



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

D4.1 – Citizenpedia Framework Specification and Architecture

WP4 Human computation
Task 4.1 Human computation framework specification
Due Date: 31/08/2016
Submission Date: 31/08/2016
Responsible Partner: Universidad de la Iglesia de Deusto (DEUSTO)
Version: 1.0
Status: Final
Author(s): Iván Pretel (DEUSTO), Unai López (DEUSTO), Enrique Sanz (DEUSTO), Diego López-de-Ipiña (DEUSTO), Vincenzo Cartelli (BENG), Giuseppe Di Modica (BENG), Orazio Tomarchio (BENG)
Reviewer(s): Vincenzo Savarino(ENG), Antonio Filograna (ENG), Alessio Palmero Aprosio (FBK)
Deliverable Type: R: Report
Dissemination Level: PU: Public

Version History

Version	Date	Author	Partner	Description
v0.1	20/04/2016	Iván, Pretel Unai, López	DEUSTO	First draft of the Table of Contents.
v0.2	24/05/2016	Unai, López	DEUSTO	Work in the Requirements section.
v0.3	30/05/2016	Iván, Pretel	DEUSTO	Work in the Functional / Logical View section.
v0.4	30/06/2016	Enrique, Sanz Iván, Pretel Unai, López	DEUSTO	Comparative analysis of participative solutions.
v0.5	04/07/2016	Vincenzo, Cartelli Giuseppe, Di Modica Orazio, Tomarchio	BENG	Requirements of the CPD.
v0.6	04/07/2016	Iván Pretel	DEUSTO	Requirements review, Actors and Roles definitions.
v0.7	04/07/2016	Vincenzo, Cartelli Giuseppe, Di Modica Orazio, Tomarchio	BENG	Actors and Roles review.
v0.8	08/07/2016	Enrique, Sanz Iván, Pretel Unai, López	DEUSTO	Work in “Top-down approach” and “Technology Choice and Usage Description”.
v0.9	15/07/2016	Iván, Pretel Unai, López Diego, López-de-Ipiña	DEUSTO	Integration and revision of all the deliverable sections.
v0.9.1	18/07/2016	Vincenzo Savarino Antonio Filograna	ENG	Internal review for pre release.
v0.9.2	29/07/2016	Iván, Pretel Unai, López	DEUSTO	Corrections on the internal review.
v0.9.3	29/08/2016	Alessio Palmero Aprosio	FBK	Internal review.
v1.0	31/08/2016	Marco 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This deliverable contains original unpublished work except where clearly indicated otherwise.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
2	Top-down approach: From user to system perspective	9
2.1	Surveys to gather users' perspective and feedback.....	10
2.2	Results.....	13
3	Technology choice and usage description	18
3.1	Comparative of existing tools suitable to be integrated or extended	18
3.1.1	Databases	18
3.1.2	Indexing engines.....	19
3.1.3	Gamification engines.....	20
3.2	Comparative analysis of participative solutions.....	21
3.2.1	Question and Answers Engines as participative solutions	21
3.2.2	Collaborative Process Modelling	23
3.3	Selection	27
4	Citizenpedia requirements specifications.....	29
4.1	Social question answering engine requirements (QAE).....	34
4.2	Collaborative procedures designer requirements (CPD)	41
4.3	Collective knowledge base requirements (CKB).....	46
4.4	Stakeholder Incentivization techniques requirements (SIT)	49
4.5	Summary of Citizenpedia framework requirements.....	51
5	Citizenpedia architecture	54
5.1	Schematic Overview ("Broad Picture")	54
5.1.1	Social question answering engine	54
5.1.2	Collaborative procedures designer	55
5.1.3	Collective knowledge base.....	57
5.1.4	Gamification Engine	58
5.2	Functional / Logical View (Interfaces)	58
6	Conclusion	60
7	References.....	61

List of figures

Figure 1 – Schematic view of the top-down approach.....	9
Figure 2 – Welcome screen for Citizenpedia survey	10
Figure 3 – Citizenpedia Survey: QAE video screenshots.....	11
Figure 4 – Citizenpedia Survey: QAE questionnaire screenshots	11
Figure 5 – Citizenpedia Survey: CPD citizen video screenshots	12
Figure 6 – Citizenpedia Survey: CPD civil servant video screenshots	12
Figure 7 – Citizenpedia Survey: CPD questionnaire screenshots (left: citizen, right: civil servant) ..	13
Figure 8 – Main citizens’ comments	17
Figure 9 – Main civil servants’ comments	17
Figure 10 – Roles hierarchy of Citizenpedia actors	31
Figure 11 – Relation between macro-requirements and requirements of Citizenpedia.....	34
Figure 12 – Example of collaboration workflow	43
Figure 13 – Macro-use cases of Citizenpedia 1	53
Figure 14 – Macro-use cases of Citizenpedia 2	53
Figure 15 – Schematic overview of Citizenpedia	54
Figure 16 – Architecture for collaboration support of process modelling	57
Figure 17 – Interactions of the main stakeholders	58

List of tables

Table 1 – Use case and age range mapping.....	14
Table 2 – Discretionary usage of the Citizenpedia framework	14
Table 3 – Usage of Question & Answers platforms.....	14
Table 4 – Motivation and rewards feedback.....	15
Table 5 – Common problems	15
Table 6 – Channel to solve the most common problems	15
Table 7 – Channel used to contact a civil servant	16
Table 8 – Flowchart understanding	16
Table 9 – Channels to suggest changes and enhancements	16
Table 10 – Q&A Engines features	23
Table 11 – Q&A Engines comparisons.....	23
Table 12 – Summarize Evaluation.....	26
Table 13 – Perspectives and Requirement fields mapping.....	29
Table 14 – Macro-requirement QAE	31
Table 15 – Macro-requirement CPD	32
Table 16 – Macro-requirement CKB.....	33
Table 17 – Macro-requirement SIT	33
Table 18 – Requirement QAE.1	34
Table 19 – Requirement QAE.2	35
Table 20 – Requirement QAE.3	36
Table 21 – Requirement QAE.4	37
Table 22 – Requirement QAE.5	37
Table 23 – Requirement QAE.6	38
Table 24 – Requirement QAE.7	39
Table 25 – Requirement QAE.8	39
Table 26 – Requirement QAE.9	40
Table 27 – Requirement CPD.1.....	41
Table 28 – Requirement CPD.2.....	42
Table 29 – Requirement CPD.3.....	43
Table 30 – Requirement CPD.4.....	44
Table 31 – Requirement CPD.5.....	44
Table 32 – Requirement CPD.6.....	45
Table 33 – Requirement CPD.7.....	46
Table 34 – Requirement CKB.1.....	46
Table 35 – Requirement CKB.2.....	47
Table 36 – Requirement CKB.3.....	47
Table 37 – Requirement CKB.4.....	48
Table 38 – Requirement CKB.5.....	49
Table 39 – Requirement SIT.1	49
Table 40 – Requirement SIT.2	50
Table 41 – Requirement SIT.3	51
Table 42 – Summary of Citizenpedia framework requirements.	51
Table 43 – Preliminary REST API methods.	59

Glossary

ACID	Atomicity, Consistency, Isolation, Durability
ANSI	American National Standards Institute
API	Application Programming Interface
BPM	Business Process Management
CKB	Collective Knowledge Base
CPD	Collaborative Procedure Designer
CS	Civil Servant
DOM	Document Object Model
HTML	HyperText Markup Language
IT	Information Technologies
JSON	JavaScript Object Notation
MIT	Massachusetts Institute of Technology
PA	Public Administration
PDF	Portable Document Format
Q&E	Question & Answer
QAE	Question Answering Engine
RDBMS	Relational DataBase Management System
REST	Representational State Transfer
SAAS	Software As A Service
SIT	Stakeholder Incentivization Techniques
SQL	Structured Query Language
UI	User Interface

Executive summary

This document is the deliverable “**D4.1 – Citizenpedia framework specification and architecture**” of the European project “SIMPATICO - SIMplifying the interaction with Public Administration Through Information technology for Citizens and cOmpanies” (hereinafter also referred to as “**SIMPATICO**”, project reference: 692819).

SIMPATICO addresses a strategic challenge towards the **innovation and modernization of the public sector**: the need to offer a more efficient and more effective experience to companies and citizens in their daily interaction with Public Administration (PA) by (i) offering a personalized delivery of PA online services; (ii) enabling a better comprehension of the complex processes and documents (forms, regulations, etc.) behind these services; (iii) engaging them to improve the administration processes and services. 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.

This report includes the results of project task T4.1 “**Human computation framework specification**”. Such task aims to match the functionality offered by the range of technical results and components in the area of Human Computation support brought forward by the different partners of the project with the functionality expectations of the different stakeholders of the envisaged solution.

This task gives place to the **specification of the Citizenpedia framework requirements**, both from a functional and non-functional perspective. A **top-down approach**, from the user perspective (outcome from a revision of the state of the art and previous experiences of the partners) to system perspective (as provided by the partners who provide components), is followed. A **methodology** and **template** used by the partners in earlier project (Volere) is used to formally specify the Citizenpedia component requirements.

Besides, a **comparative analysis** of existing solutions and components was carried out in order to select, on the one hand, those features which should be added to Citizenpedia, and on the other hand, already existing tools which could be integrated or extended.

The **high-level architecture** derived from combining the Social Question Answering Engine, Collaborative Procedures Designer and Citizenpedia Collective Knowledge base and API brought forward by the different partners is defined. This architecture follows a service-oriented approach based on REST in order to ease its integration with the other components of the SIMPATICO solution.

1 Introduction

This deliverable presents the outcome of the task T4.1 “Human computation framework specification”: the initial design and specification of the Citizenpedia platform. This document includes a formalization of the system requirements and some details on architectural aspects. Along the deliverable some references to other deliverables of the SIMPATICO project are made.

The Citizenpedia is the human computation framework that leverages the SIMPATICO project with the collaborative knowledge provided by its stakeholders. It will complement the SIMPATICO environment with a place where citizens can solve their doubts and interact in an amenable way with the public administration. It will expose mainly two tools to the stakeholders: the first one is a Question Answering Engine (QAE), where citizens will be able to post and solve doubts, and also to look up for terms and definitions. The second one is a Collaborative Procedure Designer (CPD), where civil servants will describe current e-services in the form of flowcharts/diagrams and citizens will be able to comment on them. This way, Citizenpedia will enable an easy way for citizens to take part in the design of bureaucratic procedures.

This deliverable is structured as follows: the design of Citizenpedia has been driven by a top-down approach, beginning from the developers' and user's perspective. We conducted some online surveys and used their results to drive the design of the system requirements. This process is detailed in Section 2. Section 3 provides some background on existing technologies that could be used/extended in the development of Citizenpedia, such as databases or indexing engines. Then, in Section 4, the system requirements are described using the Volere methodology. After that, description on the internal architecture of Citizenpedia is provided in Section 5, with focus on the building blocks that will be created. Finally, we draw some conclusions in Section 6.

2 Top-down approach: From user to system perspective

A top down approach to design a system should begin from the users. Following this rule, we have elaborated a methodology made up by some surveys targeting the two potential user types (civil servants and citizens) in order to complement the technical need with real ones. This section will describe this process and briefly summarize its results.

SIMPATICO is conceived as a novel ecosystem of tools, easily deployable in public administrations. In particular, the Citizenpedia framework aims to develop an innovative and open technological framework to promote the creation and management of collective knowledge to simplify the e-services of public administrations and improve their understanding. In order to implement and promote a really useful and attractive framework, Citizenpedia needs a bottom-up approach that leverages active participation of different stakeholders which take part in the requirements and framework definition. In particular, these stakeholders are citizens and civil servants, who are empowered to participate in the definition of the requirements and the main functionalities of this framework.

The performed methodology to follow this approach is based on two main sets of activities:

- **Preliminary use cases and technical definition.** Citizenpedia framework functionality demands have been inferred from the Pilot Scenarios and Public Services elicitation. However, the main inputs have been the Description of Actions, which included SIMPATICO environment preliminary specifications and the description and feedback provided by partners in the meetings already celebrated, i.e. kick-off-meeting in Trento, the 1st quarterly meeting in Sheffield in June 2016 and the several technical meetings during this period. In such meetings, the different assets brought together from previous projects have been considered when devising the technical definition.
- **Mock-ups generation and surveys definition.** The purpose of these activities was to collect needs and ideas from stakeholders in order to identify a set of requirements to be fulfilled by the Citizenpedia framework. The ideas and insights gathering was performed through online surveys (detailed in the following section) where the designed mock-ups were exposed through YouTube videos.

Having both sets of activities performed, a wide range of results were used to detail the framework requirements and create new ones that are missing. Consequently, the final requirements definition covers not only the system perspective, but also the stakeholders' one. A schematic definition of this process is depicted in the following figure.

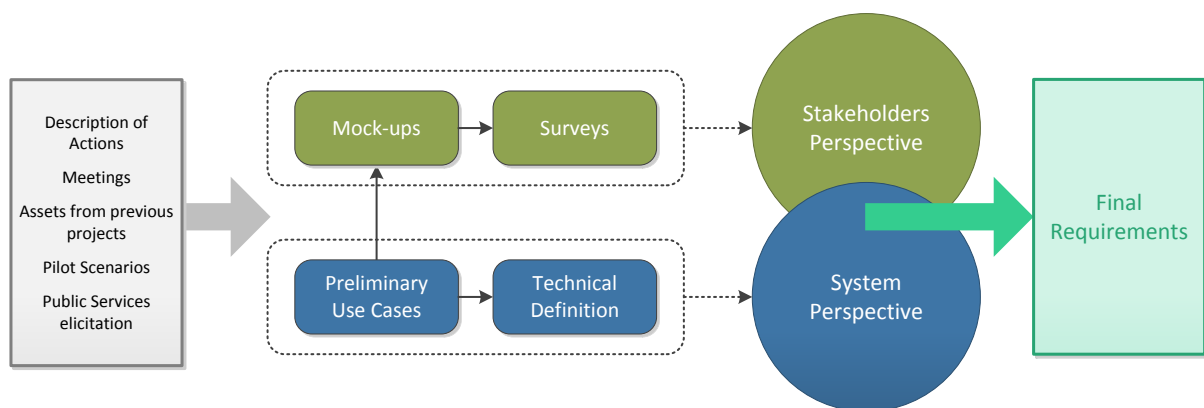


Figure 1 – Schematic view of the top-down approach

2.1 Surveys to gather users' perspective and feedback

There are several ways to gather the opinion and perspective of a user group, such as group meetings or interviews. In our case, we had several constraints in the process of approaching to the users, being the most notable one the time: we had few weeks to elaborate a test, grab users and collect their results. Thus, we chose to make personal online surveys. This way, the distribution of the surveys would be instant (just by sharing the URL to the proper collective) and we could give plenty of time for the users to fill the forms.

Given that an online survey leaves less room for explanations than a physical survey, we made our best to maximize the level of detail of what we were proposing. To that end, we included some videos where each use case was detailed by a dummy figure conducting actions in a mock-up Citizenpedia.

As a side note, we created a survey per pilot language, i.e., English, Spanish and Italian. All of the three contained the same question/answer set, translated to the proper language. In this section of the deliverable, just the English version is shown for the shake of brevity. Links to the survey forms will be provided at the end of the section.

The survey contains two parts. The first one is common to citizens and civil servants, and it is focused on the Question Answering Engine (QAE) portal of Citizenpedia. The second one is related to the Collaborative Procedure Designer (CPD). In this case, videos and questions differ for citizens and civil servants. The survey ends with some demographic questions (e.g. gender, age,...).

We will now move to a more detailed description of the survey. Initially, it presents a welcome screen with a message and two buttons, each directing the user to a survey depending on its potential role in the Citizenpedia: Citizen or Civil Servant. This welcome screen is shown as follows.

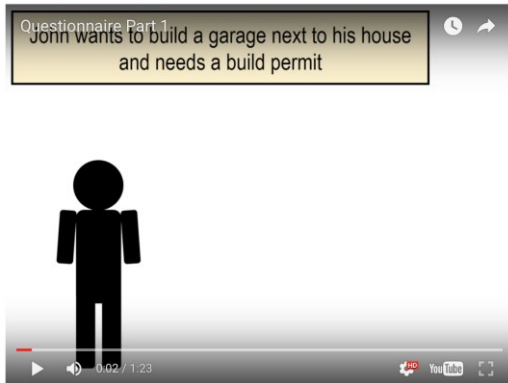


Figure 2 – Welcome screen for Citizenpedia survey

Once the user type is selected, the first part of the survey is shown: the question set about the Question Answering Engine. This part begins with a mock-up video that explains in one minute and a half the basic functionalities of the QAE. In particular, it shows a citizen called John that checks some doubts in the QAE related to a build permit. Some screenshots of the video are provided. After that, a set of questions is shown to the survey participant. We show below screenshots of some of the questions.

