WP1 -

DIGIT B1 - EP Pilot Project 645

Deliverable 2-Approach towards the execution of Task 2

Specific contract n°226 under Framework Contract n° DI/07172 – ABCIII

December 2015

DIGIT Fossa WP1 – Governance and quality of software code – Auditing of free and open source software.

D02.01-Approach towards the execution of Task 2

Author:

Disclaimer

Contents

List of Tables

List of Figures

Acronyms and Abbreviations

OSS	Open Source Software
OS	Operating System
FOSSA	Free and open Source Software Auditing
WP	Work Package
EC	European Commission

1 Introduction

1.1 Objective of this Document and Intended Audience

This document refers to the Deliverable 2 included within TASK-02: Analysis of software development methodologies used in the Open Source Software (OSS) communities. The objective of this deliverable is to provide the definition of the approach to execute this task in order to engage the necessary number of OSS communities that will offer a complete view of software development methodologies, best practices and tools in use in these communities.

This approach suggests the application of a step by step method to ensure the correct selection of the Open Source communities and the gathering of relevant information.

1.2 Structure of the Document

This document consists of the following chapters:

- Chapter 1: **Introduction**, which describes the objective of this deliverable.
- Chapter 2: **Methodological Approach,** which describes the steps that will guide the identification, engagement and analysis of the documentation from the Open Source Software communities.

1.3 Key Success Factors

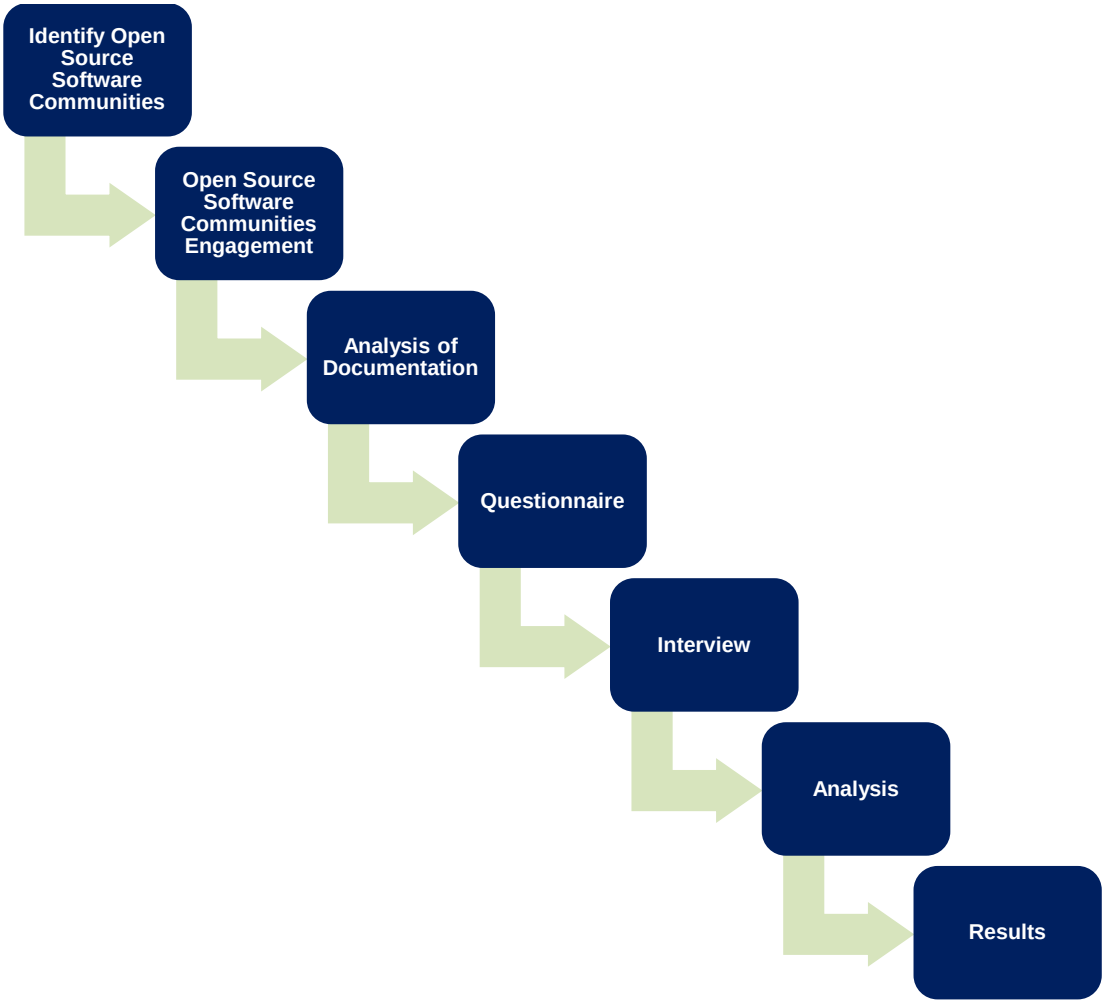
All of the steps described in Chapter 2 - Methodological approach, will ensure the fulfilment of the key success factors related to this deliverable, namely:

- To have a complete stock of methodologies used both in European Institutions and in open source communities.
- The best practices include a variety of typologies: technical, organisational and governance-related, as well as those concerned with the quality of open source software.

2 Methodological Approach

To conduct TASK-02: Analysis of software development methodologies used in the Open Source Software (OSS) communities, an approach has been defined with seven principal steps.

Figure 1. Methodological Approach - Steps



Each step is described in detail in the sections that follow.

2.1 Identify Open Source Software Communities

This is the first step of the proposed methodological approach. In order to gather enough references to ensure the correct evaluation of the methodologies used, we will provide a list of relevant Open Source Software communities and projects which meet the level of maturity required for the study. These communities will be related to Deliverable 1, which identifies the projects that are used in the European Institutions. To classify and evaluate the maturity and relevance of the identified open source communities, the following variables and metrics will be analysed:

- **Base Technology**: It represents the reference technology used by the community (Java, PHP, Linux…). The study will include well-balanced references between different technologies.
- **Category**: It represents the main category of the community: develop/support a framework, Operating System, Application, System,...
- **Major sponsors**: It defines which sponsors support the community, since when they have been supporting the community, and if they still support it. It is representative for the evaluation of the project's sustainability.
- **Latest release and release frequency**: It measures how active the community is in terms of releases. This metric monitors how fast a reported bug is fixed or a new feature is delivered.
- **Community Size**: This metric measures how many contributors support the community. It can be measured by counting the number of registered contributors and the number of contributions.
- **Activity**: This metric measures what the contribution is from different points of view (development effort, bugs reported, public discussions) to the project. It can be measured by contrasting the number of code commits, the number of tickets opened, the number of messages in the mailing list or posts on forums in a period of time.

These metrics avoid engaging non-active or volatile Open Source Software communities which may not be representative for the study.

The proposed list of OSS communities has been designed as follows.

Table 2-1 Identify Open Source Software Communities - Initial list of OSS communities

No	Community / Project / Expert	Description	Base Technology	Category
1	Spring	It is one of the most used frameworks to develop Java/JEE applications.	Java	Framework to develop Java Applications
2	Eclipse	Open source IDE mostly used for application development	Java	Integrated Development Environment
3	Jenkins	Open Source continuous integration server.	Java	Continuous Integration Server
4	Moodle	Open source web portal mostly used to create e-learning projects.	PHP	e-Learning Portal
5	Drupal	Open Source CMS Portal	PHP	Content Management System
6	WordPress	Free and Open Source CMS	PHP / MySQL	Content Management System

No	Community / Project / Expert	Description	Base Technology	Category
7	Debian	Open source community to evolve and maintain the Debian Linux distribution.	Linux	Operating System
8	MySQL	Open Source database which combines a community owned and commercial versions of the product.	MySQL	Database
9	Open Stack	Open source cloud platform.	Cloud Computing	Cloud Platform
10	Apache	Open source foundation which holds many open source software projects.	Several Technologies	Development and Support for several frameworks and projects
11	Mozilla	Free-Software community supported by Mozilla Foundation and Mozilla Corporation	Several Technologies	Development and Support for several frameworks and projects
12	Libre Office	Open Source office suite	Several Technologies	Office suite

