OPEN Project

STREP Project FP7-ICT-2007-1 N.216552

[1]

Title of Document:	Indicators for technical evaluation
Editor(s):	
Author(s):	Mattia Piunti
Contributor(s):	VF-IT, Aaalborg, Clausthal, CNR
Affiliation(s):	
Date of Document:	December, 2008
OPEN Document:	D 6.3
Distribution:	Internal
Keyword List:	Technical, Evaluation, Testing
Version:	3.0

OPEN Partners:

CNR-ISTI (Italy)
Aalborg University (Denmark)
Arcadia Design (Italy)
NEC (United Kingdom)
SAP AG (Germany)
Vodafone Omnitel NV (Italy)
Clausthal University (Germany)

Title: Indicators for technical evaluation	Id Number: D6.3

Abstract

Scope of this document is to describe the technical requirements that will be used during testing of the single applications and integration framework for the respective development phases.

This deliverable verifies in technological terms the correct integration for WP6: it defines the guidelines to test the respect of the technical requirements when the result of the design phase will be available.

The technological validation will benchmark the applications and framework against all the technical indicators of quality, while maintaining the adherence to the specifications defined in the previous WPs.

Then these indicators are to be translated for testing purposes, meaning an adaptation of these requirements so that they can be effectively verified during testing analysis with concrete steps.

This topic will be enlarged and deeply analyzed in the deliverable D6.4-Testing & Validation Methodology; the evaluation test plan, in fact, will be shaped basing on the D6.3 indicators.

Title: Indicators for technical evaluation	Id Number: D6.3

Table of Contents

Title: Indicators for technical evaluation	Id Number: D6.3

1. Introduction

A technical evaluation is usually necessary for every product/service in order to assure a correct way of working. This analysis needs some inputs, like features and values to consider, in order producing an overall evaluation and a subsequent correction of the eventual issues coming out as output.

Usual aspects of technical investigation have to be coupled with specific features of the particular technology being used, in this case the outcome of the OPEN project.

The target of this deliverable is to highlight and organize this approach in order to perform an effective technical evaluation, basing on measurable parameters and on some requirements to be necessarily tested.

Usual testing procedures for technical evaluation and specific topics strictly related to the OPEN project are described in chapter 2 and 3; the list of parameters and requirements is in fact presented with their description, and coupled with the implementation in a testing environment.

Chapter 4 will define the overall structure of the test plan (D6.4) for the technical part, underlining which parameters and details have to be necessarily inserted in a complete test description: this is primary in order to match technical parameters with testing purposes and to satisfy the specific technological goals for the OPEN project.

Therefore, this document should be considered as a reference, as it happens for the deliverables D6.1 and D6.2, for the compilation of the test plan and for the test experience, which will be integrated in the development phase of the OPEN project.

While D6.4 represents the detailed compilation of the test plan, the testing results will be collected in the following deliverables, (D6.5 / D6.7).

2. Indicators commonly used for technical evaluation

In order to perform a technical evaluation, usual aspects of investigation also comprehend a set of parameters to measure/define some contributions to the Quality of Service analysis, such as reliability, availability, performances and scalability, security, adherence to standards and in general all the main indicators of quality, while maintaining the adherence to the specifications defined within the OPEN project WPs.

A translation for testing purpose has to be defined for each of them, so that an evaluation can be realized. Note that the application after its migration is in scope of this technical evaluation as well as the migration itself, in order to underline possible degradations after successful migration

Let's start from the description of what has to be tested, so the scenarios to be analyzed in the testing phase; then the indicators will be described and translated for a real evaluation.

Gaming and business are the reference applications (see D5.1 for the application design):

- ✓ Gaming is an arcade web game (Pacman-alike) in the first phase of evaluation; then a new kind of game will be developed, mixing different services: racing game, IP TV, online betting, chat, and browsing.

- ✓ The business application is related to emergency scenarios, with different organization preserving public security. The people involved can communicate, simulate and act through this application, basing on some environmental parameters tracked by sensors.

