Online-Library

Master Test Plan

Version 1.0

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 18.04.2014 | 1.0 | Master Test Plan | Torsten, Wanping |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

1. Introduction 4

1.1 Purpose 4

1.2 Scope 4

1.3 Intended Audience 4

1.4 Document Terminology and Acronyms 4

1.5 References 4

1.6 Document Structure 4

2. Evaluation Mission and Test Motivation 4

2.1 Background 4

2.2 Evaluation Mission 4

2.3 Test Motivators 4

3. Target Test Items 4

4. Test Approach 5

4.1 Initial Test-Idea Catalogs and Other Reference Sources 5

4.2 Testing Techniques and Types 5

4.2.1 Data and Database Integrity Testing 7

4.2.2 Function Testing 8

4.2.3 Business Cycle Testing

5. Entry and Exit Criteria 21

5.1 Test Plan 21

5.1.1 Test Plan Entry Criteria 21

5.1.2 Test Plan Exit Criteria 21

5.1.3 Suspension and Resumption Criteria 21

5.2 Test Cycles 21

5.2.1 Test Cycle Entry Criteria 21

5.2.2 Test Cycle Exit Criteria 22

5.2.3 Test Cycle Abnormal Termination 22

6. Deliverables 22

6.1 Test Evaluation Summaries 22

6.2 Reporting on Test Coverage 22

7. Test Data Management 23

# Introduction

## Purpose

Der Zweck des Iteration Test Plan ist es, alle notwendigen Informationen zur Planung und Steuerung des

Testaufwands für eine bestimmte Iteration zu sammeln. Es beschreibt den Ansatz für die Prüfung der Software

und ist der Top-Level-Plan generiert und genutzt von Managern, den Testaufwand zu lenken.

Dieser Testplan für Necabatur unterstützt die folgenden Ziele:

* Automatisierte Server-Tests
* Manuelle Client-Tests
* Test-Vorgehensweise
* Test-Ressourcen und Aufwände

## Scope

* State-based Tests (Server)
* Business Cycle Tests (Server)
* User-Interface-Tests (Client)
* Last-test (System)

## Intended Audience

* Project Manager
* Project Members

## Document Terminology and Acronyms

## References

## Document Structure

# Evaluation Mission and Test Motivation

## Background

* Funktionieren der Server-Logik sicherstellen

## Evaluation Mission

* Fehler in der Logik auffinden
* Ausreichende Performance, bei Anfragen und Suchen, aber auch beim Eintragen, sicherstellen

## Test Motivators

* Use-Cases
* Performance
* Ablauf

# Target Test Items

* Client
* Server
* System

# Test Approach

## Initial Test-Idea Catalogs and Other Reference Sources

## Testing Techniques and Types

### State-based Testing (Server)

|  |  |
| --- | --- |
| Technique Objective: | [Exercise database access methods and processes independent of the UI so you can observe and log incorrect functioning target behavior or data corruption.] |
| Technique: |  [Invoke each database access method and process, seeding each with valid and invalid data or requests for data.   Inspect the database to ensure the data has been populated as intended and all database events have occurred properly, or review the returned data to ensure that the correct data was retrieved for the correct reasons.] |
| Oracles: | [Outline one or more strategies that can be used by the technique to accurately observe the outcomes of the test. The oracle combines elements of both the method by which the observation can be made and the characteristics of specific outcome that indicate probable success or failure. Ideally, oracles will be self-verifying, allowing automated tests to make an initial assessment of test pass or failure, however, be careful to mitigate the risks inherent in automated results determination.] |
| Required Tools: | [The technique requires the following tools:   * Test Script Automation Tool * base configuration imager and restorer * backup and recovery tools * installation-monitoring tools (registry, hard disk, CPU, memory, and so forth) * database SQL utilities and tools * Data-generation tools] |
| Success Criteria: | [The technique supports the testing of all key database access methods and processes.] |
| Special Considerations: | * [Testing may require a DBMS development environment or drivers to enter or modify data directly in the databases. * Processes should be invoked manually. * Small or minimally sized databases (limited number of records) should be used to increase the visibility of any non-acceptable events.] |

