Project Title	Artificial Intelligence in Secure PRIvacy-preserving computing coNTinuum
Project Acronym	AI-SPRINT
Project Number	101016577
Type of project	RIA - Research and Innovation action
Topics	ICT-40-2020 - Cloud Computing: towards a smart cloud computing continuum (RIA)
Starting date of Project	01 January 2021
Duration of the project	36 months
Website	www.ai-sprint-project.eu/

D7.2 - Quality Management Plan

Work Package	WP7 | Project management
Task	T7.3 | Quality Assurance and Risks Management
Lead author	Grzegorz Timoszuk (7BULLS)
Contributors	-
Peer reviewers	Fabrizio Amarilli, Danilo Ardagna (POLIMI), André Martin (TUD)
Version	V1.0
Due Date	30/06/2021
Submission Date	30/06/2021

Dissemination Level

X	PU: Public
	CO: Confidential, only for members of the consortium (including the Commission)
	EU-RES. Classified Information: RESTREINT UE (Commission Decision 2005/444/EC)
	EU-CON. Classified Information: CONFIDENTIEL UE (Commission Decision 2005/444/EC)
	EU-SEC. Classified Information: SECRET UE (Commission Decision 2005/444/EC)

AI-Sprint - Artificial Intelligence in Secure PRIvacy-preserving computing coNTinuum, has received funding from the European Union's Horizon 2020 research and innovation programme under Grant Agreement no. 101016577.

Versioning History

Revision	Date	Editors	Comments
0.1	22.06.2021	Grzegorz Timoszuk	initial version
0.2	25.06.2021	André Martin	review
0.3	27.06.2021	Grzegorz Timoszuk	second iteration
0.4	29.06.2021	Fabrizio Amarilli	review and third iteration
0.5	29.06.2021	Danilo Ardagna	fourth iteration
1.0	30.06.2021	Grzegorz Timoszuk	final version

Glossary of terms

Item	Description
GA	General Assembly
PSC	Project and Scientific Coordinator
SD	Scientific Director
WP	Work Package
EAB	External Advisory Board
WPL	Work Package Leaders
RTB	Research Technical Board
EC	European Commission

Keywords

Artificial Intelligence; Edge Computing; Computing Continuum, Programming Models

Disclaimer

Executive Summary

This deliverable summarizes the approach taken toward ensuring high quality in the AI-SPRINT project. The document will tackle multiple domains of quality management: technical activities, project management and communication, and deliverables production. Finally, it will summarize the approach taken, the procedures and tools used to ensure high quality.

Table of Contents

1. Introduction

The aim of the AI-SPRINT "Artificial intelligence in Secure PRIvacy-preserving computing coNTinuum" project is to develop a platform composed of design and deployment tools that will support rapid development of AI applications. AI-SPRINT will provide a cloud neutral solution and it will optimize integration with AI-based edge sensor devices.

This document overviews the practices, guidelines, and tools that will be adopted to pursue a high quality of the implemented solution and prepared deliverables. The document is organised into three main sections addressing the three domains of quality management: development of technical activities (mainly aimed at software development), project management and communication management (mainly aimed at coordinating partners' activities and sharing information within the consortium and with the funding authority), and deliverable production (mainly aimed at guaranteeing adherence of the deliverables produced to the requirements of the Grant Agreement and to quality standard expected by the funding authority). The three domains of quality management are overlapping, as software development is also part of deliverable production or quality of deliverable production is underpinned by a proper coordination of activities and knowledge sharing, that are provided by effective project management and communication.

1.1 Deliverable purpose and objectives

The present document describes the different measures, including practices, guidelines, and tools, that will be used to support a streamlined, swift execution of the project activities and communication among the consortium partners and with the funding authority and guarantee adherence of the deliverables, both scientific and non-scientific, to the required quality standards. In particular, it defines rules, tools, and procedures to ensure high quality of deliverables. The document is targeted for internal use.

