



EUROPEAN COMMISSION
Research Executive Agency (REA)
Inclusive, Innovative and Reflective Societies



Project acronym: SIMPATICO

Project full title: SIMplifying the interaction with Public Administration Through Information technology for Citizens and cOmpanies

Call identifier: EURO-6-2015

Type of action: RIA

Start date: 1 March 2016

End date: 28 February 2019

Grant agreement no: 692819

D5.5 – SIMPATICO Platform Validation Report v1

Executive Summary: The aim of this document is to present the evaluation of SIMPATICO platform, starting from the description of the adopted methodology; it includes the results of the execution of a range of integration use-cases across distinct platform components, the assessment of software quality and the analysis of the feedback collected during pilots on the deployment and configure the platform.

WP: WP5 – Integration and Environment Setup

Author(s): Antonio Filograna (ENG), Koldo Zabaleta (DEUSTO), Carolina Scarton (SCC), Raúl Santos de la Cámara (HIB), Kevin Moss (SPA)

Editor: Antonio Filograna (ENG)

Leading Partner: Engineering (ENG)

Participating Partners: FBK, DEUSTO, USFD, HIB, SPA

Version: 1.0

Status: Final

Deliverable Type: R: Report

Dissemination Level: PU: Public

Official Submission Date: 31/10/2017

Actual Submission Date: 31/10/2017

Version History

Version	Date	Author	Partner	Description
0.1	10/07/2017	A. Filograna	ENG	ToC proposal
0.2	21/07/2017	A. Filograna	ENG	Integration Use-Cases - first iteration
0.3	24/07/2017	A.Filograna, R. Santos de la Cámara	ENG HIB	Chapter 2 - first iteration
0.4	28/07/2017	R. Kazhamiakin	FBK	Quality assessment - first iteration Component Issue Tracking - first iteration
0.5	16/10/2017	A. Filograna	ENG	Chapter 5 - first iteration
0.6	17/10/2017	A. Filograna	ENG	Chapter 3 & 5 - second iteration
0.7	17/10/2017	A. Filograna, R. Santos de la Cámara	ENG HIB	Chapter 2 - second iteration
0.8	18/10/2017	R. Kazhamiakin	FBK	Quality assessment - second iteration
0.9	18/10/2017	M. Trainotti	FBK	Feedback on SIMPATICO Platform (Trento)
0.10	19/10/2017	K. Zabaleta	DEUSTO	Executive Summary
0.11	25/10/2017	R. Santos de la Cámara, K. Moss	HIB SPARTA	Feedback on SIMPATICO Platform (Galicia & Sheffield)
0.12	25/10/2017	A. Filograna	ENG	Conclusion
0.13	28/10/2017	M. Trainotti K. Moss	FBK SPARTA	Internal Review
0.14	29/10/2017	A. Filograna	ENG	Final version
1.0	31/10/2017	M. Pistore	FBK	Final quality check

Statement of originality

This deliverable contains original unpublished work except where clearly indicated otherwise. Acknowledgement of previously published material and of the work of others has been made through appropriate citation, quotation or both.

Table of contents

1	Introduction	8
1.1	SIMPATICO project	8
1.2	Platform recap	10
1.3	Structure of the deliverable	11
2	SIMPATICO platform evaluation methodology	12
2.1	Methodology used to collect data	12
2.2	Evaluation criteria definition	13
2.2.1	Integration use-cases	13
2.2.2	Quality assessment	14
2.2.3	Issue tracking.....	14
2.2.4	Feedback from users.....	14
2.2.5	SIMPATICO Platform evaluation KPIs.....	15
3	Integration Use-cases.....	16
3.1	Integration Use-Cases description	17
3.2	Integration Use-Case evaluation	25
4	Quality assessment	27
4.1	Quality assessment of the SIMPATICO components platform.....	27
4.1.1	Integration and interoperability.....	27
4.1.2	Documentation.....	28
4.1.3	Security.....	29
4.1.4	Usability.....	29
4.1.5	Source code management.....	30
4.1.6	Testing.....	31
4.1.7	Deployment.....	31
4.2	Component issue tracking.....	31
5	Feedback on platform installation and deployment	35
5.1	Trento use-case	35
5.2	Galicia use-case	37
5.3	Sheffield use-case.....	40
6	Conclusion	43
7	References.....	45

List of figures

Figure 1: SIMPATICO concept as a glance	8
Figure 2: SIMPATICO Architecture used in the first phase of experimentation	10
Figure 3: GitHub issue management.....	32
Figure 4: Creating new issue in GitHub.....	33

List of tables

Table 1: SIMPATICO platform evaluation KPIs	15
Table 2: Integration Use-Cases.....	17
Table 3: Integration Use-Case evaluation	25
Table 4 : List of KPIs associated to the integration and interoperability properties	28
Table 5: List of KPIs associated with documentation aspects.....	28
Table 6: List of KPIs associated with security aspects	29
Table 7: List of KPIs associated with usability	30
Table 8: List of KPIs associated with 4.1.5 source code management.....	30
Table 9: List of KPIs associated with testing.....	31
Table 10: List of KPIs associated with deployment.....	31
Table 11: List of KPIs associated with issue tracking.....	33
Table 12: Issue tracking information for the platform components	33
Table 13: Trento Use-Case questionnaire	35
Table 14: Galicia Use-Case questionnaire	38
Table 15: Sheffield Use-Case questionnaire.....	40
Table 16: Measured KPIs	43
Table 17: Quality Assessment KPIs.....	44

Glossary

Acronym	Definition
AAC	Authentication and Authorization Control
API	Application Programming Interface
AST	Authoring Support Tool
CDV	Citizen Data Vault
CKB	Collective Knowledge Base
CORS	Cross-Origin Resource Sharing
COTS	Commercial Off-The-Shelf
CPD	Collaborative Procedure Design
CTZP	Citizenpedia
DA	Data Analysis
DB	Dashboard
EE	Enrichment Engine
EFF	Effort
ES	Spain
eSM	e-Service Monitor
EU	Europe
GE	Gamification Engine
HTML	HyperText Markup Language
IFE	Interactive Front-End
IT	Italy
IUC	Integration Use Cases
JSON	JavaScript Object Notation

KPI	Key Performance Indicator
PA	Public Administration
PH	Person Hour
QAD	Quality Assessment of Documentation
QAE	Question Answering Engine
QAI	Quality Assessment of Integration
QAR	Quality Assessment of Deployment
QAS	Quality Assessment of Security
QASCM	Quality Assessment of Source Code Management
QAT	Quality Assessment of Testing
QAU	Quality Assessment of Usability
REST	Representational State Transfer
RO	Research Objective
RPC	Remote Procedure Call
SCM	Source Code Management
SF	Session Feedback
SSO	Single Sign-On
STEP	Standard Technical Evaluation Process
TAE	Text Adaptation Engine
UI	User Interface
UK	United Kingdom
UP	User file
WAE	Workflow Adaptation Engine

Executive summary

This document corresponds to deliverable “D5.5 – SIMPATICO platform validation report v1” of the European H2020 project “SIMPATICO – SIMplifying the interaction with Public Administration Through Information technology for Citizens and cOmpanies” (hereinafter also referred to as “SIMPATICO”, project reference 692819).

The aim of this document is to present the evaluation of SIMPATICO platform, starting from the description of the adopted **methodology**, and considering different concerns that are necessary for a thorough assessment of the platform. A definition and evaluation of **uses-cases** regarding the **integration** among the components of the SIMPATICO platform is necessary to validate the platform. It is also necessary to evaluate the quality of all the components that are part of the SIMPATICO platform according to different criteria; in particular, the **quality** of the **documentation**, **security** issues, **usability** issues and **source code** should be evaluated for each component. This deliverable presents the definition of the methodology considering these issues, the results obtained and conclusions drawn after the evaluation. Finally, the **KPIs** proposed to evaluate the SIMPATICO technical platform are presented, together with their obtained values once a workable version of the platform has been made available for phase I of evaluation.

1 Introduction

This deliverable presents the outcomes of SIMPATICO project task T5.4 “*Technical validation of the SIMPATICO platform*” in the scope of WP5 “*Integration and environment setup*”.

During the first experimentation phase of SIMPATICO within the three use-cases, the SIMPATICO platform has been evaluated not only in technical sense but also the practical ability of the platform to support the general requirements coming from T5.1. This evaluation is presented in this document.

To better understand the aim and scope of this document, in this introductory section we provide an overview of the SIMPATICO project (Section 1.1), a brief recap of each SIMPATICO component (Section 1.2) and a description of the structure of the rest of this deliverable (Section 1.3).