1st part: Question&Answer portal

Citizenpedia will have two main parts. The first one is a public question-answering portal, where you as a citizen will be able to ask any question related to the use of administrative services. This video shows our idea in a nutshell:



1st part: Question&Answer portal

Citizenpedia will have two main parts. The first one is a public question-answering portal, where you as a citizen will be able to ask any question related to the use of administrative services. This video shows our idea in a nutshell:

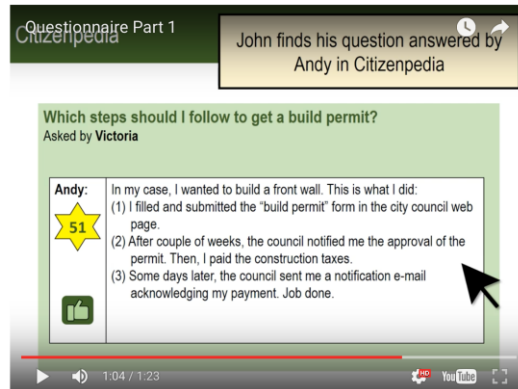


Figure 3 – Citizenpedia Survey: QAE video screenshots

Have you ever used a question-answering portal?

- Yes, to ask questions / post doubts
- Yes, to answer other's questions
- Yes, both to post and answer questions
- No, I've never used one
- Other: _____

In some question&answer portals users gain points as reward for answering questions. The more points gained, the more reputation in the portal. When a user reaches certain level of reputation, he gains permission to moderate and manage the portal. Do you see it as an attractive feature?

- Yes, it would encourage me to participate more actively in Citizenpedia
- Yes, but I don't think that it would make me be more active
- No
- Other: _____

Figure 4 – Citizenpedia Survey: QAE questionnaire screenshots

Once the part of the QAE is finished, the part related to the CPD begins. This part contains first a video and then a set of questions, which differ in the citizen and civil servant survey. The rationale behind this is that in the CPD, civil servants will be in charge of creating content (diagrams explaining bureaucratic procedures), and citizens will be able just to comment on them. Thus, we developed different videos and question sets.

The CPD video for the citizen survey shows John, the same citizen as in the QAE video, willing to make a suggestion on the build permit procedure. This one-minute video shows the basic functionalities on how a citizen can comment on a CPD diagram that represents a procedure. Some screenshots are shown below.

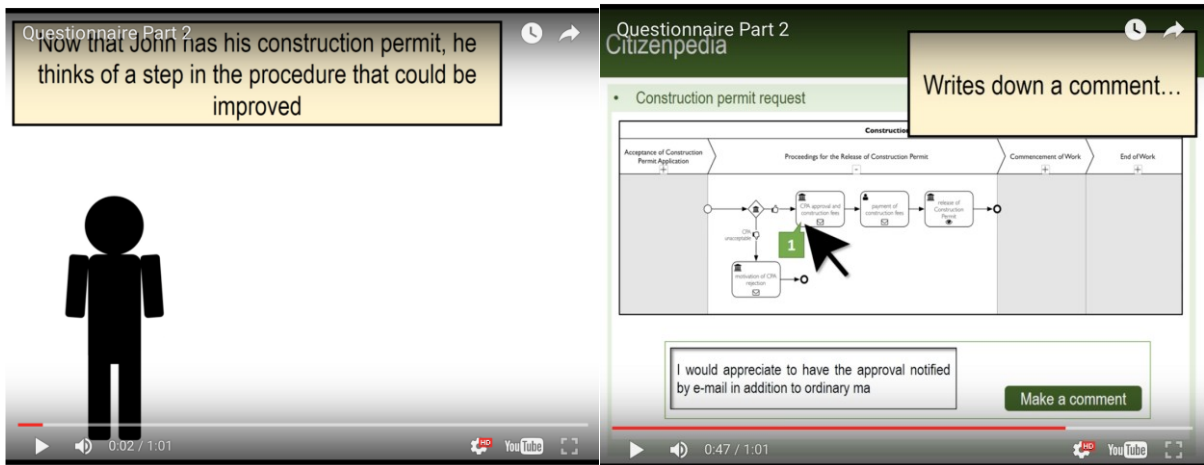


Figure 5 – Citizenpedia Survey: CPD citizen video screenshots

The CPD video for the civil servant shows Steve, a civil servant using the CPD. The minute-and-a-half video shows how Steve sees the comment made by John, and then how Steve updates the bureaucratic procedure accordingly. Screenshots are shown below.

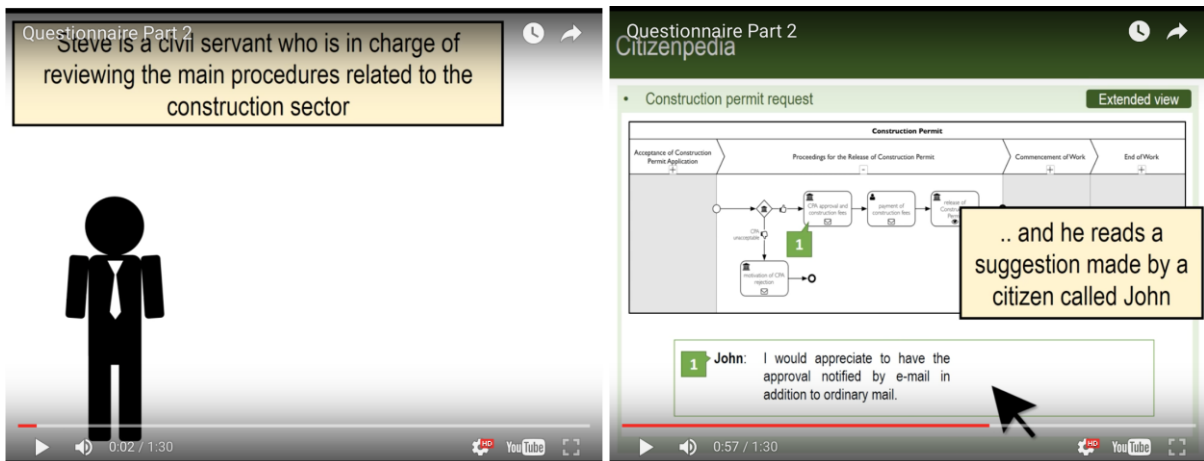
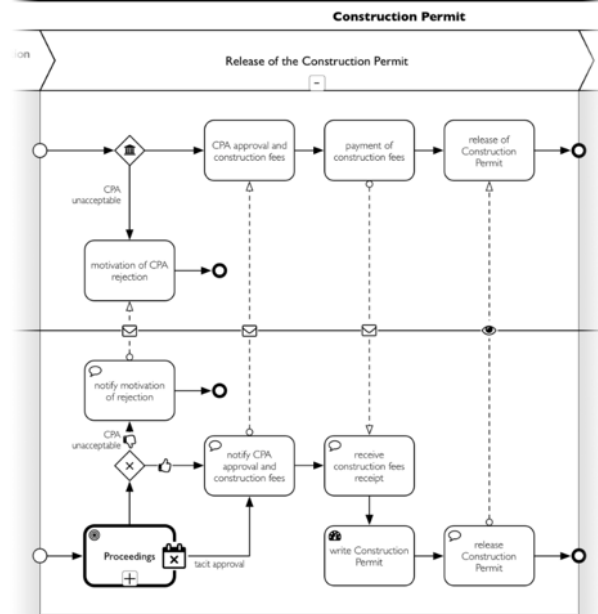
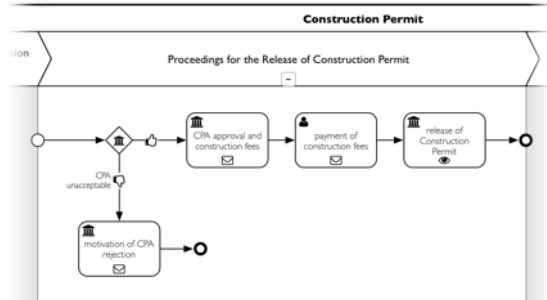


Figure 6 – Citizenpedia Survey: CPD civil servant video screenshots

As it was done in the QAE part, the CPD part of the survey shows some questions after the videos. These questions are accompanied by some diagrams to better explain the CPD concepts to the survey participant. Sample screenshots are shown below for the citizen and civil servant survey.

Sample flowchart from the video:

Sample flowchart from the video:



In the flowchart above, are you able to distinguish the internal procedure from the interactions with the citizen? (In case of a partial recognition, please select "Other" and write what needs to be clarified)

In the flowchart above, do you fully understand which steps are performed by Public Administration and which ones by citizens?

- Yes, I understand who does each of the steps
- No, I don't have a clue
- Other: _____

- Fully
- Not at all
- Other: _____

Figure 7 – Citizenpedia Survey: CPD questionnaire screenshots (left: citizen, right: civil servant)

Once the CPD part of the survey is finished, the participant is asked some demographic questions. That finishes the survey. We estimated about 10 minutes maximum for the average user to fulfill it.

The forms were created using the Google Form toolset, which eased the creation and distribution of the survey. The links to the surveys themselves can be found here for the English¹, Italian² and Spanish³ version.

2.2 Results

Having gathered the stakeholders' feedback, all the results have been grouped into three main clusters. According to the three SIMPATICO use cases (England, Italy and Spain) and the target citizen group per use case, the following table shows three main age ranges used as clustering criteria. More information on the context of each use case can be found in the SIMPATICO deliverable D6.1.

¹ <http://apps.morelab.deusto.es/citizenpedia-survey/en/>

² <http://apps.morelab.deusto.es/citizenpedia-survey/it/>

³ <http://apps.morelab.deusto.es/citizenpedia-survey/es/>

Table 1 – Use case and age range mapping

Use case	Group name	Age range (years)	Sample size
England	Youngsters	18-34	42
Italy	Middle Age	35-64	55
Spain	Elderly	65+	55

Discretionary usage measures the proportion of potential users choose to use the system. The first question exposes the following results (Table 2):

Table 2 – If you had access to such web portal to clarify these issues, would you use it?

Youngsters	Middle Age	Elderly	Response
76.19 %	76.36 %	54.55 %	Yes, as a first choice
19.05 %	14.55 %	32.73%	Just when I am not able to go physically to the public administration
2.38 %	5.45 %	9.09 %	I don't think so
2.38 %	3.64 %	3.64 %	Other

As the previous table exposes, the Citizenpedia platform would be a widely adopted solution to clarify issues related to procedures of public administration. The adoption level is directly associated to the age: the younger is the citizen the higher is the solution's adoption. Several comments were gathered through this question and were taken into account during the requirements elicitation. Several citizens mentioned the reliability of the response and a need to simplify the terms and procedures.

Another interesting finding on the results was the high usage of Q&A platforms. More than the 55% of the user of all ages report to have previously used a Q&A engine to ask/post doubts. This finding is relevant, as a relevant portion of the surveyed users would find Citizenpedia familiar. However, we also found that relatively low people devote time to answering questions in these portals. Thus, we might need to make fully exploit the gamification techniques in Citizenpedia to prevent this to happen. The detailed analysis of these findings is reported in Table 3.

Table 3 – Have you ever used a question-answering portal?

Youngsters	Middle Age	Elderly	Response
21.43 %	23.63 %	16.36 %	Yes, both to post and answer questions
47.62 %	49.09 %	40.00 %	Yes, to ask questions / post doubts
7.14 %	3.64 %	1.82 %	Yes, to answer other's questions
21.43 %	20.00 %	41.82 %	No, I've never used one
2.38 %	3.64 %	0.00 %	Other

In some Question & Answer portals users gain points as reward for answering questions. The more points gained, the more reputation in the portal. When a user reaches certain level of reputation, he gains permission to moderate and manage the portal. According to Table 4, the youngest group can be persuaded using a virtual rewarding system as well as reputation. This last term, reputation, is not an effective motivation to enhance the participation of the rest of groups. However and according to several comments of the survey's participants, it is an important concept for them because it means reliability and quality. We found this comment by a participant relevant: "For me, a high reputation of the answers owner means high quality of the answers".

Table 4 – In some question&answer portals users gain points as reward for answering questions. The more points gained, the more reputation in the portal. When a user reaches certain level of reputation, he gains permission to moderate and manage the portal. Do you see it as an attractive feature?

Youngsters	Middle Age	Elderly	Response
50.00 %	29.09 %	27.27 %	Yes, it would encourage me to participate more actively in Citizenpedia
42.86 %	29.09 %	21.82 %	Yes, but I don't think that it would make me be more active
4.76 %	40.00 %	47.28 %	No
2.38 %	1.82 %	3.64 %	Other

Regarding the most common problems which citizens experienced during public administration procedures, the following table (Table 5) exposes that the most common problems are related to the complexity of the documents and procedures. Although the three groups provides similar results, the youngest group has significantly more problems interpreting the guidelines terms. Furthermore, information architecture and usability are very common issues inside the participants comments: “I don't know which one is the correct web page”, “I am always lost browsing in e-services of the public administration”... Consequently, Citizenpeda should satisfy not only need of having a friendly description of procedures and terms, but also the need of an easy to use site.

Table 5 – Think of your previous experiences with public administration. What kind of time-costly problems have you experienced?

Youngsters	Middle Age	Elderly	Response
69.05 %	49.09 %	43.64 %	The guidelines contained many hard terms to understand
52.38 %	49.09 %	49.09 %	The guidelines were too long
45.24 %	21.82 %	16.36 %	I was not sure if I was eligible (if the administrative process was applicable to me)
35.71 %	27.27 %	10.91 %	The required documents (passport, driving license, birth certificate) were not clearly stated
4.76 %	14.55 %	9.09 %	Other

The most used channel to solve the most common problems depends on the age (see Table 6). Firstly, youngsters look up for a solution on the Internet as a main choice. The middle age group adds contacting with civil servants to the Internet browsing. Finally, the oldest group presents as a main choice a combination of asking a relative or a friend and Internet: asking somebody to find a solution through Internet because their lack of digital knowledge and experience.

Table 6 – When suffering from any of the previous problems, where did you find help?

Youngsters	Middle Age	Elderly	Response
45.24 %	34.55 %	40.21 %	I asked a relative/friend
40.48 %	41.82 %	30.91 %	I contacted with civil servants
69.05 %	41.79 %	45.45 %	I looked up for a solution on the Internet

Inside the participants who contact civil servants the most used channels are telephone and going to the Public Administration buildings. Moreover, a clear pattern can be seen: the older is the group the more digital is the chosen channel (Table 7).

Table 7 – Channel used to contact a civil servant

Youngsters	Middle Age	Elderly	Response
26.68 %	31.52 %	56.30 %	Going to the PA buildings
40.05 %	31.64 %	25.02 %	Telephone
6.67 %	5.27 %	6.16%	Internet
26.60 %	31.58 %	12.53 %	Other channel

Regarding collaborative tools to understand and model PA procedures, this kind of techniques are very useful. In particular, the younger is the citizen the clearer is a flowchart. However, less than 40% of the elderly participants do not fully understand the steps of a flowchart (Table 8). Consequently, a great amount of effort will be focused on the simplicity of the model.

Table 8 – In the flowchart, are you able to distinguish the internal procedure from the interactions with the citizen?

Youngsters	Middle Age	Elderly	Response
90.48 %	76.36 %	61.82 %	Yes, I understand who does each of the steps
9.52 %	23.64 %	38.18 %	No, I don't have a clue

In order to suggest enhancements, the favourite channel does not depend on the age (Table 9). The favourite one is leaving comments in the flowchart. Sending e-mails and posting questions are also frequent choices.

Table 9 – Imagine you were a citizen: you find something that should be improved or modified in the flowchart of a procedure. What would you do?

Youngsters	Middle Age	Elderly	Response
73.81 %	60.00 %	43.64 %	Leave a comment in the flowchart
26.19 %	25.45 %	34.55 %	Post a related question in the Q&A portal of Citizenpedia
23.81 %	27.27 %	25.45 %	Send a message/e-mail to the civil servant who created it

To sum up, the obtained feedback and the main collected comments (Figure 8) exposes several needs which have to be satisfied by the Citizenpedia requirements. The elderly group should be taken into account in terms of usability and the simplicity of the PA content. Furthermore, smartphones are the main interaction channel to perform tasks inside the digital world. As a result, Citizenpedia should take into consideration the limitations of these devices.