The objective of this document is to define the strategy that will be used for ensuring high quality. In particular it will take into account:

- Detailing activities required for preparing and conducting the various levels of testing
- Detailing activities that will be conducted during deliverable preparation
- Detailing activities executed in order to achieve high quality project management and organization
- Detailing supporting tools used in the project

1.2 Target audience

This document is intended for all the actors, project partners and external stakeholders involved in the AI-SPRINT project. Project partners include the project coordinator, researchers, and non-scientific personnel. External stakeholders comprise all the actors that, in different forms, contribute to or influence the proper execution of the project activities and include, for example, suppliers, the members of the External Advisory Board, the members of the AI-SPRINT Alliance. The indications contained in the present report will be shared (and have already been partially shared during the project proposal phase and during the preparation of the deliverable) among all partners. Partners will also take care of guaranteeing that also external stakeholders comply with the quality provisions.

1.3 Quality management strategy

AI-SPRINT is not only a software project but is a complex initiative involving multiple institutions and tackling multiple aspects of development. Three main areas of quality management have been identified:

- technical/scientific activities
- non-technical activities

- project management
- deliverable production and reporting to the European Commission

This division will be resembled in this document where every area will have a separate chapter. Each chapter will tackle tools and procedures specific to mentioned above areas. In technical/scientific areas the document will focus on testing and formal processes while for the non-technical activities, the quality management will focus on the procedures and tools. Project management will take into account meetings and tools around them. Finally, there will be a summary of quality management for deliverables for European Commission.

2. Quality procedures in Technical/Scientific activities

This chapter will be focused on the testing and technical aspects of the quality management. It will cover the project from the design phase until the acceptance tests. Additionally, it will cover the tools supporting an effective collaboration during development and testing.

2.1 Quality procedures in Design phase

A proper execution of the design phase is crucial for the success of a project. Standard quality procedures in software engineering agree on an involvement of end users and stakeholders early in the design phase. For AI-SPRINT, it is of high importance to include input from use case partners to ensure that the end result will be usable and well received. To achieve this result, there will be a strict methodology for analysis implemented. Among the several methodologies adopted for the quality design and development of technical activities the Personas technique (Cooper, Raimann, and Cronim 2007; Castro, Acuna, Juristo 2008) has been selected. This approach is used in the HCI (Human Computer Interaction) and is “useful for gathering, analysing, and synthesizing the information related to the users interacting with a software development team” (Castro, Acuna, Juristo 2008, p. 1). Personas represent detailed descriptions of end users, highlighting their characteristics and goals. They can be developed through direct engagement of end users or through the involvement of members of the software development team “playing the role” of end users. The personas approach helps focus software analysis, design, and development on the features and goals of products’ end users.

2.1.1 Requirements analysis

The first step of the analysis will be the definition of personas. Personas will reflect the target users with defined characteristics (e.g., known tools and skills possessed) and their interactions with the target system. The Personas analysis will enable the team members to clearly define needs and expectations toward the system.

To ensure transparency and retrospection, all personas will be described in the requirement analysis document (see AI-SPRINT deliverable *D1.2 Requirements Analysis*) and updated periodically maintaining the Requirements Analysis Repository.

2.1.2 Epics and use cases

The next step after personas definition is the epics and user stories creation. Epics are the bigger containers which group stories into bigger functional chunks. One user story should describe one persona’s action/interaction within the system. Having user stories once defined, functional test cases can be easily defined, and the system may be tested against it.

Additionally, we will use an UML notation to model requirements generically. A good example of its usage is presented in AI-SPRINT deliverable D1.3 Initial Architecture Design.

2.1.3 Architecture

Another important aspect of the design phase is the definition of the high-level architecture. It will be based on the personas, epic and stories defined earlier. An early architecture design will allow to define main constraints for the system and its biggest challenges. Prior to selecting target tools, the state of analysis will be conducted to ensure selection of solutions that will fit best the project's needs.

As a result, the high-level architecture diagram will emerge. The diagram will summarize the decisions taken as it will present the selected tools and main communication channels.

Additionally, sequence diagrams will be created for most crucial flows in the system. Architecture definition has been already performed at M6 (see, AI-SPRINT deliverable *D*1.3 Initial Architecture Design).

2.2 Quality procedures in Development phase

The development will be conducted in an iterative approach to maximize the feedback received from use case panthers and minimize the risk of issues arising at the end of the project. There are several important aspects to consider while implementing the solution. As the different teams will have to use different tool sets for implementation, the quality procedures will be defined in a language independent manner.