1.1 SIMPATICO project

SIMPATICO's goal is **to improve the experience of citizens and companies in their daily interactions with the public administration** by providing a personalized delivery of **e-services** based on advanced **cognitive system technologies** and by promoting an active engagement of people for the continuous improvement of the interaction with these services. The SIMPATICO approach is realized through a platform that can be deployed on top of an existing PA system and allows for **a personalized service delivery** without having to change or replace its internal systems: a process often too expensive for a public administration, especially considering the cuts in resources imposed by the current economic situation.

The goal of SIMPATICO is accomplished through a solution based on the **interplay of language processing, machine learning and the wisdom of the crowd** (represented by citizens, business organizations and civil servants) **to change for the better the way citizens interact with the PA. SIMPATICO adapts the interaction process** to the characteristics of each user; **simplifies** text and documents to make them understandable; **enables feedback for the users** on problems and difficulties in the interaction; **engages civil servants, citizens and professionals** so as to make use of their knowledge and integrate it in the system (Figure 1).

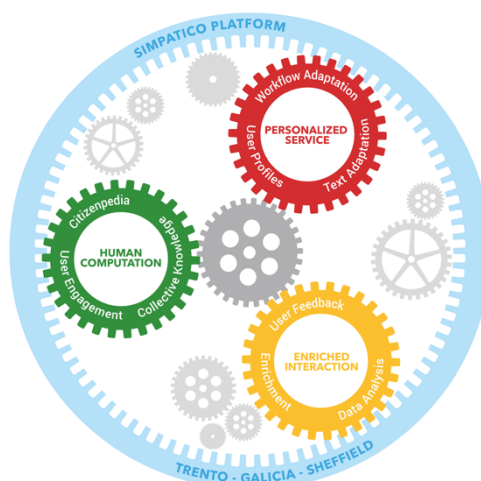


Figure 1: SIMPATICO concept as a glance

The project aims can be broken down into the following **smaller research objectives (ROs)**.

RO1. Adapt the interaction process with respect to the profile of each citizen and company (PA service consumer), in order to make it clear, understandable and easy to follow.

- A **text adaptation** framework, based on a **rich text information layer** and on machine learning algorithms capable of **inducing general text adaptation operations from few examples, and of customizing these adaptations to the user profiles**.
- A **workflow adaptation engine** that takes user characteristics and tailor the interaction according to the user's profile and needs.
- A feedback and annotation mechanism that **gives users the possibility to visualize, rate, comment, annotate, document the interaction process** (e.g., underlying the most difficult steps), so as to provide valuable feedback to the PA, further refine the adaptation process and enrich the interaction.

RO2. Exploit the wisdom of the crowd to enhance the entire e-service interaction process.

- An **advanced web-based social question answering engine (Citizenpedia)** where citizens, companies and civil servants **discuss and suggest potential solutions and interpretation for the most problematic procedures and concepts**.
- A **collective knowledge** database on e-services used to simplify these services and improve their understanding.
- An **award mechanism** that **engages users and incentivizes them to collaborate** by giving them **reputation** (a valuable asset for professionals and organizations) and **privileges** (for the government of Citizenpedia – a new public domain resource) according to their contributions.

RO3. Deliver the SIMPATICO Platform, an open software system that can interoperate with PA legacy systems.

- A platform that **combines consolidated e-government methodologies with innovative cognitive technologies** (language processing, machine learning) at different level of maturity, enabling their experimentation in more or less controlled operational settings.
- An interoperability platform that enables an **agile integration of SIMPATICO's solution with PA legacy systems** and that allows the exploitation of data and services from these systems with the SIMPATICO adaptation and personalization engines.

RO4. Evaluate and assess the impact of the SIMPATICO solution

- Customise, deploy, operate and evaluate the SIMPATICO solution on **three use-cases in two EU cities** – Trento (IT) and Sheffield (UK) – **and one EU region** – Galicia (ES).
- **Assess the impact** of the proposed solution in terms of **increase in competitiveness, efficiency of interaction and quality of experience**.

This deliverable focuses in particular on the RO4, covering the evaluation and assessment of the SIMPATICO effectiveness and impact by highlighting the pilots' point of view. Each pilot city presented its experience in using the tools available within the SIMPATICO environment.

In particular, each use-case presented strengths and weaknesses of each tool. Furthermore, pilots also proposed some enhancements for the future releases of the platform.

Before proceeding with the report, a recap about the tools offered by the SIMPATICO platform is presented. The latter is then followed by an introduction about the document's sections.

1.2 Platform recap

This section aims at recapping the definition of the SIMPATICO architecture done in the D5.1 [1] and how it works. It is important to point out that this architecture is that one used in the first phase of experimentation.

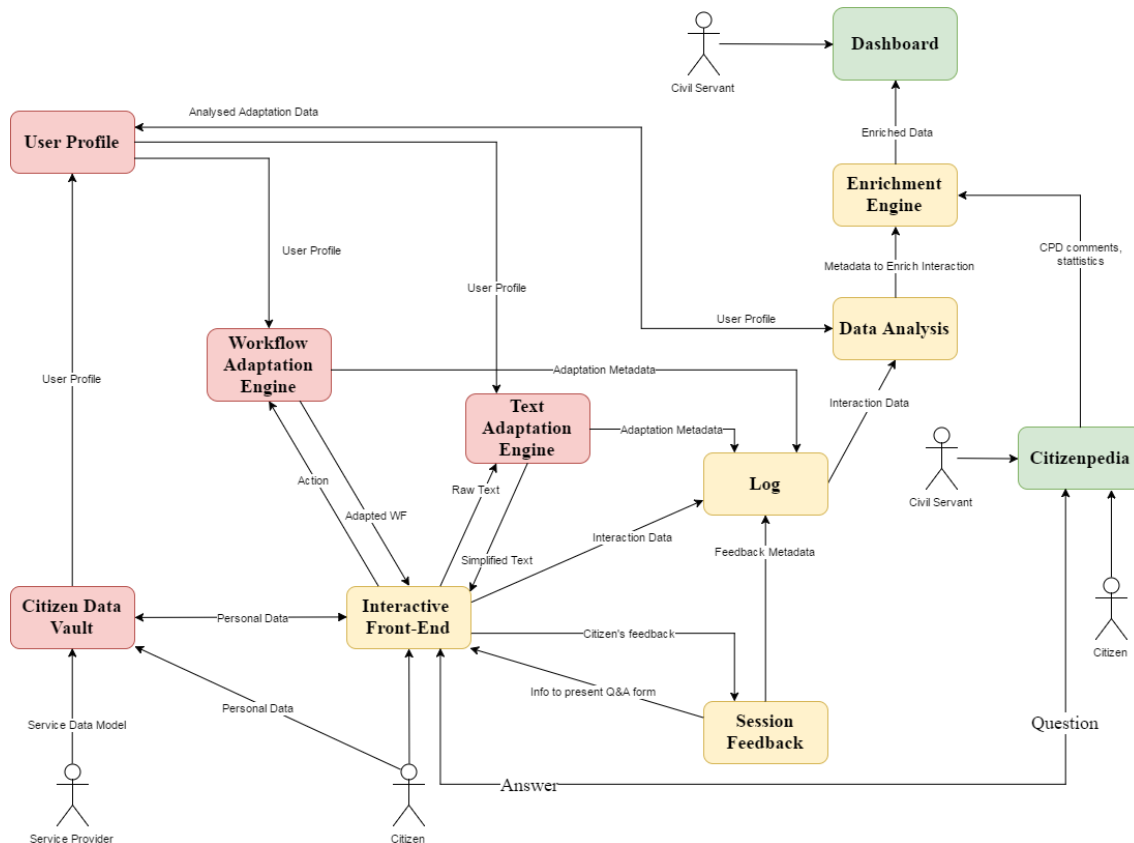


Figure 2: SIMPATICO Architecture used in the first phase of experimentation

The Figure 2 shows the SIMPATICO Architecture adopted during the first phase of experimentation (except for the User Profile component that will be put in place during the second experimentation phase). The Citizen interacting with the SIMPATICO platform can use the e-services, provided by the Public Administration, improved through the SIMPATICO techniques (e.g. Text and Workflow Adaptation Engine). Moreover, the citizen can manage his/her personal data using the Citizen Data Vault, allowing also to pre-fill in the web-form of the e-services.

All data coming from the interaction of the user with the platform are stored in the Interaction LOG, also those data generated by Session Feedback after the feedback that the citizen can leave once s/he completed the use of the e-service. All those data are analysed by Data Analysis and passed to the Enrichment Engine in charge of suggesting the improvements to be done in order to make ease the interaction of the user with the PA services. In doing so, the civil servants can monitor and analyse those data through the Dashboard.

