TEAM PROJECT SmartHome FORD CS

Course IT 538 SW-T

SmartHome FORD CS Test Specification

version v2022.1

By: N.N.  
 N.N.

N.N.

Supervisor: Ömer Karacan

Istanbul, 12.02.2022

Table of Contents

[1. Introduction 3](#_Toc95558345)

[2. Test planning 4](#_Toc95558346)

[3. Test monitoring and control 5](#_Toc95558347)

[4. Test analysis 6](#_Toc95558348)

[5. Test design 7](#_Toc95558349)

[5.1. Test Case FSM\_TC\_#1 7](#_Toc95558350)

[5.2. Test Case FSM\_TC\_#2 7](#_Toc95558351)

[6. Test implementation 8](#_Toc95558352)

[7. Test execution 9](#_Toc95558353)

[8. Test completion 10](#_Toc95558354)

# Introduction

# Test planning

**Determining the scope, objectives, and risks of testing**

**Making decisions about what to test**

**Which test activities will be carried out**

**Scheduling of test analysis, design, implementation, execution, and evaluation activities on** **particular dates**

**Selecting metrics for test monitoring and control**

**Determining the level of detail and structure for test documentation, e.g., by providing templates** **or example documents**

# Test monitoring and control

Test monitoring and control are supported by the evaluation of exit criteria, of which tasks may include:

**Checking test results and logs against specified coverage criteria!**

**Determining if more tests are needed (if yes, …)!**

# Test analysis

Test monitoring and control are supported by the evaluation of exit criteria, of which tasks may include:

**Analyzing the requirement specifications**

**Design and implementation information**

**The implementation of the component or system itself**

**Identifying features and sets of features to be tested**

# Test design

**Designing test cases**

**Identifying necessary test data to support test conditions and test cases**

**Identifying any required infrastructure and tools**

**Design black-box test techniques into test cases, if utilized during analysis**

# Test Case C\_TC\_FSM\_#1

<short description if needed>

|  |  |
| --- | --- |
| Component  Test Case ID Name | <ID, e.g., C\_TC\_FSM\_#1> <name, e.g., “Test Transition “Ready to Charge” to “Vehicle connected””> |
| Preconditions | <preconditions (operational conditions) that should prevail at the beginning of test case execution> |
| Test Steps | 1. …. |
| Post-Condition | <postconditions (operational conditions) that should prevail after test case execution. Description of this condition is mandatory if it is, at the same time, the precondition of the next test case.> |
| Test Data | <input test data that should be provided to test execution> |
| Expected Result | <prognosed data according to specification> |
| Actual Result | <parsed data after test case execution> |
| Verdict (Pass/Fail) | <result of comparison between “Expected Result” and “Actual Result”> |
| Code Coverage | <image> |

# Test Case C\_TC\_FSM\_TC\_#2

# Test implementation

**Creating automated test scripts**

**Arranging the test cases within a test execution schedule in a way that results in efficient test execution**

**Building the test environment (including, potentially, test harnesses, service virtualization, simulators, and other infrastructure items) and verifying that everything needed has been set up correctly**

**Preparing test data and ensuring it is properly loaded in the test environment**

# Test execution

**Executing tests either manually or by using test execution tools**

**Comparing actual results with expected results**

**Reporting defect**

**Logging the outcome of test executions based on the failures observed, e.g., pass, fail, blocked.**

# Test completion

**Creating a test summary report to be communicated to stakeholders,**

**Finalizing and archiving the test environment, the test data, the test infrastructure, and other testware [[1]](#footnote-1)for later reuse**

**Handing over the testware to the stakeholders**

**Analyzing lessons learned from the completed test activities to determine changes needed for future iterations, releases, and projects**

**Checking whether all work products are finalized**

1. **Testware** : Artifacts produced during the test process required to plan, design, and execute tests, such as documentation, scripts, inputs, expected results, set-up and clear-up procedures, files, databases, environment, and any additional software or utilities used in testing. [↑](#footnote-ref-1)