – see Excursus ‘Test Steps’
  - Test cases can be assigned to the respective Work Item (WI)

- Note: If Work Package (WP) is in Status ‘To be Tested’ a WI can be created directly in order to upload test cases. There’s no need to go back in the status ‘Scope extension’ to create the WI there

Steps

1. Select Work Item (GC)
2. Navigate to Documentation tab
3. Drag and drop the test cases to the Documents area from user’s local computer
4. Select New Version when using the same file name (Only for update)
5. Change test case status to ‘Released’ when finalized
6. Go to Assignment Analysis for creating one or more Test Plans
Variant B - Single Functional Test (SFT) and Acceptance Test (AT)
Create Test Plan(s)

Steps

1. Select Project and Wave.
2. Click ‘Assignment Analysis and Test Plan Generation’
3. Select Work Package(s) which are required for SFT or AT
4. Click Test Plan → ‘Save as new’
5. Possibility to load and enhance an existing Test Plan (Save)

Remarks:

Further prerequisites are the Test Case Type ‘Additive’ in Solution Documentation and a document type ‘Test Document’ in transaction ‘SOLADM’ (see Excursus ‘Technical Prerequisites’ for more details).
Variant B - Single Functional Test (SFT) and Acceptance Test (AT)

Filter Function

- Work Packages part of the Project and Wave but not covered by any Test Plan are listed here.
- The Work Package ID can be used to navigate forward and find more details.
- There should be no WP in status ‘To Be Tested’, after test plan creation activities are finished.

Remarks:
Within the assignment analysis, filters e.g. on test types (SFT, FIT, AT) are available, please make sure you maintain the respective Test Case Attributes.
Variant B - Single Functional Test (SFT) and Acceptance Test (AT)

Create Test Plan(s)

Steps

1. Input mandatory information for Test Plan

2. ‘One Test Package per Work Package’ for Test Package Creation is automatically selected due to the WP Status values

Effects

- Test Plan automatically created
- Test Packages automatically created and adjustable
- Easy Tester Assignment via embedded Assign Testers functionality
Variant B - Single Functional Test (SFT) and Acceptance Test (AT)

Tipps & Tricks: Create Test Plan(s) with Sequences

Steps

1. Input mandatory information for Test Plan

2. ‘One Test Package per Work Package’ for Test Package Creation is automatically selected due to the WP Status values

Effects

- Test Plan automatically created
- Test Packages automatically created and adjustable
- Easy Tester Assignment via embedded Assign Testers functionality
Variant B - Single Functional Test (SFT) and Acceptance Test (AT)
Assign Tester

Steps
1. Click on Test Package Details
2. Click Assign Testers link to assign Tester for each Test Package

Remark:
Tester will be automatically assigned to the Test Package if maintained in the Work Package
Variant B - Single Functional Test (SFT) and Acceptance Test (AT)

Tipps & Tricks: Multiple Tester Assignment

- The application supports selecting 'n' testers and assigning to 'm' Test Packages

![Assign Testers Diagram]
Variant B - Single Functional Test (SFT) and Acceptance Test (AT)

Tipps & Tricks: Replace Testers

- The application supports un-assignment or replacement of testers in case they become unavailable.
- If the notify flag is checked in the first screen and Business Partner details have email address maintained, Testers can be notified if their assignments are changed.
**Variant B - Single Functional Test (SFT) and Acceptance Test (AT)**

**Release Test Plan**

**Steps**

1. Test Plan > Maintain Attributes
2. Change Release Status to "Released for Test"

**Remark:**

This step is optional according to the Release Status Schema that you are using for the test plan.
Variant B - Single Functional Test (SFT) and Acceptance Test (AT)

Overview — Activities during Test Execution

The test cases were executed and Test Status values are set. If required, a Defect is created for the related test case.

In case of a defect in the application, a software correction or configuration adjustment is needed. This changes are transported to the test system, where a retest is required.

The Test Manager monitors the Test Status; at the end of the test phase, the Project Manager is informed about the status and the Test is finally signed off.
Variant B - Single Functional Test (SFT) and Acceptance Test (AT)
Execute Test Case and Document results

Steps

1. Tester opens the test case document and run the test according to test case
2. Tester uploads the result doc via ‘drag and drop’
3. Tester creates or assigns the Defects
4. Tester does retest the Test Case, when Defect Correction is solved
5. Tester updates the test status
Tester is provided with filter criteria, helping to structure the workload.

