D6.1: System and Integration Test Specification	
Document ID:	SEMIAH-WP6-System_and_Integration_Test_Specification.docx
Document version:	1.0
Document status:	Finalized
Dissemination level:	PU
Deliverable number:	D6.1
Deliverable title:	System and Integration Test Specification
WP number:	WP6
Lead beneficiary:	DEVO
Main author(s):	Bjørnar Henriksen (DEVO), Erland Kolstad (DEVO, editor), Knut Inge Normann (DEVO), Rune H. Jacobsen (AU), Stefan Siegel (FRAUNHOFER)
Nature of deliverable:	R
Delivery date from Annex 1:	M12
Actual delivery date:	16 April 2015 (M14)
Funding scheme / call:	STREP-FP7-ICT-2013-11
Project / GA number:	619560
Project full title:	Scalable Energy Management Infrastructure for Aggregation of Households
Project start date:	01/03/2014
Project duration:	36 months

Funded by the
European Union

This project has received funding from the European Union's Seventh Framework Programme for research, technological development and demonstration under grant agreement no 619560.

Executive Summary

This deliverable is the result of Work Task (WT) 6.1: Write test specifications and establish test frameworks. It provides details around the test strategy and methods chosen in the Verification and Validation Plan which has been provided in deliverable D3.1.

Decisions regarding testing techniques, tools and other requirements for testing are specified in section 5 while tasks and estimates are described in section 6.

Annex A-D specifies applicable system and integration test cases, derived from the existing user stories, functional- and non-functional requirements, and the system architecture described in deliverable D3.2 Requirements and Functional Specifications.

The list of test cases will be extended and new test cases will be added when the user stories are detailed prior to, or during, feature kickoff meetings. These test cases and the detailed specification of all test cases will be documented in D6.2.

Abbreviations

D	Deliverable
DOW	Description of Work
DSO	Distribution Service Operator
EC	European Commission
GUI	Graphical User Interface
GVPP	Generic Virtual Power Plant
HEMG	Home Energy Management Gateway
ISTQB	International Software Testing Qualifications Board
SaaS	Software as a Service
SW	Software
WP	Work Package
WPL	Work Package Leader
WT	Work Task

Contents

List of Figures

List of Tables

1 Introduction

1.1 Scope

This document elaborates the requirements described in the SEMIAH Verification and Validation Plan [2], and provides test planning and management for the integration and system tests. The relationship between this document and other test planning documents are shown in the figure 1, given below.

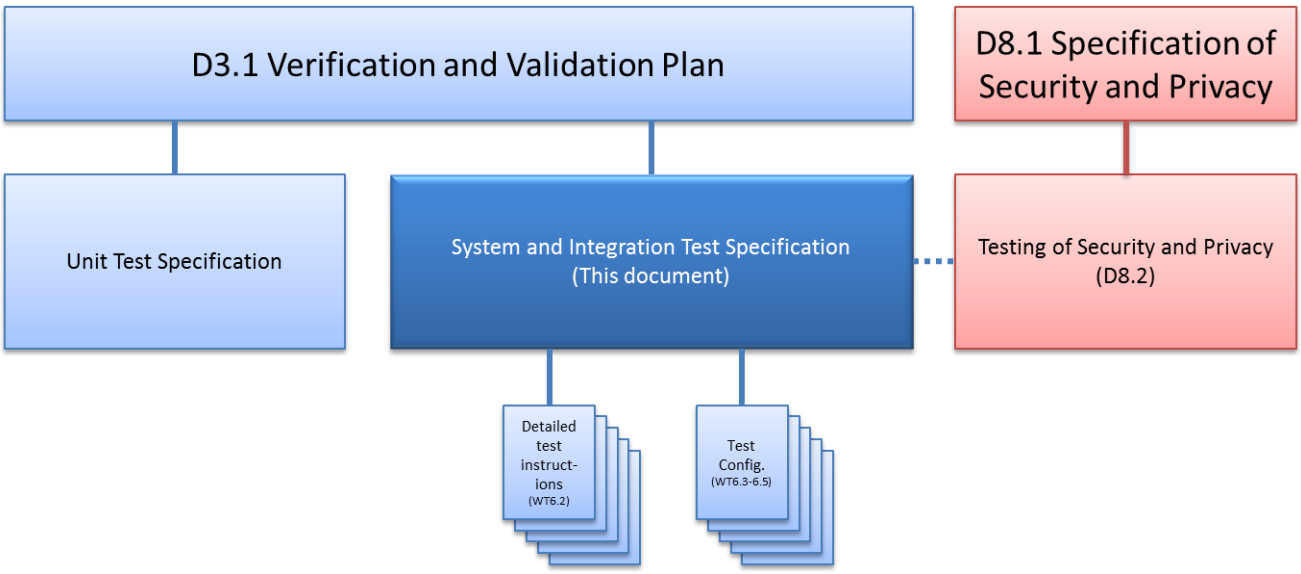

Figure 1 - Relationship between this document and other test specification and planning documents

1.2 Terms and Definitions

Unit test	Also known as component, module or program test. Search for defects in, and verifies function of, software module programs, objects, classes, etc., that are separately testable.
Integration test	Tests interfaces between components, interactions with different parts of a system, and interfaces between systems.
System test	Tests the behaviour of a whole system/product. In system testing, the test environment should correspond to the final target or production environment as much as possible.
Specification based testing	Are used to derive the test cases from the system or software requirements specifications
Boundary value analysis	Defining and testing for the boundaries of the partitions
State transition testing	Identifying all the valid states and transitions that must be tested
Exploratory testing	Exploratory testing seeks to find out how the software actually works, and to ask questions about how it will handle difficult and easy cases. The quality of the testing is dependent on the tester's skill of inventing test

cases and finding defects. The more the tester knows about the product and different test methods, the better the testing will be.[1]

2 Context of the Testing

2.1 Test Project / Test Sub-Project

The development of the SEMIAH features is conducted using agile development methodology. All features will not be made available for feature Integration at the same time, but the system will gradually build up as new features will be developed and integrated into the system. The Feature Integration and System Tests will also use this approach. As new features gets developed and integrated into the system, new versions of the system will be made available for the System Test phase. Typical test processes flow diagram is given in Fig. 2. Finally a final system test covering the whole system will be conducted before the system is released for Field Test and Pilot Operation.

[1] Kaner,C.; Bach, J.; Pettichord, B. (2001). Lessons Learned in Software Testing. John Wiley & Sons. ISBN 0-471-08112-4.