Table 2-2 Identify Open Source Software Communities - Initial list of communities or groups that supports OSS

No	Community / Project / Expert	Description	Base Technology	Category
1	Open Invention Network	Company that acquires Patents and licenses them Royalty-free	Linux	Company specialised in patents and Linux/GNU
2	OWASP	Open source project which provides advice, methodologies, tools and technologies in the web applications security field	Security	Non-profit organisation
3	Chaos Computer Club (Open source Organizations)	Association of hackers who provide information about technical and societal matters regarding technology and hacking issues	Security	Experts community
4	Free software foundation Europe	Open Source Expert Stakeholder	N/A	Official European foundation

No	Community / Project / Expert	Description	Base Technology	Category
5	IRILL	Innovation and research initiative for free software	N/A	Research centre
6	Open Forum Academy	Think Tank that examines how the openness in computing is changing the role of computers in society	N/A	Programme established by Open Forum Europe
7	INRIA	The French National Institute for computer science and applied mathematics, promotes scientific excellence for technology transfer and society	N/A	Research Center
8	CII	Core Infrastructure Initiative	Linux	Linux Foundation
9	GitHub	A web-based service which offers free access to hosting and tools for developers of free / open-source software	N/A	Web-Based services
10	SourceForge	A web-based service which offers free access to hosting and tools for developers of free / open-source software	N/A	Web-Based services

2.2 Open Source Software Communities Engagement

We will engage the previously identified Open Source Software communities by conducting the following activities:

1. Developing a list of contacts with their contact information (name, email, phone, chat? …).
2. Contacting the development teams and the main stakeholders that represent the community sending an email with the following information:
   - A one page summary briefly describing the project and the objectives, and requesting all the public documentation concerning security software development methodologies, best practices and tools that they can share in the context of the project.
   - Request their availability for an interview to discuss in more detail the development methodologies they use, the best practices and tools, and also their opinion regarding how the European Commission can contribute to the goal of ensuring that the widely used critical

software can be trusted. A questionnaire (described in 2.4) will be sent and it will act as a guide for the interview.

All the information, collected in the form of documents, opinions and advices, will be classified in order to analyse them in step 2.6.

2.3 Documentation Analysis

In order to gather information about the tools, methodologies and best practices used by the OSS communities identified and engaged during the previous step, we will analyse the information available in public repositories and gathered during the engagement process, from centres of excellence and external experts. Some of the topics that will be analysed include, but are not limited to:

- Main methodologies used in the development projects, and main aspects related to security.
- The tools, IDE, frameworks and libraries used in the development process.
- Analysis of the security requirements that the developers follow.
- Existence of security testing and its characteristics (static, dynamic).
- Metrics to assess and rank how applications add risks.
- Process to apply security upgrades and patches to all the software supporting tools and within the applications.
- Tools and methodologies to perform code reviews and quality checks.
- Governance and organisational models and guidelines.
- Identify established channels to report bugs and vulnerabilities both for developers and end-users.
- Specific topics related to the OSS Community analysed that add value to the study.

In order to get this information, the following steps will be conducted before the interview:

- **Analysis of the existing public information**: Check for public information including website, wiki, forums and non-official communities that provide trusted documentation regarding the Open Source Software community studied.
- **Analysis of the documentation provided by the community engaged**: During the engagement process we will ask for the documentation about the topics studied, which could be provided by the community's stakeholders identified.
- **Review of internal documentation coming from our development teams:** everis uses open source products and frameworks for its own projects. everis holds centres of excellence for different technologies (Java, PHP, Security), that have a broad knowledge about the Open Source products and standards. Those centres maintain a knowledge base which provides information to support projects and learning.