On the other hand, the Citizenpedia (through Question & Answer Engine and Collaborative Procedure Design) offers a community where citizens, professionals, and civil servants can cooperate

together in order to better understand how a e-service has to be used, or how a PA procedure works. Therefore, all Citizenpedia users can suggest improvements and provide feedback about e-services.

1.3 Structure of the deliverable

The remainder of the deliverable is organized as follows:

Section 2 describes the adopted methodology for evaluate the SIMPATICO platform. The section provides the process describing how the data were collected and successively analysed taking into account the evaluation criteria (SIMPATICO KPIs) there defined.

Section 3 defines the use-cases about the integration of SIMPATICO platform. Each use-case is valuated within the technical platform provided by the three cities involved in the project.

Section 4 describes the quality assessment of the SIMPATICO platform taking into account seven different macro-areas. Moreover, the component issues are collected through the GitHub repository in order to be traced.

Section 5 shows the feedback on the installation, configuration and deployment of the SIMPATICO platform gathered through a survey given to the technical team responsible to install the platform in each city and the interaction data collected during the use of the SIMPATICO components and stored in the Interaction Log component.

In the end, the **Conclusion** section summarizes all the evaluation results achieved.

2 SIMPATICO platform evaluation methodology

To evaluate the SIMPATICO Technological Platform and its related components we worked on the same line of the Standard Technical Evaluation Process (STEP) methodology [2]. The Standard Technical Evaluation Process outlines a rigorous process for technology evaluations of one or more COTS (Commercial Off-The-Shelf) products [3].

The aim of this STEP methodology is:

1. formalizing a process that can be adopted to evaluate a wide range of technological product
2. producing a set of documents where relying on the consistency of evaluation results
3. providing the evaluation team with guidelines in developing goals
4. modelling a process easy to understand and fully adopted.

Four main phases compose the STEP process, namely:

- Phase 1 - Scoping and testing strategy: declare the objectives and scope, and performs a market survey to identify potential products in the technology area.
- Phase 2 - Test preparation: a full set of evaluation criteria will be performed. The products/components will be tested against every criteria and any scenario tests that will be performed. The evaluation team then installs the products/components in the test environment
- Phase 3 - Testing, results and final report: the evaluation team tests and scores the products/components against all of the test criteria. The team must ensure that testing for each product is performed under identical conditions. The team produces a final report that incorporates the evaluation results and any supporting information.
- Phase 4 - Integration and Deployment: deploy and integrate the solution into the operational environment. Actions in this phase may include developing configuration guidance and supporting documentation.

Actually, the STEP methodology is mainly used when a company has to purchase a technological product. Since this is out of the scope of this deliverable, we will focus on the Phase 2 and 3 since they fit better with the aim of this document.

In the following two subsections, we will describe how we adopt the phase 2 and 3 of the STEP process. In particular, we will identify the KPIs to measure the platform, how we want to measure them and the evaluation process to be adopted.

2.1 Methodology used to collect data

The activities to be performed in order to evaluate the SIMPATICO platform are both manual and automatic, namely some tasks will collect data from interaction of the user with the platform, some other ones will demand directly to the users and they will be asked to respond to a survey.

Other activities foresee the evaluation of the scenarios modelled according to the functionalities provided by the platform and other ones are based on the evaluation of the quality achieved by the platform.

In particular, the **Integration Use-Cases** (see Section 3) have been depicted in order to compare the real platform operation with that one foresees. Every platform component has been described

highlighting what the component behaviour is, what the relations with other components are, and what the effects after the interaction have been produced.

Moreover, for each component the **Quality Assessment** (see Section 4.1) has been defined. Seven characteristics of software quality have been identified and the whole platform has been evaluated for each of these characteristics. In order to have a clear vision of the software development and in particular to discover and correct the bugs, an **Issue tracking** (see Section 4.2) has been adopted. In the Platform evaluation we will also measure how many bugs have been taken into account and solved, in order to understand the response of the developers of each component. Of course, the bugs can involve one or more components and the effort to solve them could be bigger.

In the end, the Platform will be evaluated through the **Feedback** (see Section 5) coming from the users during the installation, configuration and deployment of the platform within the three use-cases. On the one hand, the feedback has been collected through a survey given to the technical team responsible to install the platform. On the other hand, the data coming from the interaction of the user with the SIMPATICO functionalities has been analysed and reported. These data have been collected when the users use the Interactive Front-End component, that is in charge of sending to the Interaction Log component the interaction data of the user. The data have been collected without the direct intervention of the user and without providing explicit notice to the user that how much and which data is being collected. Data collected at this stage is typical web analytics data such as timing spent interacting with each component and each phase of the interactive process as well as actions initiated by the user such as clicks, scrolls and other interaction events.

2.2 Evaluation criteria definition

The platform evaluation has been conducted from both quantitative and qualitative point of view. The four aforementioned macro-categories have been considered taking into account different evaluation criteria. In fact, a set of KPIs have been defined for each macro-category and the reference value has been selected. The baseline has been only defined for the feedback coming from the interaction of users with SIMPATICO platform, since for all other categories defining a baseline does not make sense.

Below, the description of the KPIs for each category and their reference value are described. We have distinguished the KPI value for the first and second phase of evaluation.

2.2.1 Integration use-cases

Before starting with the pre-evaluation phase the Integration Use-Cases were defined in order to measure if the SIMPATICO Platform meet the functionalities and requirements described in the D5.1 - "SIMPATICO Platform Requirements and Architecture v1" [1].

Several use-cases were defined and for each of them a description of synopsis, the components involved, the dependencies from other components, the prerequisites, the behaviour and the effects were describe for each use-case (see Section 3 for further information).

The associated KPI (KPI_IUC_01) is defined as the percentage of the Integration Use-Cases achieved and operating. The value to be achieved for the first phase of experimentation is 75%, and 100% for the second phase of experimentation.

2.2.2 Quality assessment

In order to ensure the consistency, integrity, and usability of the platform and its components, and to facilitate their evolution and deployment, the components have been analysed across a set of quality dimensions. More specifically, each component has been characterized with respect to

- Interoperability
- Documentation
- Security
- Usability
- Source code management
- Testing
- Deployment

For each of these dimensions, a set of requirements and metrics have been associated. For example, the usability requirements amount to internationalization support, browser support range, and mobile platform support, while for the documentation it is expected to deal with the API definition and installation/configuration instructions. For each of these metrics the corresponding KPIs have been identified and applied for the component evaluation (see Section 4 for further details).

2.2.3 Issue tracking

The components of the platform are being developed and integrated by different partners of the consortium. To facilitate the identification of the problems and bugs, as well as the process of team communication and problem resolution, the project applies the unique source code management platform, namely GitHub¹, and its issue tracking management component. With these component in hand, the platform component developers have a possibility to report, assign, and discuss the different issues, being bugs, questions, enhancement requests, etc.

As a part of the platform component evaluation, the number of opened and closed issues are used as one of the evaluation metrics (see Section 4.2 for further details).

2.2.4 Feedback from users

The feedback on the SIMPATICO platform has been obtained in two ways. We gave to every technical team, responsible to install and deploy the SIMPATICO platform in each Use-Case, a questionnaire asking their feeling on the installing, configuring and deploying the platform.

We also had informal discussions with the technical staff in charge of designing and deploying the “legacy” e-services (e.g., the ones used by the PAs that we took as a starting point for our developments) and had discussions on the deployment and other technical aspects. This will be more formalised for version 2 of this document.

On the other side, we analysed the interaction data collected in the Interaction Log component that describe how the SIMPATICO users interact with the e-services provided by SIMPATICO and enriched by SIMPATICO functionalities. . As part of this interaction-driven feedback, one of the components (Session Feedback) asks for explicit feedback to the users about their feeling during the interaction.

¹ www.github.com

2.2.4.1 Feedback based on a survey

After the deployment, installation and configuration of SIMPATICO platform and each related component, every technical team responsible for the Use-case was asked about the ease, velocity and clarity of the activities to be performed to up and running the whole platform.

They replied to a multiple choice questionnaire where the possible choices are five with a scale from 1 to 5 (1. Strongly disagree; 2. Disagree; 3. Neutral; 4. Agree; 5. Strongly agree). The analysis was carried out for both every single question and the total result of the all answers.

In the first phase of experimentation the technical team replied to question about the installation, configuration and deployment of the platform; in the second phase of experimentation, they will answer about the update of the platform and component.

The associated KPI (KPI_FoS_02) is defined as the average value of the answers provided by the interviewees. For the first phase of experimentations its reference value is 3.5, for the second phase it will be 4.

2.2.5 SIMPATICO Platform evaluation KPIs

Below the Table 1 summarises the whole KPIs defined for evaluating the SIMPATICO Platform. In this table we also report the KPIs related to the effort needed to carry out some technical activities to install and maintain the whole platform.