There are three kinds of devices for each of the reference applications, as they are coming out from WP5 completion (D5.2 for the prototype to test):

- PC, Mobile phone, STB for gaming
- PC, PDA, wall-screen for business

Title: Indicators for technical evaluation	Id Number: D6.3

PD

Set Top Box

Mobile phone

Wall Screen

PC

This means six different migration scenarios for each application, in which the different indicators will be evaluated, as it is represented in the two figures below. Main issue is if to test the round trip (from a device to a second one and then again to the first one) or only the one-way migration: it depends whether the application after the round trip migration remains the same. By now the round trip should be considered as it is the most general case:
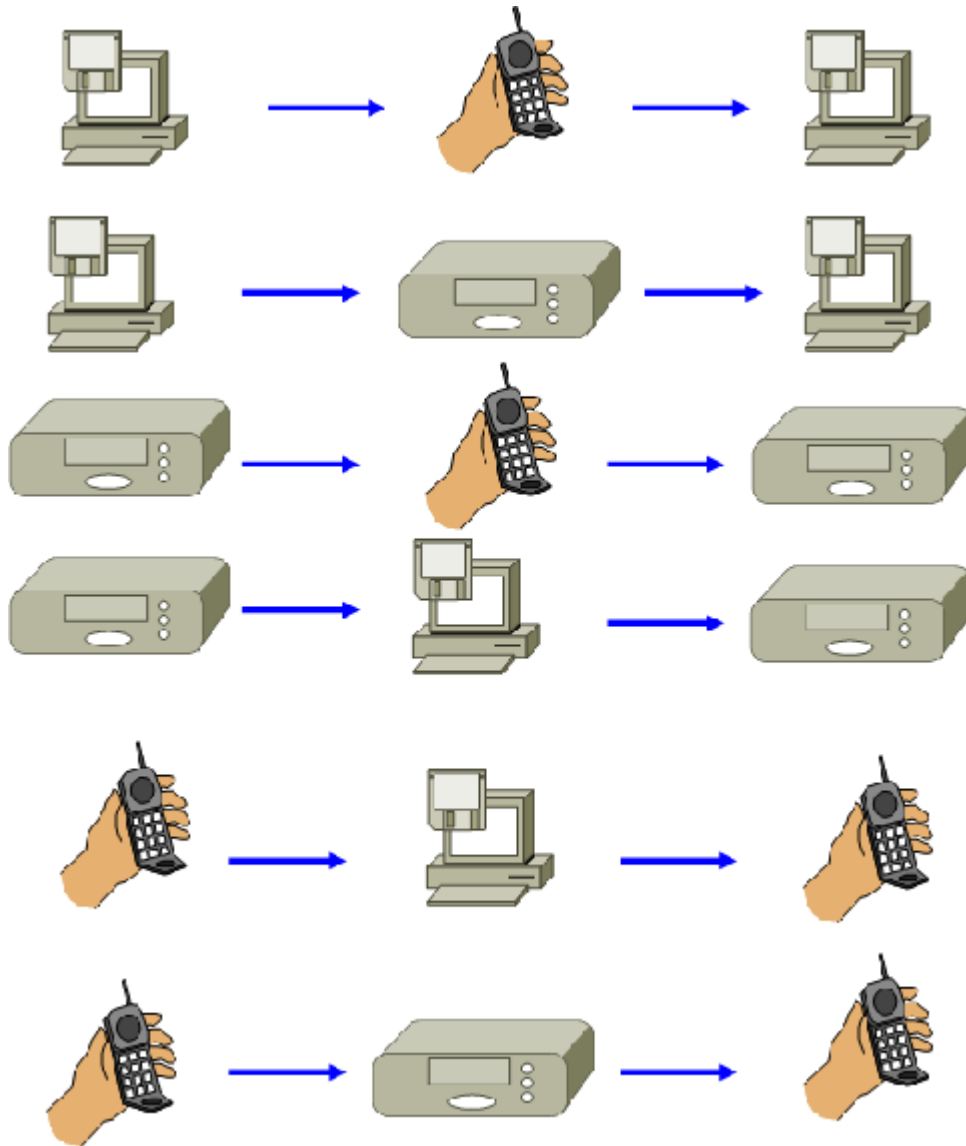
Fig.1: Scenarios for game application testing
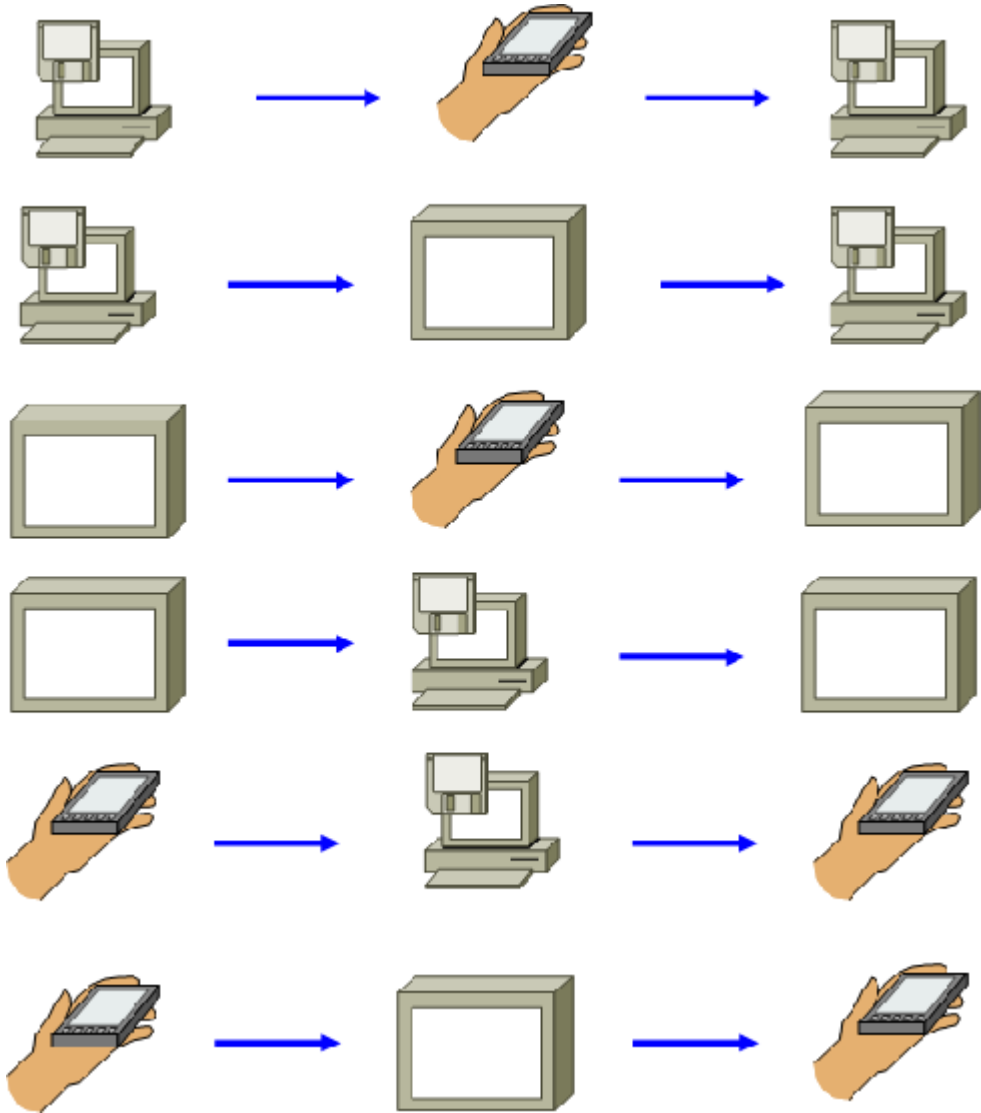
Fig.2: Scenarios for business application testing