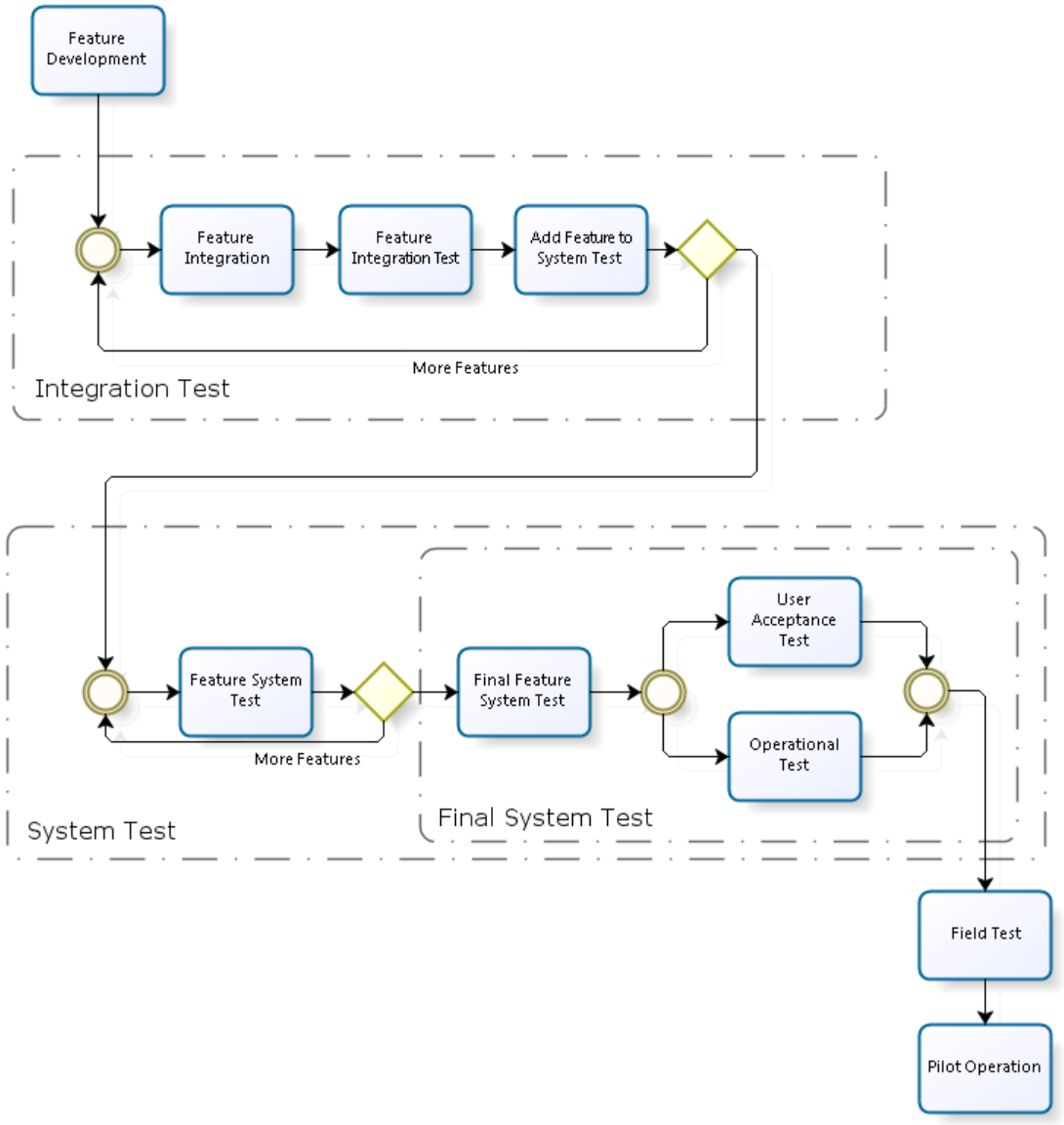

Figure 2 - Test processes

2.2 Test Item(s)

The integration test shall verify that the feature can be added to the system. It shall also verify that any interfaces or APIs get changed, or added by the feature, should be working. This test covers testing of interfaces between different sub-systems within the SEMIAH system, as well as external back-end interfaces of the system.

The system test shall be used to verify features integrated into the current system baseline. When a new feature is released from integration testing, all functionality of that feature shall be tested. In addition all functionality regression tests from earlier tested features/sub-systems shall be repeated.

The final system test is the test level where all sub-systems are connected together in a full scale test with real or simulated external components. This test will also cover user acceptance testing and operational testing of the system.

2.3 Test Scope

The integration and system test covers testing of all functionality in the SEMIAH system as described in the user stories, including internal and external interfaces. However, external wireless interfaces towards appliances or smart devices may not be included in the test scope.

The tests shall cover both features, functional and non-functional requirements.

2.4 Assumptions and Constraints

The test specification assumes all sub-systems have passed internal feature acceptance tests before entering the integration phase. Test of any features not included in the regression tests from the development phase will not be conducted in this phase.

The test plan assumes all sub-systems to be delivered with a set of test cases to be included in the feature regression tests.

It is assumed that configuration and building of the software submitted to integration testing has been verified during the pre-integration tests in the development phase.

In the plan, it is assumed that tests can be conducted remotely, or from the task leaders' premises.

All software sub-systems to enter integration tests need to be delivered no later than October 2015 (M20). Real DSO interfaces need to be available no later than January 2016 (M23), and the physical Home Energy Management Gateway with appliances need to be available not later than February 2016 (M24).

This plan assumes the SEMIAH partners to provide sufficient amount of personnel resources during the testing period. This includes testing staff, development resources for bug fixing, and management.

It is assumed all necessary documentation and specifications to be available to the system integrator and testing staff. The specification assumes detailed interface definitions, including message data details and sequences for all interfaces to be available before May 2015 (M15).

2.5 Stakeholders

List of stakeholders and their relevance to the testing, including responsibilities are shown in the SEMIAH Verification and Validation Plan [2] section 3.8.

3 Testing Lines of Communication

This section describes the lines of communication between testing and other life cycle activities (e.g. the authority for resolving issues raised by the testing tasks, and the authority for approving test products and processes), and within the testing organization.

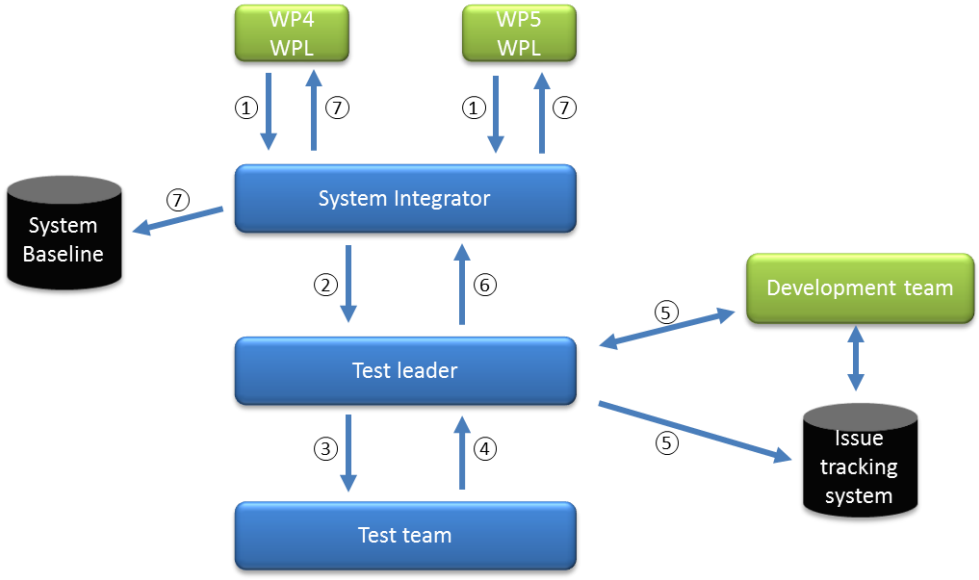

Figure 3 - Line of communications

1. Release of new feature from WP4 or WP5 for system integration
2. System integrator accepts the feature for integration testing
3. Test leader assigns test activity to test team
4. Test team reports result back to test leader
5. Test leader judges issues; updates issue tracking system, and inform the appropriate development team. Issues that cannot be resolved by the test leader and the development team are reported to the system integrator to be handled at the WP leader level.
6. Test leader reports test results back to system Integrator
7. System integrator judge test report, decides on inclusion in system baseline, and report result back to responsible development WP leader. If the feature is accepted, the System Baseline gets updated to include that new feature