Table 1: SIMPATICO platform evaluation KPIs

KPI name	KPI description	1 st phase value	2 nd phase value
KPI_IUC_01	Percentage of the Integration Use-Cases achieved and operating	75%	100%
KPI_FoS_02	The average value of the answers provided by the interviewees	3,5	4
KPI_EFF_03	Efforts needed to install the SIMPATICO infrastructure	50 PH	40 PH
KPI_EFF_04	Effort needed to connect the SIMPATICO infrastructure with repositories and/or legacy applications	80 PH	60 PH
KPI_EFF_05	Effort needed for daily operational tasks	1 days	0,5 days
KPI_PRO_06	No. of incidents encountered	10	5

Other specific KPIs on Quality Assessment and Issue Tracking will be analysed in Section 4.

3 Integration Use-cases

This section aims at defining the use-cases regarding the integration among the components belonging to the SIMPATICO platform.

The Table 2 describes all the use-cases component by component. In the table the following information are highlighted:

- Component: the component taken into account
- Use case: the short name of the use-case
- Synopsis: brief description of the use-case
- Components: the components involved in the use-case
- Dependencies: the dependencies of the current use-case from the other use-cases
- Pre Requisites: the requirements that the system has to satisfy before running the current use-case in order to be able to work
- Behaviour: the foreseen behaviour that the use-case has to perform
- Effect: what happens after the execution of the use-case

This Table 2 could change after the first experimentation phase since it will be possible to add some use-cases in order to take into account the feedback coming from the first phase. In doing so, some component could change and be improved, and the new use-cases could be needed to test the new integration features.

3.1 Integration Use-Cases description

The following table (Table 2) contains at the moment the use-cases valid for both the first and second phase of experimentation.

Table 2: Integration Use-Cases

COMPONENT	USE CASE	SYNOPSIS	COMPONENTS	DEPENDENCIES	PRE-REQUISITES	BEHAVIOUR	EFFECT
IFE	IFE1	Login	IFE, AAC			Open a service page, press login button. A new window where AAC authentication takes place. Upon window closure, the toolbar is updated and user name is shown	User is authenticated, toolbar unlocked, user name is shown
	IFE2	Session execution	IFE, LOG			User starts/terminates the e-service session	Start/end session events are logged and appear in LOG
	IFE3	E-service form compilation	IFE, LOG			User starts/terminates compilation of the e-service module	Start/end form events are logged and appear in LOG component
	IFE4	Access to an annotated element	IFE, LOG			User access any annotated element (TAE, QAE, CDV)	Click event is logged and appears in LOG component
CDV TAE	CDV1	Link Account to Service	IFE, CDV, AAC	IFE1	e-service is configured in CDV.	Click CDV button in toolbar, CDV popup is shown. Popup asks the user to create a new CDV account and link it	A new account is registered in CDV. Account is linked to the current e-service

						to the service	
	CDV2	Save user data	IFE, CDV	IFE1, CDV1	e-service is configured in CDV. the user has created a CDV account	The user fills in the eservice form and clicks on the "Save Data" button shown in CDV popup	The user data are saved in CDV
	CDV3	Read/populate user data	IFE, CDV	IFE1, CDV1, CDV2	e-service is configured in CDV. the user has already created a CDV account	During the eservice form compiling, the user can use his/her own data already stored in the CDV selecting them by drop down menu in the requested field	The user can fill in the eservice form retrieving the data in the CDV, previously saved
	TAE1	Free text simplification	IFE,TAE,LOG	IFE1		Select a phrase and click on "Free text simplification" toolbar button. A popup appears, tabs show enriched text.	Each tab show content from TAE. Phrase simplification event is registered in LOG component
	TAE2	Free word simplification	IFE,TAE,LOG	IFE1		Select a single word and click on "Free text simplification" toolbar button. A popup appears, tabs show enriched word.	Corresponding tab show content from TAE. Word simplification event is registered in LOG component
	TAE3	Annotated text	IFE,TAE,LOG	IFE1	E-service elements are	The e-service page text elements are annotated. The	Text is decorated with enrichment elements.

		simplification			annotated with a specific CSS class	user clicks the "Text simplification button". The annotated texts are highlighted. Clicking on decoration the simplified versions of text appear.	Paragraph simplification event is registered in LOG component
WAE	WAE1	Workflow simplification	IFE,WAE,LOG	IFE1	E-service workflow model is uploaded to WAE repository, form is annotated with the model URI value	Click "workflow adaptation" button. The simplified process starts (the page scrolls to the first workflow block, the block is highlighted, next/prev buttons are added). The user can execute the workflow	The workflow model is downloaded from the WAE repository, the engine starts the model execution. Workflow adaptation request event is registered in LOG component
QAE	QAE1	Login with AAC	QAE,AAC			User signs in using AAC	The user successfully logged in, the profile data is shown in QAE
	QAE2	See e-service questions	IFE,QAE,LOG	IFE1	e-service elements are annotated with a specific class corresponding to e-service paragraph	Click "QAE" toolbar button, the annotated page elements are highlighted. Clicking on the highlighted element shows the questions associated to the paragraph (if any) and a "add a question" link	The list of existing paragraph questions are associated to each annotated paragraph. Upon selecting a specific element, the 'Citizenpedia content request' event is registered in LOG

							component.
	QAE3	See specific question	IFE,QAE,LOG	IFE1,QAE2		User clicks on a specific question link. A Citizenpedia page with question details is opened in a new window.	Citizenpedia page is opened. The 'Citizenpedia question request' event is registered in the LOG component
	QAE4	Create new question	IFE,QAE,LOG	IFE1,QAE1, QAE2	e-service category is created	Within QAE2 scenario, user clicks the "add a question link". Another browser tab is opened with the precompiled Question form.	QAE is opened with the precompiled form for question creation. Once saved, the question appears in the corresponding paragraph section in IFE. The 'Citizenpedia new question' event is registered in LOG component
	QAE5	Create new answer	QAE,LOG	QAE1		Inside Citizenpedia, the user provides an answer to an existing question.	The number of answers associated to the question is incremented, the answer is saved in the DB. The 'new answer' event is registered in the LOG component

SF	SF1	SF is invoked	SF, IFE		A session is complete	The front-end side of the SF captures the appropriate event of the IFE and asks the back-end of the SF to display a feedback capture form.	The SF feedback form is presented in the front-end for the user to fill in their opinions.
	SF2	SF stores user feedback	SF, LOG	SF1	The user presses the 'Send' button in the SF UI.	The data that the user produces is compiled and stored as the current session's feedback in the LOG.	The LOG is updated with data from the last session.
	SF3	SF generates an appropriate feedback form	SF, LOG	SF1	The SF is invoked	The SF internal logic retrieves relevant data from the last session (e.g., was the text adaptation invoked at any stage) and designs an appropriate feedback form which is sent to the front-end part.	The SF front-end part receives the requested form.
CPD	CPD1	Login with AAC	CPD, AAC			User signs in using AAC	The user successfully logs in, the profile data is shown in CPD. Depending on the user role (citizen/civil servant) access to some functionality can be denied
	CPD2	Open an	CPD		The user has	The user opens a diagram from	The user is presented

		existing procedure diagram			successfully logged in	a list of available (previously created) administrative procedures	with a graphical representation of the selected procedure
	CPD3	View existing questions	CPD, QAE		The user has opened an existing procedure diagram	The user selects to view the questions posted on a procedure's activity	The user gets re-directed to the QAE, where they are presented with all the questions related to that activity
	CPD4	Submit a new question	CPD, QAE		The user has opened an existing procedure diagram	The user selects to submit a new question on a procedure's activity	The user gets re-directed to the QAE, where they are provided with a text editor to write a new question related to that activity
	CPD5	Submit a feedback	CPD		The user has opened an existing procedure diagram	The user selects to submit a new feedback (suggestion, comment) on a procedure's activity	The user is presented with a pop-up editor to post a new feedback related to that activity
	CPD6	Create a new procedure diagram	CPD		The user has successfully logged in and is a civil servant	The user selects to create a new procedure diagram	The user is presented with a canvas and a set of tools to draw the procedure phases and describe them with

							textual content
EE/DB	EE1	DB presents data	DB, EE	EE4	There is a previously generated report by the EE scheduler.		
	EE2	CPD integration	EE, CPD		The CPD API is available.	The EE back-end accesses the Collaborative Procedure Designer API to gather relevant data (e.g., e-service procedure SVG diagram).	The EE back-end obtains the required data.
	EE3	Citizenpedia integration	EE, CTZP		The CTZP API is available.	The EE back-end accesses the Citizenpedia API to gather relevant data (e.g., statistics of usage per paragraph).	The EE back-end obtains the required data.
	EE4	EE generates a report	EE	EE2, EE3	There is available data from past interactions in the LOG.	Upon invoking from the scheduling mechanism (e.g., cron) the EE back-end retrieves data from the available APIs (LOG, CPD and CTZP) and constructs a report to be displayed in the EE tab of the DB.	A EE report is stored internally for display in future invoking of the DB.
DA	DA1	Data Analysis is invoked on schedule	DA		There exists some user interaction data	The scheduling mechanism (e.g., cron) invokes the DA upon its required schedule	Aggregated processed data is stored in the LOG for further

					in the LOG that can be processed.	(e.g., each minute, each hour). The DA executes its internal rules and data available in the LOG which hasn't been processed before is processed.	analysis by DA or representation in the EE
--	--	--	--	--	-----------------------------------	---	--

3.2 Integration Use-Case evaluation

In this section the evaluation of Integration Use-Cases was carried out. The evaluation was performed in the three Use-Cases involved in the project. Both in Trento and Galicia and Sheffield the technical team has executed the use-case described in the Table 2 component by component. In the Table 3 the result of the integration use-case evaluation is reported, divided per component and city.