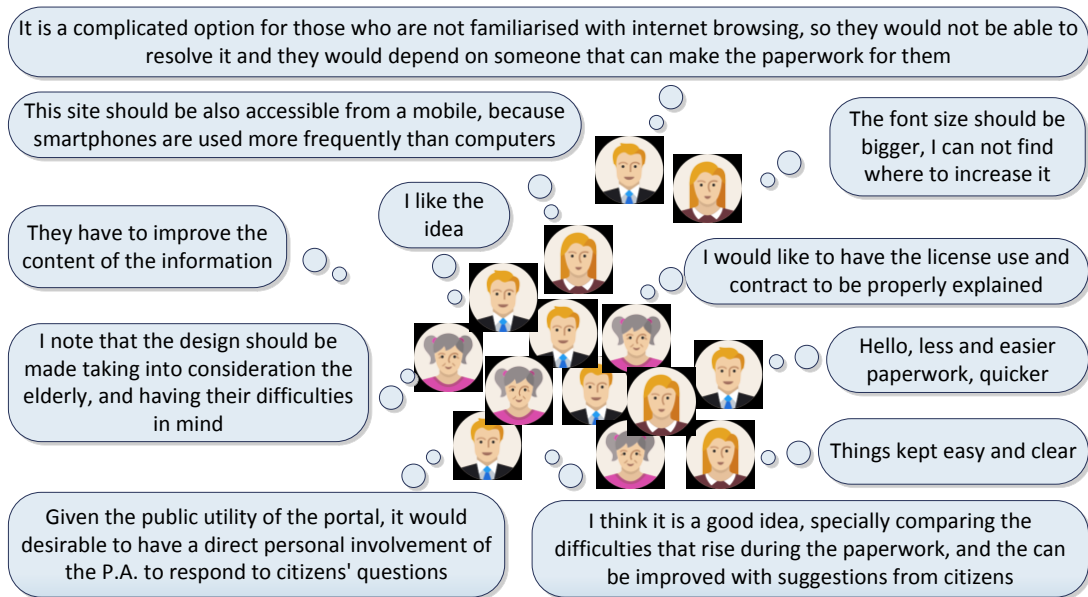


Figure 8 – Main citizens' comments

Civil servants' feedback poses a different point of view: most of their comments (Figure 9) described how the systems should be, with their stakeholders in mind. In particular, most the comments where gathered from civil servants in the Galicia region, where most PA users are elderly people. We could grab two main conclusions from these comments: first, the system should be simple and use symbols/pictures easy to recognize. A good note is how a civil servant encourages us to follow the already existing notation in physical public administrations, due to its ease of recognition. The second conclusions, beyond the IT tasks, is that we should invest time and effort in the training citizens, especially the elders, to bring them close to new technologies, in this case Citizenpedia.

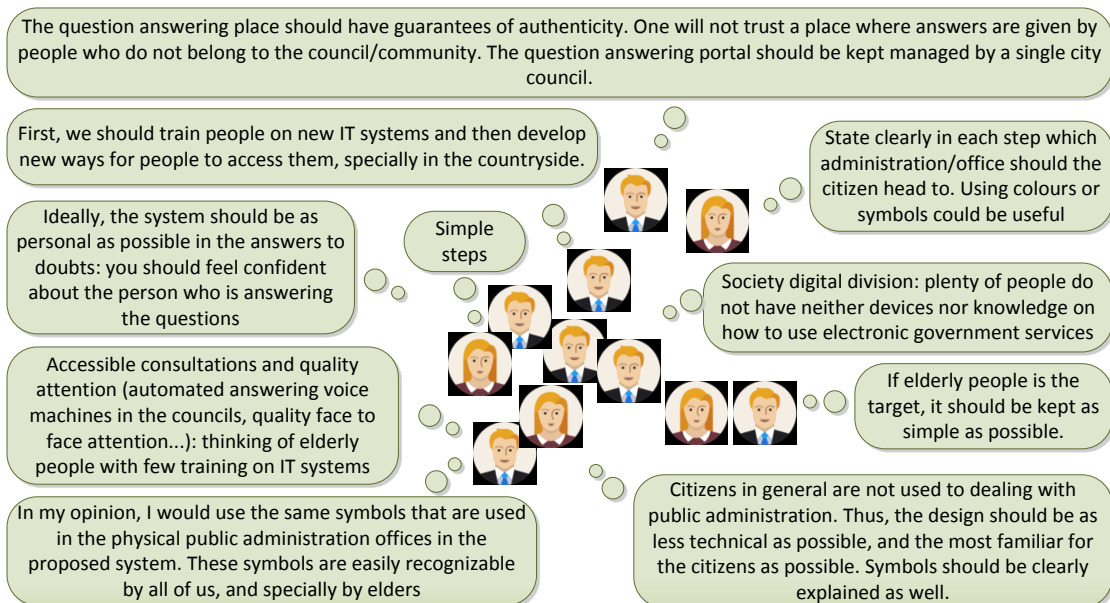


Figure 9 – Main civil servants' comments

3 Technology choice and usage description

The Citizenpedia will be built by combining different software pieces, each with a different functionality. Nowadays, the software community provides plenty of mature products/frameworks that could be used or adapted, and in this section we will review some of the most relevant ones. Due to some restrictions in the SIMPATICO project, we will focus mostly on open source tools.

This section is divided in two parts. First, we will review tools common to software engineering projects (e.g. databases) that could be integrated. Then, we will analyse participative solutions that could be extended.

3.1 Comparison of existing tools suitable to be integrated or extended

3.1.1 Databases

The database will be the software piece in charge of storing all the data generated in the Citizenpedia, from questions/answers to diagrams. There are different database mechanisms and vendors in the state of the art, and in this section we will discuss the most relevant ones.

Traditionally, databases have been relational (i.e. completely structured) and relied on using SQL (Structured Query Language) for its management. However, non-relational, NoSQL databases are gaining prominence as an alternative model for database management in the last years [1]. We will first raise a discussion on the benefits and drawbacks of each for our purposes.

SQL (Structured Query Language) is the standard programming language used to communicate with a relational database. It is used to manage, store, and retrieve data in relational databases through applications and queries either on the same computer or over a network. An SQL server consists of a relational database which comprises of a set of tables containing data with predefined categories or columns. A relational database matches data by using common characteristics found in the dataset and the resulting group is termed as Schema. There are dozens of SQL databases available, both commercial and free.

In these years of the “Big Data” era, the need to handle immense amount of data of different categories has become widespread, and the “one size fits all” approach of SQL is in question. This has led to the emergence of NoSQL, commonly referred to as “Not Only SQL”. With NoSQL, unstructured data can be stored across multiple processing nodes, without the need of fixed table schemas. The most popular NoSQL product in the community is MongoDB, an open-source document oriented database adopted by hundreds of enterprises and web sites. In July 2016, MongoDB was the fourth most widely mentioned database engine on the web, and the most popular for document stores [2].

Traditional SQL and NoSQL systems differ in some aspects:

- **Schema:** SQL databases have predefined schema whereas NoSQL databases have dynamic schema for unstructured data. E.g. new fields can be added to a table in a NoSQL database without any modification, what makes NoSQL more flexible across the time. However, this flexibility could make long term maintainability harder.
- **Scalability:** SQL databases are vertically scalable whereas the NoSQL databases are horizontally scalable. SQL databases are scaled by raising the horse-power of the hardware, i.e. increasing the RAM or hard drive capacity on a single server. In contrast, NoSQL databases are horizontally scalable: new servers can be easily added to a NoSQL database

infrastructure to handle the large traffic. In a nutshell, scaling up will mean getting a bigger box in SQL databases, and getting more boxes in a NoSQL environment.

- **Complex query handling:** Due to the flexible schema of NoSQL systems, the JOIN operator does not exist. JOIN allows the combination of multiple tables by common fields, and is a common operator in SQL databases. Due to the inexistence of this feature, queries that need to merge multiple data sources (different tables in SQL), do not fit properly in NoSQL environment.

In the case of the Citizenpedia, we consider that our data will be mostly structured, and we will probably need to perform queries that mix data from different sources. Thus, our choice will be to use a traditional relational SQL database.

The two most popular open source relational database management systems are MySQL and PostgreSQL [3]. Both have been under development for years and turned into the reference Relational Database Management Systems (RDBMS) in the community. Both are being used in hundreds of small projects, but also in big companies like Facebook or Twitter [4]. In the latter case, these companies contribute to the project with source code and documentation.

We conducted a literature review and we summarize here some of the most popular features that users look for when using a relational database:

- **ANSI (American National Standards Institute) compatibility.** Both are fully compatible with characters approved by the ANSI Institute that stand for an international standard.
- **ACID (Atomicity, Consistency, Isolation, Durability) compliant.** Both are fully compliant with the ACID properties.
- **Table changes without locking.** Both systems are able to change data of a table that is being read at the same time. This feature makes the database to operate faster.
- **Subqueries/Join capability.** When getting crossed data from various tables is needed, that is called a subquery or join. Both systems are fully capable of sub querying/joining tables, but PostgreSQL is able to do full outer joins (getting all the data from two given tables), and MySQL does not.
- **Licensing.** PostgreSQL has a MIT-style license that provides big freedom, including commercial use in open source or privative way. MySQL's client library is GPL licensed: MySQL users must pay a fee to Oracle or distribute publicly the source code of the entire application. As we are developing an open source application, both licenses are suitable.
- **User community.** Both MySQL and PostgreSQL have a large user community behind.

On this comparison, we can conclude that both MySQL and PostgreSQL fulfil all the requirements for the development of Citizenpedia.

3.1.2 Indexing engines

A search/indexing engine is a piece of software that makes content out from existing sources, such as databases or documents, and enables searching capabilities on them. They are used to provide an easy way to find content in large data sources, such as enterprise repositories or in web sites.

Nowadays, the most popular search engines are Apache Solr and Elasticsearch (ES). Both are based on Apache Lucene, a popular information retrieval library. Lucene has been under development since the mid 2000's and has been adopted by many popular web places as its indexing engine. However, Lucene does not provide parsing/crawling capabilities and alternatives like Solr/ES do. Thus, the latter have emerged as easier platforms to use for the same end.

In this section, we are focusing on Solr and ES due to their popularity and widespread adoption. Both offer similar features and are used for the same purposes, and we will conduct a brief comparison on them.

Apache Solr is currently developed by Apache, the company behind the most popular web server. Solr is a complete suite for enterprise search, including features as full-text search of faceted search. It is used by big companies as Netflix, SourceForge and eBay. It comes with a built-in GUI for its management and many plug-ins for document parsing. The community has developed some data visualization tools that work on top of Solr, such as Hue and Banana. They are used in conjunction with Solr to conduct data analytics.

ElasticSearch is a product developed by Elastic, and shipped as part of the ELK Stack: Logstash for data collection, ES for data storing/indexing, and Kibana for data visualization. Companies as Facebook, Github and Microsoft claim to be using ElasticSearch in their products. In comparison with Solr, ES is a lightweight download: is a light product that includes the essential functionalities. Other features, such as management web-console or a HTML crawler, must be installed as additional plug-ins.

In order to compare the functionalities of these products, we conducted two brief tests. We used a dataset of 1.5 GB size that contained books in different formats: HTML, Doc, PDF and ePub. We tried to achieve the same actions with Solr and ES, and compare their efficiency and ease of use.

In the first test, we made the engines (Solr and ES) to index each book one by one. This was achieved in both cases using a method of the REST API with JSON notation. We were able to index many books one by one, and prove that both engines gave the same results in terms of speed, efficiency and ease of use.

In the second test, we cleaned up both engines, and managed to index the entire dataset in a single step (and not book by book). In this test, we were able to do it with Solr out-of-the-box, but ES required the installation of a plug-in. Both products indexed all the books with no errors, and gave similar results in terms of speed.

After the two different document-indexing tests, we conducted several searches over the books. Both products worked seamlessly and returned similar search results for the out-of-the-box configuration. Additionally, we found that both Solr and ES prepared for scalability: both enabled to run multiple instances of the product, and every instance would recognize each other and work together out of the box.

3.1.3 Gamification engines

Gamification is the technique of applying game-like concepts in non-game environments. Gamification engines are the software pieces in charge of counting the “game” points per user of a system, and awarding badges/prizes/reputation skills. They are commonly used along with human computation platform, in order to increase user engagement. They will be used in the Citizenpedia as the main software piece for the Stakeholder Incentivization Techniques.

In this section, we provide an overview of some existing gamification engines in the community. There are some commercial ones (e.g. Playlyfe) but we limit our review to open source ones:

- Mambo.io is a SAAS (Software As A Service) solution. It is an open source product, but with commercial support. On the design side, it seems to be too oriented to shopping web sites, and seems difficult to adapt to our human computation framework (Citizenpedia).

- OpenBadges is product developed by Mozilla. It is oriented to educational platforms, such as web sites where users gain badges as they take courses. It is simple, as it does not include features as leader boards. It seems laborious to extend, due to the lack of basic gamification features it includes.
- UserInfuser is an open source gamification engine with many features such as badges, points, notifications and leader boards. However, it is no longer under development: the last update for the software was committed in 2011.
- The SmartCampus gamification engine is a software currently under development by the FBK research centre. As UserInfuser, it contains badges, points and a leader board. It provided a web-based UI for its management and also a REST API for programmatic accesses.

3.2 Comparative analysis of participative solutions

Understanding participative solutions inside the Citizenpedia framework as the practice of groups producing knowledge together through individual contributions, two main groups of participative solutions are taken into account: **Question and Answers Engines** as a method to augment the collective knowledge about e-services terms, doubts and procedures and the **Collaborative Process Modelling** as a method to clarify and design business models and processes related to e-services.

3.2.1 Question and Answers Engines as participative solutions

Upon the presented technologies, we present in this section an analysis of some already existing Q&A engines that could be used as a base for the development of the Citizenpedia. The state of the art presents several open source already existing software packages that could be reused, or used as reference. To that end, we reviewed several of them after searching in specialized forums.

3.2.1.1 Existing Questions and Answers Engines

We installed each of them in our own machines and conducted a series of brief tests to analyse their functionality. All of them have a common obvious set of features: the creation/update/deletion of questions and answers. Thus, in this section we will devote some lines to highlight the benefits and drawbacks we found in each solution, instead of summarizing all their features.

The first one we reviewed is Mamute, a Java-based engine built on top of VRaptor 4, an open source MVC framework for Java environments. Using Java as base platform provides stability and scalability at platform level, which makes the engine ready for large scenarios with lots of simultaneous queries. In addition, it is built with a minimal responsive UI: it detects the resolution of the screen where it is being displayed and moves the items accordingly, but it does not remove unnecessary elements or adapt the font size. It also comes with a basic gamification engine, that includes badges and reputation skill meters per user, and allow signing-up with Facebook or Google+.

On the cons side, Mamute was the slowest one in our tests: it had the largest latencies to load the different formularies in the engine. In addition, Mamute comes with no API to allow programmatic queries, and does not pose an easy way to create one. Furthermore, we found it difficult to modify and extend: it uses its own template language, instead of using a common one (e.g. Mustache or Twig). Finally, we found that the Mamute environment provides few technical documentation (on how to install and use it) and has a relatively small user community.

The second one we reviewed is Question2Answer (Q2A), an engine with a large community of users, as stated in their website: to the writing of this document they claim to have found 19,032 Q2A sites in 40 different languages. It is fully written in PHP and as opposed to Mamute, its UI runs fast in our

tests, i.e., we noticed no formulae loading latencies. It comes with a basic gamification engine, similar to the one that Mamute has, and from our tests it seemed easy to extend: it comes with several connectors that can be used to enable communication with 3rd party databases or handmade PHP scripts.

On the cons side, it does not provide a responsive interface out-of-the-box, and we noticed the absence of an API for external queries. We would like also to note that despite the large the number of sites based on Q2A, we found no user/developer community and few technical documentation on its deployment and use.

The third one is the engine we call BioStar. In contrast to the Mamute and Q2A, this engine was initially developed as a Q&A engine for a scientific community, and later its developers decided to make the source code available for free. The original BioStar community is a large website (called Biostars) with dozens of daily posts and the engine itself has been replicated for other scientific communities, such as the NeuroStars community.