**Steps**

1. Select overdue items (items where planned execution date is already past current date)
2. Select items Ready for Test
3. Select Tests which are now available for retests

**Remarks**

Text based search can be used to find a specific Test Plan or Test Package.

**Variant B - Single Functional Test (SFT) and Acceptance Test (AT)**

**Tipps & Tricks: Structure tester’s work in My Tests Execution**
All variants - Single Functional Test (SFT) and Acceptance Test (AT)
In case of an error: Report Defect

Steps
1. Input all the related information and assign the processor
2. Assign Category to specify the team
3. Reproduce the steps and attach screenshots to the Defect. Thus the Processor can get a quick analysis of the Defect

Recommendations
- Defect Type should be Defect. It is possible to prefill this field via the app Personalization
- System should be the test system, then later, if the Defect is fixed via a Defect Correction with transport, the transport will be imported to the test system via scheduled import variant of the Release Batch Import
After a Test Defect might have been dispatched to the corresponding processor, it will be processed by executing the following steps.

**Steps**

1. Select Defect

2. Click Actions → Set to ‘In Process’

3. Perform appropriate actions:
   - **Propose Solution:** The Defect does not need a code fix, but tester is provided with further/more detailed instructions
   - **Request Error Correction:** The Defect requires a code fix. Defect Correction will be created automatically, and the Work Package status will be changed to ‘In Repair’ automatically
   - **Set to ‘Tester Action’:** The Defect is missing information and need tester provide more details

**Remarks**

- Tester will get email notification when the Defect is in ‘Propose Solution’ or ‘Tester Action’*

* customer specific customizing needed
All variants - Single Functional Test (SFT) and Acceptance Test (AT)
In case of an error: Process Defect Correction, Create TR & Return to Retest

Steps

1. Select Defect Correction and set to ‘To Be Corrected’
2. Then navigate to tab ‘Transport for TR creation’
3. The related Work Package status will change from ‘To Be Tested’ to ‘In Repair’
4. Click icon + to create TR in the Dev system accordingly
5. Go to Dev system for bug fixing, and save the changes to the TR
6. Release the task in Dev system after bug fixing
7. Execute actions → Set to ‘Retest with Transport’
   • The TR will be released automatically
   • The Defect status will be changed to ‘Solution Proposal’ automatically
8. Optional: Tester will get email notification for retest (manually customizing needed)

* customer specific customizing needed
All variants - Single Functional Test (SFT) and Acceptance Test (AT) Monitoring

Steps to monitor Test Defect
1. Navigate to Test Suite Dashboard
2. Select test plan
3. Navigate to Defect Status tab in order to monitor all the related defects

Recommendation
- Tile Change & Release Management can be used to access additional details of Test Defects

Steps to monitor or distribute Defects and Defect Corrections
1. Navigate to Mass Change app
2. Select Defect Correction / Defects per Project and Wave
3. Search the Defect Correction within scope for monitoring
4. Distribute the unassigned Defects to responsible architects
All variants - Single Functional Test (SFT) and Acceptance Test (AT) Retest

Steps

1. Update the Defect status after retest
   - ‘Confirm’ Defect is fixed
   - Set to ‘In Process’ if Defect still exists

Effects

1. When the Defect is confirmed, the Defect Correction will be confirmed automatically. Thus the related TR will be ready for import to the Pre-production System.
2. However the import to Pre-production will take place after the related Work Package with assigned Work Item(s) and Defect Correction(s) will have been set to Status ‘Hand over to Release’.
3. The related Work Package status will change from ‘In Repair’ to ‘To be Tested’.