Technical parameters and requirements should be of course also a model to follow during the building up of the solution, and not only an *"a posteriori"* check. A short description is coupled here with the testing approach:

- Availability: It is the percentage of time in which the platform is correctly working; mathematically it is represented by the formula, from the fault theory:

$$MTBF/MTBF+MTTR$$

Where:

- o MTBF is the mean time between two faults,
- o MTTR is the mean repair time

→ It has to be monitored and measured for a prefixed lasting of time, also comparing it with the usual environment parameters (3G network availability …) to verify if there are some degradations from the baseline; this indicator can be only partially evaluated, since the testing timeframe will not cover a quite long period for a relevant statistic.

- Reliability: It concerns the persistence of the availability, for a prefixed lasting time, related to the execution of necessary functionalities (e.g. a game) → It has to be monitored and measured for a prefixed lasting of time too, being only partially evaluated, since the testing timeframe will not cover a quite long period for a relevant statistic.

- Performances: They are described by the so called Key Performance Indicators (KPI), some numerical values to measure, in order to give an overall evaluation, such as delays or other timings, losses, and so on → KPI values are to be appropriately measured and compared with the baseline during a prefixed lasting time, individuating possible degradations; devices should have a log of them.

Now the values to collect and to measure have to be defined, in order to evaluate the performances of the OPEN platform.

A possible approach is to analyze two layers: the overall migration performances and the performances of an application before and after the migration (see figure 3-4)
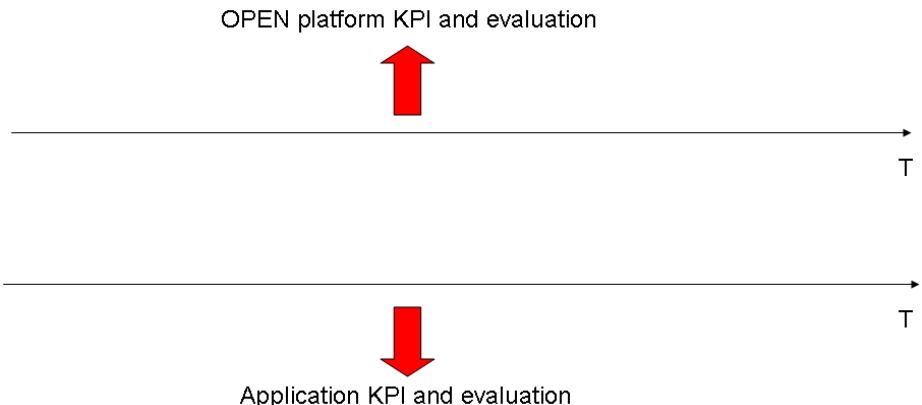

OPEN platform KPI and evaluation

T

T

Application KPI and evaluation

Figure 3: Two layers for performances analysis
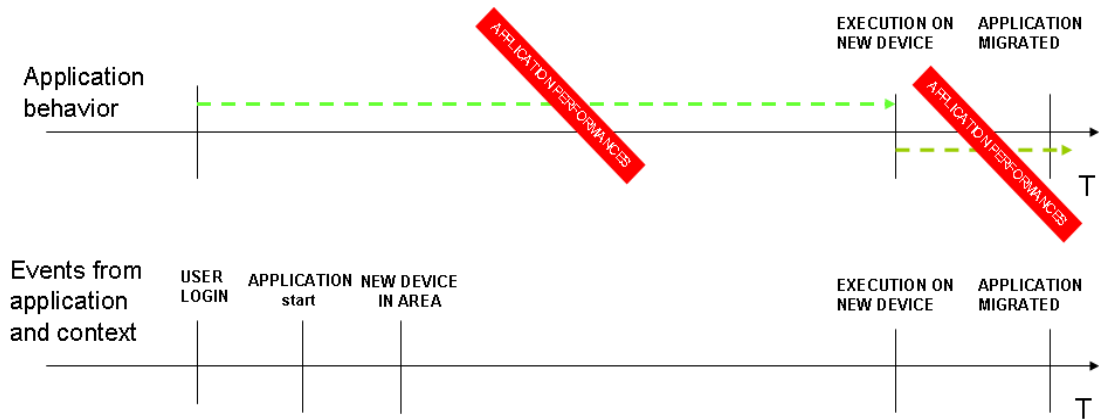
Figure 4: Application performance monitoring before and after the migration

For each migration function, described in D1.2, the necessary execution time represents a good performance indicator (the main one for computer performance); it could be coupled with other kinds of records, like happening of failures of these functionalities, to collect in the log.

This is the timing sequence of a migration, represented in figure 5; it can be analyzed to record the timing values related:

- – User asks for access (Milestone 1)
- – Security allows user entering (Milestone 2)
- – New device being in coverage (Milestone 3)
- – Device is detected (Milestone 4)
- – Context is completely retrieved (Milestone 5)
- – Trigger starts (Milestone 6)
- – Content is fully adapted (Milestone 7)
- – Application is executed on new device after the migration (Milestone 8)
- – Migration is completed (Milestone 9)
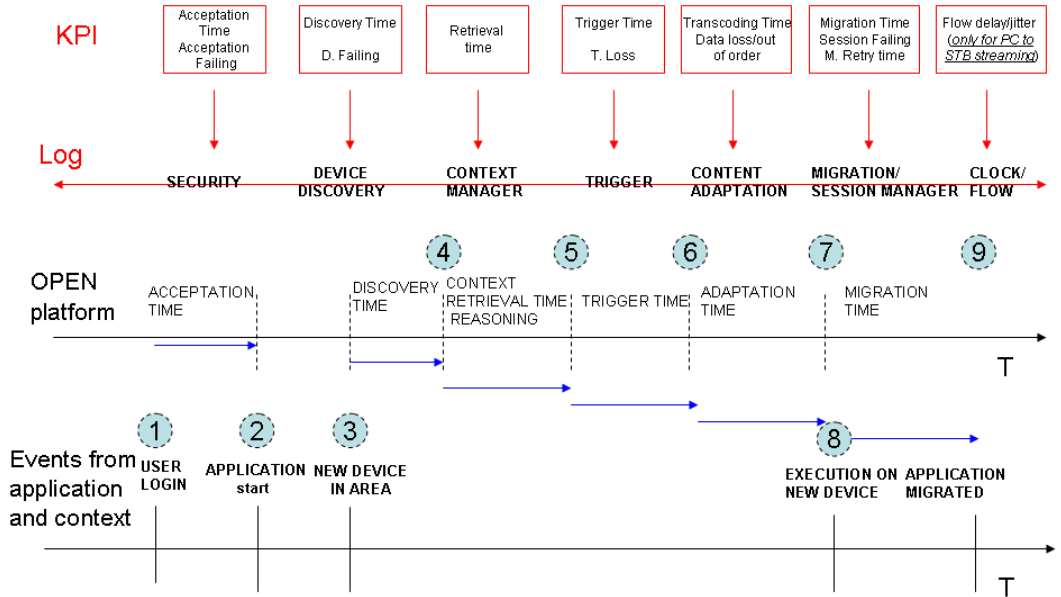
Figure 5: Timing for the performance analysis (please note that clock/flow synchronization is related only to the streaming case, PC to STB and vice versa)

The approach can be duplicated for the reference applications, gaming and business; finally, KPI collected have to be inserted in the log files of the OPEN platform and of the applications.

- Accessibility: It is the degree to which a product is accessible by as many people (from different groups and with different characteristics) as possible, the "ability to access" to its functionalities and possible benefits; it often focuses on people with disabilities, granting their right to access to different entities through the use of assistive technology. → There are several available tools the analysis can be performed through, aimed to verify this indicator:

W3C Validator for XHTML

W3C Validator for CSS

WAVE Web Access Evaluation Tool

MAGENTA

Results coming from the different evaluations can then be merged to identify possible correction areas. If the analysis is not granting satisfying effects, there are other tools, from the complete list of World Wide Web Consortium (W3C), which can be added.

- Scalability: It describes the impact (as lighter as possible) of new users/devices/applications on the platform in terms of traffic amount, necessary steps, complexity, and so on → A possible solution is to measure the growth of traffic in the system and/or the CPU usage of the migration server when new users/devices/applications are introduced in the system; this indicator can be only partially evaluated, since the users number will be quite little.

- Security: Describes the grade of inviolability from hostile acts or influences, such as sniffing, denial of service, spoofing, et cetera → It needs the introduction of the AAA procedures and a secure protocol (IPSEC-IKE?); the behavior with authorized and unauthorized users and/or devices can be verified.

- Adherence to standards: It has to be maintained for each possible environment and also to service platform standards → Basically the compliancy could be verified by the correct working of the applications towards usual specs of reference environments.

3. Specific requirements for the OPEN project

It is important to remark that previous indicators represent the main evaluation proof, but this range has to be enlarged with a set of specific features and requirements strictly related to the OPEN project, in order to define a complete and exhausting evaluation and to underline possible issues to be corrected.

So a great focus has to be given to:

1. The migration aspects
2. The reference applications, gaming and business
3. The devices that will be used and their possible limitations

These topics in fact, generate a further set of functional requirements, listed in the D1.1; in this document a subset of them (plus few additional ones) has been identified, basing on some features such as:

- ✓ Their critical importance and necessity for a correct functioning of a migration "ecosystem", made of the interaction of its components with device, applications and so on
- ✓ The feasibility of an easy way of testing these requirements, from the observation of the applications execution and of the migration
- ✓ Their relevance for testing purposes
- ✓ Their contribute to a general platform evaluation, in order to avoid requirements too context -specific

So all these requirements (listed below) will be effectively evaluated with a Y/N result, during the following test experience.