Focusing on the engine itself, it is a complete Q&A platform based on the popular Django framework. It has lots of features implemented, such as basic gamification or a responsive UI that adapts the contents and aspect of the web page to the screen that is used. It also has some social functionalities, such as the capabilities for users to follow among themselves, and also includes an API to query contents, but just for reading and not modification. It seems easy to extend, and the use of Django makes it easy to find support on the development side on forums. To our criteria, it is the most similar engine to StackOverflow we have found.

We found not too many cons on this engine, being the most notable one the absence of a developer community. However, the large BioStar user community and the Django community would make it easy to find technical support.

The last engine we reviewed is the PaizaQA engine. In contrast to the previous ones, it is a half-made engine created with educational purposes: the source code is offered along with a one-hour tutorial to build a fully featured Q&A site using the MEAN software stack (MongoDB, Express MVC, Angular.js and Node.js). The result is a fully customizable working Q&A engine with an API to enable external access programmatically. Two strong advantages on PaizaQA is the big developer community behind the MEAN stack, and the ease of extension it exposes.

On the cons side, the greatest disadvantage we found is that, due to its original educational purpose, it is lacking some functionalities that other engines have, such gamification features.

3.2.1.2 Features comparison

As a summary of this section, we provide in the following tables the most relevant features per Q&A engine we collected. The first table shows some technical details and the homepage URL for the engine. The second one summarizes some features that will be used to decide the engine to use as starting point for the Citizenpedia. The columns on the second one are:

- API: Provides an API to enable the query of the content in the engine (questions, comments...).
- Responsive: The built-in template adapts the content to the used screen.
- Easy to extend: An estimation of the difficulty to add new features.
- Gamification: The quality of the gamification features included in the engine.

Table 10 – Q&A Engines features

Q&A Engine	Base language	Default database	Homepage
Mamute	Java	MySQL	mamute.org
Question2Answer	PHP	MySQL	question2answer.org
BioStar	Python	PostgreSQL	github.com/ialbert/biostar-central
PaizaQA	JavaScript stack	MongoDB	github.com/gi-no/paizaqa

Table 11 – Q&A Engines comparisons

Q&A Engine	Built-in API	Responsive UI	Easy to extend	Gamification	Community
Mamute	No	Limited	No	Limited	Poor
Question2Answer	No	No	Yes	Limited	Poor
Biostar	Read-only	Yes	Average difficulty	Limited	Good
PaizaQA	Yes	Yes	Yes	No	Good

3.2.2 Collaborative Process Modelling

Business process modelling is, by nature, a collaborative act that involves many stakeholders. In this section, we analyse and classify several modelling tools, with regards to their collaborative features for supporting the modelling task, aiming to elicit an understanding of the collaborative aspects of process modelling.

Business Process Management (BPM) is an approach for managing, transforming and improving organisational operations [5] [6]. An integral component of BPM is conceptual modelling, which aims to graphically represent the core processes within the organisation such that they can be analysed, improved and managed. The task of creating these models is generally accomplished in a collaborative manner [7], because different stakeholders are usually involved in the task.

While significant focus has been placed on research concerning process modelling, such as modelling grammars, and methods, little is known about the modelling activity itself, i.e. how people model, what the nature of the modelling task is and how to support people with collaborative tools in their modelling endeavours. Few studies to date have investigated collaborative process modelling, and those studies have exclusively been based on prototype implementations of tools and experimental research [7] [8] [9].

We based this analysis on some of the existing commercial software products, which have a strong focus on the process modelling component.

Our main aim is to learn about the collaborative nature of business process modelling by analysing how commercially available tools support collaborative modelling in practice. This way, we develop an initial understanding of collaborative process modelling tasks as understood by tool designers, and derive a high-level architecture (see Section 5.1.2) for supporting the collaborative aspects of the SIMPATICO CPD component.

3.2.2.1 Business Process Modelling

Conceptual modelling is an approach for visually representing selected phenomena in a certain domain for the purpose of designing an information system. Process modelling distinguishes itself from the “traditional” conceptual modelling (e.g. data modelling), by focusing on phenomena enacted by humans rather than machines [10]. Here, the modelling task is a matter of capturing, in a correct and fitting way, the workplace activities and their relationships as performed by human actors. In practice, business process modelling (or process modelling) is among the highest ranked purposes for which conceptual modelling is undertaken [11].

The most elaborate area of process modelling research in Information Systems is concerned with modelling grammars (languages, techniques or paradigm). Accordingly, existing research in the field often proposes new grammars or evaluates and improves existing ones (e.g. [12], [13], [14]).

Much of the past research on process modelling is normative in nature [15], as many works “propose new artefacts [e.g., grammars, tools, or methods], make claims on benefits and performance, and advocate adoption in practice based on an illustrative example” [16] (p. 44). Perhaps this is why some authors in the field have lamented the relatively small number of empirical and theoretical studies on process modelling (e.g. [17], [18]).

3.2.2.2 Collaborative Process Modelling

While a range of works in the BPM field propose novel modelling methods or grammars that allow for describing “collaborative processes” [19] – i.e. business processes that are inherently characterised by a high degree of collaboration – collaboration in the modelling task itself remains widely unaddressed. This situation is all the more remarkable given that processes, more often than not, span geographic and organisational boundaries and so does the modelling thereof. Process modelling increasingly occurs in distributed (e.g. cross-organisational, cross-geographical) contexts, thus presenting modellers – with different background, interpretations, knowledge, and skills – with various challenges [20]. Therefore, the study of the collaborative nature of process modelling warrants researchers’ attention.

While, to the best of our knowledge, no rigorous research exists that portrays the anatomy and collaborative nature of business process modelling as a task in practice, some research on tool support for modelling tasks exist, which we adopt as a starting point. For example, Pendergast et al. [21] created a prototype process drawing tool and investigated matters of awareness needs in the modelling process. Other authors have investigated the involvement of different people in a typical modelling workflow [22], derived normative guidelines for tool use in joint modelling [23] or developed and tested certain tools for joint modelling [24].

3.2.2.3 Tool Evaluation

Overall, the support for collaborative modelling tasks, across our sample of tools, was surprisingly low and rather fragmented. No product stands out that provides adequate collaboration support with a wide range of features or good integration with the actual modelling editor. This is particularly true when comparing the analysed tools with the features provided by dedicated collaboration systems and other editors for collaborative work, e.g. for collaborative writing [25]. While no tool provides comprehensive support, each of the seven tools discussed in the following subsection provides some interesting features or follows a particular approach to facilitate collaboration in process modelling. Hence, while we did not find a good stand-alone collaborative modelling tool

among the established BPM solutions in the market, if we consider all tools in our samples taken together we begin to see a potential architecture for a collaborative process modelling suite.

3.2.2.4 Reviewed tools and their collaborative support

In the following, we provide details for each of the seven evaluated tools:

CA ERwin Process Modeller [26]. It provides a Visual Diagram Compare function, which aims to support conflict resolution, when modelling in parallel. To view differences, a colour shade is assigned to any altered diagram objects. It is possible to only show substantial changes, so that pure visual changes (e.g. repositioning of elements) are not shown. Furthermore, ERwin supports scripts with configurable criteria that can be used to find changes in the provided change list. Automatic generation of change and conflict reports furthermore complement the feature. All in all, these features support the coordination of concurrent modelling and provide awareness for changes made by other users.

ARIS Design Platform [27]. The philosophy of this tool regarding collaborative process modelling is a role-based modelling approach, where each user can access a central model repository in order to participate in the modelling process. To this end, ARIS Business Designer incorporates a change management module. Proposals for change and improvement can be made for all objects and models. Each user can directly enter their proposal with or without consulting the process manager. An improvement manager can then review the proposals and set priorities, status and persons responsible for implementing the measure. If changes are accepted, the assigned modeller will receive a modelling task in his task list. Moreover, ARIS provides functionality for displaying models on a website for sharing with relevant stakeholders, while a (public) web-based discussion board is open to all modellers, but mainly for general discussions. With the WYSIWG report builder a modeller can report about progress and send reports to colleagues for approval or commenting.

Sparx Systems Enterprise Architect [28] support the team in posting and responding to comments linked with process elements. Via the resource management module the allocation of tasks with different effort weights is possible. It is possible to define searches for changes, which supports collaborative modelling by enabling different roles to receive change reports as soon as something particular has changed and reaction is needed. Also, Enterprise Architect has a comprehensive and powerful differencing utility, which allows comparison of a model branch with a base-line model. If there are conflicting versions, automatic conflict solving is offered for simple cases and manual conflict solving otherwise.

iGrafx Process Modeller [29] provides phase management, called the “document approval process flow”. Every repository in the so-called Process Central module has a reviewer group, an approval group and an endorsement group. Within the review cycle users can review and annotate the models. In the approval cycle users approve the current model and in the endorsement cycle the users certify their understanding of an agreement with the model.

Microsoft Visio [30] lets several people work on a single diagram at the same time by uploading it to SharePoint or OneDrive, showing to the team what shapes are being edited in real time. Comments are grouped in threads and through the “commenting pane” it is possible to add, read, reply and keep track of reviewers’ comments.

IBM WebSphere Business Modeller Advanced [31]. In this tool the modelling group is supported by a BPM phase management functionality. A predefined set of roles allows task sharing between a leader, an architect or senior business analyst and a publisher. The publisher role is especially needed if a publishing server is used. In that case the person is in charge of moderating the discussions and

managing the content on the server. The publishing server displays projects using a process portal and handles comments that reviewers can enter as feedback.

Signavio Process Editor [32] is a web-based tool. One of its strength point is the sharing of models. Besides normal invitations for registered users, it is also possible to grant read-only access to diagrams for external stakeholders or to embed models into web-based systems like websites or wikis. This feature allows a larger number of contributors to be involved in reviewing and commenting on a model. Workspace awareness is created by offering automatic notification emails if a predefined model changes. Alongside the normal version comments, visual previews are shown to the user while they can browse through the distinct versions. Discussions are facilitated based on the version comments.

Summarised evaluation results are shown in the following table: CA ERwin Process Modeller (CAE), ARIS Design Platform (ARI), Sparx Systems Enterprise Architect (SPA), iGrafx Process Modeller (IGR), Microsoft Visio (MSV), IBM WebSphere Business Modeller Advanced (IBM) Signavio Process Editor (SIG).

Table 12 – Summarize Evaluation

		CAE	ARI	SPA	IGR	MSV	IBM	SIG
Process Modelling Notation	EPC		√			√		√
	BPMN		√	√	√	√	√	√
	IDEF	√			√	√		
	UML		√	√		√		√
Process Modelling Features	Asynchronous Modelling	√	√	√	√	√	√	√
	Concurrent Modelling	√	√	√	√		√	
	Synchronous Modelling							
	Templates definable	√	√	√	√	√	√	
	Framework Support	√	√	√			√	
	Phase Management			√			√	
	Correctness Checker	√	√		√	√	√	√
Commenting and Annotations	Model Comparison	√	√	√			√	√
	Links: Processes	√	√	√	√	√	√	√
	Links: Files		√	√	√	√	√	√
	Links: Web Resources		√	√	√	√	√	√
	Glossary Support	√	√	√	√		√	√
	Discussion Board		√	√				
	Member List							
	Comments: Element Level	√	√	√	√	√	√	√
	Comments: Process Level	√	√	√	√		√	√

User and Role Management	User Management	√	√	√	√		√	√
	Role Management	√		√	√		√	
	Support of Task Sharing	√	√	√			√	
	Workspace Awareness	√		√				√
Repository and Conflict Management	Repository local	√	√	√	√		√	
	Repository remote	√	√	√	√		√	√
	Version Control	√	√		√		√	
Architecture	Client/Server	√	√	√	√		√	
	Client only					√		
	Web-based							√
Import	File: Own	√	√	√	√	√	√	√
	File: PDF	√	√		√	√	√	√
	File: XPDL	√	√		√	√	√	√
	File: BPMN	√	√	√	√	√	√	√
	File: HTML	√	√	√	√	√	√	√
	File: PPT				√	√		
	File: Word/RTF		√	√	√	√	√	
	File: XLS				√	√	√	√
	Graphic: BMP			√	√	√		
	Graphic: JPEG	√		√	√	√	√	
	Graphic: PNG			√	√	√		√
Export	File: Own	√	√	√	√	√	√	√
	File: XPDL	√	√		√	√	√	√
	File: BPMN	√	√	√	√	√	√	√

3.3 Selection

According to the conducted analysis, we list in this section the technologies that will be used as starting point for the development of Citizenpedia.

Database: MySQL. It is a popular open source database developed by Oracle, with a large user community. It provides a good performance, and it is easy to use and integrate with major programming languages and frameworks. In addition, DEUSTO has used MySQL in previous projects. Thus, the learning and adoption time for MySQL in Citizenpedia will be short.

Indexing engine: Elasticsearch. It has a large user community and at the time of writing this document, ES is the most popular enterprise search engine. It will work seamlessly out of the box, it



is ease of use, it has a flexible plug-in extension system and it is lighter than Apache SolR. In addition, it is shipped with official client libraries for almost every programming language (SolR does so only for Java).

Gamification engine: SmartCampus engine. It is the most complete open source gamification engine project currently active. It is easy to use and manage, thanks to its UI and REST API. In addition, it is maintained by FBK, a partner in the SIMPATICO project.

Question answering engine: PaizaQA. It is the already existing QA engine that fulfil more features than the other reviewed ones. In addition, it is constructed using the MEAN JavaScript stack, a living user community. The only drawback is the absence of integrated gamification engine. This issue will be solved with the integration of the SmartCampus gamification engine.

Collaborative procedure designer: No tool will be initially extended. From the conducted analysis, we foresee that it will be easier and faster to develop a tool from scratch than extending one of the reviewed ones. However, we will have in mind the described tools and add every relevant feature learned from the review to the Citizenpedia CPD.

4 Citizenpedia requirements specifications

The following subsections list what will be the functional and non-functional requirements associated to the main components of the envisaged Citizenpedia framework, which is structured in the four following components:

- **Social question answering engine (QAE).** This component is made up by a responsive web interface to enable the interaction of users (citizens and civil servants) from any type of device, e.g. smartphone or PC. Techniques to promote users, questions and answers are enabled, so that the most popular questions and answers associated to them are promoted by social opinion.
- **Collaborative procedures designer (CPD).** This tool allows SIMPATICO stakeholders to enrich the collective knowledge with a graphical representation of public procedures, whose tasks represent both e-services and (non digital) services to be performed by citizens to achieve a specific goal. Thanks to this component, civil servants will be able to initially model and publish public administration procedures, implemented by both digital and no digital services, while citizens could enrich procedures diagrams with feedback and annotations.
- **Collective knowledge base (CKB).** This component manages and stores the Collective Knowledge base generated by Citizenpedia. It defines the data model of such knowledge base, including entries, contributors, and revision entities among many others. Besides, it will undertake the integration of such knowledge base with the other two components of Citizenpedia that feed information to it, i.e. the question/answer engine and the collaborative procedures designer.
- **Stakeholder Incentivization techniques (SIT).** This component offers gamification techniques and mechanisms in order to incentivize people to take an active part across the Citizenpedia framework.

The requirements listed in the following sub-sections are divided according to the listed architectural elements (components). As is described in Section 2, for the generation of these requirements several tasks to follow a top-down approach were performed. According to the Volere template detailed in the "D5.1-SIMPATICO platform requirements and architecture" deliverable, two main fields were filled from the Stakeholders perspective (provided by the analysis described in Section 2.2) and the rest from the technical point of view. The following table lists the two main perspectives and their corresponding fields.

Table 13 – Perspectives and Requirement fields mapping

Perspective	Volere Fields
System Perspective	Name Requirement Type Description Rationale Fit Criterion (Measurable) Priority Conflicts Actors Author Revision
Stakeholders Perspective	Customer satisfaction Customer dissatisfaction

After defining the main requirements (described in the following sections) the main actors and roles who are involved in the Citizenpedia framework are detailed. According to the identified requirements, these **actors** were recognised.

- **Citizen:** this is a public services customer, so he/she can use the Citizenpedia framework to solve his/her own doubts and to contribute to the collaborative knowledge. His/her contributions can be performed through the Question Answering Engine and the Collaborative Procedure Designer.
- **Civil Servant:** this is a public services employee, so he/she can use the Citizenpedia framework to help to solve the citizens' doubts and to contribute to the collaborative knowledge and the correct usage of the Citizenpedia framework. They can not only perform several tasks related to the management of the Question Answering Engine component, but also to the creation of the diagrams of procedures and management of the Collaborative Procedure Designer.