© 2020 SAP SE or an SAP affiliate company. All rights reserved. | PUBLIC
All variants - Single Functional Test (SFT) and Acceptance Test (AT)

Update Work Package Status when all related Test Cases are passed

Steps

1. 'Confirm Successful Test' for the related Work Package
2. The Work Package status will change to 'Successfully Tested', which indicates that this Work Package has passed SFT successfully
Variant B - Single Functional Test (SFT) and Acceptance Test (AT)
Track Test Status and create Test Report

Steps
1. Via Test Suite Dashboard, Test Manager is able to:
   - Get overview of all test type
   - Track all test plans, test package execution status
   - Track all Defect status via different dimensions
   - Check Traceability Matrix from Requirement to Defect
Variant B - Single Functional Test (SFT) and Acceptance Test (AT)
Track Test Status and create Test Report

Steps

- Status and progress is monitored with the Test Suite Dashboard and operational reports of type Test Execution Analytics
- Defects are analyzed by test manager in CRM and dispatched to the respective architects
- In case a correction is needed a Defect Correction is created (with or without transport)
- Defect Correction is confirmed but not handed over to release
  Note: It is important to first set the WP to Handed Over to Release by the release manager. By this the status of the Defect Correction is automatically switched to handed over to release (in case the Defect Correction is in scope of the Work Package). Otherwise you produce inconsistencies in subsequent systems.
- All corrections are done and retested
- The Test Cases are set to successfully tested
- In the Test Suite Dashboard navigate to view Traceability Matrix and select for the Project and Wave
- Select the WP status To be Tested and check test case status for the WPs
  - Check that for all test cases in status red a Defect exists
  - Check that for all test cases in status green no Defects are still open
  - Check for all test cases in status green that all Defect Corrections are in status successfully tested
- In case all SFT test for a WP are green navigate to the WP and set its status to successfully tested
## Functional Integration Test

<table>
<thead>
<tr>
<th>Test Types</th>
<th>Test Level</th>
<th>Test Requirement</th>
<th>Definition</th>
<th>Test System</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unit Test</td>
<td>Sprint</td>
<td>Work Item (Tech. Design)</td>
<td>Test to ensure that the work (e.g., WRICEF, Config) was implemented correctly</td>
<td>QAS</td>
</tr>
<tr>
<td>Single Functional Test</td>
<td>Wave</td>
<td>Single Work Package (Func. Spec)</td>
<td>Business function test to ensure that the feature was implemented correctly.</td>
<td>QAS</td>
</tr>
<tr>
<td><em>Business</em> Acceptance Test</td>
<td>Wave</td>
<td>Work Packages (Requirement)</td>
<td>Validation of requested functionality by the requestor.</td>
<td>QAS</td>
</tr>
<tr>
<td>Functional Integration Test</td>
<td>Wave(s)</td>
<td>Related Work Packages (Process with Requirements)</td>
<td>Early validation of modular processes.</td>
<td>QAS</td>
</tr>
<tr>
<td><em>(Optional)</em></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Functional Integration Test</td>
<td>Release</td>
<td>E2E Processes</td>
<td>E2E tests of the new or extended functionality, validating the end to end business process.</td>
<td>PRE</td>
</tr>
<tr>
<td><em>(Final)</em></td>
<td>Release</td>
<td>Productive Processes</td>
<td>Test of productive processes or functions</td>
<td>PRE</td>
</tr>
<tr>
<td>Regression Test</td>
<td>Release</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Functional Integration Test (FIT)
Option 1: Create Work Package with Requirement

To create a Work Package that is linked to all Requirements of a Process, use the 'Element' filter of the 'Requirement app':

Steps
1. Select the E2E Process in ‘Element’
2. Activate the ‘Search in sub-elements’ checkbox
3. Click on ‘Go’ to display the Requirements
4. Use the 'Select All' checkbox to select all filtered Requirements
5. Use ‘Work Package’ → ‘Create New Work Package’ to create a Work Package that is linked to all those Requirements
Functional Integration Test (FIT)
Option 1: Create Work Package with Requirement to have full traceability

Advantage

The Traceability Matrix is based on Projects & Requirements.

Creating a WP for the FIT Test Cases you can control, that all Requirements of a Process are integration tested and completed.
**Functional Integration Test (FIT)**

**Option 2: Create Work Package without Requirement**

Steps below are only to be performed if Test Cases for FIT are missing or to be updated.