Three possible results are allowed:

- ✓ The use-case has been executed correctly
- ✗ The execution of use-case failed
- ≡ It was impossible to perform the use-case (e.g. some component was not deployed in this particular city)

Table 3: Integration Use-Case evaluation

COMPONENT	USE CASE	TRENTO	GALICIA	SHEFFIELD
IFE	IFE1	✓	✓	✓
	IFE2	✓	✓	✗
	IFE3	✓	✓	✗
	IFE4	✓	✓	✗
CDV	CDV1	✓	✓	✗
	CDV2	✓	✓	✗
	CDV3	✓	✓	✗
TAE	TAE1	✓	≡	✓
	TAE2	✓	≡	✓
	TAE3	✓	✓	✓
WAE	WAE1	✓	≡	✓
QAE	QAE1	✓	✓	✓
	QAE2	✓	✓	✓
	QAE3	✓	✓	✓
	QAE4	✓	✓	✓
	QAE5	✓	✓	✓
SF	SF1	✓	✓	✗
	SF2	✓	✓	✗

	SF3		✓	✓	✗
CPD	CPD1		✓	✓	✓
	CPD2		✓	✓	✓
	CPD3		✓	≡	✓
	CPD4		✓	≡	✗
	CPD5		✓	≡	✗
	CPD6		✓	≡	✗
DA	DA1		✓	✓	✗
EE/DB	EE1		✓	✓	✗
	EE2		✓	✓	✗
	EE3		✓	✓	✗
	EE4		✓	✓	✗

The evaluation for the first phase of experimentation has pointed out that the distribution of the percentage of the Integration Use-Cases achieved and operating for all the use-cases was as follows:

- ✓ 73%
- ✗ 19%
- ≡ 8%

4 Quality assessment

The purpose of the quality assessment process applied to the SIMPATICO platform and its components is to

- Continuously guarantee the consistency, integrity, and stability of the components and the platform as a whole throughout the development process and the evolution of the platform.
- Facilitate the evolution and extension of the platform components without affecting the depending parts.
- Facilitate the adoption, deployment, and upgrade of the platform by the pilots.

To accomplish these goals, each component is continuously evaluated across a variety of aspects, covering all the relevant functional and non-functional properties of the platform. The deviations in these requirements (being bugs, missing features, etc) are reported through issue tracking system associated to each component and are used to monitor the resolution of those deviations.

4.1 Quality assessment of the SIMPATICO components platform

The quality of the platform components is evaluated across seven dimensions, namely:

- Integration and interoperability
- Documentation
- Security
- Usability
- Source Code Management
- Testing
- Deployment

We remark that the properties associated to these dimensions are in line with the guidelines defined for the components of the FIWARE platform [4]. That is, the SIMPATICO components that respect the metrics associated to the specified dimensions are FIWARE-compatible enablers.

Each of the dimensions defines (a set of) metric(s) that are used as quality assessment KPIs for the component.

4.1.1 Integration and interoperability

When the component is integrated with other platform components, it is important to ensure seamless and consistent integration of those tools. The integration may be performed in different ways including, in particular, API-based (when the component exposes an API to be used by other component) or via hyperlinking, client-side integration or server-side integration, etc.

The important characteristics in these regards refer to way the integration is implemented. In particular,

- for the API-based integration it is required that APIs are exposed as JSON-based REST services;
- for the hyperlink integration, it is required that the components operate in a Single Sign-On (SSO) mode so that the end user should not perform authentication passing from one component of the platform to another.
- The services that expose APIs accessible on the client side should provide adequate support for Cross-Origin Resource Sharing (CORS) in order to access the component functionality across

different domains (e.g., the one of the pilot platform deployment and the one, where the e-service is deployed).

The set of KPIs associated to the integration and interoperability properties are captured with the following table. The values of the metrics are qualitative (i.e., the requirement is satisfied or not) and applicable only to the components that should have these characteristics.

Table 4 : List of KPIs associated to the integration and interoperability properties

KPI name	KPI description	1 st phase value	2 nd phase value
KPI_QAI_01	Exposure of APIs as JSON-based REST/RPC	YES	YES
KPI_QAI_02	Support for SSO	YES	YES
KPI_QAI_03	Suport for CORS	YES	YES

4.1.2 Documentation

To facilitate the deployment and integration of the components within the platform, the documentation plays fundamental role. The aspects covered by the documentation range from the installation instructions to usage scenarios, examples, etc.

A minimal essential set of documentation elements is represented with:

- Component README file providing i) the component installation instructions; ii) the component configuration instructions; and iii) component integration instructions defining the necessary steps to set up the integration with other components.
- For the components that expose REST API, these latter should be well defined and documented. Specifically, it is required that all APIs are provided with the Swagger specification describing the available API calls, parameters, and data types.

Additionally, the documentation should provide the description of the usage scenarios of the component, examples (e.g., API call inputs and outputs, testing instructions, tutorials, howto, etc).

It is also required that the documentation provided by the component is in sync with the source code available for the component.

The metrics captured by the documentation aspect are represented in the following table.

Table 5: List of KPIs associated with documentation aspects

KPI name	KPI description	1 st phase value	2 nd phase value
KPI_QAD_01	Installation, configuration, and integration documentation in README	YES	YES
KPI_QAD_02	Swagger specification for the APIs	YES	YES
KPI_QAD_03	Additional documentation (examples, tutorials, etc)	NO	YES

4.1.3 Security

The components of the SIMPATICO platform allows for accessing sensitive user information or to perform certain operations on behalf of the user. This refers to, for instance, the information stored in the Citizen Data Vault, asking or answering questions in Citizenpedia, etc.

This information and operations should be protected in order to avoid undesired access or privacy flaws. The way the components of the platform should be protected, depends on the way they are exposed.

When the component exposes an API, the operations should be protected using the appropriate security protocols. Moreover, the use of a particular protocol should depend on whether the information is accessed on behalf of end user or in a user-independent manner. In the former case it is required that the access is performed using the OAuth2.0 access token associated to the end user (e.g. access to the user information, CDV operations, etc). In the latter case the operations should be performed in a restricted way, avoiding exposure of the methods on Internet, using the IP whitelisting, using basic authentication, etc.

When the component is exposed as a Web application to the end user, the access to the functionality should be allowed only after the user has been successfully authenticated. To accomplish this in an interoperable manner, the components of the platform should rely on a centralized authentication and authorization system, such as AAC component. This ensures

- The use of appropriate protocols, in particular OAuth2.0 protocol
- Single Sign On functionality across various platform components
- Integrated user authentication.

The functionality provided by the components may also depend on the role the user is associated with. In this way, for instance, the modification of the process models in CPD should be performed only by the civil servant using a dedicated account. The management of the different roles and their association to the platform user should be performed in a centralized way.

With regards to the security aspects, the quality assessment metrics refer to whether the component are appropriately protected.

Table 6: List of KPIs associated with security aspects

KPI name	KPI description	1 st phase value	2 nd phase value
KPI_QAS_01	API-based components exposing sensitive operations are protected using AAC OAuth2.0 implementation	YES	YES
KPI_QAS_02	Web-based components provide access to the sensitive functionality to authenticated users only.	YES	YES
KPI_QAS_03	Centralized role management and access control	NO	YES

4.1.4 Usability

This quality dimension refers to the user experience when the UI of the SIMPATICO platform components is considered. The key factors considered here include:

- Browser support. The component user interface should provide support for the wide range of widely used browsers, including IE10+, Chrome 47+, Firefox 38+, Safari 9+.
- Device support. The component UI should be properly visualized regardless the underlying platform. This also amounts to the responsive design² requirements. The platforms supported include, apart from desktop computers, mobile devices on Android 4.3+, iOS 8.3+, Windows 8.1.
- Internationalization. The UI of the component should support and should be adaptable to different user languages. This is also crucial for adoption of the components to the pilots in different countries.