  Within this step, we will retrieve and classify the information available and related to the project from the knowledge base, and we will recruit everis' experts to analyse the documentation.
- **Reviewing the most recommended practices provided by external experts in secure code:** During the OSS communities' engagement process, some secure code experts will be identified

and contacted in order to enquire about the best practices that they use or recommend. Additionally, public information provided by secure code communities and experts, such as forums, websites, wikis, blogs, webinars and workshops, will be analysed.

All gaps identified in this step will be documented, so they can be addressed during the interviews (step 2.5), Table 2 shows an example list of gaps.

Table 2-3 Documentation Analysis - Example list of gaps in the documentation

Gap	Action	Answer
The methodology to perform code reviews is not specified	Ask for the methodology and tools used for code reviews	
Tools used for the development process are not specified	Ask for relevant tools used for the development	

2.4 Questionnaire

The questionnaire is the guide for the interview that will be conducted in step 2.5. This permits to define a well-structured document to address all the questions, gaps and clarifications with regard to the documentation analysis conducted in step 2.3. Additionally, all opinions and pieces of advice made by the OSS community's contact person will be taken into account. This questionnaire will contain questions related to security in the following categories:

- Software Development lifecycle.
- Quality Assurance.
- Governance.

This step, combined with the previous ones, will ensure the fulfilment of key success factors related to this deliverable.

- To have a complete stock of methodologies used both in European Institutions and in open source communities.
- The best practices include a variety of typologies: technical, organisational and governance-related, as well as those concerned with the quality of open source software.

2.5 Interview

After the documentation analysis conducted in step 2.3, an interview will be conducted with the following attendants:

- Contact person representing each OSS community.
- everis team consultants.
- Although non-mandatory, the DIGIT Project Officer and the PMO will be invited.

This interview aims to discuss in detail topics about development methodologies, best practices, opinions, advice, tools and questions regarding the analised documentation. This interview will be guided by the questionnaire referred to in section 2.4, that will act as a guide to follow the conversation conducted either by videoconference or by phonecall. All notes, opinions, advice and extra information gathered will be classified and thoroughlystudied in step 2.6.

2.6 Analysis of Methodologies, Best Practices and Tools Used in the Open Source Software Communities

The goal of this step is to analise all the information gathered during previous steps that is relevant for the purpose of the study and provides valuable information from the perspective of the European Institutions identified in the Deliverable 1 with regard to:

- Software development methodologies in use.
- Best practices in use.
- Tools in use.
- Code Reviews.
- Security aspects
- Additional necessities identified by stakeholders and their points of view regarding how European institutions can contribute to ensure that the widely used critical software can be trusted.

The following figure shows which information sources will be taken into consideration to conduct the analisys.

Figure 2. Analysis of methodologies and tools used in the OSS communities - Information sources

- **Documentation analysis**: The documentation gathered from step 2.3 had been previously classified and analysed to define the questionnaire; however, in the event that new questions are addressed during the interview, we will fill in the gapsidentified during the documentation analysis. Once the information is complete, a thorough analysis of these documents will be conducted to extract the relevant data.
- **Recommended practices from experts:** In addition to the documentation research, recommended practices and advice from security experts will be collected and classified in step 2.3. These recommendations and advice will be taken into account for the analysis.

- **Interview results**: During the interview (step 2.5), some additional information (opinions, advice, extra information not contained in the documents) will be addressed. Since this information will be documented and classified during the interviews, it will be taken into account to complete the analysis.

2.7 Report with the Results of the Analysis of Methodologies and Tools Used in the Open Source Software Communities

A final report will be produced, with the results of the analysis performed in Step 2.6. This report will contain all the information related to the following fields:

- Detailed information about software development methodologies.
- Detailed information about best practices used.
- Detailed information about tools in use.
- Detailed information about quality assurance processes and methodologies.
- Identified governance models used by the community.
- Security aspects.
- Additional necessities identified by stakeholders and their points of view regarding how European institutions can contribute to ensure that the widely used critical software can be trusted.

This document will be structured as follows.

- Results of the analysis, grouped by communities.
- Comparison between the methodologies, tools and governance used in the communities analysed.