Two major **roles** can be adopted by the defined actors.

- **Guest:** it is an unauthenticated visitor of the Citizenpedia framework. She/he will have limited access to Citizenpedia functionalities (they can use the Citizenpedia knowledge but they cannot enhance it).
- **Authenticated user:** it is an abbreviation of "Logged in User" and represents the general type of actor in the Citizenpedia framework (they not only use the Citizenpedia knowledge, but also they enhance it).

Two main groups of roles can be distinguished inside the authenticated users who can perform different tasks related to their specific role.

The first group corresponds to the **Question Answering Engine (QAE) component** users.

- **Q&E Users:** users who have been authenticated and can perform several tasks related to create and consume information managed by Citizenpedia such as manage their own questions, answers, comments and terms definitions.
- **Q&E Moderator:** this role can use the Citizenpedia framework to manage and monitor the correct usage of the Citizenpedia framework. They are authenticated users who can perform several tasks related to the management of the QAE component, such as the management of questions, answers, comments and terms, as well as the content and the roles of the rest of moderators. Q&E Moderators would have obtained privileges to conduct more actions than citizens by contributing to Citizenpedia and gaining expertise points. Upon a certain threshold of points, they can be promoted to moderators by other moderators.

The second group corresponds to the **Collaborative Procedure Designer component** users.

- **Procedure owner:** it is a civil servant and also authenticated user who is responsible for the entire administrative procedure. It is also accountable for the implementation of the procedures and their subsequent improvements. It has to be an expert of the procedure domain, but does not have to be a technology expert. It appoints the Procedure designer, the Procedure analyst, and any potential Procedure stakeholders for each of its owned procedures.
- **Procedure designer:** it is an authenticated user and an expert in modelling and drawing procedures.

- **Procedure analyst:** it is an authenticated user and an expert in the design and improvement of procedures. It can review administrative procedure diagrams but cannot directly modify them. Its reviews can be sent back to the Procedure designer for further improvements, or sent forward to the Procedure owner for the final approval.
- **Procedure stakeholder:** it is a civil servant or a citizen who is not necessarily an expert but is affected by the proposed procedures. He/she cannot modify procedures but he/she can view and suggest changes.

According to the defined actors and roles, the following figure exposes the discussed roles hierarchy.

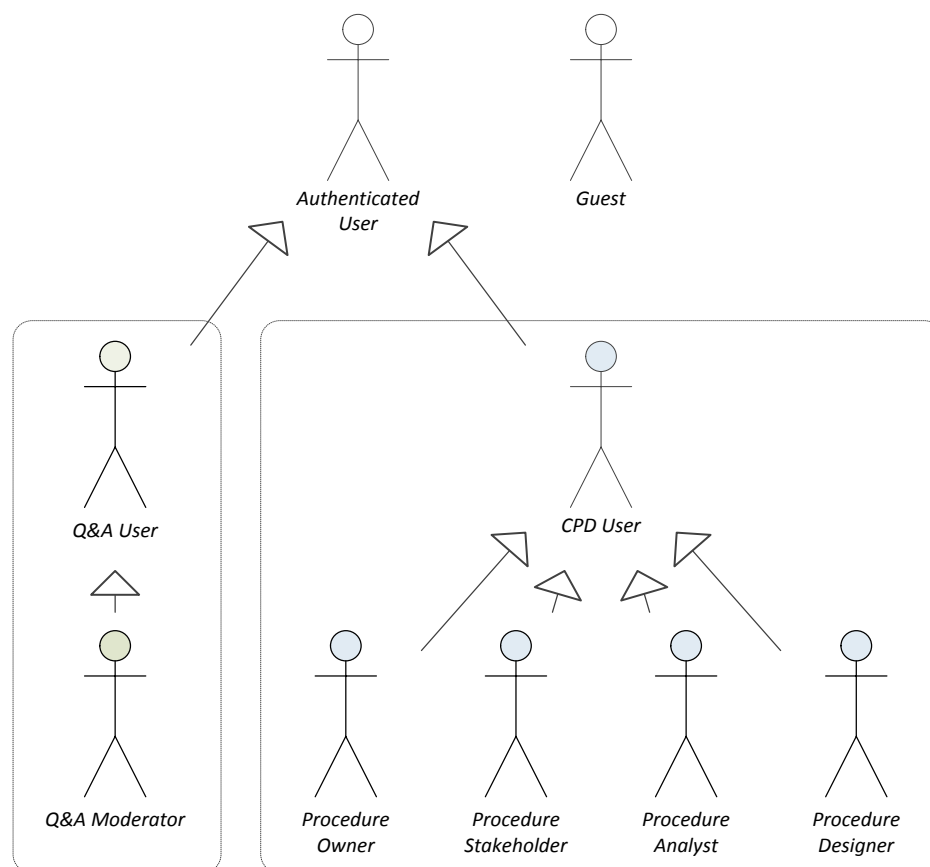


Figure 10 – Roles hierarchy of Citizenpedia actors

The Citizenpedia macro-requirements which have been extracted through the previously described methodology are mainly four. Below a brief description of them is exposed according to the Volere template detailed in the D5.1 deliverable.

Table 14 – Macro-requirement QAE

ID	QAE
Name	Enable the posting and resolution of questions and terms with ranking capabilities
Requirement Type	Functional

Description	Users can ask, answer questions and search terms and definitions. Furthermore, specific users can manage and maintain the quality of the generated knowledge through ranking and moderator tasks.
Rationale	Users will have a tool to ask and offer clarifications and interpretations that it would be impossible to derive automatically, like what is the meaning of a code of law, or the implications of a certain regulation.
Fit Criterion (Measurable)	Users can ask, answer questions and search terms and definitions. They can also rank and ensure the quality of the generated knowledge.
Customer satisfaction	3 (Scale from 1=uninterested to 5=extremely pleased).
Customer dissatisfaction	5 (Scale from 1=hardly matters to 5=extremely displeased).
Priority	4 (Scale from 1=low priority to 5=highest priority).
Conflicts	Although no other requirement is blocked by this one, it provides a key feature for the Citizenpedia framework
Actors	Citizens and Civil Servants
Author	DEUSTO Team
Revision	V1.0, 24/05/2016

Table 15 – Macro-requirement CPD

ID	CPD
Name	Enable the collaborative modelling and editing of e-services in the form of flowcharts
Requirement Type	Functional
Description	Users can start collaboration workflows to collaboratively design, review and approve administrative procedures.
Rationale	Users will have a tool where civil servants can model public administration procedures, implemented by both digital and non-digital services, while citizens can enrich procedures diagrams with feedback and annotations
Fit Criterion (Measurable)	Users can collaboratively model, review, edit and enhance public administration procedures.
Customer satisfaction	3 (Scale from 1=uninterested to 5=extremely pleased).
Customer dissatisfaction	5 (Scale from 1=hardly matters to 5=extremely displeased).
Priority	4 (Scale from 1=low priority to 5=highest priority).

Conflicts	Although no other requirement is blocked by this one, it provides a key feature for the Citizenpedia framework.
Actors	Citizens and Civil Servants
Author	DEUSTO Team
Revision	V1.0, 24/05/2016

Table 16 – Macro-requirement CKB

ID	CKB
Name	Manage the information generated by QAE and CPD
Requirement Type	Functional
Description	All the knowledge generated by citizens and civil servants using the Citizenpedia framework will be correctly managed, stored and exposed.
Rationale	The collective knowledge database on public services and procedures is extremely valuable not only for e-services, but in general for enhancing the transparency and accessibility of the public sector. This knowledge will be released as a public domain resource co-created and co-operated by users. Consequently, it will be correctly managed.
Fit Criterion (Measurable)	Knowledge generated by the QAE and the Collaborative Procedure Designer is stored and correctly exposed.
Customer satisfaction	3 (Scale from 1=uninterested to 5=extremely pleased).
Customer dissatisfaction	5 (Scale from 1=hardly matters to 5=extremely displeased).
Priority	5 (Scale from 1=low priority to 5=highest priority).
Conflicts	The QAE and CPD requirements are blocked by this one.
Actors	Citizens and Civil Servants
Author	DEUSTO Team
Revision	V1.0, 24/05/2016

Table 17 – Macro-requirement SIT

ID	SIT
Name	Incentivize users with gamification techniques
Requirement Type	Functional
Description	Techniques to promote the collaboration and enhancement of citizens and

	civil servants will be used through a well-known approach for effectively achieve human computation: gamification
Rationale	One of the main issues in collaborative platform is the users' engagement. An award mechanism that will engage users and incentivize them to collaborate by giving them reputation (a valuable asset for professionals and organizations) and privileges will be designed.
Fit Criterion (Measurable)	Users can obtain rewards and promotion according to their contributions and participation.
Customer satisfaction	4 (Scale from 1=uninterested to 5=extremely pleased).
Customer dissatisfaction	3 (Scale from 1=hardly matters to 5=extremely displeased).
Priority	4 (Scale from 1=low priority to 5=highest priority).
Conflicts	No other requirement is blocked by this one
Actors	Citizens and Civil Servants.
Author	DEUSTO Team
Revision	V1.0, 24/05/2016

Increasing the granularity of the requirements in order to obtain a more explicit and in depth approach, for each macro requirement more detailed ones were extracted (Figure 11).

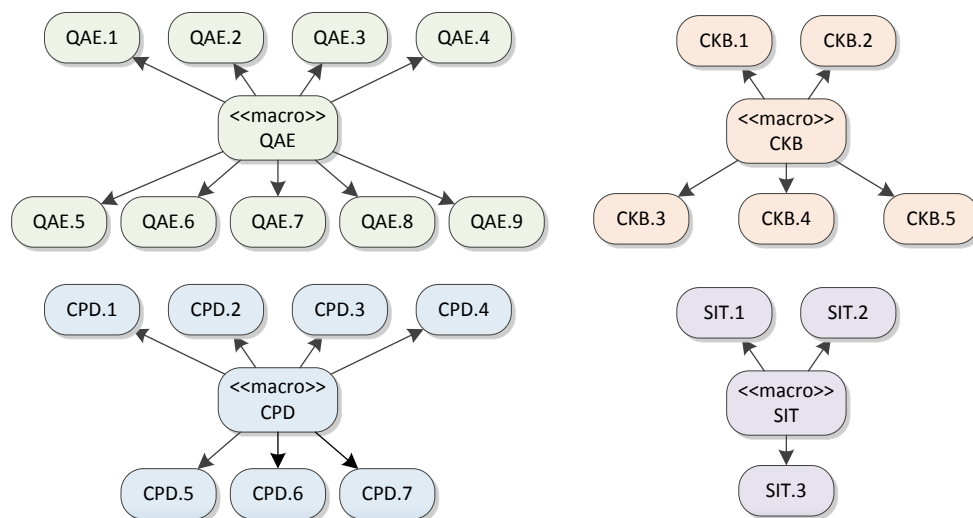


Figure 11 – Relation between macro-requirements and requirements of Citizenpedia

4.1 Social question answering engine requirements (QAE)

Table 18 – Requirement QAE.1

ID	QAE.1
Name	Creation, modification and deletion of questions

Requirement Type	Functional
Description	Users must be able to create and manage questions. Each new question should need a title and a description. Optionally, tags (topics/concepts) can be defined that describe the question in few words. A user can modify questions that he has already created, changing its title, description or tags. He can also delete a previously created question.
Rationale	This requirement is core to the QAE. It is the way to populate this part of the Citizenpedia: users shall post questions regarding their doubts in their use of public procedures.
Fit Criterion (Measurable)	Users can create questions and later modify/delete them. Moderators can modify/delete questions of other users if they are allowed to do so.
Customer satisfaction	3 (Scale from 1=uninterested to 5=extremely pleased).
Customer dissatisfaction	5 (Scale from 1=hardly matters to 5=extremely displeased).
Priority	5 (Scale from 1=low priority to 5=highest priority).
Conflicts	The following requirements are blocked by this one: QAE.2
Actors	Citizens and Citizenpedia moderators. The formers can create questions and modify/delete their own ones. The latter ones can modify and delete other citizen's questions.
Author	DEUSTO Team
Revision	V1.0, 24/05/2016

Table 19 – Requirement QAE.2

ID	QAE.2
Name	Creation, modification and deletion of answers
Requirement Type	Functional
Description	A user can post a text as an answer to an existing question. This user can later modify the text of this answer, or delete it.
Rationale	As well as QAE.1, this requirement is core to this part of Citizenpedia: along with creating questions, creating answers is the way to populate the QAE, making it useful for their users.
Fit Criterion (Measurable)	Users can create answers and later modify/delete them. Moderators can modify/delete answers of other users if they are allowed to do so.
Customer satisfaction	3 (Scale from 1=uninterested to 5=extremely pleased).

Customer dissatisfaction	5 (Scale from 1=hardly matters to 5=extremely displeased).
Priority	5 (Scale from 1=low priority to 5=highest priority).
Conflicts	No other requirement is blocked by this one, although it does fulfil a clear need to populate the QAE.
Actors	Citizens and Citizenpedia moderators. The formers can create answers and modify/delete their own ones. The latter ones can modify and delete other citizen's answers.
Author	DEUSTO Team
Revision	V1.0, 24/05/2016

Table 20 – Requirement QAE.3

ID	QAE.3
Name	Upvote or downvote an answer
Requirement Type	Functional
Description	A user marks an answer as useful (upvote) or not useful (downvote). The number of upvotes and downvotes are accumulated and a certain score is calculated per answer.
Rationale	The resulting scores (computed from the upvotes and downvotes) enables to rank the answers for a given question by relevance according to the citizens. Furthermore, the number of upvotes and downvotes of the answers will be a key factor to measure the answers quality, as well as the involvement (participation) of each user. Consequently, these quantifications will be used to measure the reputation of the users of Citizenpedia in order to grant privileges to moderate the site and ensure its governance.
Fit Criterion (Measurable)	Users can upvote and downvote answers. Answers for each question are ranked according to a score calculated from upvotes/downvotes.
Customer satisfaction	4 (Scale from 1=uninterested to 5=extremely pleased).
Customer dissatisfaction	3 (Scale from 1=hardly matters to 5=extremely displeased).
Priority	4 (Scale from 1=low priority to 5=highest priority).
Conflicts	No other requirement is blocked by this one, although it provides a useful feature for the Citizenpedia users.
Actors	Citizens
Author	DEUSTO Team

Revision	V1.0, 24/05/2016
-----------------	------------------

Table 21 – Requirement QAE.4

ID	QAE.4
Name	Create, modify or delete a comment to a question or answer
Requirement Type	Functional
Description	A user can post a text as a side comment to an existing question or answer, and modify or delete a previously done comment.
Rationale	In contrast to answers, comments should be used to clarify details to a particular answer/question, and not to provide an answer to the original question itself.
Fit Criterion (Measurable)	Users can create comments on existing questions and answers.
Customer satisfaction	3 (Scale from 1=uninterested to 5=extremely pleased).
Customer dissatisfaction	4 (Scale from 1=hardly matters to 5=extremely displeased).
Priority	3 (Scale from 1=low priority to 5=highest priority).
Conflicts	No other requirement is blocked by this one, although it provides a useful feature for the Citizenpedia users.
Actors	Citizens and moderators. The formers can create comments and modify/delete their own ones. The latter ones can modify and delete other citizen's comments.
Author	DEUSTO Team
Revision	V1.0, 24/05/2016

Table 22 – Requirement QAE.5

ID	QAE.5
Name	Promote a user to moderator
Requirement Type	Functional
Description	User is given more permissions within the QAE, e.g., he is allowed to modify or delete other users' questions or answers. This increase in the permissions is given by another moderator, according to a suggestion performed by the gamification engine.
Rationale	Users' activity in the QAE portal (e.g. creation and answering of questions)

	is monitored and recorded by the gamification engine. The gamification engine assigns points to these actions. When a user reaches certain level of points, he is able to gain more permissions.
Fit Criterion (Measurable)	A user can be promoted to moderator, gaining permissions to conduct further actions than he could previously.
Customer satisfaction	5 (Scale from 1=uninterested to 5=extremely pleased).
Customer dissatisfaction	3 (Scale from 1=hardly matters to 5=extremely displeased).
Priority	3 (Scale from 1=low priority to 5=highest priority).
Conflicts	No other requirement is blocked by this one, although it provides a way to incentivize citizens giving them more control.
Actors	Moderators.
Author	DEUSTO Team
Revision	V1.0, 24/05/2016