### Business Cycle Testing (Server)

[Function testing of the target-of-test should focus on any requirements for test that can be traced directly to use cases or business functions and business rules. The goals of these tests are to verify proper data acceptance, processing, and retrieval, and the appropriate implementation of the business rules. This type of testing is based upon black box techniques; that is verifying the application and its internal processes by interacting with the application via the Graphical User Interface (GUI) and analyzing the output or results. The following table identifies an outline of the testing recommended for each application.]

|  |  |
| --- | --- |
| Technique Objective: | [Exercise target-of-test functionality, including navigation, data entry, processing, and retrieval to observe and log target behavior.] |
| Technique: | [Execute each use-case scenario’s individual use-case flows or functions and features, using valid and invalid data, to verify that:   the expected results occur when valid data is used   the appropriate error or warning messages are displayed when invalid data is used   each business rule is properly applied] |
| Oracles: | [Outline one or more strategies that can be used by the technique to accurately observe the outcomes of the test. The oracle combines elements of both the method by which the observation can be made and the characteristics of specific outcome that indicate probable success or failure. Ideally, oracles will be self-verifying, allowing automated tests to make an initial assessment of test pass or failure, however, be careful to mitigate the risks inherent in automated results determination.] |
| Required Tools: | [The technique requires the following tools:   * Test Script Automation Tool * base configuration imager and restorer * backup and recovery tools * installation-monitoring tools (registry, hard disk, CPU, memory, and so forth) * Data-generation tools] |
| Success Criteria: | [The technique supports the testing of:   all key use-case scenarios   all key features] |
| Special Considerations: | [Identify or describe those items or issues (internal or external) that impact the implementation and execution of function test.] |

### User Interface Testing (Client)

[Business Cycle Testing should emulate the activities performed on the over time. A period should be identified, such as one year, and transactions and activities that would occur during a year’s period should be executed. This includes all daily, weekly, and monthly cycles, and events that are date-sensitive, such as ticklers.]

|  |  |
| --- | --- |
| Technique Objective: | [Exercise target-of-test and background processes according to required business models and schedules to observe and log target behavior.] |
| Technique: | [Testing will simulate several business cycles by performing the following:   * The tests used for target-of-test’s function testing will be modified or enhanced to increase the number of times each function is executed to simulate several different users over a specified period. * All time or date-sensitive functions will be executed using valid and invalid dates or time periods. * All functions that occur on a periodic schedule will be executed or launched at the appropriate time. * Testing will include using valid and invalid data to verify the following:   + The expected results occur when valid data is used.   + The appropriate error or warning messages are displayed when invalid data is used.   + Each business rule is properly applied.] |
| Oracles: | [Outline one or more strategies that can be used by the technique to accurately observe the outcomes of the test. The oracle combines elements of both the method by which the observation can be made and the characteristics of specific outcome that indicate probable success or failure. Ideally, oracles will be self-verifying, allowing automated tests to make an initial assessment of test pass or failure, however, be careful to mitigate the risks inherent in automated results determination.] |
| Required Tools: | [The technique requires the following tools:   * Test Script Automation Tool * base configuration imager and restorer * backup and recovery tools * Data-generation tools] |
| Success Criteria: | [The technique supports the testing of all critical business cycles.] |
| Special Considerations: |  [System dates and events may require special support activities.   A business model is required to identify appropriate test requirements and procedures.] |

# Entry and Exit Criteria

## Test Plan

### Test Plan Entry Criteria

Mit Implementierung der ersten Funktionalität können die Tests beginnen.

### Test Plan Exit Criteria

Der Testplan ist durch den Erfolg der manuellen Tests beendet.

### Suspension and Resumption Criteria

[Specify the criteria that will be used to determine whether testing should be prematurely suspended or ended before the plan has been completely executed, and under what criteria testing can be resumed.]

## Test Cycles

### Test Cycle Entry Criteria

[Specify the criteria to be used to determine whether the test effort for the next Test Cycle of this **Test Plan** can begin.]