2.3 Coding standards

Each team should have a predefined formalized coding standard. Usually this is a file that may be loaded into an IDE (Integrated Development Environment) and will ensure that the code will have a uniform appearance for each team member. The typical coding standard should cover the following:

- spaces vs. tabs usage for indentation
- function naming convention (e.g. camelcase vs underscores)
- variables naming convention
- code file organization (order of fields, methods, etc)
- source code line length
- comments standardization

2.4 Peer review

To ensure high code quality, the code should undergo a peer review process. The peer review should be conducted by a fellow developer if feasible. The goal of a peer review is to spot issues, to ensure using best practices and adhering to coding principles, usage of design patterns, etc.

The code review should be conducted for every single increment of code before entering a main branch in the source code repository.

2.5 Unit testing

Unit Testing tests an application's smallest building blocks. Unit tests should be conducted in isolation - other units/components behaviour should not impact the currently tested unit. A building block is defined as a single class or method of program. Unit Testing is typically implemented by the developers, and involves the testing of individual methods or functions. Tests perform checks that the code behaves as expected, and identify bugs prior to the integration of the code into the rest of the system.

Identifying bugs in the earliest stages of development is recognized as being the most cost-effective method by the software industry. Apart from reducing costs, it also allows us to fix bugs faster. Both: cost and time are essential aspects of a successful implementation.

Every language and framework provides developers with libraries that support unit testing. For example, for Pytnon pytest [1]and unittest[2] are two most popular libraries.

2.6 Integration testing

Integration testing is a higher level of technical testing. It checks if system modules or APIs are working correctly together. Integration tests should be implemented by developers or test automation engineers as this is still a technical part of the quality assurance process.

In order to ensure smooth and seamless integration testing, some system parts may be mocked. Mocking libraries will be decided by development teams depending on the technologies selected. For example the popular mocking libraries for Python are: mock[3] and unittest.mock.[4]

2.7 Quality procedures in Testing phase

As the project will be organized in an iterative approach there will be no single testing phase. The testing phases will overlap with each other during the development iterations. The goal of such an organization is to ensure high quality through the whole project and to limit the risks at the end of a project.

2.8 Functional testing

Functional testing[5] validates the features and operational behaviour of software to ensure that they correspond to their specifications. The main goal of functional testing is to ensure that certain system functions work properly from the end user perspective. The end user may be a persona identified during the analysis phase or another system. Test results from the test cycles will be reported back to the development team, allowing them to correct any issues found for a next functional test cycle. The scope of functional testing will be decided individually for each system module, process, feature or flow. It will be described in particular test cases.

[1] https://docs.pytest.org/
[2] https://github.com/Codewars/python-unittest
[3] https://pypi.org/project/mock/
[4] https://docs.python.org/3/library/unittest.mock.html
[5] https://glossary.istqb.org/en/term/functional-testing-3

2.9 Smoke testing

Smoke testing[6] is a non-exhaustive software testing method. Its goal is to ascertain that the most crucial functions of a program work and that the system is stable enough to be deployed on environments other than the development one (usually next in line testing/staging/production environment). In smoke testing, only the most crucial chosen positive execution paths/processes of the system are tested.

In an ideal world smoke tests should be executed after every release. The minimum is to test after every major release.

2.10 Tools

It is important to define a unified set of tools for activities that will be cross cutting along multiple teams. Below are summarized the tools that should be unified across all participants.

2.10.1 Source code management

GitLab[7] has been selected as the source code repository provider. This will allow all participants to use git[8] as a source code repository. Git is a modern and well optimized source code manager optimization branching strategies and ensuring seamless cooperation between multiple developers.

Proper naming convention for git commits should allow the team to easily understand the intention of code changes and track it back to the author.

Additionally, some best practices should be implemented, like:

- squashing before merge
- rebasing before merge
- using a standardized branching strategy like gitflow[9]

2.10.2 Continuous integration

Another important aspect of high quality software delivery are regular and automated builds along with automated tests executed against every build. This approach is called continuous integration as code is integrated and tested with every commit.

There are multiple tools that support such an approach but as GitLab has been selected as a SCM tool team can use built in pipelines functionality. What is more, pipelines also support a continuous delivery approach.