4 Risk Register

4.1 Product Risks

The product risks register includes test-related product risks (e.g. defects in functionality or in non-functional aspects such as performance) and provide recommendations to treat each risk.

Risk	Mitigation action
Defects in products require retesting	
Defects in stubbed interfaces	• Quality control and verification of stubbed interfaces against specification
Test case instructions are unclear or missing	• Independent review of test case instructions
Specified test cases are not covering the relevant aspects of the product	• Ensure all user stories are verifiable and clearly understood • Independent review of test cases
Unclear requirements and user stories, or non-verifiable requirements	• Close cooperation between test and development teams during feature development kickoff

Table 1 - Product risks

4.2 Project Risks

The project risks register includes test-related project risks (e.g., schedule or resources) and provide recommendations to treat each risk.

Risk	Mitigation action
Delayed delivery of features from development teams	• Updated realistic project timeline • Close coordination with WP4 and 5
Testing require more time than expected	• Ensure component testing have been performed with sufficient quality and scope for all features entering integration testing
Competent testing staff is not available	• Detailed test plans to include both time and resource requirements
Defects in products result in more retesting than expected	• Close coordination with development teams • Test reports from component testing • Use of test automation when appropriate
Development resources are not available to fix issues discovered during testing	• Detailed test plans and schedule distributed to development teams • Development teams to commit bug fixing resources
Testing tools and procedures are not available	• Quality control of test instruction development, and require testing tools to be developed as an integrated part when specialized tools are needed

Table 2 - Project risks

5 Test Strategy

5.1 Test Sub-processes

To make the tests more manageable, the tests are divided into sub-processes as shown in Figure 2.

5.1.1 Integration Test

The integration test is used to verify that features can be added to the system without introducing issues in other parts of the system. In addition, the test shall verify that any interfaces or APIs changed or added by the feature are working, and can interact with the system.

This test phase contains only one process that is repeated for every new feature/sub-system to be integrated into the system.

A list of test cases to be included in the integration test is given in Annex A. Detailed test instructions will be developed later in the project and be included in D6.2.

5.1.2 System Test

The system test is used to verify the required functionality of the feature implemented. The test shall include verification of both functional and non-functional requirements. The test is gradually built up as features are added to the system. The system test will be divided in four sub-processes:

1. Feature system test – Used as a preliminary test to verify features iteratively as they are added to the system
2. Final feature system test – Used to verify all features of the system. This is the real system feature test.
3. User acceptance test – Used to verify implementation of business case and user stories as outlined in the DOW
4. Operational test – Used to verify operational aspects of the system, e.g. manageability, installing routines in a real environment, …

Sub-process 2, 3 and 4 contribute towards the “Final System Test”.

5.1.2.1 Feature System Test

The feature system test is used to verify implemented functionality of a feature as the system is gradually built up with new features. The test will include verification of both functional and non-functional requirements.

The list in Annex B is considered an initial system test case list, as new test cases will be added as part of the system development process. As part of the feature delivery to system integration, the development teams propose a selection of test cases from their feature test for inclusion in the system tests. Non-functional test cases that are not already included are added by a joint effort between the development team and the system integrator.

5.1.2.2 Final Feature System Test

The final feature system test shall ensure the overall functionality of the system. The test will include a full regression test of all features in the system, as well as verify all functional and non-functional requirements.

The same list of test cases as developed for the feature system test will be used, and this test will retest all these tests in a full system setup.

5.1.2.3 User Acceptance Test

The user acceptance test is used to verify the general usability of the system and that the system fulfils the overall requirements given by the DOW [1]. The testing will not be conducted with real users, but the WPLs will take on the role of users to verify the systems usability and expected functionality.

Annex C gives an initial list of test cases to be included in the user acceptance test. New test cases will be added to this list as the system is being developed and configured.

5.1.2.4 Operational Test

The operational test shall include testing of installation procedures, user management, support functions, software updates and check for security vulnerabilities.

Annex D gives an initial list of test cases to be included in the operational test. New test cases will be added to this list as the system is being developed and configured.

5.2 Retesting and Regression Testing

When new functionality is integrated into the system, all components connected to this sub-system will be regression tested. The sub-system containing the functionality will also be regression tested.

The first time the system test is executed all implemented features will be included in the test. On succeeding runs, new features will be tested fully, while test cases for features added in previous increments only will be executed if they are included in the regression test scope.

The final system test will conduct a full regression test of all features in the system. Even if the system has partly passed the final system test before, a full regression test is required for retesting in this phase.

5.3 Test Design Techniques

Technical interfaces, e.g. front-end back-end integration, will be tested using specification based testing. The test design of these tests will include boundary value analysis and state transition testing.

User interfaces and apps will be tested using specification based user stories testing and “exploratory testing”.

Detailed test instructions will be developed in WT6.2 and documented in the test management system, and delivered as part of D6.2 Test scripts for system test.

The user acceptance test and the operational test will use exploratory testing. These tests will be documented in D6.3 System Test Report.

5.4 Test Completion, Suspension and Resumption Criteria

The tests will use the criteria stated in the D3.1 [2] section 3.1.

5.5 Test Data Requirements

Test data requirements will be identified as part of the test case instruction development in WT 6.2, and documented in D6.2.

5.6 Test Environment Requirements

For the back-end solution, Fraunhofer will provide two instances of their VPP-platform as Software-as-a-Service (SaaS), one dedicated to development tasks and one for production. The iterative integration tests will be conducted against the development instance. The final integration test will be included as part of the final system test and use the production instance of the VPP-platform.

The HEMG can be either virtual instances running on a PC or physical embedded units. The integration tests conducted as part of WT 6.5 will use real physical units. DEVELCO and NETPLUS are responsible for providing virtual and/or physical embedded units of their OGEMA implementation and HEMGs and also with GVPP.

A detailed test environment description will be developed as part of WT6.3, 6.4, and 6.5. D6.3 will document the used test environments.

5.7 Testing Tool Requirements

This section lists tools to be used to manage and conduct the different tests. Examples of tools included are: Test management tool, functional testing tools, simulators, stubbed interfaces, and interface testing tools.