**Prerequisite**

- Release is in status Prepare
- Quality Gate Hand-over to Release has not yet happened in order to create a new Work Package

**Steps**

1. Click Create New Work Package
2. Select WP Category ‘Non Functional’
3. Select Project and Wave
4. After the test case upload finished, switch the Work Package status to ‘Hand over to Release’

**Remark**

As Development Branch is Change Control enabled a, WI is needed to be created in order to upload FIT Test Cases.
As all the content can only be released to the parent branch via a change document, all content related to the process need to be assigned at least to one Work Item.

**Steps**

1. Assign the process structure in Work Package
2. Link other content like Process Description, User Guide, etc.
3. Create General Change (S1CG) in Scope tab. (Assign the Solution Architect or Test Manager as the Developer)
4. Then Solution Architect or Test Manager can use this Work Item type General Change to upload test cases

Note: For the General Change the availability of a Technical Design document is checked and reported in Solution Readiness Dashboard.

In case the General Change is not used to document changes on non-SAP functionalities you can switch this KPI check off.

Alternatively you can upload a dummy Technical Design document.

---

**Functional Integration Test (FIT) Optional: Link process element and its content to Work Package**

![Functional Integration Test (FIT) page from SAP documentation](image-url)
Functional Integration Test (FIT)
Optional: Upload additional or updated FIT Test cases

Steps

1. Go to Processes in scope for FIT (Make sure the current Branch is Development Branch)
2. Select the Work Item (Change Document) which used for Test case upload
3. Right click the Elements of <Process> area.
4. Click Show drop area
5. Use drag and drop to upload the FIT test case from local computer
6. Use Test Case Classification FIT for later filtering with Assignment Analysis
**Functional Integration Test (FIT)**

FIT Test Case Readiness Check

**Steps**

1. Select Solution and Development branch, click ‘Test Case Assignment’
2. Use filter for test case classification
3. Check whether there are WP without test cases of type FIT
Functional Integration Test (FIT)
Create Test plan for FIT/AT

Steps
1. Select Project and Wave
2. Click ‘Assignment Analysis and Test Plan Generation’
3. Select FIT WPs under scoped process
4. Click Test Plan → Save as new

Remarks:
- Within the assignment analysis, filters e.g. on test types (SFT, FIT, AT) are available, if the respective Test Cases Attributes are maintained. Further prerequisites are the Test Case Type ‘Additive’ in Solution Documentation and a document type ‘Test Document’ in transaction ‘SOLADM’ (see Appendix for more details)
- A Master Project can be used to build Cross Wave and Cross Project Test Plans (for the required prerequisites refer to the Focused Build Configuration Guide)
**Functional Integration Test (FIT)**

Create Test Plan

**Steps**

1. Input mandatory information for Test Plan:
   - **System Role ID**: PRE (Please select Pre-Production System as this FIT is to be performed in PRE system)
   - **Test Plan ID**
   - **Description**
   - **Test Classification**: Functional Integration Test
   - **Document Type**: Used for Test Notes document
   - **Planned Date**

2. Select ‘One Test Package per Work Package’ for Test Package Creation, then later create test package per process in test plan
   - In the Test Suite Dashboard, the test package test result will be mapped to each FIT test case upload related Work Package

Note: After test plan creation, the following activities are similar to SFT, please refer to the corresponding How-to!
# Regression Test

<table>
<thead>
<tr>
<th>Test Types</th>
<th>Test Level</th>
<th>Test Requirement</th>
<th>Definition</th>
<th>Test System</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unit Test</td>
<td>Sprint</td>
<td>Work Item (Tech. Design)</td>
<td>Test to ensure that the work (e.g. WRICEF, Config) was implemented correctly</td>
<td>QAS</td>
</tr>
<tr>
<td>Single Functional Test</td>
<td>Wave</td>
<td>Single Work Package (Func. Spec)</td>
<td>Business function test to ensure that the feature was implemented correctly.</td>
<td>QAS</td>
</tr>
<tr>
<td>(Business) Acceptance Test</td>
<td>Wave</td>
<td>Work Packages (Requirement)</td>
<td>Validation of requested functionality by the requestor.</td>
<td>QAS</td>
</tr>
<tr>
<td>Functional Integration Test (Optional)</td>
<td>Wave(s)</td>
<td>Related Work Packages (Process with Requirements)</td>
<td>Early validation of modular processes.</td>
<td>QAS</td>
</tr>
<tr>
<td>Functional Integration Test (Final)</td>
<td>Release</td>
<td>E2E Processes</td>
<td>E2E tests of the new or extended functionality, validating the end to end business process.</td>
<td>PRE</td>
</tr>
<tr>
<td>Regression Test</td>
<td>Release</td>
<td>Productive Processes</td>
<td>Test of productive processes or functions</td>
<td>PRE</td>
</tr>
</tbody>
</table>
Sample usage scenarios for Regression Tests