[6] https://glossary.istqb.org/en/term/smoke-test-1
[7] https://about.gitlab.com/
[8] https://git-scm.com/
[9] https://www.atlassian.com/git/tutorials/comparing-workflows/gitflow-workflow

2.10.3 Bugs raising and bug tracking

Bugs are an inevitable part of the software development process. It is important to properly raise a bug in order to ensure easy bug management and fixing. Every bug report should contain:

- bug identifier (can be generated automatically)
- short and descriptive summary
- system version where bug has been spotted
- reproduction steps
- clear indication of actual result versus expected result
- link to a user story/requirement that describes tested scenario
- additional resources or information (e.g., browser and its version, mobile OS version, etc.)
- bug priority and severity (if applicable)

The selection of the issue tracking tool is less crucial as all modern tools provide similar features. For example, GitLab has built in issue tracking functionality. If the team select to choose another tool there are plenty of options on the market: JIRA,[10] Redmine,[11] trac,[12] and many more.

2.11 Environments

There should be at least two environments set up: a test environment and a (pre)production environment.

Test environment should be used for regular deployments and day to day testing. The main idea of testing environment is to enable rapid testing of new features. It may use artificial data and some domain/3rd party systems may be mocked.

Then the next environment should be a staging environment. The staging environment should mirror production as closely as possible. The primary usage of a staging environment is to test all the installation/configuration/migration procedures. With this approach all major and minor upgrades to a production environment should be completed reliably.

(Pre)Production environments should be used for demos of functionalities and final acceptance tests.

[10] https://www.atlassian.com/software/jira
[11] https://www.redmine.org/
[12] https://trac.edgewall.org/

3. Quality measures in project management activities

Project management and communication management is instrumental to carrying out effective activities and creating the proper setup that allow partners to produce quality outputs. Project management activities should adhere to best practises and industry standards in order to support project executions. This section summarizes the project management structure and organizational principles that will govern the project and the practices for implementing effective communication among project partners and stakeholders.

3.1 Project management organisation structure

AI-SPRINT is a complex project, involving a consortium of 11 partners and foreseeing three pilot cases. A clear and flexible project management structure has been defined during the preparation of the project proposal and is included in the Grant Agreement. The project management structure balances rigorous decision-making processes and speeds up decision making and actions. The project structures comprising the following organisational bodies: the General Assembly (GA), the Project and Scientific Coordinator (PSC), the Scientific Director (SD), the External Advisory Body (EAB), the Work Package Leaders (WPL), and the Task Leaders (TL).

The ultimate decision-making body of the Consortium and works as the supervisory body for project execution. Second most important body is a Coordinator. The project Coordinator (POLIMI) appoints the PSC and the SD. The PSC supervises and manages all scientific aspects of the project, including the coordination of the WPLs and TLs. The PSC is supported in the project execution by an External Advisory Board (EAB) and a by supporting roles such as an Administrative Manager, an Innovation Manager, and Ethics Manager. The SD provides support to the PSC and focuses on guaranteeing a synergetic development of research and business innovation activities and in enabling progress against the technology state-of-the-art. The EAB advises on strategic issues and directions for the project and will be invited to intervene in case of technological issues providing suggestions. Work Package Leaders and Task Leaders are responsible for the coordination and monitoring of WPs and Tasks. Additionally, leaders guarantee that the required deliverables are released with the necessary quality. A detailed description of the roles and duties of the different project management bodies is included in the Grant Agreement, Annex I.

The names of the members of the different organisational bodies have been proposed in the project proposal and have been confirmed during the project Kick-off meeting. The names are provided in the AI-SPRINT deliverable *D7.1 Project Management Plan* (released at M2 of the project).

3.2 Project management procedures

Project management procedures have been defined during the project proposal preparation, are contained in Grant Agreement Annex I, and they have been updated in D7.1 Project Management Plan. Procedures comprise: management of meetings, decision taking, and resolution of conflicts. In addition to the provisions detailed in the Grant Agreement and in D7.1, the quality of coordination meetings is pursued at different levels.