Test management tools

- TestRail – used to manage test cases
- Bitbucket – used for issue tracking

Test tools

- Functional test tools to be use will be decided in WT6.2, when the test instructions will be developed. The result from this task will be documented in D6.2

Test platform

- SaaS GVPP delivered by Fraunhofer IWES
- PC or Server with Java/OSGi to host virtual HEMG
- Virtual HEMG – provided by Netplus and Develco
- Real physical HEMGs – provided by Netplus and Develco, used in WT6.5 and 6.6
- Stubbed DSO interfaces – will be developed in WT6.3 and included in the integration test deliverables
- Stubbed Appliance interfaces – will be developed in WT6.3 and included in the integration test deliverables
- Network connection between virtual Front-end and GVPP

5.8 Test Deliverables

The system test result shall be presented in a test protocol, listing all test cases run with results and remarks. Issues that are not fixed immediately are reported in the central issue tracking system.

The following documents will be created as part of the SEMIAH integration and system tests:

- Test specification – this document
- Test case instructions – will be developed in WT6.2, and documented as part of D6.2 Test Scripts for system test
- Test environment and configuration – will be developed in WT6.3, WT6.4, and WT6.5. These configurations will be documented and included in D6.3 System Test Report
- Incident Reports – Reports of any issues discovered during tests, will be documented in the issue tracking system

- Test Completion Reports – Reports to be delivered after each test phase and iteration
- Final Test Completion Report – Final test report, this will be the main part of D6.3 System Test Report

Other deliverables:

- Stubbed DSO interfaces
- Stubbed appliance interfaces
- Test tools as needed

6 Testing Tasks and Estimates

The testing effort is divided into different tasks and testing sub-processes as described in section 5.1. The task schedule is described by the Gantt-diagram shown in Fig. 4.

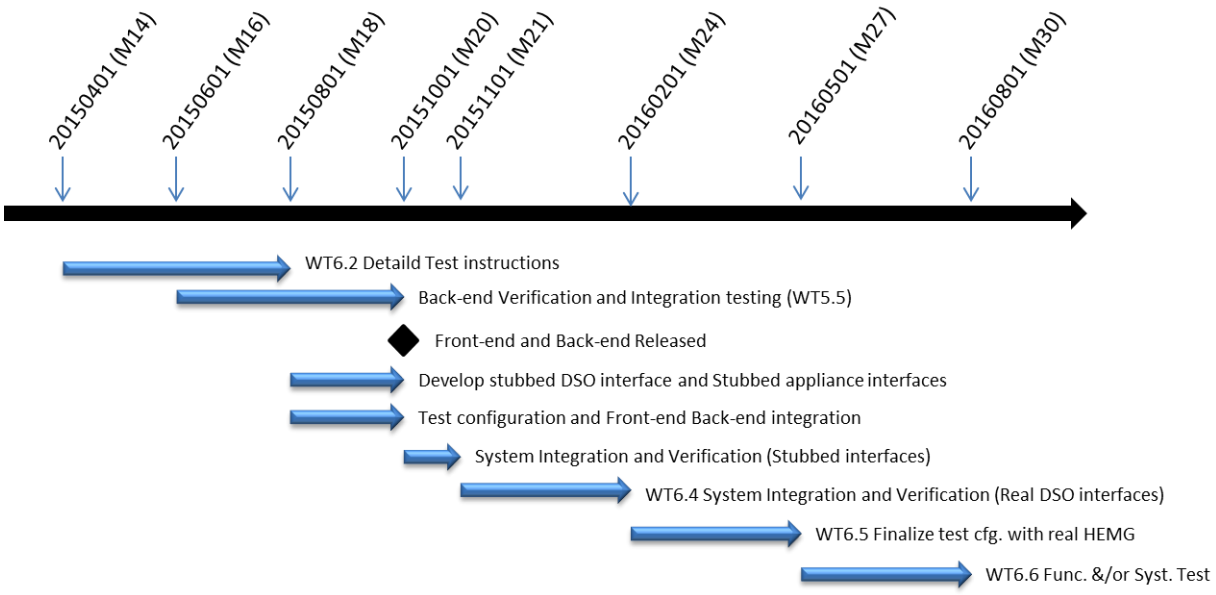

Figure 4 - Test schedule

6.1 Detailed Test Instructions

WT6.2 will elaborate the test cases in the annexes further and develop relevant test instructions. The developed test instructions will be documented in the test management tool, and in D6.2. Security testing will be covered in D8.2.

Prerequisites:

- Detailed interface definitions, including message data details and sequences. This will be provided by WP4 and WP5
- Detailed requirements from user stories and acceptance criteria

Tasks dependent on this one:

- Back-end Verification and Integration Testing
- System Integration and Verification (Stubbed Interfaces)
- System Integration and Verification (real DSO Interface)
- Finalize Test Configuration with Real HEMG
- Functional Test and/or System Test

6.2 Back-end Verification and Integration Testing

The aim of this task is to integrate the different parts of the back-end system developed in WP5, and verify their operation. Relevant test cases are included in the annexes and marked WT5.5 relevant. To reduce risks related to the front-end back-end integration, the back-end front-end interface will be included in the test using stubbed interfaces.

Prerequisites:

- Detailed Test Instructions
- Back-end system documentation
- Back-end system
- Stubbed iHousehold and iHouseholdCollection interfaces

Tasks dependent on this one:

- System Integration and Verification

6.3 Develop stubbed DSO interfaces and Stubbed Appliance Interfaces

This task shall develop stubbed DSO and Appliances interfaces to be used during the initial System Integration and Verification task. The stubbed interfaces will also be used to verify the implementation of the corresponding interfaces in the SEMIAH system.

Prerequisites:

- Detailed DSO interface and Appliance interface definitions, including message data details and sequences.

Tasks dependent on this one:

- System Integration and Verification (Stubbed Interfaces)

6.4 Test Configuration and Front-end Back-end Integration

This task shall develop a test configuration to be used in the Front-end Back-end Integration Test. This configuration will later be extended with real interfaces and real appliances. The configuration shall include documentation on how to deploy and configure the test environment, as well as needed configuration data and needed testing tools.

Prerequisites:

- Front-end system documentation
- Back-end system documentation
- Documentation of all interfaces and support systems

Tasks dependent on this one:

- System Integration and Verification (Stubbed Interfaces)
- System Integration and Verification (real DSO Interface)
- Finalize Test Configuration with Real HEMG
- Functional Test and/or System Test

6.5 System Integration and Verification (Stubbed Interfaces)

This task shall test and verify the Front-end Back-end integration and external interfaces using stubbed interfaces. The task shall also verify the user web and app interfaces developed in WT4.5[2].

Prerequisites:

- Front-end Back-end integration test configuration

[2] Because WT4.5 is scheduled to be finished after this task, there is a risk the user web and app interfaces are not available for testing during this task. This will be an issue that has been raised for the next project planning meeting.