Table 23 – Requirement QAE.6

ID	QAE.6
Name	Store and manage definitions/explanations of public-administration terms
Requirement Type	Functional
Description	Users must be able to create and manage definitions and terms related to the public-administration field. Each new term should need an explanation and related tags. A user can modify not only the definition of a term that he has already created, but also the definitions created by the rest of users. Each modification will be tracked by a changes control system.
Rationale	Due to the complexity of public procedures and the related documents, Citizens have several problems interpreting terms and definitions.
Fit Criterion (Measurable)	Users can create definitions and terms and also edit them. These changes should be tracked.
Customer satisfaction	3 (Scale from 1=uninterested to 5=extremely pleased).
Customer dissatisfaction	5 (Scale from 1=hardly matters to 5=extremely displeased).
Priority	5 (Scale from 1=low priority to 5=highest priority).
Conflicts	This requirement presents a main functionality of the Citizenpedia. The following requirements are blocked by this one: CKB.2 and CKB.3
Actors	Citizens and Civil Servants

Author	DEUSTO Team
Revision	V1.0, 27/06/2016

Table 24 – Requirement QAE.7

ID	QAE.7
Name	Low quality/spam entries monitoring
Requirement Type	Functional
Description	Moderators receive notifications about suspicious content in questions, answers and comments.
Rationale	Spam in collaborative websites is done by posting (usually automatically) random comments, copying material from elsewhere that is not original, or promoting commercial services to blogs, wikis, guest books, or other publicly accessible online discussion boards. This kind of behaviour decreases the quality of the generated knowledge as well as the number of visitors. Any web application that accepts and displays hyperlinks submitted by visitors may be a target. Consequently, Citizenpedia is a potential target.
Fit Criterion (Measurable)	This component should be able to notify suspicious content to the moderators.
Customer satisfaction	2 (Scale from 1=uninterested to 5=extremely pleased).
Customer dissatisfaction	4 (Scale from 1=hardly matters to 5=extremely displeased).
Priority	3 (Scale from 1=low priority to 5=highest priority).
Conflicts	No other requirement is blocked by this one within the Citizenpedia.
Actors	Moderators
Author	DEUSTO Team
Revision	V1.0, 24/05/2016

Table 25 – Requirement QAE.8

ID	QAE.8
Name	Access control to QAE items
Requirement Type	Non-functional
Description	This requirement expresses the need to ensure that the access to information and functionalities offered by the QAE, building blocks and

	datasets that are of private nature is restricted to the users that have the permission to access these information and functionalities. This includes the capability for moderators to control and modify the content of this component and the capability of all the logged users to control their own contents.
Rationale	In order to ensure the quality and consistency of the information generated and managed by the Citizenpedia framework, the access should be controlled.
Fit Criterion (Measurable)	Citizens can manage their own content and only moderators can perform content management tasks which are not generated by their own.
Customer satisfaction	3 (Scale from 1=uninterested to 5=extremely pleased).
Customer dissatisfaction	4 (Scale from 1=hardly matters to 5=extremely displeased).
Priority	4 (Scale from 1=low priority to 5=highest priority).
Conflicts	All the requirements which are related to the citizens and moderator tasks must comply with access control constraints.
Actors	Moderators and citizens
Author	DEUSTO Team
Revision	V1.0, 24/05/2016

Table 26 – Requirement QAE.9

ID	QAE.9
Name	Responsive user interface
Requirement Type	Non-functional
Description	The QAE will use different CSS style rules based on characteristics of the device the site is being displayed on.
Rationale	According to the heterogeneity of devices, a responsive interface will be addressed to provide an great viewing and interaction experience, easy reading and navigation with a minimum of resizing, panning, and scrolling, across a wide range of devices (from desktop computer monitors to mobile phones).
Fit Criterion (Measurable)	QAE adapts its content depending on the used device
Customer satisfaction	5 (Scale from 1=uninterested to 5=extremely pleased).
Customer dissatisfaction	3 (Scale from 1=hardly matters to 5=extremely displeased).

Priority	3 (Scale from 1=low priority to 5=highest priority).
Conflicts	No other requirement is blocked by this one, although it provides a way to incentivize citizens giving them more control.
Actors	Moderators.
Author	DEUSTO Team
Revision	V1.0, 24/05/2016

4.2 Collaborative procedures designer requirements (CPD)

Table 27 – Requirement CPD.1

ID	CPD.1
Name	Start a collaboration workflow
Requirement Type	Functional
Description	<p>In the case a new administrative procedure has to be created from scratch, the Procedure Owner selects the collaboration workflow to use from a set of collaboration workflow templates, then configures the administrative procedure settings and eventually starts the collaboration.</p> <p>In Figure 12, an example of collaboration workflow template is shown.</p> <p>In the case an existing administrative procedure has to be modified, the Procedure Owner starts the collaboration on the existing procedure.</p>
Rationale	To be able to set-up a collaborative workflow, to engage the stakeholders of the administrative procedure and to trigger the designing phase.
Fit Criterion (Measurable)	Each Procedure Owner should be able to start a collaboration either from the available templates or on an existing administrative procedure, and to input/update the administrative procedure collaborative design settings.
Customer satisfaction	5 (Scale from 1=uninterested to 5=extremely pleased).
Customer dissatisfaction	5 (Scale from 1=hardly matters to 5=extremely displeased).
Priority	5 (Scale from 1=low priority to 5=highest priority).
Conflicts	ALL the remaining requirements
Actors	Procedure Owner
Author	BEng Team
Revision	V1.0, 24/05/2016

Table 28 – Requirement CPD.2

ID	CPD.2
Name	Terminate a collaboration workflow
Requirement Type	Functional
Description	The Procedure Owner selects the collaboration workflow to terminate and confirms the termination.
Rationale	To be able to terminate an obsolete or stuck collaboration workflow instance.
Fit Criterion (Measurable)	Each Procedure Owner should be able to terminate a collaboration from all of his own collaboration workflow instances.
Customer satisfaction	4 (Scale from 1=uninterested to 5=extremely pleased).
Customer dissatisfaction	5 (Scale from 1=hardly matters to 5=extremely displeased).
Priority	4 (Scale from 1=low priority to 5=highest priority).
Conflicts	No other requirement is blocked by this one.
Actors	Procedure Owner
Author	BEng Team
Revision	V1.0, 24/05/2016

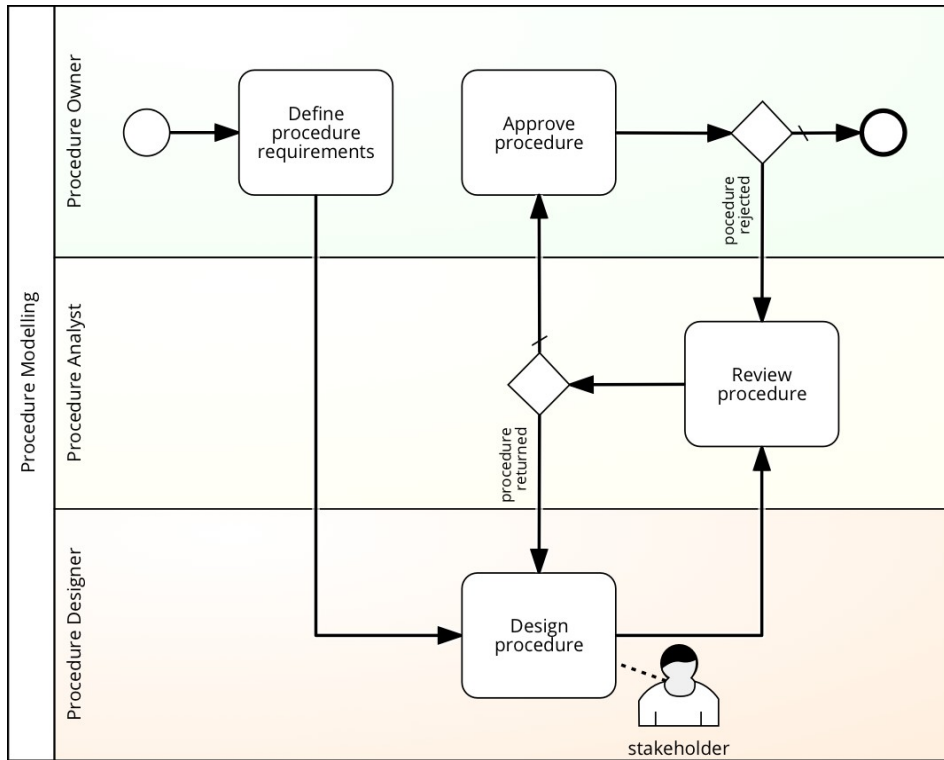


Figure 12 – Example of collaboration workflow

Table 29 – Requirement CPD.3

ID	CPD.3
Name	Collaborative design of an administrative procedure
Requirement Type	Functional
Description	Represents the “Design procedure” task of the collaboration workflow instance, that is, the graphical design operated by the Procedure Designer in collaboration with all the Procedure Stakeholders. The Procedure Designer has the responsibility to commit the modelled procedure once the design phase is complete.
Rationale	Define and document an administrative procedure in a collaborative way. The collaboration involves all the administrative procedure stakeholders.
Fit Criterion (Measurable)	The Procedure Designer should be able to draw the diagrammatic representation of the administrative procedure and all the Procedure Stakeholders (including the Procedure Designer) should be able to collaborate to the task (i.e., post comments, attach documents, etc.). Only the Procedure Designer should be able to commit the modelled procedure.
Customer satisfaction	5 (Scale from 1=uninterested to 5=extremely pleased).
Customer	5 (Scale from 1=hardly matters to 5=extremely displeased).

dissatisfaction	
Priority	5 (Scale from 1=low priority to 5=highest priority).
Conflicts	No other requirement is blocked by this one.
Actors	Procedure Designer, and ANY Procedure Stakeholder (e.g., Citizens, procedure performers, procedure owner)
Author	BEng Team
Revision	V1.0, 24/05/2016

Table 30 – Requirement CPD.4

ID	CPD.4
Name	Review of an administrative procedure
Requirement Type	Functional
Description	The Procedure Analyst reviews the new/modified administrative procedure and either sends it to the Procedure Owner for final approval or returns it to the Procedure Designer.
Rationale	Validate the modelled procedure against all the requirements before the final approval.
Fit Criterion (Measurable)	The Procedure Analyst should be able to view all the information modelled in the collaborative design task and validate its requirements compliance.
Customer satisfaction	4 (Scale from 1=uninterested to 5=extremely pleased).
Customer dissatisfaction	4 (Scale from 1=hardly matters to 5=extremely displeased).
Priority	4 (Scale from 1=low priority to 5=highest priority).
Conflicts	No other requirement is blocked by this one.
Actors	Procedure Analyst
Author	BEng Team
Revision	V1.0, 24/05/2016

Table 31 – Requirement CPD.5

ID	CPD.5
Name	Approval of an administrative procedure
Requirement Type	Functional
Description	The Procedure Owner has the faculty of approve or reject the

	new/modified administrative procedure.
Rationale	Final examination and approval of the new/modified procedure.
Fit Criterion (Measurable)	The Procedure Owner should be able to view all the information modelled in the collaborative design and eventually the review report and approve or reject it.
Customer satisfaction	4 (Scale from 1=uninterested to 5=extremely pleased).
Customer dissatisfaction	4 (Scale from 1=hardly matters to 5=extremely displeased).
Priority	4 (Scale from 1=low priority to 5=highest priority).
Conflicts	No other requirement is blocked by this one.
Actors	Procedure Owner
Author	BEng Team
Revision	V1.0, 24/05/2016

Table 32 – Requirement CPD.6

ID	CPD.6
Name	Role assignment
Requirement Type	Functional
Description	The Procedure Owner assigns particular users to the workflow roles.
Rationale	Determine who is going to participate in the collaboration.
Fit Criterion (Measurable)	The Procedure Owner should be able to associate/dissociate a set of users to/from each collaboration role.
Customer satisfaction	5 (Scale from 1=uninterested to 5=extremely pleased).
Customer dissatisfaction	4 (Scale from 1=hardly matters to 5=extremely displeased).
Priority	4 (Scale from 1=low priority to 5=highest priority).
Conflicts	No other requirement is blocked by this one.
Actors	Procedure Owner
Author	BEng Team
Revision	V1.0, 24/05/2016

Table 33 – Requirement CPD.7

ID	CPD.7
Name	Change notification
Requirement Type	Functional
Description	Each participant in the collaboration is accordingly notified about changes made by other participants.
Rationale	Keep each participant aware of the collaboration progress.
Fit Criterion (Measurable)	Each participant should be notified about changes in the workflow and/or the model state
Customer satisfaction	5 (Scale from 1=uninterested to 5=extremely pleased).
Customer dissatisfaction	5 (Scale from 1=hardly matters to 5=extremely displeased).
Priority	4 (Scale from 1=low priority to 5=highest priority).
Conflicts	No other requirement is blocked by this one.
Actors	Every participant in the collaboration workflow
Author	BEng Team
Revision	V1.0, 24/05/2016

4.3 Collective knowledge base requirements (CKB)

Table 34 – Requirement CKB.1

ID	CKB.1
Name	Store data created in QAE and CPD
Requirement Type	Functional
Description	Data created in the QAE (questions, answers, comments, terms ...) and in the CPD (flowcharts, comments...) is stored in a structured way. Furthermore, analytics on user participation, questions and answers management will be stored and provided.
Rationale	The CKB hosts a single common database for the Citizenpedia. All the generated information is stored in it, for its consumption by the Citizenpedia itself, and by other SIMPATICO components through a REST API.
Fit Criterion (Measurable)	The CKB contains the data generated in the QAE and in the Collaborative Procedures Designer.

Customer satisfaction	3 (Scale from 1=uninterested to 5=extremely pleased).
Customer dissatisfaction	5 (Scale from 1=hardly matters to 5=extremely displeased).
Priority	5 (Scale from 1=low priority to 5=highest priority).
Conflicts	This requirement is core to the entire Citizenpedia.
Actors	Citizen data sources (QAE and CPD).
Author	DEUSTO Team
Revision	V1.0, 25/05/2016

Table 35 – Requirement CKB.2

ID	CKB.2
Name	Index data modifications
Requirement Type	Functional
Description	Every time new data is created/updated/deleted in the CKB database, it must be indexed by a CKB module called indexing engine. However, data susceptible to be queried by text-based searches should only be indexed. This leaves out e.g. numeric rankings of the answers.
Rationale	The QAE should provide search capabilities beyond exact-text matching. This is achieved using an indexing engine that gathers data from a database, indexes it, and allows text-based queries over it.
Fit Criterion (Measurable)	The indexing engine must index text-based data contained in the CKB database.
Customer satisfaction	5 (Scale from 1=uninterested to 5=extremely pleased).
Customer dissatisfaction	3 (Scale from 1=hardly matters to 5=extremely displeased).
Priority	3 (Scale from 1=low priority to 5=highest priority).
Conflicts	This requirement is complementary to CKB.3
Actors	Citizens, through the QAE search feature.
Author	DEUSTO Team
Revision	V1.0, 25/05/2016

Table 36 – Requirement CKB.3

ID	CKB.3
----	-------

Name	Provide flexible searching capabilities
Requirement Type	Functional
Description	The QAE should provide search capabilities beyond exact-text matching to its user. The search function within the QAE should be able to handle typos or support derivatives (e.g. a query for “build permit” should return content matching to “construction permit” as well).
Rationale	Providing flexible search capabilities aids users in the search of what they are looking for within the Citizenpedia.
Fit Criterion (Measurable)	Searches in the QAE are able to handle typos or support derivatives.
Customer satisfaction	5 (Scale from 1=uninterested to 5=extremely pleased).
Customer dissatisfaction	3 (Scale from 1=hardly matters to 5=extremely displeased).
Priority	3 (Scale from 1=low priority to 5=highest priority).
Conflicts	This requirement is complementary to CKB.2.
Actors	Citizens, through the QAE search feature.
Author	DEUSTO Team
Revision	V1.0, 25/05/2016

Table 37 – Requirement CKB.4

ID	CKB.4
Name	Expose a RESTful API to third parties and other SIMPATICO components
Requirement Type	Functional
Description	The information contained in the CKB can be accessed through a set of REST queries, as well as the main functionalities and statistics of the usage of Citizenpedia.
Rationale	Citizenpedia is integrated within the SIMPATICO platform, what means that other components must be able to retrieve part of the information that it contains. Based on our experience, a REST API is the best way to ease the technical integration of components in a platform such as SIMPATICO.
Fit Criterion (Measurable)	Sample REST queries can retrieve information from the Citizenpedia.
Customer satisfaction	4 (Scale from 1=uninterested to 5=extremely pleased).