Meetings addressing different project issues occur with a frequency that depends on the issues tackled. Strategic issues and decisions are taken through the meetings of the General Assembly, that are foreseen every 6 months, or held in case of need. During the execution of the project GA meetings will be organised jointly with the Consortium meetings. Consortium coordination meetings are organised to align all partners on the overall progress of the project and to share key information, including also non-scientific information (e.g., reporting duties). Consortium meetings are organised every 6 months. During the Kick-off meeting, partners agreed on the organisation of a research operational coordination meeting (RTB, Research Technical Board) to monitor technical and non-technical activities at an operational level. RTB meetings are organised every 2 weeks, at least for the initial phases of the project, as a strict collaboration and alignment among partners is specially required at the beginning of the project. During the Consortium coordination meetings, partners will evaluate the opportunity of having less frequent RTB meetings. During the first 6 months of the project activities, due to the COVID19 pandemic, all meetings have been organised as online events.

Meetings are prepared in advance and meetings results and decisions are formalized and shared. Before scheduling the meeting:

- A minimal set of participants is defined
- A meeting time is agreed or the poll is issued to find best time slot (Doodle will be used to vote for the best date)
- A calendar invitation to the meeting is sent

The online nature of the meetings has not spoiled the quality and efficacy of the communication. All meetings results and decisions have been formalised through meetings' minutes and, thanks to the online, meetings have been recorded in order to offer the possibility to follow the meetings discussions to the partners who could not participate to the online events.

The types of data collected and the decisions taken during the meetings depend on the aim of the meetings: technical and scientific meetings (RTB) and coordination and progress meetings (Kick-off, Consortium meetings).

RTB meetings address the following issues:

- Discuss stage of development of activities
- Identify impediments and remove them from the project
- Tracking deliverables implementation progress
- Discuss technological choices

Kick-off meeting and consortium coordination meetings address:

- Overall stage of development
- Analyse key organisational issues
- Discuss reporting duties
- Share general information (e.g. reporting duties, administrative issues, tools used to support project activities) necessary to effectively carry out activities

In addition to specific meetings' information, in every meeting participants, minutes, and list of actions are prepared.

3.3 Project management

In order to guarantee a swift execution of project activities, during the Kick-off meeting partners agreed on the adoption of several tools such as a Google document management repository and Microsoft Teams video conference tool.

A Google repository was set-up to share all project documentation (e.g., official documents, meeting minutes, presentations, project deliverables). The repository is also instrumental to guarantee swift preparation and development of project deliverables as documents are mainly prepared in the form of online documents, that are revised in a collaborative form by the partners. Integration of the online repository with the email also guarantees that all contributors or reviewers are promptly informed in case of a required action on the documents.

Meetings are conducted via Microsoft Teams. This facilitated (and will facilitate) recording of participants and recording and sharing of the discussion in order to ensure transparency and information sharing.

3.4 Communication procedures

Proper communication and information sharing is key for a successful project and underpins all project activities. Communication should be complete, consistent, focussed on objectives, and targeted to the audience. There should be clearly defined communication channels and roles. Communication in the AI-SPRINT project encompasses scientific and technical information, administrative information, and official documentation. Scientific and technical communication aims at providing all actors involved in scientific activities with the necessary information to properly carry out tasks and produce deliverables. Administrative information is focused on financial data and reporting duties and mainly involves administrative actors in the project. Official documentation needs to be shared among all partners in order to have a clear and accepted framework for carrying out research.

In the AI-SPRINT project each type of information shared is supported by specific tools. Below is a summary of the tools used in a project.

3.4.1 Email communication

All formal and relevant communication is shared via email. This tackles meeting notes, invitations for workshops, and announcements. All project participants have shared their emails in order to ensure easy communication. Mailing lists have been created for the different types of participants (e.g., all participants, participants to WPs, administrative contacts). A repository for the proper email groups was created to avoid errors while communicating.

3.4.2 Documents repository

In order to both support document preparation and offer a unique space for all relevant documents, a repository has been created through Google drive. The repository is organised into sections dedicated to:

- Official documents of the project (e.g., Grant Agreement, Consortium Agreement, Horizon2020 call, etc.)

- Project documents (e.g., deliverables, meetings' minutes, presentations, working documents, etc.)
- Internal communication material (e.g., templates for deliverables and presentations, pictures of events or meetings)
- Monitoring tools (e.g., online archive of all communication and dissemination events).