- Front-end system documentation
- Back-end system documentation
- Documentation of all interfaces and support systems
- Verified Back-end system
- Virtual HEMG
- User interfaces
- Detailed test instructions

Tasks dependent on this one:

- System Integration and Verification (Real DSO Interfaces)
- Functional Test and/or System Test

6.6 System Integration and Verification (Real DSO Interface)

In this task the stubbed DSO interface is replaced with a connection to a real DSO. The test configuration is updated. This configuration will be used as a basis for the simulation tasks in WP7.

The integration test is extended to cover real interfaces and repeated.

Prerequisites:

- Same as System Integration and Verification (Stubbed Interfaces) with exception of stubbed DSO interface which is replaced with a real interface.

Tasks dependent on this one:

- Functional Test and/or System Test

6.7 Finalize Test Configuration with Real HEMG

The aim of this task is to replace the simulated HEMGs with physical devices and real appliances. The integration tests shall be repeated and the configuration developed will be used for the final tests.

Prerequisites:

- Integrated SEMIAH system (front-end and back-end)
- Physical HEMGs
- Appliances
- Detailed test instructions

Tasks dependent on this one:

- Functional Test and/or System Test

6.8 Functional Test and/or System Test

This task shall test the whole Demand Response system in a lab before installing it in the pilot testing households.

Prerequisites:

- Test configurations from WT6.3 or WT6.4 to be used during load testing to verify the scalability of the SEMIAH system
- Finalized test configuration with real HEMGs to be used during the final system test
- Simulator from D5.2
- Detailed test instructions

Tasks dependent on this test:

- Pilot testing in the households in Norway and Switzerland

7 Roles, Tasks, and Responsibilities

Roles, tasks, and responsibilities are described in the SEMIAH DOW. For each testing task there is a one to one mapping to work tasks in the DOW. Table 3 summarizes these responsibilities.

Task/Role	Responsible	Comments
System integration	DEVO	
Test configurations	DEVO/AU	DEVO is responsible for the initial configuration. AU is responsible for extend this configuration with real DSO interfaces
Finalized configuration	CSEM	
Integration test manager	DEVO	This task also includes iterative system tests as new functionality is added to the system
System test manager	HES-SO/DEVO	HES-SO is responsible for the final system and acceptance tests
Develop stubbed interfaces	DEVO	DEVO is responsible to make sure the stubbed interfaces are developed. It is however not decided who will perform the actual development of these interfaces.
Fix failures and issues discovered during testing	Development team who has developed the feature/sub-system with a failure or issue	

Table 3 - Roles, Tasks, and Responsibilities

8 References

[1] SEMIAH DOW (SEMIAH proposal)
[2] SEMIAH-WP3-D3.1-Verification_and_Validation_Plan Deliverable
[3] SEMIAH-WP3-D3.2-System_Requirements_and_Functional_Specifications Deliverable

Annex A Integration Test Cases

The integration test cases are derived from the technical architecture and system description in D3.2 [3].

System Under Test	Interface	Test Case ID	Test Cases	Expected result	WT5.5 (back-end) related?
1. GVPP	1. iDecisionsSupOp	INT-1.1.1	Verify the ability of the GVPP to initiate forecast services to provide forecast information to GVPP e.g., load forecasting etc. to support decisions of the GVPP and the ability of the GVPP to receive such forecasts	**1.** The GVPP is able to accept and use received forecast information e.g., load forecasting, etc. to support decisions of the GVPP	**Y**
	2. iEnergySupOp	INT-1.2.1	Verify the ability of the GVPP to initiate Energy suppliers to provide information related to energy markets such as time varying price information (Market info, Tariffs), and the ability of the GVPP to receive the information.	**1.** The GVPP is able to accept and use received information related to energy markets such as time varying price information (Market info, Tariffs)	**Y**
	3. iDsoOp	INT-1.3.1	Verify the ability of the GVPP to accept and use disseminated grid related constraints and to provide direct load control and emergency control to the DSO's	**1.**The GVPP is able to accept and use received grid related constraints	**N**
				2.The GVPP shall provide direct load control to the DSO	
				3.The GVPP shall provide emergency control to the DSO	
	4. iAlgorithms	INT-1.4.1	Verify the ability of the GVPP to receive forecasts	The GVPP is able to accept and use received forecast information	**Y**
		INT-1.4.2	Verify the ability of the GVPP to receive measurements	The GVPP is able to accept and use received measurements information	
		INT-	Verify the ability of the GVPP to receive	The GVPP is able to accept and use received	

		1.4.3	constraints	constraints information	
		INT-1.4.4	Verify the ability of the GVPP to receive electricity load aggregation	The GVPP is able to accept and use received aggregated electricity load information	
		INT-1.4.6	Verify the ability of the GVPP to receive load scheduling	The GVPP is able to accept and use received load scheduling information	
	5. iHousehold	INT-1.5.1	Verify the ability of the GVPP to receive Flexibility offers from the household appliances through the HEMG	The GVPP is able to accept and use received Flexibility offers from the household appliances through the HEMG	Y
		INT-1.5.2	Verify the ability of the GVPP to receive Measurements from the household appliances through the HEMG	The GVPP is able to accept and use received measurements information from the household appliances through the HEMG	
		INT-1.5.3	Verify the ability of the GVPP to receive Reports from the household appliances through the HEMG	The GVPP is able to accept and use received reports from the household appliances through the HEMG	
		INT-1.5.4	Verify the ability of the GVPP to provide Schedules to the household appliances through the HEMG	The schedules are provided according to specification	
	6. iHouseholdCollection	INT-1.6.1	Verify the ability of the GVPP to receive Flexibility offers for a collection of e.g., households and/or appliances through the HEMG	The GVPP is able to accept and use received Flexibility offers from collection of e.g., households and/or appliances through the HEMG	Y
		INT-1.6.2	Verify the ability of the GVPP to receive Measurements for a collection of e.g., households and/or appliances through the HEMG	The GVPP is able to accept and use received measurements information from collection of e.g., households and/or appliances through the HEMG	
		INT-1.6.3	Verify the ability of the GVPP to receive Reports for a collection of e.g., households and/or appliances through the HEMG	The GVPP is able to accept and use received reports from collection of e.g., households and/or appliances through the HEMG	
		INT-1.6.4	Verify the ability of the GVPP to provide Schedules for a collection of e.g., households and/or appliances through the HEMG	The schedules are provided according to specification	