The corresponding KPIs are captured in the following table.

Table 7: List of KPIs associated with usability

KPI name	KPI description	1 st phase value	2 nd phase value
KPI_QAU_01	Minimal browser support.	NO	YES
KPI_QAU_02	Multi-platform support and responsiveness	NO	YES
KPI_QAU_03	Internationalization	YES	YES

4.1.5 Source code management

For the development, issue tracking, and documentation of the SIMPATICO platform component, GitHub[5] has been selected as a source code management repository. The rationale behind this choice is as follows:

- In case of open source projects, GitHub allows for unlimited and free creation of repositories;
- It relies on state of art code management protocol, namely Git;
- It is equipped with a range of suitable tools extending the basic code management, including issue tracking, planning, documentation, notification, etc;
- GitHub is a tool of choice for FIWARE platform components.

Besides adhering to GitHub, there are other requirements that the components should satisfy. This includes

- Issue tracking for the bug, enhancement, and evolution of the components (discussed in details in Section 4.2);
- FIWARE recommendations [4] for the code management. This includes the guidelines for the source code branching, tagging, pull requests, etc.
- Appropriate version management including tagging the stable release versions, documenting the issues resolved with the release and implemented features, etc.

The corresponding KPIs are captured in the following table.

Table 8: List of KPIs associated with 4.1.5 source code management

² https://en.wikipedia.org/wiki/Responsive_web_design

KPI name	KPI description	1 st phase value	2 nd phase value
KPI_QASCM_01	Use GitHub for SCM and issue tracking	YES	YES
KPI_QASCM_02	Adhere to FIWARE SCM guidelines	YES	YES

4.1.6 Testing

To ensure the quality of the components, appropriate testing activities should be defined and implemented by the component providers and the platform as a whole. Apart from end-to-end integration test defined above, the components should provide a series of unit test with appropriate coverage regarding the APIs used by other components. The presence of unit tests is also fundamental for the adoption of Continuous Integration environment. Based on the revised platform architecture and the interaction requirements, these tests will be introduced as a part of the software components released during the second phase of the project. For this reason the KPI applies to the 2nd phase evaluation only.

Table 9: List of KPIs associated with testing

KPI name	KPI description	1 st phase value	2 nd phase value
KPI_QAT_01	Unit test coverage	-	80%

4.1.7 Deployment

To facilitate the deployment of the components by the pilot cities, the components should clearly define the deployment procedure, the system (software and hardware) requirements, and, in case of integration with other SIMPATICO components, the integration configuration. It is also expected that the installation of the component is completely autonomous and may be performed in isolation from the other components.

To further improve the deployment procedure allowing for targeting different Cloud environments, it is required that the components provide the corresponding Docker[6] container specification. This requirement is in line with the FIWARE recommendations for the component release management.

Table 10: List of KPIs associated with deployment

KPI name	KPI description	1 st phase value	2 nd phase value
KPI_QAR_01	Docker containers provided	NO	YES

4.2 Component issue tracking

As explained in previous section, the management of software issues is performed with GitHub issue management. This choice is tailored mainly to the fact that in this way the issues are related and explicitly linked to the software code and its changes. This also facilitates the realization of FIWARE guidelines for the issue management.

The issue management on GitHub is not limited only to the component developers, but is open to all GitHub users, even outside of the SIMPATICO project (for what concerns the issue creation). The creation of a new issue requires the following information:

- Component assigned (implicit to the component repository);
- Name and description of the issue;
- Issue type. The principal ones refer to Bug and Enhancement, where the former refers to a problem or malfunctioning of the component, while the later refers to the missing or new required functionality.

It is responsibility of the component owner to manage the issues providing:

- Feedback or clarification request in the form of comments;
- Modification of the issue type (e.g., wontfix, duplicate, etc) with the explanation of the reason for change
- Assigning a specific developer (assignee) for the issue resolution.

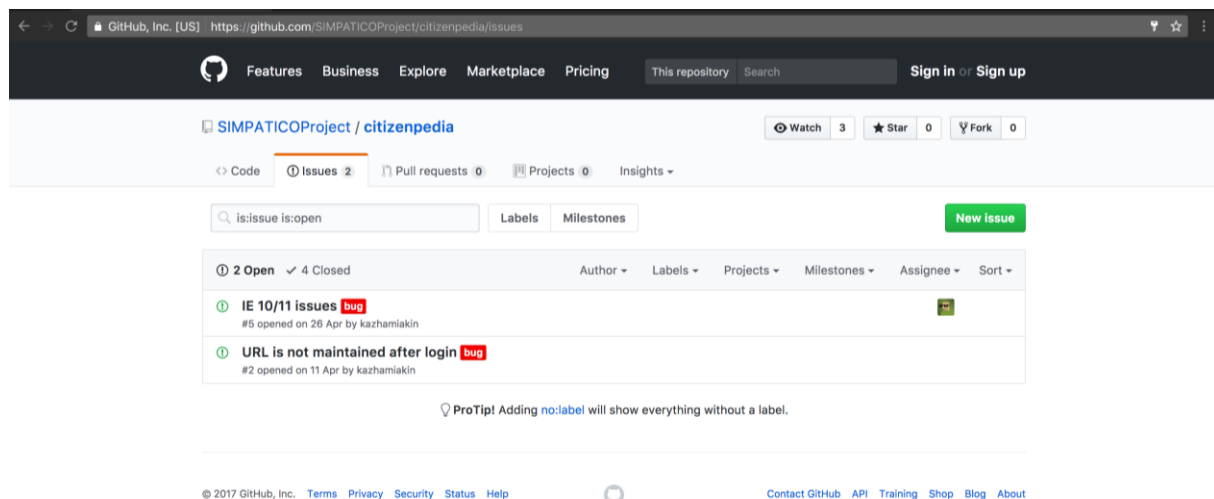


Figure 3: GitHub issue management

When the issue is resolved, the resolution should be linked via git commit message. The closed issues should be verified by the person who has reported the issue.

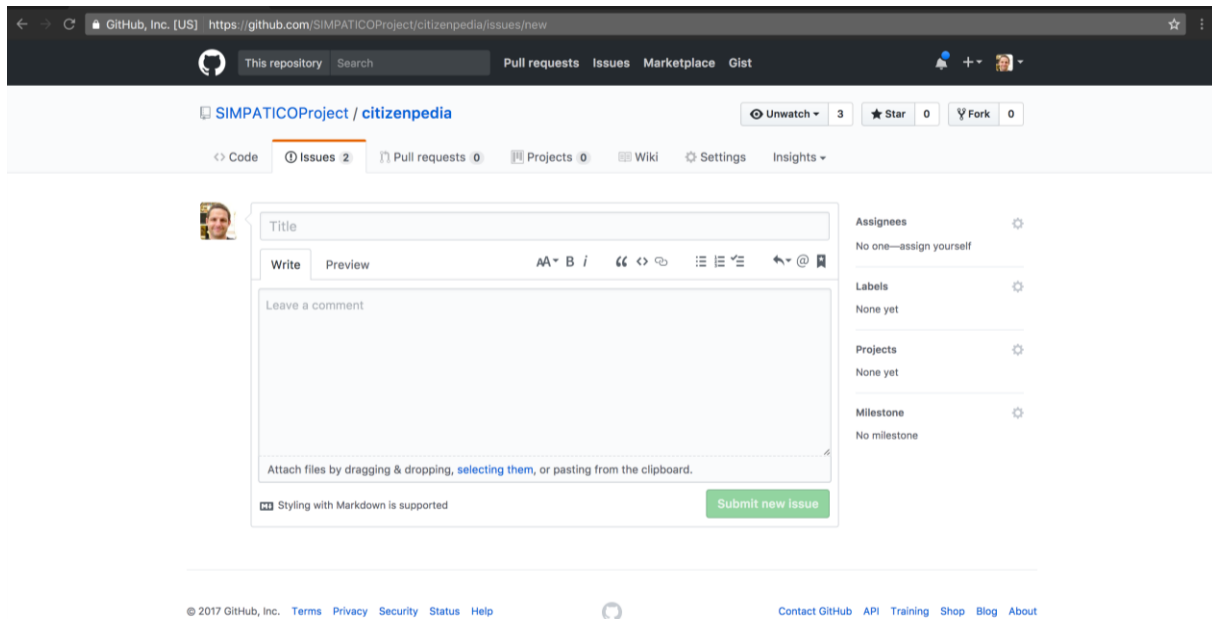


Figure 4: Creating new issue in GitHub

With respect to the issue tracking, the corresponding KPI refers to the percentage of the resolved issues for the component.

