Tools4LEAs

**A project of the European Anti-Cybercrime Technology Development Association (EACTDA)**

EACTDA

# How Software Quality is managed within the Tools4LEAs project: an overview

This project has received funding from the European Union's ISF Police programme, under grant agreement no 101036219.

# 1. Software Quality within the Tools4LEAs project: an overview

The Tools4LEAs project(s) aim at conducting "last-mile" development projects that deliver "fully-tested" and "operational-ready" software products with no license cost and with access to the source code for EU public entities fighting cybercrime.

The Tools4LEAs project does not start from scratch, and it establishes a minimum maturity equivalent to TRLs 6-7 to start with. This is, the "last-mile" development projects start from "prototypes", normally results of security research and innovation actions, and further develop them to the level of "fully-tested" and "operational-ready" tools, an equivalent to TRL 8.

Those "Technology Providers" who own the "prototypes" (or "pre-existing tools") and that agree with the licensing scheme of the Tools4LEAs project[1], can offer them so that they are included in a catalogue of pre-existing tools. This catalogue is regularly reviewed and prioritised by the End-User Advisory Board of the Tools4LEAs project[2]. Those pre-existing tools on top of the prioritised list are then further developed in the "last-mile" development projects of the Tools4LEAs project[2].

There is a great variety within the pre-existing tools ("prototypes") that are further developed, and therefore, it is essential for the Tools4LEAs project to make use of standards to make sure that the final products delivered ("fully-tested" and "operational-ready") are of the highest quality possible.

Software is defined as collection of data, programs, procedures, associated documentation and rules that does not have any mass, volume and colour and it does not degrade over time.

And software quality is defined by the ISO/IEC 25010:2011[3] standard as the degree to which the (software) system satisfies the stated and implied needs of its various stakeholders, and thus provides value. In this standard the stakeholders' needs are represented in the software quality model, which categorises the software into eight characteristics and thirty-two sub-characteristics.

When defining detailed business requirements for the "last-mile" development projects, or when designing and conducting software testing and software evaluation activities, all the software characteristics previously described are taken into consideration.

The table below, with the list of characteristics and sub characteristics, is an extract of the one used within the Tools4LEAs project to define the business requirements that are specific to each of the "last-mile" development projects. It is presented here for illustrative purposes, so that the reader can get an understanding on how the business requirements are prepared, and subsequently how the development and testing activities are prepared and conducted.

[1] For more information about it, please contact capabilitymanager@eactda.eu

[2] Chaired by Europol (EC3 and Innovation Lab) and with +20 members from 14 different EU Member States, all of them from specialized units of European public entities fighting cybercrime.

[3] https://www.iso.org/standard/35733.html

| Software characteristics | Software sub-characteristics | KPIs and other metrics | Project specific requirements |
|---|---|---|---|
| **Functional suitability** *Degree to which a product or system provides functions that meet stated and implied needs when used under specified conditions* | • **Functional Completeness** - *Degree to which the set of functions covers all the specified tasks and user objectives* | • … | |
| | • **Functional Correctness** - *Degree to which a product or system provides the correct results with the needed degree of precision* | • Accuracy<br>• Precision<br>• … | |
| | • **Functional Appropriateness** - *Degree to which the functions facilitate the accomplishment of specified tasks and objectives* | • Response time<br>• … | |
| **Performance efficiency** *Performance relative to the amount of resources used under stated conditions* | • **Time Behaviour** - *Degree to which the response and processing times and throughput rates of a product or system, when performing its functions, meet requirements* | • … | |
| | • **Resource Utilization** - *Degree to which the amounts and types of resources used by a product or system, when performing its functions, meet requirements* | • | |
| | • **Capacity** - *Degree to which the maximum limits of a product or system parameter meet requirements* | • | |
| **Compatibility** *Degree to which a product, system or component can exchange information with other products,systems or components, andor perform its required funtions while sharing* | • **Co-existence** - *Degree to which a product can perform its required functions efficiently while sharing a common environment and resources with other products, without detrimental impact on any other product* | • | |
| | • **Interoperability** - *Degree to which two or more systems, products or components* | • Data exchange formats supported | |