	7. iSemiahOp	INT-1.7.1	Verify the ability of the GVPP to allow operation and maintenance of the SEMIAH system over the iSemiahOp interface	The SEMIAH system can be operated and maintained through the provided interface	**Y**
	8. iVppOp	INT-1.8.1	Verify the ability of the GVPP to allow operation and maintenance of the IWES.vpp components planned for use in the SEMIAH pilot over the iVppOp interface	The IWES.vpp components planned for used in the SEMIAH pilot can be operated and maintained	**Y**
2. HEMG	1. iHousehold (consumer)	INT-2.1.1	Verify the ability of the HEMG to provide Flexibility offers to the GVPP and/or another HEMG	The Flexibility offers are provided according to specification	**N**
		INT-2.1.2	Verify the ability of the HEMG to provide Measurements to the GVPP and/or another HEMG	The measurements are provided according to specification	
		INT-2.1.3	Verify the ability of the HEMG to provide Reports to the GVPP and/or another HEMG	The created reports are provided according to specification	
	2. iHousehold (provider)	INT-2.2.1	Verify the ability of the HEMG to receive Flexibility offers from another HEMG	The HEMG used to offload the GVPP is able to accept and use received Flexibility offers from the household appliances through the HEMG	**N**
		INT-2.2.2	Verify the ability of the HEMG to receive Measurements from another HEMG	The HEMG used to offload the GVPP is able to accept and use received measurements information from the household appliances through the HEMG	
		INT-2.2.3	Verify the ability of the HEMG to receive Reports from another HEMG	The HEMG used to offload the GVPP is able to accept and use received reports from the household appliances through the HEMG	
		INT-2.2.4	Verify the ability of the HEMG to provide Schedules to another HEMG	The schedules are provided according to specification	
	3. iHouseholdCollection (consumer)	INT-2.3.1	Verify the ability of the HEMG to provide Flexibility offers for a collection of e.g., households and/or appliances to the	The Flexibility offers are provided according to specification	**N**

			GVPP and/or another HEMG		
		INT-2.3.2	Verify the ability of the HEMG to provide Measurements for a collection of e.g., households and/or appliances to the GVPP and/or another HEMG	The measurements are provided according to specification	
		INT-2.3.3	Verify the ability of the HEMG to provide Reports for a collection of e.g., households and/or appliances to the GVPP and/or another HEMG	The created reports are provided according to specification	
	4.iHouseholdCollection (provider)	INT-2.4.1	Verify the ability of the HEMG to receive Flexibility offers for a collection of e.g., households and/or appliances from another HEMG	The HEMG used to offload the GVPP is able to accept and use received Flexibility offers from collection of e.g., households and/or appliances through the HEMG	**N**
		INT-2.4.2	Verify the ability of the HEMG to receive Measurements for a collection of e.g., households and/or appliances from another HEMG	The HEMG used to offload the GVPP is able to accept and use received measurements information from collection of e.g., households and/or appliances through the HEMG	
		INT-2.4.3	Verify the ability of the HEMG to receive Reports for a collection of e.g., households and/or appliances from another HEMG	The HEMG used to offload the GVPP is able to accept and use received reports from collection of e.g., households and/or appliances through the HEMG	
		INT-2.4.4	Verify the ability of the HEMG to provide Schedules for a collection of e.g., households and/or appliances to another HEMG	The schedules are provided according to specification	
		INT-2.4.5	Verify the ability of the HEMG to Offload the GVPP or other HEMG for a collection of e.g., households and/or appliances	The load on the GVPP is reduced as the amount of changes in flexibility offers and load are reduced	
	5. Provisioning (provider)	INT-2.5.1	Verify the ability of the HEMG to allow Installation from a provisioning service	1. Software/services can be installed on the HEMG using the provided interface	**N**
				2. Installed software/services on the HEMG	

				can be configured using the provided interface	
		INT-2.5.2	Verify the ability of the HEMG to allow SW updates from a provisioning service	The software on the HEMG can be updated using the provided interface	
		INT-2.5.3	Verify the ability of the HEMG to allow Remote support from a provisioning service	The service provider can provide remote support through the provided interface	
		INT-2.5.4	Verify the ability of the HEMG to allow Equipment monitoring from a provisioning service	The service provider can request and receive equipment monitoring data through the provided interface	
	6. UserGUI (provider)	INT-2.6.1	Verify the ability of the user to perform Profile updates through the GUI	It is possible to change the user profile through the GUI	**N**
		INT-2.6.2	Verify the ability of the user to Accept incentives through the GUI	It is possible to Accept incentives through the GUI	
		INT-2.6.3	Verify the ability of the user to perform Device configuration through the GUI	It is possible to change the device configuration through the device	
		INT-2.6.4	Verify the ability of the user GUI to provide Monitoring information to the user	The GUI provides equipment monitoring information through the GUI	
		INT-2.6.5	Verify the ability of the user GUI to provide information regarding Green points to the user	Provide achieved green points to the user through the GUI	
		INT-2.6.6	Verify the ability of the user GUI to provide Incentives to the user	Provide incentives offers to the user through the GUI	
3. SEMIAH Algorithms	1. iAlgorithms (consumer)	INT-3.1.1	Verify the ability of the SEMIAH Algorithms to provide flexibility schedules to the GVPP	1. The Algorithm is able to accept and use received information to produce new schedules	**Y**
				2. The schedules are provided according to specification	
4. EnergyOn	1. iAlgorithms (consumer)	INT-4.1.1	Verify the ability of ENERGY ON to provide flexibility schedules to the GVPP	1. The Algorithm is able to accept and use received information to produce new schedules	**Y**
				2. The schedules are provided according to specification	

5. Provisioning service	1. Provisioning (consumer)	INT-5.1.1	Verify the ability of the provisioning service to perform Installations on the HEMG	1. Possible to perform Installation of software/services on the HEMG	**N**
				2. Possible to perform configuration of installed software/services on the HEMG	
		INT-5.1.2	Verify the ability of the provisioning service to perform SW updates on the HEMG	Possible to perform SW update of HEMG	
		INT-5.1.3	Verify the ability of the provisioning service to perform Remote support on the HEMG	Possible to perform Remote support of HEMG	
		INT-5.1.4	Verify the ability of the provisioning service to perform Equipment monitoring on the HEMG	Possible to do Equipment monitoring of HEMG	

Annex B System Test Cases

The system test cases are derived from the use cases in D3.2 [3]

Main Actors	Test Case ID	Test Cases	Expected result	WT5.5(back-end) related?
1. Distribution System operator (DSO)	**SYS-1.1**	Schedule a global power profile move in the consumption pattern on a particular feeder	**1.** Congestion on that feeder is suppressed / avoided	**N**
			2. Power quality will stay within the permitted range	**N**
			3. No disconnection of intermittent renewables will be required	**N**
			4. Grid reinforcement is delayed / made unnecessary	**N**
	SYS-1.2	Schedule a global power profile move in the generation pattern on a particular feeder	**1.** Congestion on that feeder is suppressed / avoided	**N**
			2. Power quality will stay within the permitted range	**N**
			3. No disconnection of intermittent renewables will be required	**N**
			4. Grid reinforcement is delayed / made unnecessary	**N**
	SYS-1.3	Schedule a global power profile move in the storage pattern on a particular feeder	**1.** Congestion on that feeder is suppressed / avoided	**N**
			2. Power quality will stay within the permitted range	**N**
			3. No disconnection of intermittent renewables will be required	**N**
			4. Grid reinforcement is delayed / made unnecessary	**N**
	SYS-1.4	Manage the capacity reserve for power profile moves on a feeder	**1.** I know the achievable margin on that feeder in case of congestion or power quality	**N**
			2. I can adapt the margin on that feeder to my needs.	**N**