3.1. List of requirements:

Requirements are below listed through a classification based on the functional elements they are related to, in order to possibly define and isolate some issues that could arise during the test with the respect of them; for each element a table contains the requirements with their reference number from D1.1 and their typology.

Title: Indicators for technical evaluation	Id Number: D6.3

3.1.1. Application

Reference from D1.1	Requirement	Typology
7	The user must be enabled to watch a program using his set top box and multiple screens	MSP
78	Gaming anywhere, anytime, anyhow	User Interface/Migration Service Platform
117	OPEN enables the viewing and browsing of information for different users with different devices at the same time	UI
144	The OPEN platform should be able to handle, e.g. co-ordinate and synchronize, inputs from multiple-users, not only in gaming scenarios, but for others application too	MSP
152	When several users share the same screen in a multiplayer game, there must be a perfect synchronism in the input elaboration	UI/MSP

3.1.2. Migration orchestration

Reference from D1.1	Requirement	Typology
86	Migration should be triggered by the user	MSP/UI
6	System should be able to trigger a migration	MSP
82	Migration should be automatic / system triggered. Based on previous settings by the user	MSP/UI
81	Binary implementations of the services must be downloadable into the target device – A downlink is required	Network
62	Users want to use the migration process for triggering application actions, e.g. for joining a game	(MSP/UI
157	The OPEN platform should be installed and listening for any device requesting migration	MSP

54	It must be possible to continue my current service seamlessly across multiple devices	MSP

3.1.3. Context management

Reference from D1.1	Requirement	Typology
34	Service content should be provided in a context aware manner	UI/Appl. Log.
162	The OPEN platform should be able to maintain the data inserted by the user in the source device and show them in a consistent way after migration on the target device	UI/MSP
163	The OPEN platform should present the last data inserted by the user on the source device in the first presentation provided to the user in the target device	UI/MSP
106	OPEN should let me know where my data is. After it has migrated several times	MSP/UI/Appl. Log.
74	Users must be able to migrate identified parts of the application to other devices e.g. high score list	MSP/UI
61	The user does not want to care about networking aspects when trying to migrate	Network
63	OPEN should work with and without internet connection	Network

3.1.4. Content Adaptation

Reference from D1.1	Requirement	Typology
115	OPEN enables the user to get, what s/he individually can handle, i.e. the information remains not only complete, but in terms of	UI

	perceived complexity understandable after a migration	
90	The user must be able to select which content he wants to migrate to the low-end device	MSP
156	The input devices must be able support the same actions	UI
87	I should be able to migrate more than the user interface, i.e. codec, computation tasks…	MSP/Appl. Log.
Additional	Image size must fit the screen of every kind of device allowed	MSP/Network
Additional	Page has to be entirely loaded for a good user experience	MSP/Network

3.1.5. Trigger management

Reference from D1.1	Requirement	Typology
131	The offline-online migration must be triggered by network QoS parameters too	Network
Additional	The offline-online migration must be triggered by battery too	Network

3.1.6. Policy manager

Reference from D1.1	Requirement	Typology
79	The user must be able to instruct the system, not to be interrupted, e.g. by somebody waiting to join. The user wants to control who can join the game, e.g. at play time by a list	MSP/Appl. Log.
80	Users must be able to accept or deny a migration from a to b	MSP/Appl. Log.
66	The user must be able to specify migration policies, e.g. automatic migration when switched	MSP/Appl. Log.

	off	
75	Users must be able to push and pull user interfaces	MSP

3.1.7. Security

Reference from D1.1	Requirement	Typology
38	My private information should be kept safe	MSP/UI
Additional	Use of secure protocol (e.g. IPSEC/IKE)	Network

3.1.8. Clock/flow synchronization

Reference from D1.1	Requirement	Typology
Additional	Periodic actions of the applications maintain their phasing	MSP/Network

3.1.9. Session manager

Reference from D1.1	Requirement	Typology
43	Recording of sessions	MSP/Network
123	OPEN enables the users to have a complete ex-post emergency analysis ()	UI

Title: **Indicators for technical evaluation**	Id Number: **D6.3**

3.1.10. Mobility support

Reference from D1.1	Requirement	Typology
91	OPEN should predict the data and applications needed when going mobile. Possible migration also for non-OPEN service providers	Network/MSP

3.1.11. Device discovery

Reference from D1.1	Requirement	Typology
20	Users need to discover devices in the vicinity.	Network
33	Devices in geographical range (but not network range) should be usable to migrate to	Network

3.1.12. Service enablers interface

Reference from D1.1	Requirement	Typology
Additional	User status for Presence service maintained after migration	MSP/Network

4. Test plan

Test plans usually define the execution of a technical evaluation, based on testing;, since the technical evaluation methods have been described here, also the guidelines for test plan drafting should be defined; at least, this document should identify which are the necessary info and parameters to agree, among the OPEN partners, during the making of D6.4 deliverable which, in fact, will include the test plan and its translation into detailed test specifications.