Customer dissatisfaction	5 (Scale from 1=hardly matters to 5=extremely displeased).
Priority	5 (Scale from 1=low priority to 5=highest priority).
Conflicts	No other requirement is blocked by this one within the Citizenpedia.
Actors	Citizens, through other SIMPATICO components and 3rd parties platforms
Author	DEUSTO Team
Revision	V1.0, 25/05/2016

Table 38 – Requirement CKB.5

ID	CKB.5
Name	Privacy and access control to data
Requirement Type	Non-functional
Description	This requirement expresses the need to ensure that the public or private nature of the datasets published is kept. Besides, it demands that in the dataset management access control is ensured. Different users and types of users will have different rights associated to registration/deregistration or CRUD operations.
Rationale	This component must be able to provide the data access, which complements the functionality related to the management of the content
Fit Criterion (Measurable)	Authorized users must have access to the management of the corresponding information. Likewise, non-authorized users should never have access to them.
Customer satisfaction	4 (Scale from 1=uninterested to 5=extremely pleased).
Customer dissatisfaction	4 (Scale from 1=hardly matters to 5=extremely displeased).
Priority	4 (Scale from 1=low priority to 5=highest priority).
Conflicts	Several requirements are blocked by this one. Concretely QAE.8 and CPD.6
Actors	Citizens, through other SIMPATICO components
Author	DEUSTO Team
Revision	V1.0, 25/05/2016

4.4 Stakeholder Incentivization techniques requirements (SIT)

Table 39 – Requirement SIT.1

ID	SIT.1
----	-------

Name	Register an action of a user
Requirement Type	Functional
Description	Every time a citizen makes an action in the Citizenpedia (e.g. create a question, post an answer...), it will be registered in the Gamification engine. Each action has a set of points associated, and each new registered action will update the user profile (also known as “player”) in the gamification engine.
Rationale	By registering every action, the Gamification engine will have a record of the action per users and a computed score. This score will be used to rank most active users and promote some of them to moderators.
Fit Criterion (Measurable)	Each time a user makes an action, its “player” profile is updated in the gamification engine.
Customer satisfaction	2 (Scale from 1=uninterested to 5=extremely pleased).
Customer dissatisfaction	4 (Scale from 1=hardly matters to 5=extremely displeased).
Priority	3 (Scale from 1=low priority to 5=highest priority).
Conflicts	This requirement is core to the gamification engine, as it provides the way to populate it.
Actors	Citizens through the QAE and CPD.
Author	DEUSTO Team
Revision	V1.0, 25/05/2016

Table 40 – Requirement SIT.2

ID	SIT.2
Name	Retrieve the state of a user in the gamification engine
Requirement Type	Functional
Description	The gamification engine returns the score achieved by a certain “player”, which is a user of the Citizenpedia, as well as the rewarding and incentives, budgets, ratings and contributions needed to progress to moderator.
Rationale	The gamification engine provides a way to query the state of a user
Fit Criterion (Measurable)	The status of a user has been computed according to the actions he has conducted in Citizenpedia
Customer satisfaction	4 (Scale from 1=uninterested to 5=extremely pleased).

Customer dissatisfaction	3 (Scale from 1=hardly matters to 5=extremely displeased).
Priority	3 (Scale from 1=low priority to 5=highest priority).
Conflicts	This requirement is complementary to QAE.5
Actors	Citizens
Author	UDEUSTO Team
Revision	V1.0, 25/05/2016

Table 41 – Requirement SIT.3

ID	SIT.3
Name	Retrieve the notifications of a user in the gamification engine
Requirement Type	Functional
Description	The gamification engine returns the set of actions and notifications (e.g. achievement of certain score threshold) for a certain user.
Rationale	The gamification engine provides a query to query the events associated to a user.
Fit Criterion (Measurable)	The actions related to a user have been registered.
Customer satisfaction	4 (Scale from 1=uninterested to 5=extremely pleased).
Customer dissatisfaction	3 (Scale from 1=hardly matters to 5=extremely displeased).
Priority	3 (Scale from 1=low priority to 5=highest priority).
Conflicts	This requirement is complementary to QAE.5
Actors	Citizens
Author	UDEUSTO Team
Revision	V1.0, 25/05/2016

4.5 Summary of Citizenpedia framework requirements

According to the previously-detailed requirements the following table summarizes them:

Table 42 – Summary of Citizenpedia framework requirements.

ID	Name	Priority
QAE.1	Creation, modification and deletion of questions	5
QAE.2	Creation, modification and deletion of answers	5

QAE.3	Upvote or downvote an answer	4
QAE.4	Create, modify or delete a comment to a question or answer	3
QAE.5	Promote a user to moderator	3
QAE.6	Store and manage definitions/explanations of public-administration terms	5
QAE.7	Low quality/spam entries monitoring	3
QAE.8	Access control to QAE items	4
QAE.9	Responsive user interface	3
CPD.1	Start a collaboration workflow	5
CPD.2	Terminate a collaboration workflow	4
CPD.3	Collaborative design of an administrative procedure	5
CPD.4	Review of an administrative procedure	4
CPD.5	Approval for an administrative procedure	4
CPD.6	Role assignment	4
CPD.7	Change notification	4
CKB.1	Store data created in QAE and CPD	5
CKB.2	Index data modifications	3
CKB.3	Provide flexible searching capabilities	3
CKB.4	Expose a RESTful API to third parties and other SIMPATICO components	5
CKB.5	Privacy and access control to data	4
SIT.1	Register an action of a user	3
SIT.2	Retrieve the state of a user in the gamification engine	3
SIT.3	Retrieve the notifications of a user in the gamification engine	3

According to the defined actors and roles, the following figures exposes the major macro use cases diagrams to clarify the main functionalities of the Citizenpedia framework.

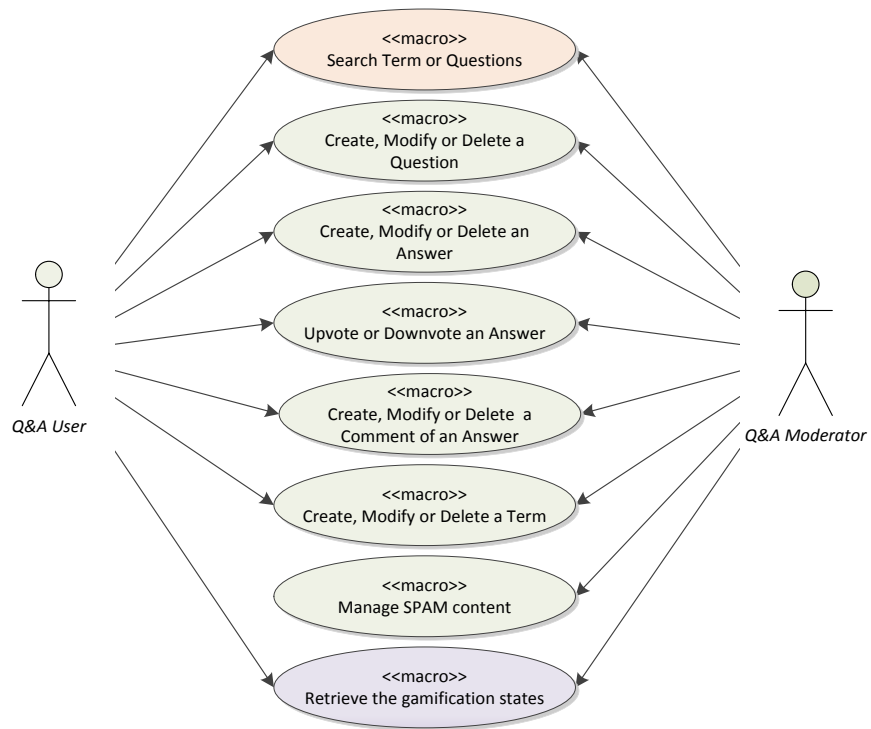


Figure 13 – Macro-use cases of Citizenpedia 1

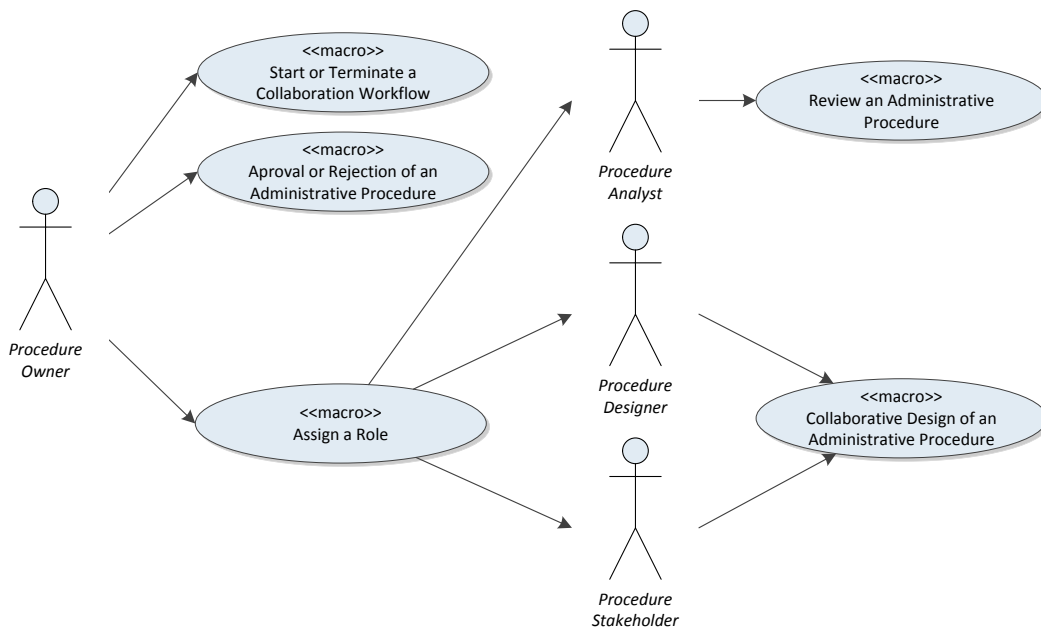


Figure 14 – Macro-use cases of Citizenpedia 2

5 Citizenpedia architecture

According to the specified requirements the architecture of the Citizenpedia framework is exposed. First, a schematic overview and description of each building block is provided. Then, a description of the interfaces for the interaction with other SIMPATICO components or third party applications is shown. Further information on internal interactions with the other SIMPATICO components will be provided in D5.1 deliverable.

5.1 Schematic Overview (“Broad Picture”)

The Citizenpedia framework will be composed of several building blocks, each providing a functionality. Some of them will provide a UI for citizens/civil servants and other will provide support for storage/queries/management. In addition, we consider two main ways of accessing the Citizenpedia: through a web user interface (mostly for citizens/civil servants) or through a REST API (aimed for the communications with other SIMPATICO components or 3rd party applications).

The architecture is defined in the following Figure 15:

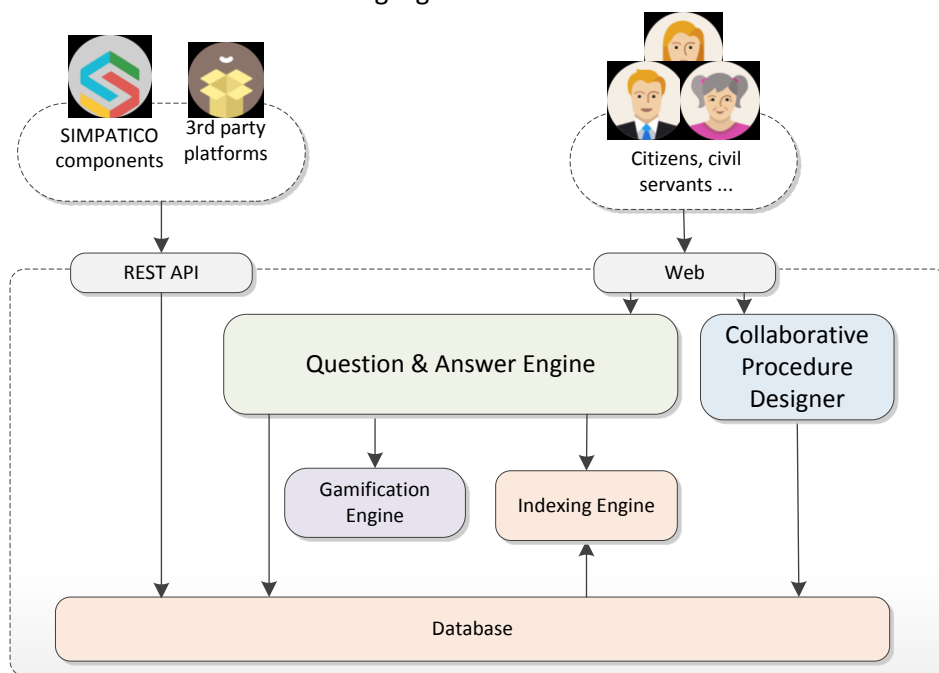


Figure 15 – Schematic overview of Citizenpedia

Next, a detailed component description and a review of the main functionality expectations for each of the depicted components is given:

5.1.1 Question Answering Engine

This part of Citizenpedia will provide a place where citizens will be able to post and resolve doubts regarding e-services and public administration. The chosen look-and-feel for this place is to develop a Question Answering engine, whose design will be inspired by existing ones such as Stack Overflow. We already presented in Section 3.2.1 a review of open source QAE tools, and the final design of Citizenpedia will result on a combination of the most useful features of them.

The main functionality of QAE will be to create and answer questions in a public manner. We will encourage users to communicate in a public manner, with the aim for all the generated information to remain over the time. This is usual in QAE places in the field of engineering (e.g. Stack Overflow), where sometimes an answer written two or three years past in time is useful for the user looking for a doubt.

Our initial design considers to have two main roles: user and moderator. Initially, every citizen will be a user and every civil servant a moderator. Users should be able to search along Citizenpedia and post content. Moderators will have higher privileges, i.e., permission to edit/delete content from other users.

We consider to implement a rewarding and reputation mechanism. Each time a user conducts an action (e.g. posting/answering a question, leaving a comment...), it will be recorded and several points will be given. Upon certain amount of points, badges will be given. This will enable to users to gain reputation and distinguish most active participants in the community. In addition, we consider that once a user reaches certain level of reputation, he/she would turn into moderator.

On the user interface (UI) design side, we consider it to be as simple as possible. Simple UIs are always better adopted by the user community, following the “less is more” principle. Furthermore, this was confirmed by the surveys conducted on potential stakeholders (see Section 2 of this deliverable): several users, especially the elderly ones, reported to consider Citizenpedia a good idea but only if it was intuitive to use. This translates into having just the necessary buttons/elements in the forms, and having a clean template.

In addition, we consider that UI must be responsive: the content must be adapted according to the screen of the device where it is being displayed. The importance of this requirement was highlighted in a conversation with the public administration of the pilot city Sheffield: they claimed that the percentage of user that accessed their services using smartphones instead of laptops/PCs have been increasing substantially over the last few years. Thus, having a responsive UI would increase the adoption of Citizenpedia among citizens.

Focusing on the QAE as part of the Citizenpedia architecture, it is connected to other three components (Figure 15): the database, the indexing engine and the gamification engine. The aim of each interaction is the following:

- The **database** will be in charge of storing all the data created in the QAE, from questions/answers to user profile information.
- The **indexing/search engine** will answer the text-based queries in the QAE, e.g. “steps for a build permit”.
- The **gamification engine** will be in charge of registering every action conducted in the QAE (e.g. answering a question), and computing the reputation skills.

5.1.2 Collaborative procedures designer

Drawing from the tools comparison in Section 3.2.2, against the background of a social construction of technology perspective, tool designers perceive modelling as predominantly asynchronous; i.e. no product allows to model synchronously on the same object. Moreover, the tool descriptions above show that no product provides comprehensive, integrated support for collaborative business process modelling. However, each of the seven tools exhibits (i) a shared representation that is uniquely

understandable by all the stakeholders in the modelling task (i.e. a **human-understandable graphical notation**) and (ii) some features that are relevant and useful in the context of joint process modelling initiatives. Taken together, these features allow us to work towards an architecture for supporting collaborative process modelling. To this end, we group the features into three dimensions, namely, **modelling roles & workflow**, **awareness creation**, and **communicative support**, which we outline in the following sub sections.