	SYS-1.5	Get informed on the success / failure of my global power profile moves	I have a feedback on my power profile move requests.	**N**
	SYS-1.6	Integrate the flexibility service into the control center operation	The flexibility service is part of my distribution system operation procedures.	**N**
	SYS-1.7	Have some control on the power profile moves requested by energy suppliers	I avoid that power profile moves requested by energy suppliers put my grid at risk	**N**
	SYS-1.8	Implement the flexibility service on a service independent framework	**1.** I can later on implement on the framework a service to collect load curves and transmit them to the market clearing entity (activity as metering system operator)	**N**
			2. I can later on implement on the framework a service allowing me to use local devices as sensors (e.g. meters) or as actuators (e.g. infeed inverters)	**N**
2. Electricity energy supplier	**SYS-2.1**	Schedule a global power profile move in the consumption pattern of my customers	**1.** I can minimize the acquisition cost of energy	**Y**
			2. I can adapt my global power profile to my intermittent generation	**Y**
			3. I can minimize the balance energy for my balance group	**Y**
			4. I can successfully respond to call control reserve activation	**Y**
	SYS-2.2	Schedule a global power profile move in the generation pattern of my customers	**1.** I can minimize the acquisition cost of energy	**Y**
			2. I can adapt my global power profile to my intermittent generation	**Y**
			3. I can minimize the balance energy for my balance group	**Y**
			4. I can successfully respond to call control reserve activation	**Y**
	SYS-2.3	Schedule a global power profile move in the storage pattern of my customers	**1.** I can minimize the acquisition cost of energy	**Y**
			2. I can adapt my global power profile to my intermittent generation	**Y**
			3. I can minimize the balance energy for my balance group	**Y**

			4. I can successfully respond to call control reserve activation	**Y**
	SYS-2.4	Manage the capacity reserve for power profile moves on my customers' premises	**1.** I can increase / decrease the margin for energy purchase / sales operations	**Y**
			2. I can be sure to dispose of a big enough margin to clear the balance energy within a calculation period	**Y**
			3. I can reliably offer bids for control reserve	**Y**
	SYS-2.5	Be informed on the success / failure of my global power profile moves	I have a feedback on my power profile move requests	**Y**
	SYS-2.6	Be informed on power profile move operations requested by DSOs on my customers' premises	I can take actions to mitigate the effect of these operations	**Y**
	SYS-2.7	Integrate the flexibility service into my market operation	The flexibility service can be used in a way similar to other market instruments.	**Y**
	SYS-2.9	Package a flexibility product for my clients (communication, rewarding scheme, contract...)	I can increase the loyalty of my current clients and acquire new ones.	**Y**
	SYS-2.12	Verify the ability to request a demand reduction on a day when supplies are too low	This is a general user story. The details can come later. The request for reduction (towards the prosumers) must come as late as possible, based on the latest predictions	**Y**
3. Prosumer	**SYS-3.1**	Synchronize my consumption / local production with the intermittent generation of renewables	I can bring a concrete contribution to the energy turnaround	**N**
	SYS-3.2	Be informed in a simple, concise and unbiased way on my electrical energy consumption, including its flexible use	**1.** I can understand my local energy consumption profile.	**N**
			2. I can understand my local energy generation profile.	**N**
			3. I can understand my local energy storage profile.	**N**
	SYS-3.3	Allow external entities to automatically exploit the flexibility of my processes	My electricity bill is reduced.	**Y**
	SYS-3.4	Verify the ability to sell my flexibility to the grid.	It shall be possible to sell my flexibility to the grid.	**N**
4. Telecom operator	**SYS-4.1**	On the basis of my telecom gateway, deploy and manage a distributed infrastructure for the operation of value added home services	**1.** I can increase my ARPU (Average Revenue Per User)	**N**
			2. I can increase the loyalty of my clients	**N**

5. Virtual power plant owner	**SYS-5.1**	Verify the ability, as a virtual power plant owner, to trade the flexible energy capacity from on the market	It shall be possible, as a virtual power plant owner, to trade the flexible energy capacity from on the market	**Y**

B.1 Non-functional Requirements for Scalability, Availability, and Interoperability

Req. id.	Test case ID	Description	Test cases	WT5.5(back-end) related?
6.1.1.1	**SYS-6.1.1.1**	Scalability: Development of aggregation, forecasting and scheduling algorithms capable of managing at least 200 000 households	Verify that aggregation, forecasting and scheduling algorithms are capable of managing at least 200 000 households	**N**
6.1.1.2	**SYS-6.1.1.2**	Availability: Integration and verification of back-end system to ensure 24/7 operation.	Verify back-end system to ensure 24/7 operation.	**N**
6.1.1.3	**SYS-6.1.1.3**	Scalability: Deployment of demonstrator in 200 households	Verify a demonstrator in 200 households	**N**
6.1.1.4	**SYS-6.1.1.4**	Scalability: Development of a large-scale simulator to emulate the behavior of 200 000 households	Verify a large-scale simulator to emulate the behavior of 200 000 households	**N**
6.1.1.5	**SYS-6.1.1.5**	Scalability: Shift of peak electrical consumption and possibility to fulfill with distributed generator	Verify shifting of peak electrical consumption in off peak period	**N**
6.1.1.6	**SYS-6.1.1.6**	Scalability: Reduce peak loads	Verify the reduction of peak loads	**N**
6.1.1.7	**SYS-6.1.1.7**	Interoperability: Solution will be compliant with the IEEE 2030-2011 standard for Smart Grid interoperability	Verify the compliance with the IEEE 2030-2011 standard for Smart Grid interoperability	**N**
6.1.1.8	**SYS-6.1.1.8**	Scalability: SEMIAH should be able to manage entire systems and millions of appliances and TWhs of energy by designing an architecture that is inherently parallelizable, scalable, reliable and robust.	Verify that SEMIAH is able to manage entire systems and millions of appliances and TWhs of energy by the design of an architecture that is inherently parallelizable, scalable, reliable and robust.	**N**
6.1.1.9	**SYS-6.1.1.9**	Performance: The system will have real-time Demand-Response within less than 5 minutes response time	Verify that the system has real-time Demand-Response within less than 5 minutes response time	**N**

Annex C User Acceptance Test Cases

The user acceptance test cases are derived from the business requirements in D3.2 [3]

Test Case ID	Test Cases	WT5.5(back-end) related?
ABT-1.1	Verify that the project has developed a novel Information and Communication Technology (ICT) infrastructure for the implementation of Demand Response (DR) in households.	**Y**
ABT-2.1	Verify that the DR infrastructure enables shifting of energy consumption to off-peak periods.	**Y**
ABT-2.2	Verify that the DR infrastructure enables shifting of energy consumption to periods with high generation of electricity from Renewable Energy Sources (RES).	**Y**
ABT-3.1	Verify that ICT framework is open.	**Y**
ABT-3.2	Verify that the ICT framework promotes an environment for the development and innovation of smart grid services in households.	**N**
ABT-4.1	Verify that the project has developed a centralized system for DR service provisioning.	**Y**
ABT-4.2	Verify that the DR system is based on electricity consumption	**Y**
ABT-4.3	Verify that the DR system supports aggregation.	**Y**
ABT-4.4	Verify that the DR system supports forecasting.	**Y**
ABT-4.5	Verify that the DR system supports scheduling.	**Y**
ABT-5.1	Verify that the project delivers a hardware solution for households.	**N**
ABT-5.2	Verify that the hardware solution enables control of electricity loads.	**N**
ABT-5.3	Verify that the hardware solution has a competitive price.	**N**
ABT-6.1	Verify that the project integrates security functions to prevent that the system is compromised.	**N**
ABT-6.2	Verify that the project integrates privacy functions to ensure that the privacy and integrity of the system users is not compromised.	**N**
ABT-7.1	Verify that the project develops new business models for electricity players to quantify costs and benefits for players in the value chain.	**N**
ABT-7.2	Verify that the project develops new business models for residential customers to quantify costs and benefits for players in the value chain.	**N**
ABT-8.1	Verify that the project contributes to the reduction of the residential user's electricity bills.	**N**
ABT-8.2	Verify that the project improves integration of RES.	**N**
ABT-8.3	Verify that the project contributes to higher stability of the electricity grid.	**N**