Scenarios

**Scenario 1 – Regression Test before Deploy**

- Re-execute selected Test Cases across the Test Types executed before to ensure no side-effects on test results of earlier Test Types occurred
- Scope: All units, functionalities and processes of the entire release
- Recommendation: Focus on high priority Test Cases

**Scenario 2 – Regression Tests after Waves or Sprints**

- Re-execute selected Test Cases across the Test Cases in former Waves and Sprints in order to make sure respective test results are still 'OK' after new developments or bugfixes have been introduced
- Scope: Selected units, functionalities and processes of former Waves and Sprints
- Recommendation: Focus on high priority Test Cases
Regression Test Recommendations

- Regression Test activities such as Creation of Test Plans, Test Execution can be handled very similar to the procedure described for FIT.
- To further facilitate those activities SAP Solution Manager offers:
  - Business Process Change Analyzer (BPCA) to define an optimized test scope
  - Test Automation Framework (TAF) including Component Based Test Automation (CBTA) to reduce manual testing efforts.
Regression Test Support
Business Process Change Analyzer

- Change impact analysis for business processes resulting from software change events
- Use cases:
  - Impact analysis for customizing and code changes
- Benefits:
  - Precise impact analysis and significant test scope reduction

Identify your test scope and significantly reduce your test effort for regression tests with BPCA!
Regression Test Support
BPCA Prerequisites

Solution Documentation

- Minimum: Executable Library

Usage data setup

- SAP Solution Manager Configuration: Scenario Usage Logging (root flag!)

Usage data collection

- ABAP Call Monitor (SCMON) or UPL data will be collected and extracted from managed system

BPCA Self Check

- Checks BPCA prerequisites in SAP Solution Manager and managed systems

Semi-dynamic TBOM

- Where: Execution Library or scope in Solution Documentation
- How: Background Job to generate semi dyn. TBOMs using UPL or SCMON data

Dynamic TBOM

- Where: high-priority process steps

Completeness Reports

- Check: Process Steps without (dynamic) TBOMs
Regression Test Support
Test Automation Framework

Test Automation Framework

Test Design
- Seamless integration between SAP Solution Manager and test tools from SAP and Partners
- Test Data Container (TDC) provide test data for test script parameters
- Central management of System under Tests

Test Execution
- Start of automated tests from Workcenter Test Management
- Ad-hoc start or scheduler for unattended execution
- Integrated test execution logs from partner tools

Test Result Analysis
- Test Manager can view test results from manual and automated tests in one environment
- Status and progress reports
- Integrated partner reports

Accelerated Repair
- Workflow between Tester and Test Engineer to trigger test case repair
- Rich environment for Test Engineer to analyze problems and perform repair activities
Regression Test Support
Flow to create new automated Test Configuration

Solution Documentation
Select a Business Process, Process Step or an Executable

- Order to Cash V1 - Standard Order
- CRM Sales with ERP Processing
- Create Quotation
- Create Sales Order from Quotation
- Create Outbound Delivery
- Post Goods Issue
- Create Billing Document

Test Composition Environment (TCE)
Create new Test Configuration

Select SAP or non-SAP tools
Create CBTA Test Configuration

CBTA – Test Creation Wizard
Quotation ➔ Sales Order ➔ Delivery

CBTA – Test Components / Test Script

Test Composition Environment (TCE)
Finalize Test Configuration

Test Data Assignment Wizard
Assign test data

Test Data Container

- Test Script Steps
  - Type
  - Name
  - CBTA Test Script
  - Z_CBTA_CREATE_QUOTATION
  - CBTA Test Script
  - Z_CBTA_CREATE_SALES_ORDER
  - CBTA Test Script
  - Z_CBTA_CREATE_DELIVERY
  - CBTA Test Script
  - Z_CBTA_CREATE_TRANSFER_ORDER
  - CBTA Test Script
  - Z_CBTA_POST_GOODS_ISSUE
  - CBTA Test Script
  - Z_CBTA_CREATE_INVOICE_DOCUMENT