5.1.2.1 Workflow, role models and task distribution

Several tools feature the differentiation of roles with regards to particular responsibilities in the modelling process, as well as a predefined modelling workflow. Hence, modelling is seen as an act of coordinating various stakeholders working on the same set of models over time, i.e. in a certain sequence. Our analysis suggests that tool designers view a **central repository** as an important feature for facilitating collaborative process modelling. Models are held centrally and are made accessible to the modelling team. The repository is then complemented by **role management** and a (predefined) **workflow**, which is meant to facilitate effective coordination of the collaborative modelling process. The need for role differentiation is recognised in the literature as well; Dean et al. [22] have shown that the involvement of knowledgeable individuals in different stages in the workflow fosters model completeness and quality.

5.1.2.2 Awareness creation and conflict resolution

A workflow and role model will only be the basis for facilitating collaboration on the actual object of attention, i.e. the model. What is further needed is support for coordinating changes to models or parts thereof. Thus, tools need to facilitate awareness for what various stakeholders (roles) change in order to allow others to review changes, make comments and agree or disagree with the changes. Pendergast et al. [21] found a similar requirement – they conclude that awareness with regards to what has been altered in a model is a key factor for efficient convergence to a single model solution. Some of the above products display certain features in this respect, such as **notification** features, **change reports**, as well as **differencing functionality** or dedicated **conflict resolution support** (which allows models to deal with the side effects of otherwise uncoordinated, concurrent modelling). However, no tool provides holistic and comprehensive awareness support. We conclude that awareness features help to stay on top of changes by other users in concurrent modelling situations.

5.1.2.3 Communication and discussion

Our analysis reveals that some designers appear to acknowledge, through their design, that modelling is a communication-intensive task. Some tools allow commenting on changes by others, facilitating a simple form of text-based discussion, although most tools do not provide any means for targeted discussions about certain elements, parts or areas of a process model. Generally, communication features do not exceed commenting or communication similar to simple discussion boards. We see this finding as problematic since recent research has stressed the necessity to facilitate communication.

5.1.2.4 Draft architecture for collaboration support in process modelling

The above three feature areas will only be truly effective if they are intertwined and applied in conjunction, i.e. bearing on each other. Communication should tie in with the workflow and role-based responsibilities; different roles need different kinds of awareness throughout the stages of the modelling process, while roles should have an influence on how discussions evolve. While we have identified the importance of these three dimensions, their interplay, as well as specific applications and integration with existing products, they need to be the focus of future research. We

acknowledge that our model only represents a high-level outline of an architecture, which requires further development.

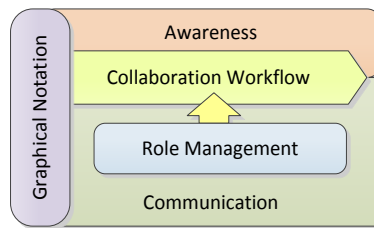


Figure 16 – Architecture for collaboration support of process modelling

The Figure 16 presents a sketch of the proposed architecture. First, it shows that role management is intimately related to the process development workflow. This role concept would not only incorporate access rights, but also determine responsibilities in different stages of the workflow. The second concept, communication, supplements the role concept by supporting dispersed teams, which have to rely on asynchronous modelling and have a high need for communication and coordination support. The third concept, awareness, further supports these needs. The awareness functionalities clarify who (which role) did what (e. g. model changes), when (time), and where (concrete model). Communication, Awareness and the Collaboration itself are supported by a common and objective language: the Graphical Notation.

Each of these concepts is interdependent; it relies on and strengthens the others. If, for example, a modeller releases a new model, the awareness concept should notify the respective role in charge of reviewing the model of any changes by using the communication functions. The reviewer might comment on the changes, which would then be communicated to the modeller and the role for approving the final model.

5.1.3 Collective knowledge base

This module covers the storage of the information and its interface to Citizenpedia modules in several ways, and to other components of the SIMPATICO platform.

The main piece of this module is the database, which will store all of the information of the QAE and the CPD. This includes user profiles, questions/answers from the QAE and flowcharts/comments from the CPD. During the course of the SIMPATICO project we will leverage possible issues on the storage of user profiles in the Citizenpedia database with the user profiles of the SIMPATICO platform.

The database will serve its information in two ways. The first one is by direct queries to it from QAE and CPD. We consider that these will be simple queries such as getting user profiles or private messages. These queries will be done using the API that the selected database provides.

Text-based queries, such as “steps for a build permit”, will be committed to the indexing engine instead of the database. This engine will index every text-based data that is pushed into the database, such as questions or comments, and allow text-based searches in a more flexible way, i.e., allowing text mistakes (e.g. typing “build permit” instead of “build permit”) and fuzzy searches (e.g. giving “construction permit” as answer when “build permit” is searched).

In addition, the information contained in the database must be exposed to other SIMPATICO components and third party applications. This will be done through a RESTful API, which will be a key factor to simplify the integration.

5.1.4 Gamification Engine

The gamification engine is the main software piece of the Stakeholder Incentivization Techniques. It allows to define and modify a scoreboard and several badges to the Citizenpedia users. Each time a user performs an action, e.g. answering a question, it will be registered in the gamification engine, and its reputation skill will be computed. During the course of the project several reputation skills will be defined, being the greatest reward to become a moderator of Citizenpedia.

5.2 Functional / Logical View (Interfaces)

This section provides a description of the purposed interfaces that will be provided by the Citizenpedia framework. According to the requirements and the main macro-use cases defined the following diagram shows the interactions of the main external stakeholders with the Citizenpedia framework.

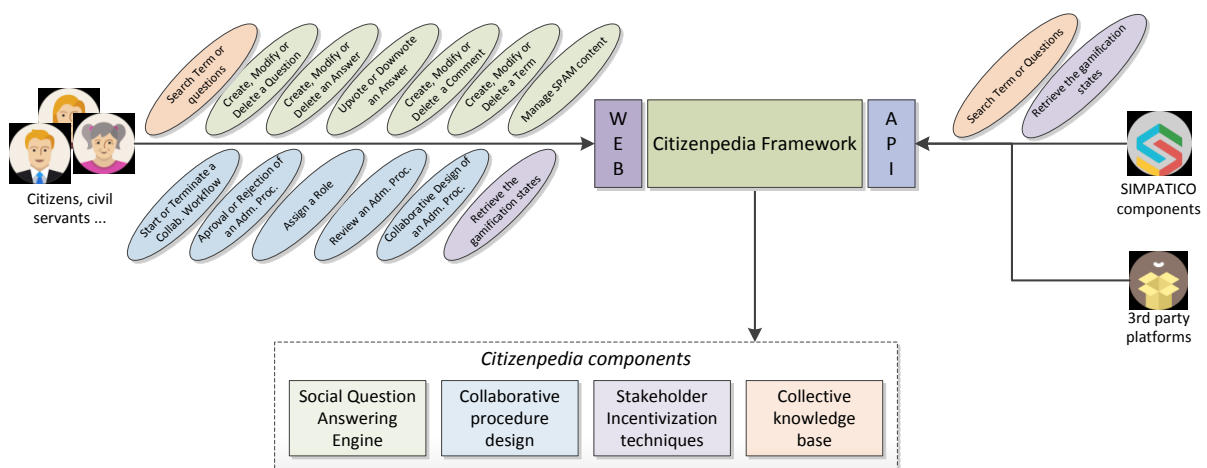


Figure 17 – Interactions of the main stakeholders

As can be observed, the SIMPATICO components may programmatically access the Citizenpedia framework through a provided RESTful API. In addition, citizens and civil servants might also access the framework functionality through the Citizenpedia Web UI which offers a user interface to access all functionalities of the framework.

According to the main detected needs of the components of the SIMPATICO platform, a preliminary description of the possible exposed functionality is shown through the following table.

Table 43 – Preliminary REST API methods.

URL	HTTP Method	Description	Provider
/qae/question/all	GET	Get all the questions	QAE
/qae/question/search?q=term1+termN	GET	Get all the questions related to the queried terms	QAE
/qae/question	POST	Creates a new question	QAE
/qae/question/{question_id}	GET	Gets all the data of a question	QAE
/qae/question/{question_id}	PUT	Updates a question	QAE
/qae/question/{question_id}	DELETE	Removes a question	QAE
/qae/question/{question_id}/answer/all	GET	Gets all the answers of a question	QAE
/qae/question/{question_id}/answer	POST	Creates a new answer of a question	QAE
/qae/stats	GET	Gets statistics about the QAE	QAE
/sit/execute	POST	Execute an action	SIT
/sit/state/{gameId}/{playerId}	GET	Read player state	SIT
/sit/state/{gameId}	GET	Read all player state in game	SIT
/sit/notification/{gameId}/{playerId}	GET	Read notifications about a player	SIT
/sit/stats	GET	Get all the statistics of the SIT	SIT

6 Conclusion

The Citizenpedia is the human computation framework within the SIMPATICO platform. It will provide a framework where citizens will be able to post and solve doubts related to e-services, and also to interact with the public administration. This deliverable describes the functional and technical aspects of Citizenpedia.

The functionality of Citizenpedia has been put into evaluation by its main stakeholders: citizens and civil servants. We created several online survey forms, and distributed them among stakeholders in the three pilots of the SIMPATICO project: Spain, Italy and England. The results of the survey have been analysed to later describe the Citizenpedia requirements accordingly. This process related to the surveys has been covered in Section 2.

The Citizenpedia framework will be a bespoke development for the SIMPATICO project. However, it will require the integration of already existing software pieces, such as databases or indexing engines. We conducted a review of the most popular software tools suitable to be integrated within Citizenpedia, highlighting their advantages and drawbacks, in Section 3.

The design stage of the Citizenpedia requires defining the functional and non-functional requirements that the framework will fulfil. These requirements have been created from two perspectives: a technical one, from the point of view of the developers, and a stakeholders' one, gathered from the surveys we created. These requirements have been described using the popular Volere methodology. This part of the work is covered in Section 4.

Finally, we wanted to go beyond the Volere-based requirements in the description of the Citizenpedia. Thus, we defined which building blocks will compose Citizenpedia, and provide a deeper description of each of them in Section 5. We also include a description of the public interfaces that Citizenpedia will provide for its interconnection with other SIMPATICO components and third party applications.

7 References

- [1] L. Yishan and S. Manoharan, “A performance comparison of SQL and NoSQL databases,” in *IEEE Pacific Rim Conference on Communications, Computers and Signal Processing (PACRIM)*, 2013.
- [2] “Knowledge Base of Relational and NoSQL Database Management Systems,” [Online]. Available: <http://db-engines.com/en/ranking>.
- [3] “MySQL vs PostgreSQL,” [Online]. Available: https://www.wikivs.com/wiki/MySQL_vs_PostgreSQL.
- [4] J. Lin and D. Ryaboy, “Scaling big data mining infrastructure: the twitter experience,” *ACM SIGKDD Explorations Newsletter 14.2*, pp. 6-19, 2013.
- [5] M. Hammer, “What is Business Process Management? Handbook on Business Process Management,” Berlin, Springer, 2010.
- [6] M. Weske, “BPM Concepts, Languages, Architectures,” Heidelberg, Springer, 2007.
- [7] P. Rittgen, “Collaborative Modeling – A Design Science Approach,” in *42nd Hawaii International Conference, Hawaii, IEEE*, 2009.
- [8] P. Rittgen, “Collaborative Business Process Modeling – Tool Support for Solving Typical Problems,” in *Proceedings of the Conf-IRM 2010 “Collaboration and Community in a Global World*, Montego Bay, Jamaica, 2010.
- [9] K. Riemer, J. Holler and M. Indulska, “Collaborative process modelling — tool analysis and design implications,” in *ECIS, 2011*, 2011.
- [10] B. Curtis, M. Kellner and J. Over, “Process Modeling,” *Communications of the Association for Computing Machinery*, vol. 35, no. 9, pp. 75-90, 1992.
- [11] I. Davies, P. Green, M. Rosemann, M. Indulska and S. Gallo, “How Do Practitioners Use Conceptual Modeling in Practice?,” *Data & Knowledge Engineering*, vol. 58, no. 3, pp. 358-380, 2006.
- [12] R. Aguilar-Savén, “Business process modelling: Review and framework,” *International Journal of Production Economics*, vol. 2, no. 90, pp. 129-149, 2004.
- [13] D. Georgakopoulos, M. Hornick and A. Sheth, “An Overview of Workflow Management: From Process Modeling to Workflow Automation Infrastructure,” *Distributed and Parallel Databases*, vol. 2, no. 3, pp. 119-153, 1995.
- [14] G. Giaglis, “A Taxonomy of Business Process Modeling and Information Systems Modeling Techniques,” *The International Journal of Flexible Manufacturing Systems*, vol. 13, no. 2, pp. 209-228, 2001.
- [15] D. Moody, “Theoretical and Practical Issues in Evaluating the Quality of Conceptual Models: Current State and Future Directions,” *Data & Knowledge Engineering*, vol. 3, no. 55, pp. 243-276, 2005.
- [16] J. Recker, *Understanding Process Modelling Grammar Continuance: A Study of the Consequences of Representational Capabilities*, Queensland University of Technology, Brisbane, Australi: Dissertation, 2008.
- [17] T. Eikebrokk, J. Iden, D. Olsen and A. Opdahl, “Exploring Process-Modelling Practice: Towards a Conceptual Model,” in *Proceedings of the 41st Hawaii International Conference on System*

Sciences (HICCS'08), Big Island, HI, USA, 2008.

- [18] M. Indulska, J. Recker, M. Rosemann and P. Green, "Business Process Modeling: Current Issues and Future Challenges," in *Proceedings of the 21st International Conference on Advanced Information Systems (CAIS'09)*, Amsterdam, The Netherlands., 2009.
- [19] K. Ryu and E. Yücesan, "CPM: A collaborative process modeling for cooperative manufacturers," *Advanced Engineering Informatics*, vol. 2, no. 21, 2007.
- [20] E. Adamides and N. Karacapilidis, "A knowledge centred framework for collaborative business process modelling.," *Business Process Management Journal*, vol. 5, no. 12, p. 2006, 557-575.
- [21] M. Pendergast and K. Aytes, "Supporting the group creation of formal and informal graphics during business process modeling.," *Interacting with Computers*, vol. 4, no. 11, pp. 355-373, 1999.
- [22] D. Dean and R. Orwig, "Modeling with a group modeling tool: group support, model quality, and validation," in *Twenty-Seventh Hawaii International Conference*, Wailea, HI, USA, 1994.
- [23] D. Dori and D. Beimel, "OPCATeam – collaborative business process modeling with OPM," *Business Process Management*, pp. 66-81, 2004.
- [24] A. Meire and M. Borges, "Supporting multiple viewpoints in collaborative graphical editing," *Multimedia Tools and Applications*, vol. 2, no. 32, pp. 185-208, 2007.
- [25] K. Riemer, "E-Collaboration Systems – Identification of System Classes using Cluster Analysis," *International Journal of E-Collaboration*, vol. 5, no. 3, pp. 1-24, 2009.
- [26] "CA Process Automation," [Online]. Available: <http://www.ca.com/us/products/ca-process-automation.html>.
- [27] "ARIS Architect & Designer," [Online]. Available: http://www.softwareag.com/corporate/products/aris_alfabet/bpa/products/architect_design/overview/.
- [28] "Ultimate Modeling Power," [Online]. Available: <http://www.sparxsystems.com/products/ea/index.html>.
- [29] "iGrafx Application Areas - Process Modeling & Analysis," [Online]. Available: <http://www.igrafx.com/products/process-modeling-analysis>.
- [30] "Microsoft Visio," [Online]. Available: <https://products.office.com/visio/flowchart-software>.
- [31] "WebSphere Business Modeler Advanced," [Online]. Available: <http://www.ibm.com/software/products/it/modeler-advanced>.
- [32] "Signavio Process Editor," [Online]. Available: <http://www.signavio.com/products/process-editor/>.
- [33] "Apache Solr," [Online]. Available: <http://lucene.apache.org/solr/>.
- [34] "Elasticsearch | Elastic," [Online]. Available: <https://www.elastic.co/products/elasticsearch>.
- [35] "ARIS Community!," [Online]. Available: <http://www.ariscommunity.com>.