Table 11: List of KPIs associated with issue tracking

KPI name	KPI description	1 st phase value	2 nd phase value
KPI_IT_01	Percentage of issues resolved	75%	100%

In its current state, the GitHub issue management system reports the following information for each of the SIMPATICO components:

Table 12: Issue tracking information for the platform components

COMPONENT	Open Issues	Closed Issues	KPI_IT_01
IFE	0	4	100%
CDV	1	6	85%
TAE	4	4	50%
WAE	0	4	100%
QAE	3	6	66%
SF	1	5	83%
CPD	2	3	60%
EE/DB	1	4	80%

DA		0	0	100%
AAC		0	4	100%

5 Feedback on platform installation and deployment

In order to evaluate the ease to deploy and install the SIMPATICO platform in the three different use-cases, each city technical team has answered to a survey.

The questionnaire was focused on the activities to be undertaken to deploy, install and in case customize the SIMPATICO platform. The most of questions foresee a closed answer selecting one of the following values:

1. strongly disagree
2. disagree
3. neutral
4. agree
5. strongly agree

One question gives the possibility to insert free text to describe briefly the experience and give more feedback and suggestions in order to improve the platform. For this reason, a field Note was added to each question, if the survey editor wants to add some important issue.

Last four questions focus on the effort needed to complete all the actions needed to make the platform operating and maintain it.

In the following three sub-sections all questions and answers were reported use-cases per use-cases.

5.1 Trento use-case

In Trento use case SIMPATICO platform and tools are going to be tested in a production environment. More precisely, as part of its “smart city” strategy, Trento is working on the deployment of new e-service portal. The portal serves as a “one-stop shop” or unique access point that offers integrated and facilitated access to all the various services. With this new portal, it is possible for citizens and businesses to authenticate using smart service cards or one-time password devices, and to complete the interaction online.

SIMPATICO platform extends Trento e-service portal; in particular the e-service HTML template structure has been modified in order to inject the integration with SIMPATICO IFE component. The phase 1 architecture deploys CDV on a machine in the Trento data centre meanwhile all other SIMPATICO components are deployed on a cloud infrastructure managed by FBK. As far as citizen and professional authentication, both Trento e-service portal and AAC SIMPATICO component are based on the Provincia Autonoma di Trento Citizen Authentication Service.

For Trento the integration of the SIMPATICO platform and tools in the e-service portal was straight forward.

Table 13: Trento Use-Case questionnaire

n.	Question	Possible Answer	Your Answer	Please add some notes
1	The provided documentation to guide the installation and deployment of the platform and its related components was helpful	5. strongly agree 4. agree 3. neutral 2. disagree 1. strongly disagree	4	

2	Every single component was easy to install and deploy	5. strongly agree 4. agree 3. neutral 2. disagree 1. strongly disagree	4	
3	The installation and set-up of every single component was quick	5. strongly agree 4. agree 3. neutral 2. disagree 1. strongly disagree	4	
4	The integration of all components was simple to be performed	5. strongly agree 4. agree 3. neutral 2. disagree 1. strongly disagree	4	
5	The SWAGGER APIs helped me to test the component functionalities and facilitated to carry out the integration	5. strongly agree 4. agree 3. neutral 2. disagree 1. strongly disagree	3	
6	The problems occurred during the platform installation and deployment were already well documented and easy to solve	5. strongly agree 4. agree 3. neutral 2. disagree 1. strongly disagree	3	
6.a	Referring to the previous question, the intervention of a specialized responsible for a specific component due to an occurred error was prompt and professional	5. strongly agree 4. agree 3. neutral 2. disagree 1. strongly disagree	4	
7	The integration to connect the SIMPATICO infrastructure with repositories and/or legacy applications was simple	5. strongly agree 4. agree 3. neutral 2. disagree 1. strongly disagree	3	
8	Have you had to customize the APIs of your repositories and/or legacy applications for the connection with SIMPATICO?	1. No, I've used existing APIs without any changes 2. Yes, I had to adjust the APIs (please provide more details)	2	The adjustments primarily concern IFE component as this is the point of integration with the e-service UI and in our case was strongly related to the UI of the hosting web portal.

9	Please provide us with some sentences describing your experience with the SIMPATICO Platform installation and deployment. Describe briefly the occurred problems and some hints to improve the platform installation and deployment	free text		The main issue is that the platform is made independent components made with heterogeneous technologies. This required some expertise to bootstrap these technologies and obtain the right configuration. Using a virtualization technology (like, e.g., Docker) might help with the set up hiding the complexity and heterogeneity of the components.
10	Efforts needed to install the SIMPATICO infrastructure	person/hours	8	
11	Effort needed to connect the SIMPATICO infrastructure with repositories and/or legacy applications	person/hours	40	The effort is due to specific customization requested for the pilot.
12	Effort needed for daily operational tasks related to the technical maintenance of the SIMPATICO infrastructure	days	1	
13	No. of incidents encountered	number	0	After production deployment

5.2 Galicia use-case

In Galicia use-case SIMPATICO platform and tools are going to be tested in a controlled environment. For this controlled environment two e-services have been replicated from the ones already available in the Xunta digital platform. The e-services selected are the BS607A (Grants for the attendance to wellness and spas program) and BS613B (Individual grants for personal autonomy and complimentary personal assistance for disabled people).

For this use case Galician elderly community, concretely FEGAUS, COGAMI and ATEGAL associations, and Xunta civil servants have been involved.

The replicated e-services have been extended using SIMPATICO platform. For this, the e-service HTML template structure has been modified in order to inject the integration with SIMPATICO IFE component. All the components of the platform are deployed on a cloud infrastructure managed by Hi-Iberia.

For Galicia use-case the integration of the SIMPATICO platform and tools in the replicated e-service portal was straight forward.

Table 14: Galicia Use-Case questionnaire

n.	Question	Possible Answer	Your Answer	Please add some notes
1	The provided documentation to guide the installation and deployment of the platform and its related components was helpful	5. strongly agree 4. agree 3. neutral 2. disagree 1. strongly disagree	4	Quality of the documentation and its format is a bit uneven. Maybe some guidelines would be helpful to homogenise.
2	Every single component was easy to install and deploy	5. strongly agree 4. agree 3. neutral 2. disagree 1. strongly disagree	4	No major problems and found issues were quickly ironed out.
3	The installation and set-up of every single component was quick	5. strongly agree 4. agree 3. neutral 2. disagree 1. strongly disagree	3	Since it was the first version, some aspects needed to be fixed by developers.
4	The integration of all components was simple to be performed	5. strongly agree 4. agree 3. neutral 2. disagree 1. strongly disagree	3	Some issues for the first version.
5	The SWAGGER APIs helped me to test the component functionalities and facilitated to carry out the integration	5. strongly agree 4. agree 3. neutral 2. disagree 1. strongly disagree	5	This was very helpful in the design, testing and deployment phases.
6	The problems occurred during the platform installation and deployment were already well documented and easy to solve	5. strongly agree 4. agree 3. neutral 2. disagree 1. strongly disagree	3	Some problems were found only during deployment/testing
6.a	Referring to the previous question, the intervention of a specialized responsible for a specific component due to an occurred error was prompt and professional	5. strongly agree 4. agree 3. neutral 2. disagree 1. strongly disagree	5	Everybody responded quickly to our calls for help. The issue system was highly functional.
7	The integration to connect the SIMPATICO infrastructure with repositories and/or legacy	5. strongly agree 4. agree 3. neutral	N/A	N/A for Galicia, everything was developed from scratch including the

	applications was simple	2. disagree 1. strongly disagree		services.
8	Have you had to customize the APIs of your repositories and/or legacy applications for the connection with SIMPATICO?	1. No, I've used existing APIs without any changes 2. Yes, I had to adjust the APIs (please provide more details)	N/A	N/A for Galicia.
9	Please provide us with some sentences describing your experience with the SIMPATICO Platform installation and deployment. Describe briefly the occurred problems and some hints to improve the platform installation and deployment	free text		There were issues with some components: - CPD had to be updated with new APIs prior to the first test deployment. - CDV needed to be updated to comply with the rest of components. Eventually there were some problems during the pre-evaluation that were addressed during the next version (deployment in progress) - TAE provided results for Spanish that were not useable during pre-evaluation. This is addressed in the next version. - AAC had several problems with special characters of the Spanish language such as acute accents (á) that propagated from the user accounts used in Google.
10	Efforts needed to install the SIMPATICO infrastructure	person/hours	60	For the pre-evaluation, estimate is: 60 person/hours for the system (including asking for fixes and re-deploying) For 1st prototype, estimate is: 15 person/hours for the complete system (same developers as in pre-

				evaluation so some lessons learnt have applied)
11	Effort needed to connect the SIMPATICO infrastructure with repositories and/or legacy applications	person/hours	N/A	N/A as the Galicia system does not link to real e-services provided by Xunta.
12	Effort needed for daily operational tasks related to the technical maintenance of the SIMPATICO infrastructure	days	1	
13	No. of incidents encountered	number	10	Estimate, not explicitly tracked during the deployment. Figure refers to issues that required communication with the developers of the components and/or new versions to be deployed.