© 2020 SAP SE or an SAP affiliate company. All rights reserved. | PUBLIC
Regression Test Support
Flow to execute automated Test Configurations

1. User starts test execution
2. Test data selection
3. Test execution: handover of Test Script + Test Data + SUT info
4. Logon and automated business process execution
5. Test results and logs
Summary
# SAP Solution Manager - Test Suite

## Capabilities

<table>
<thead>
<tr>
<th>Solution Documentation</th>
<th>Change Impact Analysis</th>
<th>Test Planning</th>
<th>Test Execution and Reporting</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Solution Documentation</strong></td>
<td><strong>Change Impact Analysis</strong></td>
<td><strong>Test Planning</strong></td>
<td><strong>Test Execution and Reporting</strong></td>
</tr>
<tr>
<td>• Executable Library</td>
<td>• Test Plan Management</td>
<td>• Test Plan</td>
<td>• Test Execution</td>
</tr>
<tr>
<td>• Process Step Library</td>
<td>• Test Plan</td>
<td>• Test Plan</td>
<td>• Manual Test Execution</td>
</tr>
<tr>
<td>• Business Processes</td>
<td>• Test Packages</td>
<td>• Test Packages</td>
<td>• Automated Test Execution</td>
</tr>
<tr>
<td>• Test Cases</td>
<td>• Test Sequences</td>
<td>• Test Sequences</td>
<td>• Defect Management</td>
</tr>
</tbody>
</table>

### Test scope estimation before upgrade

**Scope and Effort Analyzer (SEA)**

### Test scope optimization during upgrade

**Business Process Change Analyzer (BPCA)**

**Test Automation Framework**

- CBTA (Component Based Test Automation)
- eCATT
- Micro Focus UFT
- Worksoft Certify
- Tricentis Tosca
- other 3rd party test automation tools

### Supported UI technologies by SAP test tools:
- SAPGUI, WebDynpro, CRM Web-Client, SAPUI5, Fiori, ...

### Partner test tools:
- all other UI technologies

Optional 3rd party software

© 2020 SAP SE or an SAP affiliate company. All rights reserved. I CUSTOMER
SAP Solution Manager - Test Suite
Capabilities of SAP Solution Manager Test Suite and Focused Build

On Top Functionality of Focused Build for SAP Solution Manager

- Requirements / Work Package based Test Plan Generation
- Manual Testing: Test Steps
- Test Execution: My Test Executions
- Test Suite Dashboard

SAP Solution Manager – Test Suite

- Manual Test Cases
- Automated Test Cases
- Test Automation Framework with 3rd party test tool integration
- Test Composition Environment to manage SAP and partner tools
- Test Data Management
- Change Impact Analysis
- Test Plan and Test Package Management
- Test Execution of manual and automated tests
- Gap and Completeness Analytics
- Test Execution Analytics
What’s New with Focused Build SP6
Test Suite
Test Suite – Focused Build SP06 features

Where-used List

Applications

Test Steps Designer

Feature details

- List of all Test Plans, where Test Case is part of
- Direct access from the Test Case to the linked Test Plan, Package and Sequence
- The Version indicates a needed update in the Test Plan

Benefits

Transparency about Test Case usage and versions in Test Plans.
Test Suite – Focused Build SP06 features
Support multiple Document Status Schemas

Applications
Test Steps Designer

Feature details
• Test Cases can have different Document Status Schemas
• Status Schema is set at creation time and cannot be changed afterwards
• Document Status Schemas need to be activated for Test Steps in SPRO (Focused Build > Test Suite Extensions > Test Steps > General Settings for Test Steps Application)

Benefits
Possibility to differ between Test Cases which require a special Document Status Schema e.g. with a digital signature and other Test Case types.
# Test Suite – Focused Build SP06 features

## Mass Update of Test Steps Header details

### Applications

Test Steps Designer

### Feature details

- Mass Update of Header details
- Test Steps test cases in changed controlled branches or in a locked status cannot be changed via Mass Update
- Mass Download of selected Test Steps Test Cases

### Benefits

Easy and fast update of general header information for more than one test case.