Google Drive also offers the possibility of co-editing documents. This approach proves to be effective in:

- Presentation preparation. During the preparation of meetings, all partners collaborate on the creation of the materials to be illustrated. A quality procedure adopted foresees that the organiser of a meeting shares the agenda of the meeting and a presentation deck in advance before the meeting and invites all key contributors to work on the presentation. This approach serves both the purpose of facilitating collaboration (avoiding for instance lack of time due to "partners working on different versions of a document") and of guaranteeing consistency of the presentations, as all presentations are prepared based on a standard template.
- Preparation of deliverables. Deliverables are created in a collaborative form. The partner responsible for the deliverable prepares a document that is shared on the Google repository and invites all contributors to provide their inputs. The same approach is followed during the revision process. As outlined in the following section a quality process has been implemented to guarantee high quality of the documents produced.

3.4.3 Instant messaging

Efficient day to day communication in a dispersed team during the pandemic is a challenge. In order to facilitate that an instant messaging solution should be selected for the project. It should provide a topic related chats/groups, peer to peer communication, and voice messaging. In the AI-SPRINT project, Slack has been selected as the instant messaging solution. It fulfils all the mentioned above requirements.

4. Quality procedures in deliverable production and reporting to the European Commission

It is essential for the AI-SPRINT project to ensure that all required deliverables are of high quality. Quality encompasses several aspects such as adherence of documents to the required objectives (i.e., deliverables' contents much be compliant to deliverables' descriptions contained in the Grant Agreement); homogeneity (i.e., all documents must follow a similar and standard structure, in order to facilitate the reading process and provide a consistent communication to the funding authority); harmonisation (i.e., deliverables should be complementary and offer consistent information); timeliness (i.e., deliverables should be prepared on time for quality revisions and for submission to the funding authority via the Participant portal). In order to match all these quality objectives, the quality procedures defined in the project are described in the following paragraphs.

4.1 Quality procedure for documents

It is of paramount importance that all documents prepared during the project are consistent and of the highest quality. These requirements affect both the project deliverables and the presentations prepared for project meetings. Two quality aspects are addressed by procedures: content and consistent styling. The following procedures have been adopted in the preparation and release of the documents: a peer review process, the definition of standard templates, a documents' naming strategy.

4.1.1 Deliverables' peer review process

All project documents are prepared using the online repository providing collaborative editing. A rigorous review process was defined based on the following elements:

- Each deliverable leader is responsible for the development of the document, in collaboration with the required partners.
- Two peer reviewers are appointed for each deliverable. Their role is to provide comments and suggestions for the improvement of the deliverable. The appropriate experts coming from the consortium are chosen as peer reviewers. The appointment of the peer reviewers is defined during the RTB meetings based on the suggestions of WP leaders.
- The deliverable revision process foresees that an initial draft of the deliverable containing at least the Table of Contents of the document is to be sent by the partners to the reviewers at least 50 days before the deliverable submission deadline. 10 days before the deliverables' deadlines a complete version of the documents is shared with reviewers for their revisions. Reviewers and deliverables' leaders work collaboratively on the finalisation of the documents. WP leaders are responsible for the final approval of deliverables to be submitted to the funding authority.

Partners agreed on a standard approach for the review process. Reviews of deliverables address the following aspects:

- Document objective and purpose are clearly defined.
- Document leader and supporting authors are defined.
- Document reviewers are identified.
- Document is being prepared in a timely manner and within the agreed deadline.
- Document is sent for reviews and the comments are raised.
- Document comments are cleared or discussed.
- Document's final formatting is applied and the last read is executed.
- Document is ready for publication.

4.1.2 Templates

Project documents comply to quality standards in terms of homogeneity of structures and visual identity. In order to achieve these two templates were defined:

- Document template containing key sections of the document (e.g., initial table to track the review process, executive summary, etc.) as well as graphical aspects such as fonts, headings styles, text styles, headers, footnotes.
- Presentation template containing visual presentation elements and style.