ABT-9.1	Verify that the project enables savings in CO2 emissions.	**N**
ABT-9.2	Verify that the project enables savings in fuel costs.	**N**
ABT-9.3	Verify that the project contributes to reduced investments in electricity network expansions.	**N**
ABT-9.4	Verify that the project contributes to reduced investments in electricity peak generation plants.	**N**

Annex D Operational Test Cases

The operational test cases are derived from the use cases in D3.2 [3]

Main Actors	Test Case ID	Test Cases	Expected result	WT5.5 (back-end) related?
1. Distribution System operator (DSO)	**SYS-1.9**	Have strong access control for controlling services and deploying updates	I avoid malicious intrusion. This includes the possibility to change digital certificates and encryption algorithms if these have been compromised. Note that simple password protection is not considered strong access control on its own.	**N**
	SYS-1.10	Protect the data of the company (both in transit and at rest) by using encryption	I can keep data confidential. This includes protecting technical information about the services which may be used to attack the system.	**N**
2. Electricity energy supplier	**SYS-2.10**	The service to be reliable and also be resistant to market manipulations and fraud.	I can provide a sustainable business and deliver reliable services to my customers and users.	**N**
	SYS-2.11	Market sensitive information to be kept confidential and ensure strong access control and ensure non-repudiation on use of the energy trading interface	The risk of fraud is reduced and I can provide a transparent and auditability service to my customers and users	**N**
	SYS-2.8	Implement the flexibility service on a service independent framework	It can be later on reused for other services to my customers (information on own consumption / production, sustainability of consumed energy, context based advices for energy efficiency…).	**N**
3. Prosumer	**SYS-3.5**	Avoid that my local generation is disconnected from the grid	The locally generated energy is not lost.	**N**
	SYS-3.6	Have control over the distribution of my personal data	I can manage my privacy as I want	**N**

	SYS-3.7	Verify that my appliances are operated in an appropriate way	The flexibility system would not damage them nor reduce their efficiency.	**N**
	SYS-3.4	To be able to take precedence over the automated flexibility control system	I can take the reins of the system if I find the behavior of the flexibility control system inappropriate.	**N**
	SYS-3.8	To be able to manage additional appliances to the flexibility control system	I have the (full) control over my premises.	**N**
	SYS-3.9	To be able to manage removal of appliances from the flexibility control system	I have the (full) control over my premises.	**N**
	SYS-3.10	To be able to configure the flexibility control system	My comfort level is not lowered below my expectations.	**N**
	SYS-3.11	Have a simple and fair contract with the entity(ies) managing my flexibility	I can trust that benefits are shared.	**N**
	SYS-3.12	Have trustworthy and secure energy management solution	Malicious actors cannot use the service as a bridgehead to monitor or attack elements in my home or computer network, malicious actors cannot exploit information about my behavior to do damage or break into my house.	**N**
	SYS-3.14	Verify the ability to connect a new smart device to the system	It shall be possible to connect a new smart device to the system	**N**
	SYS-3.15	Verify the ability for a flexibility operator to add a new prosumer	It shall be possible for a flexibility operator to add a new prosumer	**N**
	SYS-3.16	Verify the ability for a flexibility operator to monitor equipment on customer premises and keep it healthy	It shall be possible for a flexibility operator to monitor equipment on customer premises and keep it healthy	**N**
4. Telecom operator	**SYS-4.2**	Gather as many data as possible (data related to my customers)	Valorize them (through services, which are mostly unknown yet).	**N**

Annex E Selection of Test Management Tool

In the selection process of a test management tool we have considered a list of test tool requirements. We first made an initial consideration regarding the budget for such a tool and found that Spira test and TestRail were two relevant tools to consider. The table below represents the requirements and what is supported by these tools.

E.1 Comparisons

Test tool requirements	Comments	SpiraTest	TestRail
Usability and ease of use		Requires some adjustments and configuration. Can be a bit confusing before you get to know the tool.	Nice representation and easy to start using.
Possible to integrate with issue tracker (Bitbucket)	At least have a field to note bug-ID	Not possible to integrate with Bitbucket Issue Tracker, but can define a field for issue id's.	Possibility to integrate with Bitbucket Issue Tracker.
Test case management		Yes	Yes
Includes Test Case Instructions		Yes	Yes
Revision handling of test cases (with instructions)		No	Possible to see all changes to a test case (including descriptions). Not possible to easily revert to an earlier version.
Possibility to export Test Case Specifications	No matter export format it needs to be adjusted in order to be included in a report.	OK, html word etc.	OK, Excel, CSV, XML
Test reporting and possibility to export results	No matter export format it needs to be adjusted in order to be included in a report.	OK, html word etc.	OK, Excel, CSV, XML
Possibility for test status reporting		Yes	Yes

Easy to reuse and/or rerun a test case several times and show this in a summary.		Yes	Yes
Dashboard / Easy to see status of tests and progress.		Yes	Yes
Gives a good overview of a project divided in different test areas.	Relies on how the users organize their project.	Yes, flexible in how the user wants to arrange the project.	Yes, flexible in how the user wants to arrange the project.
Availability, distributed access.		Yes, can be provided as a hosted service.	Yes, can be provided as a hosted service.
Possible with different rights for different users/roles.		No	Easy to setup different users/roles with different rights both on projects and areas.
Possibility to measure coverage of requirements in written/executed test cases.		OK. Have a requirement handling section that can be connected with test cases.	Does not have a separate requirement handling section, but it can be solved by including requirements as references and measure coverage of references in a report.
Optional			
Possible to run automated tests from the test tool.	This is not considered important as we do not see that test automation will give high value to such a short project.	Includes a simple test automation framework that can start scripts and check the output (passed/failed).	Possible to integrate TestRail with an other automation system, though it will require some work and insight to do this.

E.2 Conclusion

TestRail is easier to use, the user interface has a nice presentation and provides better integration towards the selected issue tracker. It also provides better possibilities for access control for different roles/users.

SpiraTest have more functionality included in the tool but requires more to start using. Also the extra functionality is something that we don't think will be needed in this project.

Considering the totality of the results we have decided that **TestRail** is the best choice for SEMIAH.