---

<table>
<thead>
<tr>
<th>Test Case Name</th>
<th>Folder</th>
<th>Status</th>
<th>Priority</th>
<th>Owner</th>
</tr>
</thead>
<tbody>
<tr>
<td>Order</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Demo Test Case</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Order-to-Cash</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Order-to-Cash</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Order Processing</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Order Processing</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

![Test Steps Designer](image-url)
Test Suite – Focused Build SP06 features

Test Steps Parameters

Applications
Test Steps Designer, My Test Executions

Feature details
• Additional app to maintain Test Data for Test Steps test cases centrally
• Test Data could be defined as dependent from Test Plans, Test Package and Tester
• Parameters are available via right-mouse click in all long-text fields of a test case

Benefits
Central maintenance for Test Data related to Test Steps test cases and Test Plans.
Test Suite – Focused Build SP06 features
Landing Page Quick Filters

Applications

My Test Executions

Feature details

- Tabs for quick filtering of the Worklist of a Tester
  - Assigned: all Test Packages assigned to the User
  - Ready to Test: all Test Packages with Test Cases which are “Ready to Test”
  - Released: all Test Packages with Test Cases which are not “Ready to Test”
  - Protected: all Test Packages which are “Locked” (e.g. if a Test Plan is not allowed for execution)

Benefits

Increase the Productivity of Testers by seeing Test Packages which are Ready to Test first.
Test Suite – Focused Build SP06 features
Additional dialog for Defect Type selection

Applications
My Test Executions

Feature details
• Selection dialog appears before Defect creation
• The Defect dialog can be adjusted via Personalization (Test Suite App catalog) for each user

Benefits
Automatic Defect Type Determination based on Customizing/Personalization.
Test Suite – Focused Build SP06 features
Default Defect Priority

Applications
My Test Executions

Feature details
• The Defect Priority is automatically set related to specific rules. The following order is considered:
  • Test Case Priority
  • Test Package Priority (if no Test Case Priority exists)
  • CRM Customizing (if no Test Package Priority exists)
  • Priority: Medium (if no CRM Customizing Priority exists)

• The Defect Priority can still be changed manually

Benefits
Easier estimation of the Defect Priority.
Test Suite – Focused Build SP06 features
Take over Evidence Attachment as Defect Attachment

Applications
My Test Executions

Feature details
• Option to take over evidence document as Defect attachment, if this is uploaded for the Step with Errors
• The evidence document is attached to the Defect while creating it

Benefits
Faster assignment of Defect Attachments.
Test Suite – Focused Build SP06 features
Further improvements

Application
My Test Executions

Feature details
- Support of Web Assistant
- Limit Business Partner value help to valid partners
- UI changes on Attachments and Results tab

Application
Test Steps Designer

Feature details
- Support navigation via keyboard (tab key)
- Support of Web Assistant
- Automatic setting of Test Case Owner
- Testing Mode not changeable if used in Test Plan
- Limit Business Partner value help to valid partners
- UI changes for display mode
Test Suite – Focused Build SP06 features
Further improvements

Application
Assignment Analysis

Feature details
- Warn when removing Test Cases with execution data
- Automatic 1:1 relation between Test Package and Work Package in certain statuses

Application
Test Suite Dashboard

Feature details
- Support Category in Traceability Matrix
- Provide correct filter options for text fields
Excursus – Create Test Steps
Test Steps
Test Step Designer

Design time application to create and maintain manual test cases of type Test Steps.

Entry points
- Standalone – start from SAP Solution Manager Launchpad
- Solution Documentation (Process Steps and Executables can automatically transferred into your Test Case)

Test Steps
- Steps + sub-steps
- Descriptions, Executables
- Partner
- Attachments
- Evidence
- Result Attributes
- File upload using (csv)
Excursus – Technical Prerequisites
Check Report

With the check report you can check your project for test management Requirements.
Test Case Attribute Additive
Document Type – Test Document