4.1.3 Document naming strategy

Clear naming conventions support better readability and help reduce waste of time due, for example, to ambiguity of messages, need to such for information, need to search for the right documents to work on. Below there are naming conventions for most commonly used document types in project:

- Deliverables

  AI-SPRINT [WP] [Deliverable] [Document name]_[Version]_[Status]

  For example: AI-SPRINT WP7 D7.3 Data Management Plan_v0.1_DRAFT
- Meeting notes and slides

  AI-SPRINT [Date] [Document name]

  for example: AI-SPRINT May-25-2021-RTB-Minutes

4.2 Quality procedure for project meetings

All project meetings are supported by shared documents, in the form of presentations or texts. The same quality principles adopted for project deliverables are applied also to meetings' documents. In particular, documents will be prepared and shared through the online repository and a collaborative editing approach is adopted. The organiser of a meeting is responsible for preparing the initial draft of the document to support the meeting discussion and shares it with participants in advance before the meeting in order to collect feedback, in case some relevant issue is missing, and to give to the required contributors sufficient time to provide input.

Special attention is dedicated to the preparation of Review meetings. The funding authority, the European Commission (EC), conducts reviews on the proper implementation of the project. Review meetings are defined in the Grant Agreement. In the AI-SPRINT two reviews have been identified at the end of the two reporting periods, at M18 and at M36. These reviews tackle both scientific and administrative aspects of the project implementation. In addition to that, it is common practice to have an additional technical review early in the project implementation, usually within the first year of activity, to monitor the progress of scientific aspects. Review meetings involve project reviewers, who, after reading the project deliverables, have the opportunity of assessing the activities carried out, interact with project partners, and ask for clarifications or offer suggestions and recommendations. Project reviewers are appointed by the EC. Due to the relevance of the review meetings, special provisions are required to offer to the funding authority and to the project reviewers the most clear and effective view of the project.

4.2.1 Preparation

The preparation for review meetings will comprise the following steps:

- After the EC notice of a review meeting is received, the Project Scientific Coordinator and the EC project officer define a meeting detailed agenda.
- The PSC asks all partners to prepare their contributions, either in the form of presentations or of project outputs (e.g., software demo, video, measurements, etc.). A document template will be used and several steps of preparation of contributions and revisions are foreseen.
- Presentations must be finalized and shared in the document repository for review of their consistency and the quality not later than a week before the meeting.
- At least one rehearsal meeting is organised before the project review meeting. In case the review meeting foresees also demonstrations of the software developed during the project, different meetings to test the software are organised.
- The EC project officer and the project reviewers receive all documents no later than two weeks prior to the meeting, while the review meeting material, presentations or project outputs, are shared during the meeting. In case some deliverables are “live documents”, i.e. they are released according to the project schedule, but are continuously updated during the project activities (e.g., deliverable D7.3 on DMP is released at M6, but is continuously updated during the project as the data collected and elaborated during the project may change), the latest and updated versions of the deliverables might be shared with the EC and with the reviewers through an online repository. The sharing of any material with project reviewers will necessarily be agreed with and approved by the EC project officer.

4.2.2 Agenda of Project Review Meeting

Project review meeting agenda may include presentations, live demons, and videos that report achievements. As a general indication, the agenda should include at least following contents:

- Welcome/Opening
- Introduction
    - Presentation of the beneficiaries
    - Presentation of project objectives
    - Presentation of project organisation
- Management summary

    - Summary of results achieved vs planned objectives
    - Financial reporting
- Technical summary
    - Major results achieved since last Review (and adaptations)
    - Technical presentation of major results (presentations, documents, demos)
- Dissemination and exploitation efforts
- Conclusions (and plans for the next period if applicable)

The agenda for the project review meeting will be developed jointly with and approved by the EC project officer early in advance in order to have sufficient time for preparing the review material.

References

Aoyama, M. (2007, October). Persona-scenario-goal methodology for user-centered requirements engineering. In 15th IEEE International Requirements Engineering Conference (RE 2007) (pp. 185-194). IEEE.

Castro, J. W., Acuña, S. T., & Juristo, N. (2008, October). Integrating the personas technique into the requirements analysis activity. In 2008 Mexican International Conference on Computer Science (pp. 104-112). IEEE.

Castro, J. W., Acuña, S. T., and Juristo, N. (2008). Enriching requirements analysis with the personas technique.

Cooper, A., Reimann, R., and Cronin, D. (2007). About face 3: the essentials of interaction design. John Wiley & Sons.

Schneidewind, L., Hörold, S., Mayas, C., Krömker, H., Falke, S., and Pucklitsch, T. (2012). How personas support requirements engineering. In 2012 First International Workshop on Usability and Accessibility Focused Requirements Engineering (UsARE) (pp. 1-5). IEEE.