VF-IT, as leader of WP6, will be directly involved in testing experience, with the other partners within the OPEN project cooperating according to the DoW recommendations.

The execution of technical evaluation testing will depend on the OPEN platform development timing (currently the D6.5, containing the evaluation results, is supposed to be completed in M20); after the completion of testing, finally results coming from test experience will be collected in the following WP6 deliverables.

Two testing timeframes are foreseen, in order to:

- First, act as an input for a following phase of software development, correcting technical issues that will arise;

- Then, definitively demonstrate the technical solidness of OPEN platform.

4.1. Test plan format

Test plan formats could be very different according to the nature of test; however, some typical sections to include can be identified; for this evaluation a procedure commonly used for testing within GSMA members can satisfy OPEN project needs.

Now a brief description of each section follows:

1. **Test Plan Identifier** - It is a unique way to refer as to the test plan, related to the ongoing project

2. **References** - The set of documents within the ongoing project to which the test plan refers

3. **Introduction** - Represents the status of the ongoing project in which the test experience is planned

4. **Lesson learned from previous experiences** - If testing is divided in more phases, each phase can contribute to modify the approach during the following, both at high level and in detail

5. **Test items** - It is a categorization of the whole testing evaluation in different areas

6. **Risk** - Potential risks, which must imply mitigation actions

7. **Features to be tested** - In-scope functionalities

8. **Features to be not tested** - Out of scope

9. **Approach** - Describes the cooperation of partners executing the evaluation and their role/actions during testing

10. **Entry/Exit criteria** - Describes, if it is feasible, how a partner can join/leave an ongoing test – OUT OF SCOPE FOR OPEN PROJECT

11. **Test environment** - Describes the testing environment with a special section dedicated to possible limitations

12. **Item pass/fail criteria** - Identifies how to declare a test case passed/failed; usually where there are multiple steps then the case shall fail if any of the steps fail

13. **Test deliverables** Suspension criteria and resumption requirements - Describes, if it is feasible, why partners can stop and then resume testing, and with what kind of requirements – OUT OF SCOPE FOR OPEN PROJECT

14. **Test deliverables and reporting** - Identifies the deliverable/report files that will be used both during and after testing to collect the results

15. **Remaining test tasks** - If testing is divided in more phases, this section describes what can be the added features to verify in the following ones

16. **Staffing and training needs** - Describes the resources and the know how needed to participants

17. **Roles and responsibilities** - Identifies precise actions for people involved in the test experience

18. **Schedule** - Defines the timing of testing, agreed among partners

19. **Post trial analysis** - Describes, if there, some analysis/evaluations, to do after the closure of testing, such as tracking analysis, statistics, and so on

Title: Indicators for technical evaluation	Id Number: D6.3

5. Conclusions

This deliverable focused on:

- What are the indicators commonly used for technical evaluation
- How to apply them to the OPEN Project for testing purposes
- A set of requirements to be set in order to perform a more complete evaluation
- What technical evaluation test procedures should be inserted in the following D6.4

Testing experience will be performed in order to evaluate and verify all the requirements and indicators that have been described here.

6. References

[1]   IEEE   http://ieeexplore.ieee.org/book/0818668520.excerpt.pdf
http://www.ieee.org/portal/site

[2] WP1 for architecture description (D1.1 for the requirements, D1.2 for the functional elements)

[3] D5.1 for applications design description

[4] D1.2 for functional elements description

[5] Wikipedia website for accessibility and performance definition:
http://en.wikipedia.org/wiki/Web_accessibility
http://en.wikipedia.org/wiki/Computer_performance

[6] Accessibility evaluation tools:

http://validator.w3.org

http://jigsaw.w3.org/css-validator

http://wave.webaim.org/

http://giove.isti.cnr.it:8080/Magenta/

Complete W3C list: http://www.w3.org/WAI/ER/tools/complete.

[7] Testing procedures for GSMA members