### Test Cycle Exit Criteria

[Specify the criteria that will be used to determine whether the test effort for the current Test Cycle of this **Test Plan** is deemed sufficient.]

### Test Cycle Abnormal Termination

[Specify the criteria that will be used to determine whether testing should be prematurely suspended or ended for the current test cycle, or whether the intended build candidate to be tested must be altered.]

# Deliverables

[In this section, list the various artifacts that will be created by the test effort that are useful deliverables to the various stakeholders of the test effort. Don’t list all work products; only list those that give direct, tangible benefit to a stakeholder and those by which you want the success of the test effort to be measured.]

## Test Evaluation Summaries

[Provide a brief outline of both the form and content of the test evaluation summaries, and indicate how frequently they will be produced.]

## Reporting on Test Coverage

[Provide a brief outline of both the form and content of the reports used to measure the extent of testing, and indicate how frequently they will be produced. Give an indication as to the method and tools used to record, measure, and report on the extent of testing.]

## Perceived Quality Reports

[Provide a brief outline of both the form and content of the reports used to measure the perceived quality of the product, and indicate how frequently they will be produced. Give an indication about to the method and tools used to record, measure, and report on the perceived product quality. You might include some analysis of Incidents and Change Request over Test Coverage.]

## Incident Logs and Change Requests

[Provide a brief outline of both the method and tools used to record, track, and manage test incidents, associated change requests, and their status.]

## Smoke Test Suite and Supporting Test Scripts

[Provide a brief outline of the test assets that will be delivered to allow ongoing regression testing of subsequent product builds to help detect regressions in the product quality.]

## Additional Work Products

[In this section, identify the work products that are optional deliverables or those that should not be used to measure or assess the successful execution of the **Test Plan**.]

### Detailed Test Results

[This denotes either a collection of Microsoft Excel spreadsheets listing the results determined for each test case, or the repository of both test logs and determined results maintained by a specialized test product.]

### Additional Automated Functional Test Scripts

[These will be either a collection of the source code files for automated test scripts, or the repository of both source code and compiled executables for test scripts maintained by the test automation product.]

### Test Guidelines

[Test Guidelines cover a broad set of categories, including Test-Idea catalogs, Good Practice Guidance, Test patterns, Fault and Failure Models, Automation Design Standards, and so forth.]

### Traceability Matrices

[Using a tool such as Rational RequisistePro or MS Excel, provide one or more matrices of traceability relationships between traced items.]

# Test Data Management

[Provide an outline of the workflow to be followed by the Test team in the development and execution of this **Test Plan**.]

The specific testing workflow that you will use should be documented separately in the project's Development Case. It should explain how the project has customized the base RUP test workflow (typically on a phase-by-phase basis). In most cases, we recommend you place a reference in this section of the **Test Plan** to the relevant section of the Development Case. It might be both useful and sufficient to simply include a diagram or image depicting your test workflow.

More specific details of the individual testing tasks are defined in a number of different ways, depending on project culture; for example:

* defined as a list of tasks in this section of the **Test Plan**, or in an accompanying appendix
* defined in a central project schedule (often in a scheduling tool such as Microsoft Project)
* documented in individual, "dynamic" to-do lists for each team member, which are usually too detailed to be placed in the **Test Plan**
* documented on a centrally located whiteboard and updated dynamically
* not formally documented at all

Based on your project culture, you should either list your specific testing tasks here or provide some descriptive text explaining the process your team uses to handle detailed task planning and provide a reference to where the details are stored, if appropriate.

For Master Test Plans, we recommend avoiding detailed task planning, which is often an unproductive effort if done as a front-loaded activity at the beginning of the project. A Master Test Plan might usefully describe the phases and the number of iterations, and give an indication of what types of testing are generally planned for each Phase or Iteration.

**Note**: Where process and detailed planning information is recorded centrally and separately from this Test Plan, you will have to manage the issues that will arise from having duplicate copies of the same information. To avoid team members referencing out-of-date information, we suggest that in this situation you place the minimum amount of process and planning information within the Test Plan to make ongoing maintenance easier and simply reference the "Master" source material.]