5.3 Sheffield use-case

In Sheffield use case SIMPATICO platform and tools are going to be tested in a controlled environment. For this environment 3 e-services have been replicated from those that are available on the current Sheffield City Council Website. The website enables simple access to a number of services available to citizens.

The SIMPATICO platform gives the citizen extra benefits to allow a better user experience within the site, in particular the e-service HTML template structure has been modified in order to inject the integration with SIMPATICO IFE component. The phase 1 architecture deploys CDV on a machine in the Sheffield City Council data centre meanwhile all other SIMPATICO components are deployed on a cloud infrastructure hosted by SPARTA. In regards to authentication, all e-services are authenticated via the AAC SIMPATICO component.

In regards to integration of the SIMPATICO platform there were some issues in regards to the complexities of the current Sheffield City Council website however once these were overcome the integration of the tools were relatively straightforward.

Table 15: Sheffield Use-Case questionnaire

n.	Question	Possible Answer	Your Answer	Please add some notes
1	The provided documentation to guide the installation and deployment of the platform and its related components was helpful	5. strongly agree 4. agree 3. neutral 2. disagree 1. strongly disagree	4	

2	Every single component was easy to install and deploy	5. strongly agree 4. agree 3. neutral 2. disagree 1. strongly disagree	3	We required to install and deploy in Docker files so was not as easy as conventional methods.
3	The installation and set-up of every single component was quick	5. strongly agree 4. agree 3. neutral 2. disagree 1. strongly disagree	3	
4	The integration of all components was simple to be performed	5. strongly agree 4. agree 3. neutral 2. disagree 1. strongly disagree	4	
5	The SWAGGER APIs helped me to test the component functionalities and facilitated to carry out the integration	5. strongly agree 4. agree 3. neutral 2. disagree 1. strongly disagree	4	
6	The problems occurred during the platform installation and deployment were already well documented and easy to solve	5. strongly agree 4. agree 3. neutral 2. disagree 1. strongly disagree	3	
6.a	Referring to the previous question, the intervention of a specialized responsible for a specific component due to an occurred error was prompt and professional	5. strongly agree 4. agree 3. neutral 2. disagree 1. strongly disagree	4	
7	The integration to connect the SIMPATICO infrastructure with repositories and/or legacy applications was simple	5. strongly agree 4. agree 3. neutral 2. disagree 1. strongly disagree	2	It was very difficult to connect with our legacy application
8	Have you had to customize the APIs of your repositories and/or legacy applications for the connection with SIMPATICO?	1. No, I've used existing APIs without any changes 2. Yes, I had to adjust the APIs (please provide more details)	1	

9	Please provide us with some sentences describing your experience with the SIMPATICO Platform installation and deployment. Describe briefly the occurred problems and some hints to improve the platform installation and deployment	free text		Many of the components were from different technologies so it was difficult to install them together. We felt the easier approach would have been to contain the components in Docker files to enable quick and simple deployment.
10	Efforts needed to install the SIMPATICO infrastructure	person/hours	80	
11	Effort needed to connect the SIMPATICO infrastructure with repositories and/or legacy applications	person/hours	120	Mainly due to the complexes of the legacy system.
12	Effort needed for daily operational tasks related to the technical maintenance of the SIMPATICO infrastructure	days	1	
13	No. of incidents encountered	number	5	Not tracked. Mainly with the components but solved after communications with developers.

6 Conclusion

This document aims at firstly defining the technical KPIs to evaluate the SIMPATICO technical platform, and secondly measuring those KPIs to understand the evolution of the whole platform in the three different use-cases (Trento, Galicia, Sheffield) and produce the related improvements starting from the feedback received from technical teams.

The SIMPATICO platform was evaluated basing on three different parameters. The first one is the feasibility of the all Integration Use-Cases to evaluate the integration among all the platform components; the second one is the Quality Assessment to evaluate mainly the quality of the development of the components, and the last one is focused on the Feedback received from the technical team of each city.

The measure KPIs are listed in the Table 16.

The KPI_IUC_01 "Percentage of the Integration Use-Cases achieved and operating" shows that the Sheffield had some integration problems due to the complexity of the legacy system adopted in the city. The common feedback coming from the use cases pilot on the deployment and installation of the platform (KPI_FoS_02) points out the need to use the Docker container [6] for the most of SIMPATICO components to facilitate the integration, avoiding to know and understand several technologies used by the different components.

The effort needed to deploy and install the platform in this first phase has involved too many resources for Galicia and Sheffield. It will be interesting to know what value will have the same KPIs in the second phase when the SIMPATICO platform will be updated according to the new requirements described in D5.2 [7].

Table 16: Measured KPIs

KPI name	KPI description	Target	Trento	Galicia	Sheffield
KPI_IUC_01	Percentage of the Integration Use-Cases achieved and operating	75%	100%	77%	60%
KPI_FoS_02	The average value of the answers provided by the interviewees	3,5	3,44	3,85	3,11
KPI_EFF_03	Efforts needed to install the SIMPATICO infrastructure	50 PH	8 PH	60 PH	80 PH
KPI_EFF_04	Effort needed to connect the SIMPATICO infrastructure with repositories and/or legacy applications	80 PH	40 PH	N/A	120 PH
KPI_EFF_05	Effort needed for daily operational tasks	1 day	1 day	1 day	1 day
KPI_PRO_06	No. of incidents encountered	10	0	10	5

The Quality Assessment KPIs are listed in Table 17 and they meet the target values at 99%.

Table 17: Quality Assessment KPIs

KPI name	KPI description	Target	Trento	Galicia	Sheffield
KPI_QAI_01	Exposure of APIs as JSON-based REST/RPC	YES	YES	YES	YES
KPI_QAI_02	Support for SSO	YES	NO ³	YES	YES
KPI_QAI_03	Support for CORS	YES	YES	YES	YES
KPI_QAD_01	Installation, configuration, and integration documentation in README	YES	YES	YES	YES
KPI_QAD_02	Swagger specification for the APIs	YES	YES	YES	YES
KPI_QAD_03	Additional documentation (examples, tutorials, etc)	NO	NO	NO	NO
KPI_QAS_01	API-based components exposing sensitive operations are protected using AAC OAuth2.0 implementation	YES	YES	YES	YES
KPI_QAS_02	Web-based components provide access to the sensitive functionality to authenticated users only.	YES	YES	YES	YES
KPI_QAS_03	Centralized role management and access control	NO	NO	NO	NO
KPI_QAU_01	Minimal browser support.	NO	NO ⁴	NO4	NO4
KPI_QAU_02	Multi-platform support and responsiveness	NO	NO	NO	NO
KPI_QAU_03	Internationalization	YES	YES	YES	YES
KPI_QASCM_01	Use GitHub for SCM and issue tracking	YES	YES	YES	YES
KPI_QASCM_02	Adhere to FIWARE SCM guidelines	YES	YES	YES	YES
KPI_QAR_01	Docker containers provided	NO	NO	NO	NO

³ Partial due to restrictions of the SPID protocol

⁴ CPD, CTZ do not completely support IE10/11

7 References

- [1] A. Filograna, "SIMPATICO Platform Requirements and Architecture v1," SIMPATICO Project, 2016.
- [2] S. Brown, "Standardized Technology Evaluation Process (STEP) User's Guide and Methodology for Evaluation Teams," May 2007.
- [3] "Standardized Technical Evaluation Process," [Online]. Available: <http://www2.mitre.org/work/sepo/toolkits/STEP/>. [Accessed 05 May 2017].
- [4] FIWARE (Core Platform of the Future Internet), "Developer Guidelines," 2017. [Online]. Available: https://forge.fiware.org/plugins/mediawiki/wiki/fiware/index.php/Developer_Guidelines.
- [5] GitHub. [Online]. Available: <http://www.github.com>.
- [6] Docker. [Online]. Available: <https://www.docker.com/>.
- [7] A. Filograna, "SIMPATICO platform requirements and architecture v2," SIMPATICO Project, 2017.
- [8] M. Pistore, "SIMPATICO evaluation report v1," SIMPATICO Project, 2017.
- [9] R. Santos de la Cámara, "Advanced methods and tools for user interaction automation," SIMPATICO Project, 2017.