| Software characteristics | Software sub-characteristics | KPIs and other metrics | Project specific requirements |
|---|---|---|---|
| *the same hrdware or software environmet* | *can exchange information and use the information that has been exchanged* | • … | |
| **Usabilit**<br>*Degre to which a product or systm can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use* | • **Appropriateness Recognizability** - *Degree to which users can recognize whether a product or system is appropriate for their needs* | • | |
| | • **Learnability** - *Degree to which a product or system can be used by specified users to achieve specified goals of learning to use the product or system with effectiveness, efficiency, freedom from risk and satisfaction in a specified context of use* | • Average time it takes to learn how to use the software<br>• Average satisfaction level<br>• … | |
| | • **Operability** - *Degree to which a product or system has attributes that make it easy to operate and control* | • … | |
| | • **User Error Protection** - *Degree to which a system protects users against making errors* | • Use of confirmation dialogs<br>• Use of standard design conventions<br>• Undo/Reset… feature<br>• … | |
| | • **User Interface Aesthetics** - *Degree to which a user interface enables pleasing and satisfying interaction for the user* | • | |
| | • **Accessibility** - *Degree to which a product or system can be used by people with the widest range of characteristics and capabilities to achieve a specified goal in a specified context of use* | • Degree to which the software can be used by persons with disabilities (visual, auditory, speech, cognitive, physical, neurological) | |

| Software characteristics | Software sub-characteristics | KPIs and other metrics | Project specific requirements |
|---|---|---|---|
| **Reliability**<br>*Degree to which a system, product or component performs specified functions under specified conditions for a specified period of time* | • **Maturity** - *Degree to which a system, product or component meets needs for reliability under normal operation* | • # of known critical/major/minor defects<br>• … | |
| | • **Availability** - *Degree to which a system, product or component is operational and accessible when required for use* | • Availability = Uptime/downtime<br>• … | |
| | • **Fault Tolerance** - *Degree to which a system, product or component operates as intended despite the presence of hardware or software faults* | • … | |
| | • **Recoverability** - *Degree to which, in the event of an interruption or a failure, a product or system can recover the data directly affected and re-establish the desired state of the system* | • … | |
| **Security**<br>*Degree to which a product or system protects information and data so that persons or other products or systems have the degree of data access appropriate to their types and levels of authorization* | • **Confidentiality** - *Degree to which a product or system ensures that data are accessible only to those authorized to have access*<br>• **Integrity** - *Degree to which a system, product or component prevents unauthorized access to, or modification of, computer programs or data*<br>• **Non-repudiation** - *Degree to which actions or events can be proven to have taken place so that the events or actions cannot be repudiated later* | • Integration with Identity Management Systems (e.g., Open LDAP, OAuth 2, SAML 2)<br>• File verification (e.g., hash)<br>• Digital signatures<br>• Logging<br>• Digital traces/footprints<br>• … | |

| Software characteristics | Software sub-characteristics | KPIs and other metrics | Project specific requirements |
|---|---|---|---|
| | • **Authenticity** - *Degree to which the identity of a subject or resource can be proved to be the one claimed*<br>• **Accountability** - *Degree to which the actions of an entity can be traced uniquely to the entity* | | |
| **Maintainability**<br>*This characteristic represents the degree of effectiveness and efficiency with which a product or system can be modified to improve it, correct it or adapt it to changes in environment, and in requirements.* | • **Modularity** - *Degree to which a system or computer program is composed of discrete components such that a change to one component has minimal impact on other components* | • Modular architecture (e.g., multi-tier, the module software design pattern)<br>• … | |
| | • **Reusability** - *Degree to which an asset can be used in more than one system, or in building other assets* | • High quality documentation (technical specifications, installation and user manuals)<br>• Use of commonly used programming languages<br>• … | |
| | • **Analysability** - *Degree of effectiveness and efficiency with which it is possible to assess the impact on a product or system of an intended change to one or more of its parts, or to diagnose a product for deficiencies or causes of failures, or to identify parts to be modified* | • Source code availability<br>• Analysis of the Software Bill of Materials (SBOM), for security/vulnerabilities as well as for licensing/IPR compliance purposes.<br>• … | |
| | • **Modifiability** - *Degree to which a product or system can be effectively and efficiently modified without introducing defects or degrading existing product quality* | • Configuration changes requiring a restart (or not)<br>• … | |

| Software characteristics | Software sub-characteristics | KPIs and other metrics | Project specific requirements |
|---|---|---|---|
| | • **Testability** - *Degree of effectiveness and efficiency with which test criteria can be established for a system, product or component and tests can be performed to determine whether those criteria have been met* | • Benchmark data availability<br>• Test data availability<br>• … | |
| **Portability**<br>*Degree of effectiveness and efficiency with which a system, product or component can be transferred from one hardware, software or other operational or usage environment to another* | • **Adaptability** - *Degree to which a product or system can effectively and efficiently be adapted for different or evolving hardware, software or other operational or usage environments* | • … | |
| | • **Installability** - *Degree of effectiveness and efficiency with which a product or system can be successfully installed and/or uninstalled in a specified environment* | • High quality installation manuals / tutorials | |
| | • **Replaceability** - *Degree to which a product can replace another specified software product for the same purpose in the same environment* | • … | |